Beating Two-Thirds For Random-Order Streaming Matching

Sepehr Assadi
Department of Computer Science, Rutgers University, Piscataway, NJ, USA

Soheil Behnezhad
Department of Computer Science, University of Maryland, College Park, MD, USA

Abstract

We study the maximum matching problem in the *random-order* semi-streaming setting. In this problem, the edges of an arbitrary n-vertex graph $G = (V, E)$ arrive in a stream one by one and in a random order. The goal is to have a single pass over the stream, use $O(n \cdot \text{polylog}(n))$ space, and output a large matching of G.

We prove that for an absolute constant $\varepsilon_0 > 0$, one can find a $(2/3 + \varepsilon_0)$-approximate maximum matching of G using $O(n \log n)$ space with high probability. This breaks the natural boundary of $2/3$ for this problem prevalent in the prior work and resolves an open problem of Bernstein [ICALP'20] on whether a $(2/3 + \Omega(1))$-approximation is achievable.

2012 ACM Subject Classification Theory of computation → Graph algorithms analysis; Theory of computation → Streaming, sublinear and near linear time algorithms

Keywords and phrases Maximum Matching, Streaming, Random-Order Streaming

Digital Object Identifier 10.4230/LIPIcs.ICALP.2021.19

Category Track A: Algorithms, Complexity and Games

Related Version *Full Version*: `https://arxiv.org/abs/2102.07011`

Funding *Sepehr Assadi*: Research supported in part by the NSF CAREER award CCF-2047061 and a gift from Google Research.
Soheil Behnezhad: Research supported by Google Ph.D. Fellowship.

Acknowledgements We thank Aaron Bernstein for helpful conversations on the random-order streaming matching problem and several insightful comments that helped us in improving the presentation of the paper. We also thank the anonymous ICALP reviewers for their valuable comments.

1 Introduction

A matching in a graph $G = (V, E)$ is any collection of vertex-disjoint edges and in the maximum matching problem, we are interested in finding a matching of largest size in G. This problem has been a cornerstone of algorithmic research and its study has led to numerous breakthrough results in theoretical computer science. In this paper, we study the maximum matching problem in the *semi-streaming* model of computation [9] defined as follows.

▶ **Definition 1.** *Given a graph $G = (V, E)$ with n vertices $V = \{1, \ldots, n\}$ and m edges in E presented in a stream $S = \langle e_1, \ldots, e_m \rangle$, a semi-streaming algorithm makes a single pass over the stream of edges S and uses $O(n \cdot \text{polylog}(n))$ space, measured in words of size $\Theta(\log n)$ bits, and at the end outputs an approximate maximum matching of G.*

The greedy algorithm for maximal matching gives a simple $1/2$-approximation algorithm to this problem in $O(n)$ space. When the stream of edges is adversarially ordered, this is simply the best result known for this problem, while it is also known that a better than

$\frac{1}{1+\ln 2} \sim 0.59$-approximation is not possible [14] (see also [11,13]). Closing the gap between these upper and lower bounds is among the most longstanding open problems in the graph streaming literature.

Going beyond this "doubly worst case" scenario, namely, an adversarially-chosen graph and an adversarially-ordered stream, there has been an extensive interest in recent years in studying this problem on **random order streams**. This line of work was pioneered in [16] who showed that the $1/2$-approximation of greedy can be broken in this case and obtained an algorithm with approximation ratio $(1/2 + 0.003)$ for this problem. Since [16], there has been two main lines of attack on this problem. Firstly, [8,10,15] followed up on the approach of [16] and improved the approximation ratio all the way to $6/11$ [8]. In parallel, [1] built on the sparsification approach of [6,7] in dynamic graphs to achieve an (almost) $2/3$-approximation but at the cost of $\widetilde{O}(n^{1.5})$ space, which is no longer semi-streaming. A beautiful work of [5] then obtained a semi-streaming (almost) $2/3$-approximation by showing how a generalization of the sparsification approach in [1] can be found in $\widetilde{O}(n)$ space.

The $2/3$-approximation ratio of the algorithm of [5] is the best possible among all prior techniques for this problem: the first line of attack in [8, 10, 15, 16] is based on finding length-3 augmenting paths and even finding *all* these paths does not lead to a better-than-$2/3$-approximation[1]. The second line in [1, 5] is based on finding an edge-degree constrained subgraph (EDCS) which hits the same exact barrier as there are graphs whose EDCS does not provide a better than $2/3$-approximation (see [6]). Finally, even for an algorithmically easier variant of this problem, the one-way communication problem, which roughly corresponds to only measuring the space of the algorithm when crossing the midpoint of the stream, the best known approximation ratio is still $2/3$ which is known to be tight for adversarial orders/partitions [11].

