Online Matching With Recourse: Random Edge Arrivals

Aaron Bernstein*[1] and Aditi Dudeja†[1]

[1]Rutgers University

Abstract

The matching problem in the online setting models the following situation: we are given a set of servers in advance, the clients arrive one at a time, and each client has edges to some of the servers. Each client must be matched to some incident server upon arrival (or left unmatched) and the algorithm is not allowed to reverse its decisions. Due to this no-reversal restriction, we are not able to guarantee an *exact* maximum matching in this model, only an approximate one.

Therefore, it is natural to study a different setting, where the top priority is to match as many clients as possible, and changes to the matching are possible but expensive. Formally, the goal is to always maintain a maximum matching while minimizing the number of changes made to the matching (denoted the *recourse*). This model is called the online model with recourse, and has been studied extensively over the past few years. For the specific problem of matching, the focus has been on vertex-arrival model, where clients arrive one at a time with all their edges. A recent result of Bernstein et al. [BHR19] gives an upper bound of $O\left(n \log^2 n\right)$ recourse for the case of general bipartite graphs. For trees the best known bound is $O(n \log n)$ recourse, due to Bosek et al. [BLSZP18]. These are nearly tight, as a lower bound of $\Omega(n \log n)$ is known.

In this paper, we consider the more general model where all the vertices are known in advance, but the edges of the graph are revealed one at a time. Even for the simple case where the graph is a path, there is a lower bound of $\Omega(n^2)$. Therefore, we instead consider the natural relaxation where the graph is worst-case, but the edges are revealed in a *random* order. This relaxation is motivated by the fact that in many related models, such as the streaming setting or the standard online setting without recourse, the matching problem becomes easier when the input comes in a random order. Our results are as follows:

- Our main result is that for the case of general (non-bipartite) graphs, the problem with random edge arrivals is almost as hard as in the adversarial setting: we show a family of graphs for which the expected recourse is $\Omega\left(\frac{n^2}{\log n}\right)$.
- We show that for some special cases of graphs, random arrival is significantly easier. For the case of trees, we get an upper bound of $O\left(n \log^2 n\right)$ on the expected recourse. For the case of paths, this upper bound is $O\left(n \log n\right)$. We also show that the latter bound is tight, i.e. that the expected recourse is at least $\Omega\left(n \log n\right)$.

*bernstei@gmail.com. This work was done while funded by NSF Award 1942010 and the Simon's Group for Algorithms & Geometry

†aditi.dudeja@rutgers.edu

1 Introduction

The online matching problem models a scenario in which a set of servers is given in advance, and a set of clients arrive one at a time, with each client incident to some of the servers. In the standard version of this model, the arriving client must be immediately matched to a free server or be left unmatched, and this decision is irrevocable. Due to this constraint, it is not possible to guarantee an *exact* matching, so the goal is to guarantee the best possible approximation. (See the work of Karp et al. [KVV90], which shows that we can't get better than $1-\frac{1}{e}$ approximation.)

But there are several applications where the top priority is to match *all* the clients (or at least to have a maximum matching), and the irreversibility condition of the standard online model is too restrictive; in applications such as streaming content delivery, web hosting, job scheduling, or remote storage it is preferable to reallocate the clients provided the number of reallocations is small (see [CDKL09] for more details). Therefore, over the past decade there have been many papers on the so-called online model with *recourse*, where the goal is to maintain an *exact* solution the problem, while making as few changes to this solution as possible.

In the case of matching in particular, existing results focus on the vertex-arrival model, which is analogous to the similar model in online matching without recourse. In this model, clients arrive one at a time and ask to be matched to a server. The algorithm is allowed to change the matching over time and must always maintain a maximum matching: the goal is then to minimize the total number of changes made to the matching, denoted the *recourse*. Note that the trivial recourse bound is $O(n^2)$ (n changes per client), but one can do significantly better. This model has been studied extensively (see for example, [GKKV95, CDKL09, BLSZ14, BLSZ15, GKS14, BLSZP18, BHR19]), and the state of the art is an upper bound of $O(n\log^2 n)$ on the total recourse [BHR19]) in bipartite graphs. For the special case of trees, the best known upper bound is $O(n\log n)$ due to [BLSZP18]. These upper bounds nearly match the lower bound of $\Omega(n\log n)$ for trees due to [GKKV95].

In this paper, we consider a more general model where the graph can be non-bipartite and, more importantly, the *edges* in the graph are revealed one at a time; the algorithm must again maintain a maximum matching at all times. Unfortunately, we have very strong lower bounds when the order in which the edges arrive is adversarial; even for the simplest possible case of a path, $\Omega(n^2)$ recourse is necessary. To overcome this lower bound, we consider a natural relaxation of this model where the adversary can still choose the graph, but edges arrive in a *random* order. One of the motivations behind this relaxation is that in several related models, such as the online model without recourse or the streaming model, the matching problem is shown to be easier when the input arrives in a random order. (See [KMT11, MY11] for online model without recourse, and [KMM12, KKS14, GKMS19, FHM$^+$20] for the streaming model).

Our results show that for the case of trees and paths, we can do significantly better in the random edge-arrival model: in particular, we show an upper bound of $O(n\log n)$ on the expected recourse in the case of paths (which we show is tight), and a bound of $O\left(n\log^2 n\right)$ in the case of trees. But our main result is that in general graphs, the random arrival setting is provably almost as hard as the adversarial setting. We state our main results formally:

Theorem 1. There is a family of (non-bipartite) graphs G_n with n vertices and $\Theta(n\log n)$ edges, such that if edges of the graph arrive in a random order, then the total expected recourse taken by **any** algorithm that maintains a maximum matching in the graph is $\Omega\left(\frac{n^2}{\log n}\right)$.

Theorem 2. Let T be a tree and let the edges of T arrive one at a time in a random order. Then, the expected total recourse taken by **any** algorithm that maintains a maximum matching in T is at most $O(n\log^2 n)$.

Theorem 3. Let P be a path on n vertices, and let the edges of P arrive in a random order. The expected total recourse taken by **any** algorithm that maintains a maximum matching in P is $O(n \log n)$. Moreover, this bound is tight: the expected recourse taken by any algorithm is $\Omega(n \log n)$.

Remark 4. For the lower bounds of Theorems 1 and 3, when we say that *any* algorithm has expected recourse $\Omega(T)$, this holds even if the algorithm knows the random permutation in advance. That is, the lower bound holds even if the algorithm is optimal for every possible ordering of the edges.

Remark 5. For the upper bounds in Theorems 2 and 3, the algorithm we use simply changes the matching along an augmenting path whenever such a path becomes available due to the insertion of some edge. If there are multiple augmenting paths the algorithm can take, it chooses between them arbitrarily; the upper bound holds regardless of the choice of path.

We leave as an intriguing open problem whether our lower bound in Theorem 1 also holds for *bipartite* graphs, or whether these graphs allow for expected $o(n^{2-\varepsilon})$ recourse when edges arrive in a random order. See Section 5 for more details.

2 Preliminaries

Let G be an unweighted graph. A matching in G is a set of vertex-disjoint edges. Given any matching M of G, we say that a vertex v is matched if it incident to an edge in M, and free otherwise. Given any two matchings M and M', we use $M \oplus M'$ to denote the symmetric difference. We study the model of online matching with recourse under random edge arrivals. In this model, the adversary fixes any graph $G = (V, E)$ with m edges and n vertices. The vertex set is given in advance, but the edges arrive one at a time; the arrival order $e_1, \ldots, e_m$ is a *random* permutation of E. The goal of the algorithm is to maintain a sequence of matchings $M_1, \ldots M_m$, such that M_i is a maximum matching in the graph $(V, \{e_1, \ldots, e_i\})$. The total recourse of the algorithm is $\sum_{i=1}^{m-1} |M_i \oplus M_{i+1}|$, which is the total number of changes made to the matching throughout the entire sequence of insertions.

Intuitively, an algorithm that minimizes recourse should only change the matching when the maximum matching in the graph increases in size. We formalize this notion below.

Definition 6. We say that an algorithm is only-augmenting if for every $1 \leq i \leq m-1$, either $M_i = M_{i+1}$, or $M_i \oplus M_{i+1}$ consists of a single augmenting path; that is, $M_i \oplus M_{i+1}$ consists of an odd-length path P in $\{e_1, \ldots, e_{i+1}\}$ such that every second edge of P is in M_i, but the first and last edges of P are not in M_i.

Simplifying Assumption: Throughout the paper, we only consider algorithms that are only-augmenting.

The above assumption is clearly justified for upper bounds, since we just need to present *some* algorithm with bounded recourse. For lower bounds, we justify the assumption with the following lemma, which is proved in Section B of the appendix.

Lemma 7. (Justification of Simplifying Assumption) Let G be some graph whose edges arrive in a random order. Say that we can prove that any only-augmenting algorithm that maintains a maximum matching in G has expected recourse $\Omega(T)$. Then any algorithm (possibly not only-augmenting) that maintains a maximum matching in G has expected recourse $\Omega(T)$.

3 Lower Bound on Expected Recourse in General Graphs

This section will be devoted to proving Theorem 1, the main result of our paper. Our proof will proceed as follows. In Section 3.1 we define our candidate graph G_n (we will refer to it as G from now). The main step will be to show that between the times when half the edges of the graph have arrived and a three-quarters of the edges have arrived, the graph induced by non-isolated vertices contains a perfect matching or a near perfect matching throughout (see Definition 10 for a definition of near perfect matching). We will then use this fact to prove Theorem 1.

3.1 The Graph

We use n to denote the number of vertices in our graph. In this write-up, $s = 400 \log n$ and $t = \frac{n}{500 \log n}$. Let K_s denote the complete graph on s vertices. Our graph is called G (see Figure 1) and it consists of t copies of K_s that we index as $K_s^{(i)}$ for $1 \leq i \leq t$. The remaining $\frac{n}{5}$ vertices are partitioned into t sets $\left\{D^{(i)}\right\}_{1 \leq i \leq t}$ of size $100 \log n$ each. The graph G contains the following edges.