Given this state-of-affairs, the $2/3$-approximation ratio for random-order streaming matching has emerged as natural barrier [5, 15]. In particular, [5] posed obtaining a $(2/3 + \Omega(1))$-approximation to this problem as an important open question. We resolve this question in the affirmative in our work.

1.1 Our Contributions

Our main result is a semi-streaming algorithm for maximum matching in random-order streams with approximation ratio strictly-better-than-$2/3$.

▶ **Theorem 2** (Main Result). *Let G be an n-vertex graph whose edges arrive in a random-order stream. For an absolute constant $\varepsilon_0 > 0$, there is a single-pass streaming algorithm that obtains a $(\frac{2}{3} + \varepsilon_0)$-approximate maximum matching of G using $O(n \log n)$ space w.h.p.*

Theorem 2 breaks the $2/3$-barrier of all prior work in [1, 5, 8, 10, 15, 16]. Moreover, even though the improvement over $2/3$ is minuscule in this theorem (while we did not optimize for constants, the bound on ε_0 is only $\sim 10^{-14}$ at this point), it still proves that $(2/3)$-approximation is not the "right" answer to this problem. This is in contrast to some other problems of similar flavor such as one-way communication complexity of matching (on adversarial partitions) [3,11] or the fault-tolerant matching problem [3] which are both solved using similar techniques (see the unifying framework of [3] based on EDCS) and for both $2/3$-approximation is provably best possible.

[1] The work of [8] also considers length-5 augmenting paths. However, these paths are used *instead of* length-3 paths "missed" by the algorithm *not in addition to* length-3 paths and thus the same shortcoming persists.

Beyond (2/3)-approximation. Breaking this 2/3-barrier naturally raises the question on what is the right bound on the approximation ratio of random-order streaming matching. In particular, is $(1-\varepsilon)$-approximation possible? We make progress toward settling this question as well by showing that no "completely" space-efficient algorithm exists for this latter problem: there is provably no semi-streaming algorithm for the matching problem even on bipartite graphs that can achieve a $(1-\varepsilon)$-approximation in $O(2^{(1/\varepsilon)^{0.99}} \cdot n \cdot \text{polylog}n)$ space; in other words, if one hopes for achieving a $(1-\varepsilon)$-approximation, an exponential dependence on $(1/\varepsilon)$ in the space is unavoidable. As the main focus of our work is on the algorithm in Theorem 2, we provide the details of the lower bound only in the full version [2].

1.2 Overview of Techniques

Prior work. As stated earlier, there has been two main lines of attacks on the streaming matching problem in random-order streams. The first approach aims to find a *large* matching of the graph G early on in the stream, and then spends the rest of the stream *augmenting* this matching. For instance, [16] showed that in order for the greedy algorithm to fail to find a better-than-1/2-approximation, the algorithm should necessarily pick many "wrong" edges early on in the stream. As such, in instances where greedy is not beating the 1/2-approximation itself, we already have an almost 1/2-approximation by the *middle* of the stream, and we can thus focus on augmenting this matching in the remainder half to beat 1/2-approximation. The work of [15] then improved this result further by showing that a modified greedy algorithm, when unsuccessful in obtaining a large matching itself, finds an almost 1/2-approximation when only $o(1)$-fraction of the stream has passed (as opposed to middle), which gives us more room for augmentation. Finally, [8] built on this approach and further improved the augmentation phase.

The second approach to this problem was based on obtaining an EDCS, a subgraph defined by [6, 7] and studied further in [3], that acts as a "matching sparsifier". On a high level, an EDCS is a sparse subgraph satisfying the following two constraints: (i) edge-degree of edges in the EDCS cannot be "high", while (ii) edge-degree of missing edges cannot be "low". These constraints ensure that an EDCS always contains an almost 2/3-approximate matching of the graph and has additional robustness properties. For instance, [1] proved that union of several EDCS computed on different parts of a random stream, is itself an EDCS for the entire stream. This allowed them to compute an EDCS of the input in $\widetilde{O}(n^{1.5})$ space and directly obtain their almost 2/3-approximation. Finally, [5] gave an elegant proof that weakening the requirement of EDCS allows one to still preserve the almost 2/3-approximation but now recover this subgraph in only $O(n \log n)$ space. More specifically, the algorithm of [5] first finds a subgraph only satisfying property (i) of the EDCS in the first $o(m)$ edges of the stream, and then picks *all* (potentially) necessary edges for satisfying property (ii) in the remainder; the proof then shows that this set of potentially necessary edges is of size only $O(n \log n)$.