(a) For $1 \leq i \leq t-1$, we introduce edges between every vertex of $K_s^{(i)}$ and every vertex of $K_s^{(i+1)}$. Additionally, edges are also introduced between every vertex of $K_s^{(1)}$ and every vertex of $K_s^{(t)}$.

(b) For $1 \leq i \leq t$, we fix an arbitrary set $U^{(i)} \subset K_s^{(i)}$ of size $100 \log n$. Introduce an arbitrary matching between $U^{(i)}$ and $D^{(i)}$. Call this matching $M^{(i)}$. Let $M = \cup_{i=1}^{t} M^{(i)}$; we add the edges of M to G. For any $u \in D \cup U$, we define $M(u)$ to be the vertex that u is matched to. We also let $U = \cup_{i=1}^{t} U^{(i)}$ and $D = \cup_{i=1}^{t} D^{(i)}$.

We denote the number of edges in G by m. Note that $m = \Theta(n \log n)$.

3.2 Relating G_p and $G^{p \cdot m}$

Definition 8. Let $p \in [0,1]$. We define $E_p \subset E(G)$ to be the set of edges obtained by sampling each $e \in E(G)$ with probability p. Let $V_p = V(G) \setminus \{v \in D \text{ such that } (v, M(v)) \notin E_p\}$; note that V_p excludes isolated vertices in D. Let G_p be the induced subgraph $G[V_p]$.

Definition 9. Let $E^{p \cdot m} \subset E(G)$ be the set of edges obtained by sampling $p \cdot m$ random edges of $E(G)$. Let $V^{p \cdot m} = V(G) \setminus \{v \in D \text{ such that } (v, M(v)) \notin E^{p \cdot m}\}$; note that $V^{p \cdot m}$ excludes isolated vertices in D. Let $G^{p \cdot m}$ be the induced subgraph on $G[V^{p \cdot m}]$.

Definition 10. Let H be a graph with an odd number of vertices. Let $\mathcal{M}$ be any matching of H that leaves exactly one vertex unmatched. Then, $\mathcal{M}$ is called a near perfect matching of H.

We state the main theorem that we want to prove in this section:

Theorem 11. Let $p \in [0.5, 0.75]$, then, the graph $G^{p \cdot m}$ contains a perfect matching or a near perfect matching with probability at least $1 - O\left(\frac{1}{n^3}\right)$.

To prove this theorem, we claim that it is sufficient to prove the following theorem:

Theorem 12. Let $p \in [0.5, 0.75]$, then, graph G_p contains a matching or a near perfect matching with probability at least $1 - O\left(\frac{1}{n^4}\right)$.

To show that Theorem 12 implies Theorem 11, we prove the following lemma:

Lemma 13. Let $p \in [0.5, 0.75]$, and let $G^{p \cdot m}$ and G_p be as described above, and let $\mathcal{G}$ be the set of graphs that contain a perfect matching or a near perfect matching, then,

$$\Pr\left(G^{p \cdot m} \notin \mathcal{G}\right) \leq 10\sqrt{m} \cdot \Pr\left(G_p \notin \mathcal{G}\right).$$

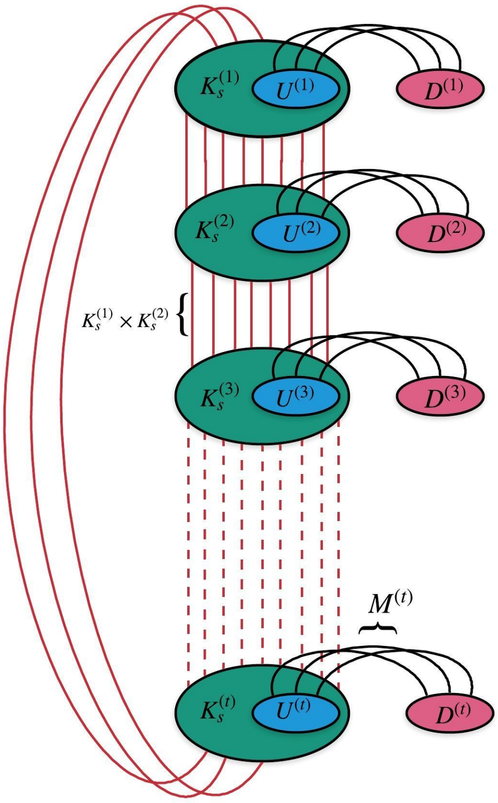

Figure 1: Graph G

We refer the reader to Section A of the appendix for a proof of Lemma 13. For now, we prove Theorem 11 assuming Theorem 12 and Lemma 13:

Proof (Theorem 11). It follows from Lemma 13 that:

$$
\begin{aligned}
\Pr\left(G^{p \cdot m} \text{ does not contain a matching}\right) &\leq 10\sqrt{m} \cdot \Pr\left(G_p \text{ does not contains a perfect matching}\right) \\
&= 10\sqrt{m} \cdot O\left(\frac{1}{n^4}\right) \qquad \text{(Due to Theorem 12)} \\
&= O\left(\frac{1}{n^3}\right). \qquad \text{(Since } m = \Theta\left(n \log n\right)\text{)}
\end{aligned}
$$

□

The following corollary follows from Theorem 11, via a union bound:

Corollary 14. Consider the graphs $\mathcal{G} = \{G^{p \cdot m}\}_{p \in [0.5, 0.75]}$. The probability that every $G \in \mathcal{G}$ contains a perfect matching or a near perfect matching is at least $1 - O\left(\frac{1}{n}\right)$.

The bulk of our paper is proving Theorem 12. But first, we provide some intuition for our choice of G by sketching how Corollary 14 implies our main result (Theorem 1).

Proof sketch of Theorem 1. Recall the edges $M \subset E(G)$ that connect the vertices in D, where $|M| = \Theta(n)$ (see 3.1). Consider how the graph $G^{p \cdot m}$ evolves from for $p = \frac{1}{2}$ to $p = \frac{3}{4}$. Let us assume without loss of generality that $G^{\frac{1}{2} \cdot m}$ contains an even number of vertices. Whenever an edge (d, x) from M is inserted into the graph, $d \in D$ is added to $V(G^{p \cdot m})$ (See Definition 8). Since we know from Corollary 14 that $G^{p \cdot m}$ contains a perfect matching whenever $V(G^{p \cdot m})$

is even, we know that after every two edges (d, x) and (d', x') added to M, there is a perfect matching in the resulting graph; thus, the algorithm must take some augmenting path from d to d'. Because G consists of $\Omega\left(\frac{n}{\log(n)}\right)$ consecutive layers, it is easy to see that with probability $\frac{1}{2}$, the shortest path from d to d' has length $\Omega\left(\frac{n}{\log(n)}\right)$. We expect to add $\frac{|M|}{4} = \Omega(n)$ edges to M between $G^{\frac{1}{2}\cdot m}$ and $G^{\frac{3}{4}\cdot m}$, so we have $\Omega(n)$ augmenting paths of expected length $\Omega\left(\frac{n}{\log(n)}\right)$, which implies total augmenting path length $\Omega\left(\frac{n^2}{\log(n)}\right)$. See Section 3.5 for full proof. $\square$

3.3 Proving G_p has a Near-Perfect Matching

We now turn to proving Theorem 12. To this end, we introduce some notation:

Definition 15. Given G_p, we define the active subgraph A of G_p as follows: let $V(A) = V(G_p) \setminus \{u \in D \cup U : (u, M(u)) \in G_p\}$. The active subgraph A is the induced subgraph $G[V(A)]$.

Definition 16. We define $A^{(i)}$ to be the induced subgraph on $V(A) \cap V\left(K_s^{(i)}\right)$ for $1 \leq i \leq t$. For $1 \leq i \leq t$, let $|V\left(A^{(i)}\right)| = a_i$. Then,

(a) If a_i is even, then let $P^{(i)} \cup Q^{(i)}$ be an arbitrary $\frac{a_i}{2}$ by $\frac{a_i}{2}$ bipartition of $V(A^{(i)})$.

(b) If a_i is odd, then let $v^{(i)}$ be an arbitrary vertex in $V(A^{(i)})$ and let $P^{(i)} \cup Q^{(i)}$ be an arbitrary $\lfloor \frac{a_i}{2} \rfloor$ by $\lfloor \frac{a_i}{2} \rfloor$ bipartition of $V(A^{(i)}) \setminus v^{(i)}$.

We denote $G(P^{(i)}, Q^{(i)})$ to be the bipartite graph between $P^{(i)}$ and $Q^{(i)}$

Claim 17. We observe that $V(A) \cap D = \emptyset$. This follows from the following two facts:

(a) Consider any $u \in D$ such that $(u, M(u)) \notin G_p$. Then, $u \notin V(G_p)$. This follows immediately from Definition 8.

(b) By Definition 15, we know that any u such that $(u, M(u)) \in G_p$ is not included in $V(A)$.

Claim 18. From Definition 15, we know that $a_i \geq 400 \log n - |U^{(i)}|$. Since $|U^{(i)}| = 100 \log n$ (see Section 3.1 (b)), it follows that $a_i \geq 300 \log n$.

In order to prove Theorem 12, it is sufficient to prove the following theorem:

Theorem 19. The active subgraph, A contains a perfect matching or a near perfect matching with probability at least $1 - O\left(\frac{1}{n^4}\right)$.