Our work. Our approach can be seen as a natural combination of these two mostly disjoint lines of work. The first part comes from a better understanding of EDCS. We present a rough characterization of when an EDCS cannot beat the 2/3-approximation, which shows that in these instances, we can effectively ignore the second constraint of EDCS. As a result, we obtain that the only way for the algorithm of [5] to fail to achieve a better-than-2/3-approximation, is if it already picks an almost 2/3-approximation in the first $o(m)$ edges. Note that this is conceptually similar to the first line of work on random-order streaming matching, but the techniques are entirely disjoint. In particular, our proof is a deterministic property of EDCS not a randomized property of a greedy algorithm on a particular ordering.

We are now in the familiar territory of having a large matching very early on in the stream, and we can spend the remainder of the stream augmenting it. The main difference however is that starting from an almost $2/3$-approximation matching, there is essentially no length-3 paths for us to augment and we instead need to handle length-5 augmenting paths. The key challenge is to find the middle edge of these length-5 augmenting paths. Indeed, we note that the $2/3$-approximation lower bound of [11] for *adversarial* order streams gives away a $2/3$-approximate matching early on for free, yet it is provably impossible to augment it in the remainder of the stream using $\widetilde{O}(n)$ space. To get around this, we crucially use the random arrival assumption again. Particularly, we regard any length-5 augmenting path whose middle edge arrives after its two endpoint edges as a "discoverable" path and then find a constant fraction of such paths. Since the edges arrive in a random order, a constant fraction of length-5 augmenting will be discoverable and thus we are able to beat $2/3$-approximation in our setting.

2 Notation and Preliminaries

General notation. For a graph $G = (V, E)$ and $v \in V$, we use $\deg_G(v)$ to denote the degree of v in G and $N_G(v)$ to denote the neighborset of v (when clear from the context, we may drop the subscript G). For any edge $e = (u, v) \in E$, we define the edge-degree of e in G as $\deg(u) + \deg(v)$. We use $\mu(G)$ to denote the *size* (i.e., the number of edges) of the maximum matching in G.

For integer $k \geq 1$ and $p \in [0, 1]$, we use $\mathcal{B}(k, p)$ to denote the *binomial distribution* with parameters k and p. That is, $\mathcal{B}(k, p)$ is the discrete probability distribution of the number of successful experiments out of k independent experiments each with probability p of success.

Random-order streams. We consider the random-order streaming setting where the edges of G arrive one by one in an order chosen uniformly at random from all possible orderings. Let e_i be the i-th edge that arrives in the stream. For any two parameters a, b satisfying $1 \leq a < b \leq m$ we use $G[a, b]$ to denote the subgraph of G on vertex-set V and edge-set $\{e_a, \ldots, e_b\}$. We may also use $G_{<a}$ and $G_{\geq a}$ respectively for $G[1, a-1]$ and $G[a, m]$.

For the input graph G defined by the stream, we can assume w.l.o.g. that $\mu(G) \geq c \log n$ for any desirably large constant c. The reason is that any graph can be easily shown to have at most $2n \cdot \mu(G)$ edges and if $\mu(G) = O(\log n)$ then we can store the whole input in the memory and report an optimal solution using $O(n \log n)$ space. We further assume throughout the paper that the number of edges m is known by the algorithm in advance. This is a common assumption in the literature and can be removed via standard techniques by guessing m in geometrically increasing values at the expense of multiplying the space by an $O(\log n)$ factor.

2.1 Preliminaries

Hall's theorem. We use the following standard extension of the Hall's marriage theorem for characterizing maximum matching size in bipartite graphs.

▶ **Fact 3** (Extended Hall's Theorem [12]). *Let $G = (L, R, E)$ be a bipartite graph with $|L| = |R| = n$. Then, $\max\Big(|A| - |N(A)|\Big) = n - \mu(G)$, where A ranges over L or R, separately. We refer to such set A as a* ***witness set****.*

Fact 3 follows from Tutte-Berge formula for matching size in general graphs [4, 17] or a simple extension of the proof of Hall's marriage theorem itself.[2]

Alternating and augmenting paths. Given a matching M, an *alternating path* P for M is a path whose edges alternatively belong to M and do not belong to M. An *augmenting path* for M is an alternative path that starts and ends with edges that do not belong to M. Given an augmenting path P for M, we use notation $M \oplus P := (M \setminus P) \cup (P \setminus M)$ to denote the matching obtained by flipping the containment of edges of P in M. Given two matchings M and M', their *symmetric difference* $M \Delta M'$ is a graph including only the edges that belong to exactly one of M and M'.