Proof (Theorem 12). Given a perfect (resp. near-perfect) matching $\mathcal{M}(A)$ of A, we will construct a perfect (resp. near perfect) matching $\mathcal{M}(G_p)$ of G_p. Consider any $u \in V(G_p) \setminus V(A)$. Note that $M(u) \in V(G_p) \setminus V(A)$ and $(u, M(u)) \in G_p$. So we may match u to $M(u)$ in G_p. In particular, $\mathcal{M}(G_p) = \mathcal{M}(A) \cup \{(u, M(u)) \text{ where } u \in V(G_p) \setminus V(A)\}$. Thus, $\mathcal{M}(G_p)$ is a perfect (or a near perfect matching) of G_p if $\mathcal{M}(A)$ is a perfect (or a near perfect matching) of A. $\square$

3.4 Near Perfect Matching in Active Subgraph

To prove Theorem 19, we need Chernoff bound, and some existing results on matchings in random bipartite graphs.

Theorem 20. [JLR00] Define $B(n, n, p)$ to be the bipartite graph obtained by deleting edges from $K_{n,n}$ independently with probability $1 - p$. Then,

$$\Pr\left(B(n, n, p) \text{ does not contain a perfect matching }\right) = O\left(ne^{-np}\right).$$

Theorem 21 (Chernoff Bounds)**.** Let $X_0, \cdots, X_k$ be $0-1$ random variables that are independent. Let $\mu = \mathbb{E}\left[\sum_{i=1}^{k} X_i\right]$. Then,

$$\Pr\left(\sum_{i=1}^{k} X_i \leq (1-\delta)\mu\right) \leq e^{-\frac{\delta^2\mu}{2}} \text{ and,} \tag{1}$$

$$\Pr\left(\sum_{i=1}^{k} X_i \geq (1+\delta)\mu\right) \leq e^{-\frac{\delta^2\mu}{3}}. \tag{2}$$

Consider the $A^{(i)}$'s in Definition 16. We mentioned that for some of these $A^{(i)}$'s the corresponding a_i's might be odd. Let $\left\{A^{(i_1)}, \cdots, A^{(i_k)}\right\}$ be this set, with $i_1 < \cdots < i_k$. Let $v^{(i_j)}$ be the vertex left out of the bipartition $P^{(i_j)} \cup Q^{(i_j)}$ of $A^{(i_j)}$ for $1 \leq j \leq k$ (see Definition 16(b)). We define the following events:

Definition 22. For $1 \leq i \leq t$, let $\mathcal{A}_i$ be the event that $G(P^{(i)}, Q^{(i)})$ contains a perfect matching (or a near perfect matching). Let $\mathcal{A} = \cap_{i=1}^{t} \mathcal{A}_i$.

Definition 23. Let $\mathcal{M}'_i$ be the maximum matching of $G\left(P^{(i)}, Q^{(i)}\right)$ for $1 \leq i \leq t$. Let $\mathcal{M}' = \cup_{i=1}^{t} \mathcal{M}'_i$.

Definition 24. For $1 \leq m \leq \left\lfloor \frac{k}{2} \right\rfloor$, let $\mathcal{B}_m$ be the event that there is an augmenting path between $v^{(i_{2m-1})}$ and $v^{(i_{2m})}$ with respect to $\mathcal{M}'$. Let $\mathcal{B} = \bigcap_{i=1}^{\lfloor \frac{k}{2} \rfloor} \mathcal{B}_m$.

In order to prove Theorem 19, we follow these steps:

(a) We will prove that each $\mathcal{A}_i$ happens with high probability, and therefore by union bound, $\mathcal{A}$ happens with high probability also.

(b) We prove that each $\mathcal{B}_m$, conditioned on $\mathcal{A}$ happens with high probability, and by union bound, $\mathcal{B}$ conditioned on $\mathcal{A}$ also happens with high probability.

This will imply that the active graph, A contains a perfect matching or a near perfect matching with high probability.

Before we move on to proving *(a)* and *(b)*, we note that $G(P^{(i)}, Q^{(i)})$ and $V(A^{(i)})$ are both random variables. In particular, $V(A^{(i)}) = \left(V(K_s^{(i)}) \setminus U^{(i)}\right) \cup S$, where S is a uniformly random subset of $U^{(i)}$. However, if we fix the vertex set $V\left(A^{(i)}\right)$, then the edges of $G(P^{(i)}, Q^{(i)})$ are precisely equivalent to those of a random bipartite graph; we remind the reader that $P^{(i)} \cup Q^{(i)}$ is an arbitrary bipartition of $A^{(i)}$ (see Definition 16). Formally:

Observation 25. For $1 \leq i \leq t$, $G\left(P^{(i)}, Q^{(i)}\right)$ conditioned on $V\left(A^{(i)}\right) = S$, where $|S| = a_i$, has the same distribution as $B\left(\left\lfloor \frac{a_i}{2} \right\rfloor, \left\lfloor \frac{a_i}{2} \right\rfloor, p\right)$.

Now we prove the following lemma:

Lemma 26. For $1 \leq i \leq t$, $\Pr(\neg\mathcal{A}_i) = O\left(\frac{1}{n^5}\right)$. Moreover, $\Pr(\neg\mathcal{A}) = O\left(\frac{1}{n^4}\right)$.

Proof. We know that:

$$\begin{aligned}
\Pr(\neg\mathcal{A}_i) &= \sum_T \Pr\left(\neg\mathcal{A}_i \,\middle|\, V\left(A^{(i)}\right) = T\right) \cdot \Pr\left(V\left(A^{(i)}\right) = T\right) \\
&= \sum_T O\left(|T| \cdot e^{-|T|}\right) \cdot \Pr\left(V\left(A^{(i)}\right) = T\right) \\
&\quad \text{(Follows from Observation 25 and Lemma 20)} \\
&= \sum_T O\left(\frac{1}{n^5}\right) \cdot \Pr\left(V\left(A^{(i)}\right) = T\right)
\end{aligned}$$

(Follows from Claim 18 that $a_i \geq 300 \log n$)

$$= O\left(\frac{1}{n^5}\right).$$

(Since we are summing over disjoint events)

By union bound it follows that, $\Pr(\neg\mathcal{A}) = O\left(\frac{1}{n^4}\right)$. $\square$

Theorem 27. For $1 \leq m \leq \lfloor \frac{k}{2} \rfloor$, $\Pr(\neg\mathcal{B}_m \mid \mathcal{A}) = O\left(\frac{1}{n^8}\right)$. Therefore, by union bound it follows that $\Pr(\neg\mathcal{B} \mid \mathcal{A}) = O\left(\frac{1}{n^7}\right)$.

Proof. To bound $\Pr(\neg\mathcal{B}_m \mid \mathcal{A})$, we consider two cases:

(a) **Case 1: $\mathbf{v_{i_{2m-1}}}$ and $\mathbf{v_{i_{2m}}}$ are in consecutive layers.** That is, $i_{2m} = i_{2m-1} + 1$. We will give an overview of what we are about to do. We will use v to denote $v_{i_{2m-1}}$, v' to denote $v_{i_{2m}}$, P and P' to denote $P^{(i_{2m-1})}$ and $P^{(i_{2m})}$, Q and Q' to denote $Q^{(i_{2m-1})}$ and $Q^{(i_{2m})}$ respectively.

Let $N_P(v)$ (resp. $N_{P'}(v')$) denote the set of vertices in P (resp. P') adjacent to v (resp. v'). Let $\deg_P(v)$ (resp. $\deg_{P'}(v')$) denote $|N_P(v)|$ (resp. $|N_{P'}(v')|$).

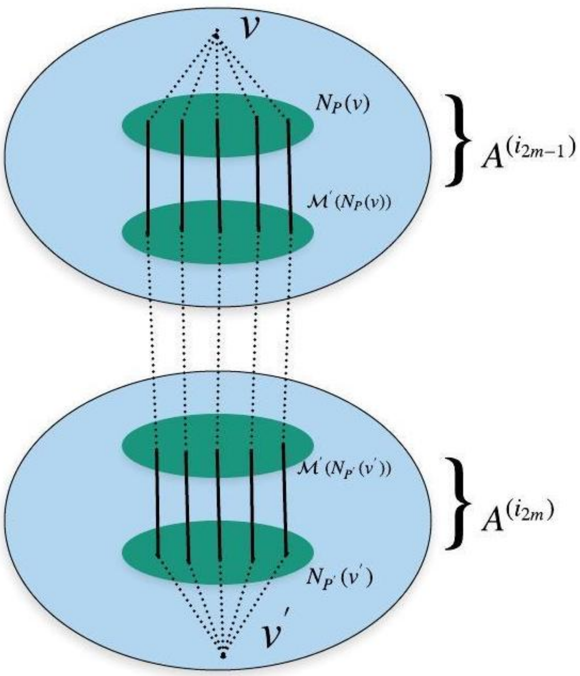

Figure 2: Case (a): When unmatched vertices are in consecutive layers

For a set of vertices S, let $\mathcal{M}'(S)$ denote the set of vertices matched to S in $\mathcal{M}'$ (refer to Definition 23 for the definition of $\mathcal{M}'$). We will prove that with high probability $|\mathcal{M}'(N_P(v))|$ and $|\mathcal{M}'(N_{P'}(v'))|$ are large. Conditioned on these sizes being large, we will prove that there is an edge (x, x') in A where $x \in \mathcal{M}'(N_P(v))$ and $x' \in \mathcal{M}'(N_{P'}(v'))$. It follows there is an augmenting path $\mathcal{P} = (v, \mathcal{M}'(x), x, x', \mathcal{M}'(x'), v')$ in A (note that $\mathcal{M}'(x) \in N_P(v)$ and $\mathcal{M}'(x') \in N_{P'}(v')$). (See Figure 2)

To show this, we first show that $|N_P(v)|$ and $|N_{P'}(v')|$ are large with high probability. We will condition on $\mathcal{A}$, so $|\mathcal{M}'(N_P(v))|$ and $|\mathcal{M}'(N_{P'}(v'))|$ will consequently be large with high probability. It then follows that one of the edges between these two sets is in A with high probability.

We now turn to the formal proof of case (a). Let X_v and $X_{v'}$ be the random variables denoting $\deg_P(v)$ and $\deg_{P'}(v')$ respectively. Each edge incident on v and v' in A is sampled independently with probability $p \in [0.5, 0.75]$. This is true even if we condition on the event $\mathcal{A}$. Consequently, $\mathbb{E}[X_v \mid \mathcal{A}] = \mathbb{E}[X_v] \geq 75 \log n$. Since X_v is the sum of $0-1$ independent random variables, we may apply Chernoff bound (see Theorem 21). It follows that:

$$\Pr(X_v \leq 25 \log n \mid \mathcal{A}) = O\left(\frac{1}{n^8}\right).$$

Similarly, we have:

$$\Pr(X_{v'} \leq 25 \log n \mid \mathcal{A}) = O\left(\frac{1}{n^8}\right).$$

Define $\mathcal{Y}$ to be the event that $|\mathcal{M}'(N_P(v))| \geq 25 \log n$ and $|\mathcal{M}'(N_{P'}(v'))| \geq 25 \log n$. Observe that,

$$\begin{aligned}\Pr(\neg\mathcal{Y} \mid \mathcal{A}) &\leq \Pr(X_v \leq 25 \log n \mid \mathcal{A}) + \Pr(X_{v'} \leq 25 \log n \mid \mathcal{A}) \\ &= O\left(\frac{1}{n^8}\right).\end{aligned}$$

Define $\mathcal{Z}$ to be the event that there is an edge between $\mathcal{M}'(N_P(v))$ and $\mathcal{M}'(N_{P'}(v'))$. Observe that,

$$\begin{aligned}\Pr(\neg\mathcal{Z} \mid \mathcal{A}) &\leq \Pr(\neg\mathcal{Y} \mid \mathcal{A}) + \Pr(\neg\mathcal{Z} \mid \mathcal{Y}, \mathcal{A}) \\ &= O\left(\frac{1}{n^8}\right) + \frac{1}{n^{O(\log n)}}.\end{aligned}$$

The second term follows from the fact that each edge appears independently with probability $p \in [0.5, 0.75]$, and there are $\Omega(\log^2 n)$ edges between $\mathcal{M}'(N_P(v))$ and $\mathcal{M}'(N_{P'}(v'))$ conditioned on $\mathcal{Y}$. It follows that $\Pr(\neg\mathcal{B}_m \mid \mathcal{A}) \leq \Pr(\neg\mathcal{Z} \mid \mathcal{A}) = O\left(\frac{1}{n^8}\right)$. This proves our claim for this case.

(b) **Case 2: $\mathbf{i_{2m} > i_{2m-1} + 1}$.** We denote $v_{i_{2m-1}}$ by v, $P^{(i_{2m-1})}$ by P and $v^{(i_{2m})}$ by v'. Let $f = i_{2m} - i_{2m-1}$. For $1 \leq j \leq f$, let $P^{(i_{2m-1}+j)}$ be denoted by $P + j$. We similarly define Q and $Q + j$. We also define the following sets:

$$\begin{aligned}S_0 &= N_P(v) \\ S_j &= N_{P+j}(\mathcal{M}'(S_{j-1})) \text{ for } 1 \leq j \leq f.\end{aligned}$$

For $0 \leq j \leq f$, let $\mathcal{X}_j$ be the event that $|\mathcal{M}'(S_j)| \geq 25 \log n$. Let $\mathcal{E}$ be the event that there is an edge between v' and $\mathcal{M}'(S_f)$. It is easy to check that the occurrence of $\mathcal{X}_0, \mathcal{X}_1 \cdots, \mathcal{X}_f$ implies that there is an alternating path from v to a large set of vertices (at least $\Omega(\log n)$) in $Q + j$ for all $j \in [f]$. Note that $\mathcal{E}$ implies that there is an edge from $Q + f$ to v'. Combined, $\mathcal{X}_1 \cdots, \mathcal{X}_f, \mathcal{E}$ imply an augmenting path from v to v'. We thus have:

Observation 28. Let $\mathcal{B}_m$ and $\mathcal{X}_1, \cdots, \mathcal{X}_f, \mathcal{E}$ be as defined above (refer to Definition 24 for a definition of $\mathcal{B}_m$), then:

$$\Pr(\mathcal{B}_m \mid \mathcal{A}) \geq \Pr\left(\cap_{k=0}^{f} \mathcal{X}_k \cap \mathcal{E} \;\middle|\; \mathcal{A}\right).$$

From the above observation, we deduce that in order to upper bound $\Pr(\neg\mathcal{B}_m \mid \mathcal{A})$, it is sufficient to upper bound $\Pr\left(\cup_{k=0}^{f}\neg\mathcal{X}_k \cup \neg\mathcal{E} \;\middle|\; \mathcal{A}\right)$. We know that:

$$\Pr\left(\cup_{k=0}^{f}\neg\mathcal{X}_k \cup \neg\mathcal{E} \;\middle|\; \mathcal{A}\right) \leq \sum_{k=0}^{f} \Pr\left(\neg\mathcal{X}_k \mid \cap_{k=0}^{i-1}\mathcal{X}_k \cap \mathcal{A}\right) + \Pr\left(\neg\mathcal{E} \;\middle|\; \cap_{k=0}^{f}\mathcal{X}_k \cap \mathcal{A}\right).$$

(Follows from the definition of conditional probability)

We computed $\Pr(\neg\mathcal{X}_0 \mid \mathcal{A})$ in case (a). We remind the reader this is just the probability that $|\mathcal{M}'(S_0)| \leq 25\log n$. We now show how to compute $\Pr(\neg\mathcal{X}_j \mid \mathcal{A}, \mathcal{X}_0, \cdots, \mathcal{X}_{j-1})$. Consider any $w \in P + j$. We want to compute the probability that w is in the set $N_{P+j}(\mathcal{M}'(S_{j-1})) = S_j$ conditioned on the event $\mathcal{X}_{j-1}$ and $\mathcal{A}$ i.e. $|\mathcal{M}'(S_{j-1})| \geq 25\log n$. Since every edge on w is present in the active graph A independently with probability p:

$$\begin{aligned} \Pr(v \notin S_j \mid \mathcal{A}, \mathcal{X}_0, \cdots, \mathcal{X}_{j-1}) &\leq (1-p)^{25\log n} \\ &\leq \left(\frac{1}{2}\right)^{25\log n} \end{aligned} \tag{3}$$

(Follows from the fact that $p \geq 0.5$)

This implies that:

$$\mathbb{E}\left[|S_j| \mid \mathcal{A}, \mathcal{X}_0, \cdots, \mathcal{X}_{j-1}\right] \geq 100\log n$$

Since $|S_j|$ is a sum of $0-1$ random variables (it is the sum of $\mathbb{1}_{\{v\in S_j\}}$, that take value 0 with probability $O\left(\frac{1}{n^{25}}\right)$ (due to eq. (3)) and 1 otherwise), we can apply Chernoff bounds (Theorem 21):

$$\Pr\left(|S_j| \leq 25\log n \mid \mathcal{A}, \mathcal{X}_0, \cdots, \mathcal{X}_{j-1}\right) = O\left(\frac{1}{n^9}\right)$$

Since we condition on $\mathcal{A}$ (that is a perfect or, a near perfect matching being present), we know that:

$$|\mathcal{M}'(S_j)| = |S_j|$$

Consequently, we have:

$$\begin{aligned} \Pr\left(|\mathcal{M}'(S_j)| \leq 25\log n \;\middle|\; \mathcal{A}, \mathcal{X}_0, \cdots, \mathcal{X}_{j-1}\right) &= \Pr\left(|S_j| \leq 25\log n \mid \mathcal{A}, \mathcal{X}_0, \cdots, \mathcal{X}_{j-1}\right) \\ &= O\left(\frac{1}{n^9}\right) \end{aligned}$$

Finally, we want to bound $\Pr(\neg\mathcal{E} \mid \mathcal{A}, \mathcal{X}_0, \cdots, \mathcal{X}_f)$. This can be upper bounded:

$$\Pr(\neg\mathcal{E} \mid \mathcal{A}, \mathcal{X}_0, \cdots, \mathcal{X}_f) \leq \left(\frac{1}{2}\right)^{25\log n}$$

(Edges on v' appear independently with probability $p \geq 0.5$)

$$= O\left(\frac{1}{n^{25}}\right)$$

It is immediate from Observation 28 that:

$$\Pr(\neg\mathcal{B}_m \mid \mathcal{A}) = O\left(\frac{1}{n^8}\right)$$

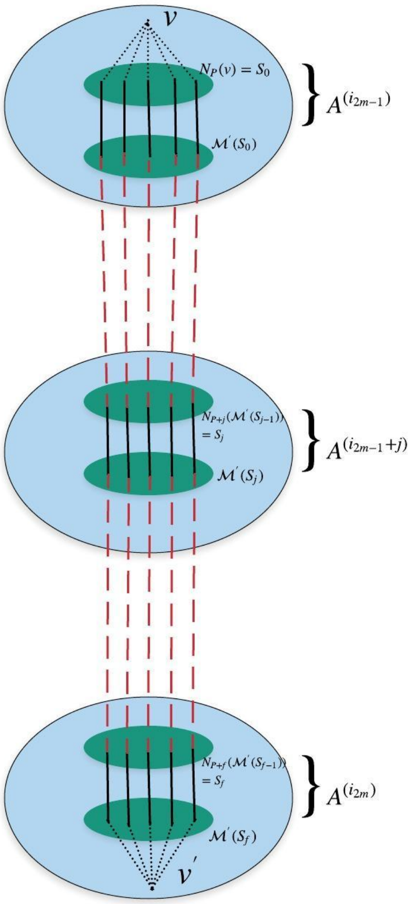

Figure 3: Case (b): When v and v' are not in consecutive layers

.