2.2 Bernstein's Algorithm

We briefly review the parameters and guarantees of the algorithm of Bernstein [5] that we need. In the following, we slightly increase the constants in the parameters which is needed for our results.

▶ **Definition 4 (Parameters).** *For some small $\varepsilon \in (0, \frac{1}{2})$ to be determined later, let*

$$\lambda := \frac{\varepsilon}{128}, \qquad \beta_+ := 64 \cdot \lambda^{-2} \log(1/\lambda), \qquad \beta_- = (1-\lambda) \cdot \beta_+.$$

A high level overview of the algorithm of [5] is as follows:

Algorithm 1 Bernstein's Algorithm [5].

The algorithm of [5] proceeds in two phases as follows:

- Phase I terminates within the first εm edges of the stream. At the end of Phase I, the algorithm constructs a subgraph $H \subseteq G_{<\varepsilon m}$ (using the algorithm of Lemma 5) that in particular guarantees that for all $(u, v) \in H$, $\deg_H(u) + \deg_H(v) \leq \beta_+$. Also let U be the set of *all* edges in $G_{\geq \varepsilon m}$ such that $\deg_H(u) + \deg_H(v) < \beta_-$.
- In Phase II, the algorithm simply stores U in the memory and at the end of the stream returns a maximum matching of $H \cup U$.

The following lemma is all we need from [5] in our paper.

▶ **Lemma 5** (See Lemma 4.1 of [5]). *There is a way of constructing the subgraph H of $G_{<\varepsilon m}$ such that with probability at least $1 - n^{-3}$, $|H \cup U| = O(n \log(n) \cdot \text{poly}(1/\varepsilon))$.*

3 Finding an Almost (2/3)-Approximation Early On

We start by characterizing the tight instances of the algorithm of [5] (Algorithm 1). Roughly speaking, we show that the only way for Algorithm 1 to end up with a (2/3)-approximation is if in its Phase I it computes a subgraph H that already has an almost (2/3)-approximate matching. This will then be used by our algorithm in the next section to obtain a strictly better-than-(2/3)-approximation by augmenting this already-large matching. We start by presenting and proving this result for bipartite graphs which is the main part of the proof; we then extend the result to general graphs (with no considerable loss of parameters for our purpose) using the probabilistic method approach of [3] for the original EDCS.

[2] Simply add $n - \mu(G)$ vertices to each side of the graph and connect them to all the original vertices; then apply original's Hall's theorem for perfect matching to this graph as this graph now has one.

3.1 Bipartite Graphs

In this section we prove the following structural result:

▶ **Theorem 6.** *Let $\lambda \in (0, 1/2)$ and $\beta_- \leq \beta_+$ be such that $\beta_+ \geq \frac{10}{\lambda}$ and $\beta_- \geq (1-\lambda)\beta_+$. Suppose $G = (L, R, E)$ is any bipartite graph and:*

(i) *H is a subgraph of G where for all $(u, v) \in H$: $\deg_H(u) + \deg_H(v) \leq \beta_+$; and*

(ii) *U is the set of all edges (u, v) in $G \setminus H$ such that $\deg_H(u) + \deg_H(v) < \beta_-$.*

Then, for any parameter $\delta \in (0, 1)$, either:

$$\mu(H) \geq (1-4\lambda) \cdot \left(\frac{2}{3} - \delta\right) \cdot \mu(G) \quad \text{or} \quad \mu(H \cup U) \geq (1-2\lambda) \cdot \left(\frac{2}{3} + \frac{\delta^2}{18}\right) \cdot \mu(G).$$

Let us define the following (see Figure 1 for an illustration):

- Let M^* be a maximum matching of G and define $M^*_U := M^* \cap U$ and $M^*_{\bar{U}} := M^* \setminus U$.
- A is a Hall's theorem witness set in $H \cup M^*_U$ (as in Fact 3) and $B := N_{H \cup M^*_U}(A)$. Without loss of generality we assume $A \subseteq L$ and define $\bar{A} := L \setminus A$ and $\bar{B} := R \setminus B$.

We start with the following simple claim that follows easily from Fact 3. See the full version of the paper for details [2].

▷ Claim 7. For the witness set A:

(i) $|\bar{A}| + |B| \leq \mu(H \cup U)$.