From case (a) and case (b), we know that by union bound, $\Pr(\neg\mathcal{B} \mid \mathcal{A}) = O\left(\frac{1}{n^7}\right)$ $\square$

Proof (Theorem 19). From Lemma 26 and Theorem 27 we have that:

$$\begin{aligned}\Pr(A \text{ does not contain a perfect matching }) &\leq \Pr(\neg\mathcal{A}) + \Pr(\neg\mathcal{B} \mid \mathcal{A}) \\ &= O\left(\frac{1}{n^4}\right).\end{aligned}$$

$\square$

3.5 Lower Bound On Lengths of Augmenting Paths

We start with some definitions:

Definition 29. For $i \in \{1, \cdots, m\}$, we denote by e_i, the edges arriving at time i. Let $S = \{e_{0.5m}, \cdots, e_{0.75m}\}$.

This section will be devoted to proving that among the edges in S, $\Omega(n)$ edges will be forced to augment along paths of expected length $\Omega\left(\frac{n}{\log n}\right)$. Formally,

Theorem 30. With high probability, there exists $S' \subset S$, $|S'| \geq \frac{n}{100}$ such that each $e \in S'$ is forced to augment along a path of expected length at least $\Omega\left(\frac{n}{\log n}\right)$, irrespective of the algorithm used to find the maximum matching in the graph.

We first give a proof of Theorem 1 using Theorem 30:

Proof (Theorem 1). For $i \in [m]$, let $\mathcal{Z}_i$ be the random variable denoting the length of the augmenting path that we augment along when the edge e_i joins. Let $\mathcal{Z} = \sum_{i=1}^{m} \mathcal{Z}_i$, which is the random variable denoting the total length of the augmenting paths taken during the course of the algorithm. We want to compute the quantity $\mathbb{E}[\mathcal{Z}]$. We note that:

$$\begin{aligned}\mathbb{E}[\mathcal{Z}] &= \sum_{i=1}^{m} \mathbb{E}[\mathcal{Z}_i] \\ &\geq \sum_{j \in S'} \mathbb{E}[\mathcal{Z}_j] \\ &= |S'| \cdot \Omega\left(\frac{n}{\log n}\right) \\ &= \Omega\left(\frac{n^2}{\log n}\right) \\ &\text{(Due to Theorem 30).}\end{aligned}$$

□

Before we prove Theorem 30, we need certain observations, and the following version of Chernoff for negatively associated random variables:

Theorem 31. [DP09] Let $X_0, \cdots, X_k$ be $0-1$ random variables that are negatively associated. Let $\mu = \mathbb{E}\left[\sum_{i=1}^{k} X_i\right]$. Then,

$$\Pr\left(\sum_{i=1}^{k} X_i \leq (1-\delta)\mu\right) \leq e^{-\frac{\delta^2 \mu}{2}} \text{ and,} \tag{4}$$

$$\Pr\left(\sum_{i=1}^{k} X_i \geq (1+\delta)\mu\right) \leq e^{-\frac{\delta^2 \mu}{3}}. \tag{5}$$

We remind the reader of the edges M in graph G between D and U (refer to Section 3.1(b)). Note that $|M| \geq \frac{n}{5}$. Further, $M = \cup_{i=1}^{t} M^{(i)}$, and $|M^{(i)}| \geq 100 \log n$ for all $i \in [t]$.

We now prove the following claim about S:

Claim 32. Let $\mathcal{R}$ be the event that for all $i \in [t]$, $|M^{(i)} \cap S| \geq 10 \log n$. Then, $\Pr(\mathcal{R}) \geq 1 - O\left(\frac{1}{n^3}\right)$.

Proof. Consider any $M^{(i)}$, and let $e \in M^{(i)}$. Let Z_e be a $0-1$ random variable that takes value 1 if $e \in S$, and 0 otherwise. Let $Z = \sum_{e \in M^{(i)}} Z_e$. This is the random variable that denotes $|M^{(i)} \cap S|$. Further, Z is a sum of negatively associated random variables, and therefore obeys the condition of Theorem 31. We note the following:

$$\begin{aligned}\Pr(Z_e = 1) &= \frac{1}{4} \\ \mathbb{E}[Z] &= 25 \log n\end{aligned}$$

It follows that:

$$\begin{aligned}\Pr\left(Z \leq 10\log n\right) &\leq \exp\left(-\left(0.6\right)^2\left(0.5\right)25\log n\right)\\ &\leq \exp\left(-4.5\log n\right)\\ &= O\left(\frac{1}{n^4}\right).\end{aligned}$$

Due to union bound, we know that $\Pr\left(\mathcal{R}\right) \geq 1 - O\left(\frac{1}{n^3}\right)$. $\square$

We also have the following corollary due to Claim 32:

Corollary 33. With probability at least $1 - O\left(\frac{1}{n^3}\right)$, $|M \cap S| \geq \frac{n}{50}$.

We are ready to define the candidate set S' in Theorem 30. Let $M \cap S = \left\{e_{i_1}, \cdots, e_{i_q}\right\}$. Let us assume without loss of generality that before the arrival of e_{i_1}, the set $V(G_{i_1-1})$ is even, so by Theorem 11 the graph G_{i_1-1} has a perfect matching. We define S' to contain every second edge of $M \cap S$: that is, $S' = \left\{e_{i_2}, e_{i_4}, \cdots, e_{i_{2\lfloor\frac{q}{2}\rfloor}}\right\}$. For the rest of the proof we proceed as follows: we will show that with high probability, when $e_{i_{2s}}$ arrives, it will join an augmenting path ending at $e_{i_{2s-1}}$ where $s \in \left\{1, \cdots, \left\lfloor\frac{q}{2}\right\rfloor\right\}$. Let $e_{i_{2s}} \in M^{(j)}$ and $e_{i_{2s+1}} \in M^{(j')}$. Then, the length of the augmenting path that $e_{i_{2s-1}}$ joins is at least $d(e_{i_{2s-1}}, e_{i_{2s}}) = \min\left\{t - |j' - j|, |j' - j|\right\}$. We prove that the expected value of this quantity is at least $\Omega\left(\frac{n}{\log n}\right)$.

We prove the following observation:

Lemma 34. For all $s \in \left\{1, \cdots, \left\lfloor\frac{q}{2}\right\rfloor\right\}$, $\mathbb{E}\left[d\left(e_{i_{2s-1}}, e_{i_{2s}}\right)\right] \geq \frac{n}{2000\log n}$.

Proof. Consider $e_{i_{2s-1}}$ then the number of edges in M at a distance k from $e_{i_{2s-1}}$ is $200\log n$. This implies that:

$$\begin{aligned}\Pr\left(d\left(e_{i_{2s-1}}, e_{i_{2s}}\right) = k\right) &= \frac{1000\log n}{n}\\ \mathbb{E}\left[d\left(e_{i_{2s-1}}, e_{i_{2s}}\right)\right] &= \sum_{k=0}^{\frac{t}{2}} k \cdot \Pr\left(d\left(e_{i_{2s-1}}, e_{i_{2s}}\right) = k\right)\\ &\geq \frac{t}{4}\\ &= \frac{n}{2000\log n}.\end{aligned}$$

$\square$

Lemma 35. If $G^{p\cdot m}$ contains a perfect matching or a near perfect matching for all $p \in [0.5, 0.75]$, then for all $s \in \left\{1, \cdots, \left\lfloor\frac{q}{2}\right\rfloor\right\}$, $e_{i_{2s}}$ is forced to augment along a path that ends in $e_{i_{2s-1}}$.

Proof. We remind the reader that $|V\left(G^{p\cdot m}\right)|$ is a random variable (check Definition 9) and it's value increases if and only if the edges in M arrive. Recall the assumption that $|V\left(G_{i_1-1}\right)|$ is even. Upon the arrival of e_{i_1}, we have a near perfect matching in the graph, and this remains the case until e_{i_2} arrives. At this point under our assumption, there must be a perfect matching in the graph, and the only unmatched vertices are the end points of e_{i_1} and e_{i_2} in D. (Here we use the simplifying assumption from the preliminaries that the algorithm is only-augmenting, so since the arrival of e_{i_1} does not increase the size of the maximum matching, and since the algorithm only changes the matching via augmenting paths, the endpoint of e_{i_1} in D remains free until the arrival of e_{i_2}.) It follows that these endpoints are joined together by an augmenting path. Continuing this way, we can prove the theorem for any $s \in \left\{1, \cdots, \left\lfloor\frac{q}{2}\right\rfloor\right\}$. $\square$

Proof (Theorem 30). Let $\mathcal{F}$ be the event that there is an $S' \subset S$, $|S'| \geq \frac{n}{100}$ such that each $e \in S'$ augments along a path of expected length at least $\Omega\left(\frac{n}{\log n}\right)$. Note that the event $\mathcal{F}$ fails to happen if one of these go wrong:

(a) $|S'| \leq \frac{n}{100}$. We call this event $\neg\mathcal{U}$. We know from Corollary 33 that $\Pr(\neg\mathcal{U}) = O\left(\frac{1}{n^3}\right)$. This is because S' just takes alternate elements from S.

(b) Let $\mathcal{V}$ be the event that for all $p \in [0.5, 0.75]$, $G^{p \cdot m}$ contain a perfect matching or a near perfect matching. Then, from Lemma 35 we know that $\mathcal{V}$ implies that for all $s \in \left\{1, \cdots, \left\lfloor \frac{q}{2} \right\rfloor\right\}$, $e_{i_{2s-1}}$ is forced to join an augmenting path ending in $e_{i_{2s}}$. From Lemma 34, we know all these paths have expected length at least $\frac{n}{2000 \log n}$. We know from Corollary 14, that $\Pr(\neg\mathcal{V}) = O\left(\frac{1}{n}\right)$.

It follows that the occurrence of $\mathcal{A}$ and $\mathcal{B}$ implies the occurrence of $\mathcal{F}$. Consequently, $\Pr(\mathcal{F}) \geq 1 - \Pr(\neg\mathcal{U}) - \Pr(\neg\mathcal{V}) \geq 1 - O\left(\frac{1}{n}\right)$. □

4 Upper And Lower Bounds On Trees

4.1 Upper Bound on Trees

This section will be devoted to proving results on trees. Let T be a tree on n vertices, for $i \in [n-1]$, we denote by e_i the edge arriving at time $n-1-i$.

We recall Theorem 2:

Theorem 2. Let T be a tree and let the edges of T arrive one at a time in a random order. Then, the expected total recourse taken by **any** algorithm that maintains a maximum matching in T is at most $O(n \log^2 n)$.

Definition 36. Consider an edge e. Let $\mathcal{Z}_e^k$ be the event that e on arrival joins an augmenting path of length k; that is, when e joins, there is an augmenting path of length k that contains e. Let $\mathcal{W}_e^k$ be the event that when e joins, there is *some* path of length k that contains e. Similarly, we will use $\mathcal{Z}_e^{\geq k}$ to denote the event that e on arrival joins an augmenting path of length at least k and $\mathcal{W}_e^{\geq k}$ to denote the event that e on arrival joins a path of length at least k.

Definition 37. Let e be any edge. Let AP_e be a random variable denoting the length of the augmenting path that e joins, and let P_e be a random variable denoting the length of the path that e joins.

The following observation follows from the fact that if e_i on arrival joins an augmenting path of length k, then it joins a path of length k:

Observation 38. Consider edge e_i, that is, the edge that arrives at time $n-1-i$, then:

$$\Pr\left(\mathcal{Z}_{e_i}^k\right) \leq \Pr\left(\mathcal{W}_{e_i}^k\right) \text{ and,}$$
$$\mathbb{E}\left[AP_{e_i}\right] \leq \mathbb{E}\left[P_{e_i}\right].$$

In order to prove Theorem 2, it is sufficient to prove the following lemma:

Lemma 39. Let P_{e_i} be as defined above, then, $\mathbb{E}\left[\sum_{i=1}^{N} P_{e_i}\right] = O\left(n \log^2 n\right)$.

We state the proof of Theorem 2 using Lemma 39:

Proof (Theorem 2). Let AP be the random variable denoting the expected total length of the augmenting paths taken by the algorithm. Then, $AP = \sum_{i=1}^{n} AP_{e_i}$. This implies that:

$$\mathbb{E}[AP] = \sum_{i=1}^{n-1} \mathbb{E}\left[AP_{e_i}\right]$$

$$\leq \sum_{i=1}^{n-1} \mathbb{E}\left[P_{e_i}\right]$$
$$= O\left(n \log^2 n\right).$$

(Follows from Lemma 39).

□

We now prove Lemma 39 to complete the proof of Theorem 2:

Proof Lemma 39. Let $e_i = e$, and consider a fixed path L of length k such that $L \subseteq T$ and $e \in L$; note that T refers to *all* the edges in the tree, not just those that have arrived so far. In order for e to join L when e arrives, each edge on L must have appeared before e. That is, the edges of this k-length path must all be among $e_{i+1}, \cdots, e_{n-1}$. Now, fixing $k = \frac{4 \cdot n \cdot \log n}{i}$ we have:

$$\begin{aligned}
\Pr\left(\text{All edges of } L \text{ appear before } e \mid e_i = e\right) &\leq \prod_{e' \in L} \Pr\left(e' \text{ appears before } e \;\middle|\; e_i = e\right) \\
&\leq \left(\frac{n-i-1}{n-1}\right)^k \\
&\leq e^{-\left(\frac{i \cdot k}{n-1}\right)} \\
&= O\left(\frac{1}{n^4}\right).
\end{aligned}$$

Using the fact there are at most n^2 paths at any point in the tree:

$$\begin{aligned}
\Pr\left(\mathcal{W}_{e_i}^{\geq k}\right) &\leq \sum_{e \in T} \Pr\left(\mathcal{W}_{e_i}^{\geq k} \;\middle|\; e_i = e\right) \\
&= n^2 \cdot O\left(\frac{1}{n^4}\right) \\
&= O\left(\frac{1}{n^2}\right).
\end{aligned}$$

Recalling that $k = \frac{4 \cdot n \cdot \log n}{i}$, we have:

$$\begin{aligned}
\mathbb{E}\left[P_{e_i}\right] &\leq n \cdot \Pr\left(\mathcal{W}_{e_i}^{\geq k}\right) + \frac{4 \cdot n \cdot \log n}{i} \\
&\leq \frac{1}{n} + \frac{4 \cdot n \cdot \log n}{i} \\
\mathbb{E}\left[\sum_{i=1}^{n-1} P_{e_i}\right] &= O\left(n \log^2 n\right).
\end{aligned}$$

This finishes the proof of our Lemma. □

4.2 Upper Bound on Paths

We next prove our theorem for paths, recalling Theorem 3:

Theorem 3. Let P be a path on n vertices, and let the edges of P arrive in a random order. The expected total recourse taken by **any** algorithm that maintains a maximum matching in P is $O(n \log n)$. Moreover, this bound is tight: the expected recourse taken by any algorithm is $\Omega\left(n \log n\right)$.

Let $\{e'_1, \cdots, e'_{n-1}\}$ be the edges of the path P. We remind the reader of the definition of the event $\mathcal{W}^k_{e'_i}$ (see Definition 36). We have the following simple observation:

Observation 40. For any $i \in [n-1]$, $\Pr\left(\mathcal{W}^k_{e'_i}\right) \leq \frac{2}{k^2}$. Therefore, $\mathbb{E}\left[P_{e'_i}\right] = O(\log n)$.

Proof. Consider any path $L = \left\{e'_{j_1}, \cdots, e'_{j_k}\right\}$ of length k that e'_i can join. Let $e'_i = e'_{jl}$ for some $l \in [k]$. Then,

$$\Pr\left(e'_i \text{ on arrival joins } L\right) = \frac{2 \cdot (k-1)!}{(k+2)!} \leq \frac{2}{k^3}.$$

The denominator corresponds to all possible orderings of $e'_{i_1-1}, \cdots, e'_{i_k+1}$. On the other hand, if we want e'_{i_l} to join L, then $e'_{i_1}, \cdots, e'_{i_l-1}, e'_{i_l+1}, \cdots, e'_{i_k}$ must appear before e'_{i_l} and e'_{i_1-1} and e'_{i_k+1} must appear after e'_{i_l}. The numerator refers to the number of such orderings. Since the number of choices for L are k, we get $\Pr\left(\mathcal{W}^k_{e'_i}\right) \leq k \cdot \frac{2}{k^3} \leq \frac{2}{k^2}$, and $\mathbb{E}\left[P_{e'_i}\right] \leq \sum_{k=1}^{n} k \cdot \frac{1}{k^2} = O(\log n)$, as desired. □

With this, we have the proof of the upper bound in Theorem 3:

Proof (Theorem 3). Let $AP = \sum_{i=1}^{n-1} AP_{e'_i}$. Due to observation 38, we know that:

$$\begin{aligned}\mathbb{E}\left[\sum_{i=1}^{n-1} AP_{e'_i}\right] &\leq \mathbb{E}\left[\sum_{i=1}^{n-1} P_{e'_i}\right] \\ &= O(n \log n).\end{aligned}$$

□

4.3 Lower Bound on Paths

This subsection will be devoted to proving the lower bound in Theorem 3. We note that each edge on arrival joins two paths, so we have the following definition:

Definition 41. We use R_i and Q_i to denote the two paths joined by e'_i when it arrives. (R_i and/or Q_i could be empty if e'_i is incident to a previously isolated vertex.) We also use $Q_i^{+\frac{1}{2}}$ (respectively, $Q_i^{-\frac{1}{2}}$) to denote the half of Q_i that is away (respectively, near) e'_i (see Figure 4).

Definition 42. We say that two subpaths Q,R of P are *connected* by edge $e = (v_i, v_{i+1})$ if Q,R are disjoint, one of the paths has v_i as an endpoint, and the other has v_{i+1} as an endpoint. Now, fix any subpaths Q, R that are connected by the edge e'_i. We define $\mathcal{D}^i_{Q,R}$ to be the event that $Q_i = Q$ and $R_i = R$.

Definition 43. For a path L, we use $|L|$ to denote the number of edges on L. For any two edges e and f, we define $\delta(e, f)$ to be the number of edges on the path P between e and f. For example, $\delta\left(e'_1, e'_{n-1}\right) = n - 1$.

Next, we state the following simple observation:

Observation 44. If $|R_i|$ and $|Q_i|$ are even, then e'_i on arrival joins creates an augmenting path between two unmatched vertices u and v, where $u \in R_i$ and $v \in Q_i$.

Definition 45. Suppose we condition on the event $\mathcal{D}^i_{Q,R}$ where Q and R are fixed paths that are connected by the edge e'_i. We let t_Q and t_R denote the times when the last edge on Q and the last edge on R arrived. Let e_{t_Q} and e_{t_R} denote the last edges to arrive on Q and R respectively (see Figure 4).