(ii) There is a matching $\bar{M} \subseteq M^*_{\bar{U}}$ between A and $\bar{B}$ in G with size $|\bar{M}| = \mu(G) - \mu(H \cup M^*_U)$.

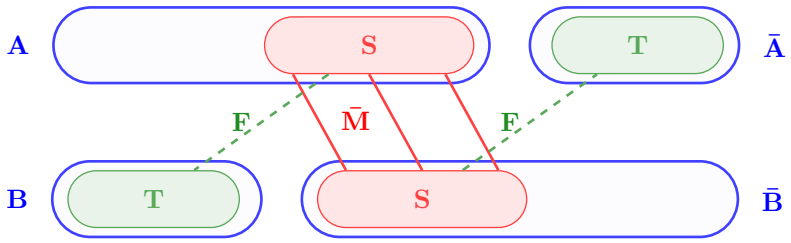

Figure 1 An illustration of the Hall's witness set and our notation in the proof of Theorem 6. Note that there are no edges between A and $\bar{B}$ in $H \cup M^*_U$, and matching $\bar{M}$ belongs entirely to $M^*_{\bar{U}}$.

Consider any edge $(u, v) \in \bar{M}$ defined in Claim 7. As $\bar{M} \subseteq M^*_{\bar{U}}$, by property (ii) of the statement of Theorem 6, we have, $\deg_H(u) + \deg_H(v) \geq \beta_-$. We arbitrarily remove the edges on u and v until the above inequality becomes tight for every edge. We let F be the remaining edges. Note that any edge in F is incident on exactly one vertex of $\bar{M}$ as there are no edges in $H \cup M^*_U$ between the endpoints of $\bar{M}$. We record these properties as follows:

$$\forall (u, v) \in \bar{M} : \ \deg_F(u) + \deg_F(v) = \beta_- \quad \text{and} \quad |F| = |\bar{M}| \cdot \beta_-. \tag{1}$$

We refer the reader to the full version of the paper [2] for illustrative examples that highlight the ideas for proving Theorem 6. Here for space constraints, we only provide the formal proof.

In Lemma 8, we prove a lower bound on $\mu(H)$. This lemma can then be used as follows: if the degrees of most edges in $\bar{M}$ are "balanced", i.e., both endpoints have degree $\approx \beta_-/2$, then $\mu(H)$ will already be of size $2 \cdot |\bar{M}|$ which is sufficient for the first condition of Theorem 6.

▶ **Lemma 8** (matching of H is large). *We have* $\mu(H) \geq \frac{\beta_-}{1+4\lambda} \cdot \sum_{(u,v)\in \bar{M}} \frac{1}{\max\{\deg_F(u)\,,\,\deg_F(v)\}}$.

Proof. For every edge $(u,v) \in \bar{M}$, define $F(u,v)$ as set of edges in F that are incident on u or v. We define the following fractional matching $x \in \mathbb{R}^F$ on edges of F:

- for any edge $e \in F(u,v)$: set $x_e := \frac{1}{1+4\lambda} \cdot \frac{1}{\max\{\deg_F(u)\,,\,\deg_F(v)\}}$.

Let us now prove this is indeed a valid fractional matching. For any vertex w matched by $\bar{M}$,

$$x_w := \sum_{e \ni w} x_e \leq \deg_F(w) \cdot \frac{1}{1+4\lambda} \cdot \frac{1}{\deg_F(w)} < 1,$$

thus satisfying the fractional matching constraint.

Now fix a vertex w not matched by $\bar{M}$. Let $u_1, \ldots, u_{\deg_F(w)}$ denote the neighbors of w in F. By definition, all these vertices are matched by $\bar{M}$. Let $v_1, \ldots, v_{\deg_F(w)}$ be the matched pairs of these vertices. We need the following simple claim proved in the full version.

▷ Claim 9. For every $i \in [\deg_F(w)]$, $\deg_F(w) \leq (1+4\lambda) \cdot \max\{\deg_F(u_i)\,,\,\deg_F(v_i)\}$.