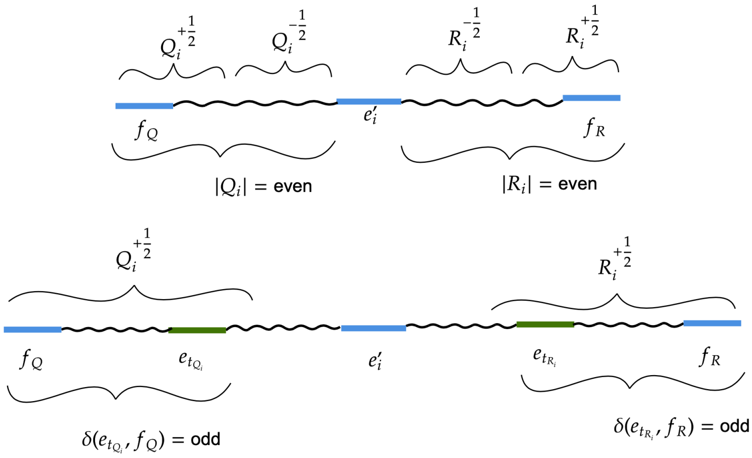

Figure 4: Conditions for a long augmenting path to be created

Our intuition for proving the lower bound is the following. Suppose e_{t_Q} ends up being in $Q^{+\frac{1}{2}}$. Before the arrival of e_{t_Q}, $Q^{+\frac{1}{2}}$ was broken into two segments. Suppose the segment further from e_i' is of even length, then this segment contains an unmatched vertex. Further, the arrival of e_{t_Q} doesn't alter the matched/unmatched status of this vertex. This is because the overall length of Q is even, so one vertex will be unmatched. The same argument holds for e_{t_R} also if it ends up being in $R^{+\frac{1}{2}}$. This will imply that when e_i' arrives, it creates an augmenting path between some vertex in $Q^{+\frac{1}{2}}$ and some vertex in $R^{+\frac{1}{2}}$, so the length of this augmenting path is about half the length of the path $Q \circ e_i' \circ R$ that e_i' joins. We state this intuition formally as a lemma:

Lemma 46. Consider the edge e_i', and let Q and R be the even-length paths connected by e_i'. Let f_Q and f_R be the edges of Q and R respectively, that are the furthest from e_i'. Let $\mathcal{E}^i_{Q,R} \subset \mathcal{D}^i_{Q,R}$ be the event that $e_{t_Q} \in Q^{+\frac{1}{2}}$, $e_{t_R} \in R^{+\frac{1}{2}}$, and $\delta\left(f_Q, e_{i_Q}\right)$, $\delta\left(f_R, e_{i_R}\right)$ are odd. Then, the occurrence of $\mathcal{E}^i_{Q,R}$ implies that e_i' on arrival joins two unmatched vertices $v \in Q^{+\frac{1}{2}}$ and $v' \in R^{+\frac{1}{2}}$, and the length of the augmenting path between v and v' is at least $\frac{k-1}{2}$.

Proof. Note that since $\delta\left(f_Q, e_{t_Q}\right)$ is odd, the segment of $Q^{+\frac{1}{2}}$ that f_Q is contained in before the arrival of e_{t_Q}, has even length. This implies that it contains an unmatched vertex, v. The same argument holds for v' as well. Since $v \in Q^{+\frac{1}{2}}$ and $v' \in R^{+\frac{1}{2}}$, we know that the augmenting path between v and v' has at least $\frac{k-1}{2}$ edges. $\square$

We are now ready to prove the lower bound in Theorem 3.

Proof (Theorem 3). We remind the reader of the definition of $AP_{e_i'}$ (see Definition 37). We let $i \in \left[\frac{n}{3}+1, \frac{2n}{3}\right]$ and $k \leq \frac{n}{3}$. Using Observation 44 and Lemma 46, we arrive at following inequality:

$$\mathbb{E}\left[AP_{e_i'}\right] \geq \sum_{k \leq \frac{n}{3}} \left(\frac{k-1}{2}\right) \cdot \Pr\left(\bigcup_{\substack{|Q|+|R|=k-1 \\ |Q|,|R| \text{ even}}} \mathcal{E}^i_{Q,R}\right)$$

$$= \sum_{k \le \frac{n}{3}} \sum_{\substack{|R|+|Q|=k-1 \\ |Q|,|R| \text{ even}}} \left(\frac{k-1}{2}\right) \cdot \Pr\left(\mathcal{E}^i_{Q,R}\right)$$

To see why the above equation is true, note that Observation 44 tells us that if the paths being joined are even length paths, then they each contain an unmatched vertex. Lemma 46 states that the occurrence of the event $\mathcal{E}^i_{P,Q}$ ensures that these unmatched vertices are far apart. Here are we using the simplifying assumption from the preliminaries: since the algorithm only changes the matching via augmenting paths, those unmatched vertices remained the same until the arrival of e'_i. Therefore, the probability of occurrence of an augmenting path of length $\frac{k-1}{2}$ involving e'_i is lower bounded by $\Pr\left(\bigcup_{\substack{|Q|+|R|=k-1 \\ |Q|,|R| \text{ even}}} \mathcal{E}^i_{Q,R}\right)$. The equality in the next line follows from the fact that the events $\left\{\mathcal{E}^i_{Q,R}\right\}$ are disjoint. Now, let v and v' be the unmatched vertices in Q and R, where $|Q| + |R| = k - 1$, and $|Q|, |R|$ are even. Then,

$$\Pr\left(\mathcal{E}^i_{Q,R}\right) \ge \frac{2}{k \cdot (k+1) \cdot (k+2)} \cdot \left(\frac{1}{8}\right) \cdot \left(\frac{1}{8}\right)$$

The first term in the expression is due to $\Pr\left(\mathcal{D}^i_{Q,R}\right)$(we refer the reader to Observation 40 for a formal proof). The second term is due to the fact that conditioned on $\mathcal{D}^i_{Q,R}$, there are $\left\lfloor \frac{|Q|}{4} \right\rfloor$ choices for e_{i_Q} that satisfy the condition of $\mathcal{E}^i_{Q,R}$ (that is, only $e \in P^{+\frac{1}{2}}$ with $\delta(e, f_Q)$ odd, satisfy the condition of $\mathcal{E}^i_{Q,R}$). The same argument holds for choices for e_{i_R} as well. It follows from the two equations above that:

$$\begin{aligned} \mathbb{E}\left[AP_{e'_i}\right] &\ge \sum_{k \le \frac{n}{3}} \left(\frac{k+1}{2}\right) \cdot \left(\frac{k-1}{2}\right) \cdot \left(\frac{1}{8}\right) \cdot \left(\frac{1}{8}\right) \cdot \frac{2}{k \cdot (k+1) \cdot (k+2)} \\ &= \Omega(\log n). \end{aligned}$$

Note that in the first inequality, we multiply by $\frac{k+1}{2}$ because for every choice of odd k, there are $\frac{k+1}{2}$ ways of removing an edge so that it gets split into two paths of even length.

The proof of the theorem then follows from the fact that $\mathbb{E}\left[\sum_{i=\frac{n}{3}+1}^{\frac{2n}{3}} AP_{e'_i}\right] = \Omega(n \log n)$. □

5 Conclusion and Open Problems

We consider the problem of maximum matching with recourse in the random edge-arrival setting. The goal is to compute the expected recourse. As mentioned in the introduction, there are strong lower bounds of $\Omega\left(n^2\right)$ in the adversarial edge-arrival model, even for the case of simple paths. For random edge-arrivals, we can do significantly better for special classes of graphs: we prove an upper bound of $O(n \log n)$ for the case of paths and $O\left(n \log^2 n\right)$ for the case of trees. This bound is tight up to $\log n$ factors, since we prove that for the case of paths, any algorithm must take expected total recourse of $\Omega(n \log n)$. But for general graphs, we show that random arrival is basically as hard as adversarial arrival: we give a family of graphs for which the expected recourse is at least $\Omega\left(\frac{n^2}{\log n}\right)$.

An interesting open question is the case of *bipartite* graphs: if edge-arrivals are random, can we prove a similar lower bound of $\Omega\left(\frac{n^2}{\text{polylog}(n)}\right)$ on the expected recourse? Our current lower-bound construction seems hard to extend to the bipartite case, as our proof crucially relies on the fact that after a constant fraction of the edges have arrived, if we focus only on the non-isolated vertices in the lower-bound graph G, then G contains a perfect matching with high probability. This allowed us to force the adversary to take an augmenting path between every new pair of non-isolated vertices. But in the case of bipartite graphs, it seems difficult to guarantee a perfect matching between the non-isolated vertices because the number of non-isolated vertices on the left might not be equal to the number on the right; in fact, they are likely to differ by a $\Theta\left(\sqrt{n}\right)$ factor.

A Omitted Proofs

In this section, we give a proof of Lemma 13. In order to do so, we state the following simple observation that relates the distribution of G_p with $G^{p \cdot m}$:

Claim 47. The distribution of G_p conditioned on $|E(G_p)| = p \cdot m$ is the same as the distribution of $G^{p \cdot m}$.

Proof. Let G_0 be any subgraph of G containing $p \cdot m$ edges. Then, we know that:

$$\begin{aligned}
\Pr\left(G_p = G_0 \mid |E(G_p)| = p \cdot m\right) &= \frac{\Pr\left(G_p = G_0\right)}{\Pr\left(|E(G_p)| = p \cdot m\right)} \\
&= \frac{(p)^{p \cdot m}\left(1-p\right)^{(1-p) \cdot m}}{\binom{m}{p \cdot m}(p)^{p \cdot m}\left(1-p\right)^{(1-p) \cdot m}} \\
&= \binom{m}{p \cdot m}^{-1} \\
&= \Pr\left(G^{p \cdot m} = G_0\right).
\end{aligned}$$

□

Now, we prove Lemma 13:

Proof (Lemma 13). We know that:

$$\begin{aligned}
\Pr\left(G_p \notin \mathcal{G}\right) &= \sum_{k=0}^{m} \Pr\left(G_p \notin \mathcal{G} \mid |E(G_p)| = k\right) \cdot \Pr\left(|E(G_p)| = k\right) \\
&\geq \Pr\left(G_p \notin \mathcal{G} \mid |E(G_p)| = p \cdot m\right) \cdot \Pr\left(|E(G_p)| = p \cdot m\right) \\
&\geq \Pr\left(G^{p \cdot m} \notin \mathcal{G}\right) \cdot \binom{m}{p \cdot m} \cdot (p)^{p \cdot m} \cdot (1-p)^{(1-p) \cdot m}. \\
&\text{(Follows from Claim 47)}
\end{aligned}$$

Using Stirling Approximation:

$$k! \approx \sqrt{2\pi k}\left(\frac{k}{e}\right)^{k}$$

We have:

$$\Pr\left(G_p \notin \mathcal{G}\right) \geq \Pr\left(G^{p \cdot m} \notin \mathcal{G}\right) \cdot \frac{1}{10\sqrt{m}}.$$

□

B Justifying the Assumption from Preliminaries

In this section, we justify the simplifying assumption from Section 2. For convenience, we restate the corresponding Lemma below.

Lemma 48. (Justification of Simplifying Assumption) Let G be some graph whose edges arrive in a random order. Say that we can prove that any only-augmenting algorithm that maintains a maximum matching in G has expected recourse $\Omega(T)$. Then any algorithm (possibly not only-augmenting) that maintains a maximum matching in G has expected recourse $\Omega(T)$.

Proof. Recall that the edges of G arrive in the order $e_1, \ldots, e_m$. Define $E_i = \{e_1, \ldots, e_i\}$, $E_0 = \emptyset$, and $G_i = (V, E_i)$. Define $\mu(G_i)$ be the maximum matching size in graph G_i, and let $\eta = \mu(G_m)$ be the maximum matching size in the whole graph G.

Now, we define the following set of *crucial* indices $C = \{j \mid \mu(G_j) = \mu(G_{j-1}) + 1\}$. Let $M_1, \cdots, M_m$ be the matchings maintained by the algorithm at every turn. Note that since the algorithm always maintains a maximum matching, we have $|M_j| = |M_{j-1}| + 1$ for all $j \in C$ and $|M_j| = |M_{j-1}|$ otherwise. We denote the indices inside C as $C = \{i_1, \ldots, i_\eta\}$, and define $i_0 = 0$.

Now, let ALG be the (possibly not only-augmenting) algorithm whose expected recourse we are trying to bound. Let $M_1, \ldots, M_m$ be the sequence of matchings maintained by ALG. Note that the total recourse of ALG is at least $\sum_{j=0}^{\eta-1} |M_{i_j} \oplus M_{i_{j+1}}|$. Our proof hinges on the following definition and claim.

Definition 49. Define a sequence $M^*_{i_0}, M^*_{i_1}, \cdots, M^*_{i_\eta}$ to be only-augmenting if $M^*_{i_0} = \emptyset$, each $M^*_{i_j}$ is a maximum matching in G_{i_j}, and each symmetric difference $M^*_{i_j} \oplus M^*_{i_{j+1}}$ consists of a single augmenting path.

Claim 50. Consider any sequence of matchings $M_{i_0}, M_{i_1}, \cdots, M_{i_\eta}$, where $M_{i_0} = \emptyset$ and M_{i_j} is a maximum matching in G_{i_j}. Then, there exists an only-augmenting sequence $M^*_{i_0}, \cdots, M^*_{i_\eta}$ such that

$$\sum_{j=0}^{\eta-1} |M_{i_j} \oplus M_{i_{j+1}}| \geq \sum_{j=0}^{\eta-1} |M^*_{i_j} \oplus M^*_{i_{j+1}}|$$

.

Before proving the claim, let us quickly observe that it completes the proof of the lemma. Let σ be some permutation of the edge set, and define $r(\sigma)$ to be the best possible recourse achievable on this permutation. Let $r^*(\sigma)$ be the best possible recourse of an only-augmenting algorithm. The assumption of the lemma states that $\mathbb{E}_\sigma[r^*(\sigma)] = \Omega(T)$. The claim above clearly implies that $r(\sigma) \geq r^*(\sigma)$, which implies that $\mathbb{E}_\sigma[r(\sigma)] = \Omega(T)$, thus completing the lemma.

Proof of Claim 50. We use a proof by induction. To make the induction step work, we actually prove a slightly stronger claim. Namely, that

$$\sum_{j=0}^{\eta-1} |M_{i_j} \oplus M_{i_{j+1}}| \geq \sum_{j=0}^{\eta-1} |M^*_{i_j} \oplus M^*_{i_{j+1}}| + |M_{i_\eta} \oplus M^*_{i_\eta}|$$

To prove the claim for $\eta = 1$, we set $M^*_{i_1} = M_{i_1}$. Now, say that the claim is true for some η, and consider $\eta + 1$. Let $M^*_{i_0}, \ldots, M^*_{i_\eta}$ be the sequence guaranteed by the induction hypothesis. We now need to find a suitable $M^*_{i_{\eta+1}}$. Consider the symmetric different $M^*_{i_\eta} \oplus M_{i_{\eta+1}}$. Because $M_{i_{\eta+1}}$ is a maximum matching and $|M^*_{i_\eta}| = \eta = |M_{i_{\eta+1}}| - 1$, it is clear that $M^*_{i_\eta} \oplus M_{i_{\eta+1}}$ consists of a single augmenting path, plus some disjoint alternating paths and cycles. Let a be the augmenting path and let ρ be the total length of all the other paths plus cycles. We set $M^*_{i_{\eta+1}}$

to be $M^*_{i_\eta} \oplus a$. Note that $|M^*_{i_{\eta+1}} \oplus M_{i_{\eta+1}}| = \rho$ and that any sequence of changes from $M^*_{i_\eta}$ to $M_{i_{\eta+1}}$ has size at least $|a| + \rho$. We thus have:

$$\begin{aligned}
\sum_{j=0}^{\eta} |M_{i_j} \oplus M_{i_{j+1}}| &\geq |M_{i_\eta} \oplus M_{i_{\eta+1}}| + |M^*_{i_\eta} \oplus M_{i_\eta}| + \sum_{j=0}^{\eta-1} |M^*_{i_j} \oplus M^*_{i_{j+1}}| \\
&\text{(follows due to the induction hypothesis)} \\
&\geq \rho + |a| + \sum_{j=0}^{\eta-1} |M^*_{i_j} \oplus M^*_{i_{j+1}}| \\
&\text{(follows from the sequence of changes from } M^*_{i_\eta} \text{ to } M_{i_{\eta+1}}) \\
&\geq |M^*_{i_{\eta+1}} \oplus M_{i_{\eta+1}}| + \sum_{j=0}^{\eta} |M^*_{i_j} \oplus M^*_{i_{j+1}}|
\end{aligned}$$

□

We have thus completed the proof of the lemma. □

References

[BHR19] Aaron Bernstein, Jacob Holm, and Eva Rotenberg. Online bipartite matching with amortized $o(\log^2 n)$ replacements. *Journal of the ACM (JACM)*, 66(5):1–23, 2019.

[BLSZ14] Bartlomiej Bosek, Dariusz Leniowski, Piotr Sankowski, and Anna Zych. Online bipartite matching in offline time. In *2014 IEEE 55th Annual Symposium on Foundations of Computer Science*, pages 384–393. IEEE, 2014.

[BLSZ15] Bartłomiej Bosek, Dariusz Leniowski, Piotr Sankowski, and Anna Zych. Shortest augmenting paths for online matchings on trees. In *International Workshop on Approximation and Online Algorithms*, pages 59–71. Springer, 2015.

[BLSZP18] Bartłomiej Bosek, Dariusz Leniowski, Piotr Sankowski, and Anna Zych-Pawlewicz. A tight bound for shortest augmenting paths on trees. In *Latin American Symposium on Theoretical Informatics*, pages 201–216. Springer, 2018.

[CDKL09] Kamalika Chaudhuri, Constantinos Daskalakis, Robert D Kleinberg, and Henry Lin. Online bipartite perfect matching with augmentations. In *IEEE INFOCOM 2009*, pages 1044–1052. IEEE, 2009.

[DP09] Devdatt P Dubhashi and Alessandro Panconesi. *Concentration of measure for the analysis of randomized algorithms.* Cambridge University Press, 2009.

[FHM+20] Alireza Farhadi, Mohammad Taghi Hajiaghayi, Tung Mai, Anup Rao, and Ryan A. Rossi. Approximate maximum matching in random streams. In *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms, SODA 2020, Salt Lake City, UT, USA, January 5-8, 2020*, pages 1773–1785. SIAM, 2020.

[GKKV95] Edward F Grove, Ming-Yang Kao, P Krishnan, and Jeffrey Scott Vitter. Online perfect matching and mobile computing. In *Workshop on Algorithms and Data Structures*, pages 194–205. Springer, 1995.

[GKMS19] Buddhima Gamlath, Sagar Kale, Slobodan Mitrovic, and Ola Svensson. Weighted matchings via unweighted augmentations. In *Proceedings of the 2019 ACM Symposium on Principles of Distributed Computing*, pages 491–500, 2019.

[GKS14] Anupam Gupta, Amit Kumar, and Cliff Stein. Maintaining assignments online: Matching, scheduling, and flows. In *Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms*, pages 468–479. SIAM, 2014.

[JLR00] Svante Janson, Tomasz Luczak, and Andrzej Rucinski. *Random graphs*. Wiley-Interscience series in discrete mathematics and optimization. Wiley, 2000.

[KKS14] Michael Kapralov, Sanjeev Khanna, and Madhu Sudan. Approximating matching size from random streams. In *Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms*, pages 734–751. SIAM, 2014.

[KMM12] Christian Konrad, Frédéric Magniez, and Claire Mathieu. Maximum matching in semi-streaming with few passes. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pages 231–242. Springer, 2012.

[KMT11] Chinmay Karande, Aranyak Mehta, and Pushkar Tripathi. Online bipartite matching with unknown distributions. In *Proceedings of the forty-third annual ACM symposium on Theory of computing*, pages 587–596, 2011.

[KVV90] Richard M Karp, Umesh V Vazirani, and Vijay V Vazirani. An optimal algorithm for on-line bipartite matching. In *Proceedings of the twenty-second annual ACM symposium on Theory of computing*, pages 352–358, 1990.

[MY11] Mohammad Mahdian and Qiqi Yan. Online bipartite matching with random arrivals: an approach based on strongly factor-revealing lps. In *Proceedings of the forty-third annual ACM symposium on Theory of computing*, pages 597–606, 2011.