To finalize Lemma 8, for any vertex w not matched by $\bar{M}$, we have,

$$x_w := \sum_{e=(w,u_i)} x_e = \sum_{u_i} \frac{1}{1+4\lambda} \cdot \frac{1}{\max\{\deg_F(u_i)\,,\,\deg_F(v_i)\}} \underset{\text{Claim 9}}{\leq} \sum_{u_i} \frac{1}{\deg_F(w)} = 1,$$

thus satisfying the fractional matching constraint. This implies that x is a valid fractional matching. Finally, the value of this fractional matching is:

$$\begin{aligned}\sum_{e\in F} x_e &= \sum_{(u,v)\in N} \sum_{e\in F(u,v)} x_e = \sum_{(u,v)\in N} \frac{\deg_F(u) + \deg_F(v)}{(1+4\lambda)\cdot \max\{\deg_F(u)\,,\,\deg_F(v)\}} \\ &= \frac{\beta_-}{1+4\lambda} \cdot \sum_{(u,v)\in N} \frac{1}{\max\{\deg_F(u)\,,\,\deg_F(v)\}},\end{aligned}$$

where the last equation is by Eq (1). As the integrality gap of the matching polytope on bipartite graphs is one, we obtain the desired lower bound on $\mu(H)$. ◀

We now prove that if on the other hand most edges of $\bar{M}$ are "unbalanced", then $\mu(H \cup U)$ should be sufficiently large. To continue, we need a quick definition. Let S denote the endpoints of the matching $\bar{M}$ and T be the neighborset of these vertices in F. Recall that by Eq (1), S and T are disjoint (see Figure 1).

▶ **Lemma 10** (matching of $\mu(H \cup U)$ is large). *We have* $\mu(H \cup U) \geq \frac{|\bar{M}|^2 \cdot {\beta_-}^2}{|\bar{M}| \cdot \beta_- \cdot \beta_+ - \sum_{s\in S} (\deg_F(s))^2}$.

Proof. Since $F \subseteq H$, by property (i) of Theorem 6, we have that

$$|F| \cdot \beta_+ \geq \sum_{(u,v)\in F} \deg_F(u) + \deg_F(v) = \sum_{s\in S} (\deg_F(s))^2 + \sum_{t\in T} (\deg_F(t))^2. \tag{2}$$

We can lower bound the second term of the RHS as follows. Recall that sum of quadratics is minimized over all-equal terms. As $\sum_{t\in T} \deg_F(t) = |F|$, this implies that,

$$\sum_{t\in T} (\deg_F(t))^2 \geq \sum_{t\in T} \left(\frac{|F|}{|T|}\right)^2 = |T| \cdot \left(\frac{|F|}{|T|}\right)^2 = \frac{|F|^2}{|T|}.$$

By plugging in this bound in Eq (2) and moving the terms around, we have that

$$|T| \geq \frac{|F|^2}{|F| \cdot \beta_+ - \sum_s (\deg_F(s))^2} = \frac{|\bar{M}|^2 \cdot {\beta_-}^2}{|\bar{M}| \cdot \beta_- \cdot \beta_+ - \sum_s (\deg_F(s))^2}. \qquad \text{(as } |F| = |\bar{M}| \cdot \beta_- \text{ by Eq (1))}$$

Finally, $T \subseteq \bar{A} \cup B$ (as there are no edges between A and $\bar{B}$) and thus by Claim 7, $|T| \leq \mu(H \cup U)$ which finalizes the proof. ◀

Lemma 10 can be used as follows: when degree of most edges in $\bar{M}$ are "balanced", the quantity $\sum_s (\deg_F(s))^2$ will be close to $|\bar{M}| \cdot (\beta_-)^2/2$ which implies that $\mu(H \cup U)$ will be almost $2 \cdot |\bar{M}|$; however, when degrees of edges in $\bar{M}$ are "unbalanced", the quantity $\sum_s (\deg_F(s))^2$ *cannot* decrease all the way to $|\bar{M}| \cdot (\beta_-)^2/2$ and thus we can get a higher lower bound on the value of $\mu(H \cup U)$ which breaks the (2/3)-approximation.

To finalize the proof of Theorem 6, we need the following claim for lower bounding $\sum_{s \in S} (\deg_F(s))^2$ in the RHS of Lemma 10, in the cases where RHS of Lemma 8 is small.

▷ Claim 11. Suppose $\sum_{(u,v) \in \bar{M}} \frac{\beta_-}{\max\{\deg_F(u), \deg_F(v)\}} = (2-\gamma) \cdot |\bar{M}|$ for some $\gamma \in [0,1)$; then $\sum_s (\deg_F(s))^2 \geq |\bar{M}| \cdot \left(\frac{(2+\gamma^2-2\gamma) \cdot {\beta_-}^2}{4+\gamma^2-4\gamma}\right)$.

The proof of Claim 11 is given in the full version [2]. We are now ready to prove Theorem 6.

Proof of Theorem 6. Let us pick $\gamma \in [0,1)$ such that $\sum_{(u,v) \in \bar{M}} \frac{\beta_-}{\max\{\deg_F(u), \deg_F(v)\}} = (2-\gamma) \cdot |\bar{M}|$ (as the max-term is at least $\beta_-/2$, such a γ always exist). By plugging in the bound of Claim 11 in Lemma 10, we have that,

$$\begin{aligned}
\mu(H \cup U) &\geq \frac{|\bar{M}|^2 \cdot {\beta_-}^2}{|\bar{M}| \cdot \beta_- \cdot \beta_+ - |\bar{M}| \cdot \left(\frac{(2+\gamma^2-2\gamma) \cdot {\beta_-}^2}{4+\gamma^2-4\gamma}\right)} \\
&\geq (1-2\lambda) \cdot |\bar{M}| \cdot \frac{1}{1 - \left(\frac{(2+\gamma^2-2\gamma)}{4+\gamma^2-4\gamma}\right)} && \text{(as } \beta_- \geq (1-\lambda)\beta_+\text{)} \\
&= (1-2\lambda) \cdot |\bar{M}| \cdot \frac{4+\gamma^2-4\gamma}{2-2\gamma} = (1-2\lambda) \cdot |\bar{M}| \cdot (2 + \frac{\gamma^2}{2-2\gamma}).
\end{aligned}$$

Considering $|\bar{M}| \geq \mu(G) - \mu(H \cup U)$ by Claim 7, we obtain that

$$\mu(H \cup U) \geq (1-2\lambda) \cdot \mu(G) \cdot \left(\frac{2}{3} + \frac{\gamma^2}{18 - 18\gamma + 3\gamma^2}\right) \geq (1-2\lambda) \cdot \mu(G) \cdot \left(\frac{2}{3} + \frac{\gamma^2}{18}\right).$$

Now if for the parameter δ in Theorem 6, we already have $\gamma \geq \delta$, we will obtain the second condition. Further, without loss of generality, we can assume that $|\bar{M}| \geq (\frac{1}{3} - \frac{\delta}{3}) \cdot \mu(G)$ as otherwise $\mu(H \cup M^*_U) \geq (\frac{2}{3} + \delta) \cdot \mu(G)$ by Claim 7 which is stronger than the second condition of Theorem 6.

Suppose $\gamma < \delta$ and $|\bar{M}| \geq (\frac{1}{3} - \frac{\delta}{3}) \cdot \mu(G)$ then. In this case, by the definition of γ and Lemma 8,

$$\begin{aligned}
\mu(H) &\geq \frac{1}{1+4\lambda} \cdot (2-\gamma) \cdot |\bar{M}| \geq \frac{1}{1+4\lambda} \cdot (2-\delta) \cdot \left(\frac{1}{3} - \frac{\delta}{3}\right) \cdot \mu(G) \\
&\geq (1-4\lambda) \cdot \left(\frac{2}{3} - \delta\right) \cdot \mu(G),
\end{aligned}$$

thus satisfying the first condition. This concludes the proof. ◀

We can also extend the guarantee of Theorem 6 to general (non-bipartite) graphs following the probabilistic method technique of [3] (see the full version [2] for the exact guarantee) without incurring any loss to the guarantees up to constants.

4 An Improved Algorithm via Augmentation

In this section, we show that the maximum matching of the subgraph H constructed in the early part of the stream of Algorithm 1 can be augmented well via the remaining edges. Combined with our guarantee of Section 3, we complete in this section the proof of Theorem 2. Namely, we show that for some parameter $\varepsilon_0 > 0$, there is a single-pass random-order streaming algorithm (formalized as Algorithm 2) that obtains a $(\frac{2}{3} + \varepsilon_0)$-approximate maximum matching of G using $O(n \log n)$ space with high probability of $1 - 1/\text{poly}(n)$.

4.1 The Algorithm

Our starting point is Algorithm 1. Recall that this algorithm stores two subgraphs H and U of G of size $O(n \log n)$. Subgraph H is constructed early on, after merely observing εm edges of the stream. In addition to H and U, here we store an additional subset of edges that we use to augment a matching of H with. Particularly, let M_H be an arbitrary maximum matching of H. Having matching M_H early on, in our algorithm we augment M_H using the edges that arrive in the rest of the stream (i.e., Phase II) in parallel to storing U. The augmenting paths that we find may be of size up to *five*. This is crucial since we may not have enough augmenting paths of length smaller than five to go beyond (2/3)-approximation. Now by plugging our bound of Section 3, it can be shown that either $H \cup U$ includes our desired approximation of strictly better that 2/3, or M_H is almost a (2/3)-approximate matching which coupled with the augmenting paths that we find for it in Phase II leads to our better-than-(2/3)-approximation.

To find these augmenting paths, we divide the $(1-\varepsilon)m$ edges of Phase II into Phase II.A and Phase II.B. To do this, we first draw a random variable $\tau \sim \mathcal{B}((1-\varepsilon)m, \gamma)$. Phase II.A will then proceed on the edges that arrive up to the τ-th edge of Phase II and Phase II.B proceeds on the rest of the edges. Drawing random variable τ (instead of having a fixed threshold) is particularly useful in the analysis: Conditioned on the edges that are to arrive in Phase II (but not their ordering), each edge now belongs to Phase II.A *independently* with probability γ and to Phase II.B otherwise. Note that with a fixed threshold, we do not get this independence.

For Phase II.A, let us define G_H to be the subgraph of G whose edges arrive in Phase II.A and have exactly one endpoint matched by M_H. Note that G_H is bipartite (even though G may not be) with one partition corresponding to vertices $V(M_H)$ and another to $V \setminus V(M_H)$. In Phase II.A, we only consider the edges of G_H and greedily construct a maximal $(2, b)$-matching T of G_H (for some constant $b \geq 2$). It is the vertices in part $V(M_H)$ of G_H that have maximum degree 2 in T and those in the other partition can have degree up to b. In our analysis, we show that the edges of T can be used as the two endpoint edges of many augmenting paths of length three or five for M_H (see Figure 2).

In Phase II.B, we first let $M \leftarrow M_H$ and upon arrival of each edge e, we iteratively augment M via length-up-to-five augmenting paths using the edges in $T \cup \{e\}$ until no such path is left. In our analysis, we use the edges of Phase II.B either as the middle edge of length-five augmenting paths or as the single edge of the length-one augmenting paths the algorithm may find (see Figure 2).

At the end of the stream, we return a maximum matching of $M \cup H \cup U$. The algorithm outlined above is formalized as Algorithm 2.

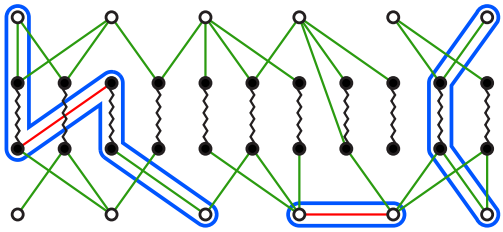

Figure 2 An example of an execution of Algorithm 2. Here the black zig-zagged edges denote M_H which is fixed by the end of Phase I and we would like to augment it. The black nodes are those matched by M_H and the white ones are those left unmatched by M_H. The edges between white and black nodes (colored green) are the edges in T. Each black node has at most two edges in T and the green nodes can have up to b. The red edges are those that arrive in Phase II.B. Three augmenting paths of length one, three, and five that are discoverable by the algorithm are highlighted by blue.

Algorithm 2 Our final random-order streaming matching algorithm with approximation ratio strictly-better-than-$2/3$.

Parameters: $\gamma = 2/3$, $b = 500$, and a sufficiently small constant $\varepsilon < 0.01$ to be fixed later.

(1) In Phase I of the algorithm, which consists of the first εm edges of the stream, we construct a subgraph H of G as in Phase I of Algorithm 1. At the end of Phase I, we fix an arbitrary maximum matching M_H of H.

(2) In Phase II, which includes all the edges that arrive after Phase II, we store subgraph U using Phase II of Algorithm 1. In addition, we store another subset of edges that we use to augment M_H. These edges are constructed in two sub-phases Phase II.A and Phase II.B.

(3) Draw random variable τ from the Binomial distribution $\mathcal{B}((1-\varepsilon)m, \gamma)$. Note that this can be done in $O(m)$ time and $O(1)$ space as we only need to count the successes.

(4) Phase II.A starts after Phase I and ends upon arrival of the τ'th edge of Phase II.

- **a.** Let $G_H(V_H, U_H, E_H)$ be a bipartite subgraph of G where $V_H := V(M_H)$ is the set of vertices matched in M_H, $U_H := V \setminus V(M_H)$ is the set of vertices left unmatched in M_H, and E_H is the edges of G between V_H and U_H that arrive in Phase II.A.
- **b.** We initialize $T \leftarrow \emptyset$ and upon arrival of an edge $e = (u, v)$ of G_H with $u \in U_H$ and $v \in V_H$, if $\deg_T(v) < 2$ and $\deg_T(u) < b$ we add e to T. That is, T is a maximal $(2, b)$-matching of G_H which requires $O(nb)$ space to store.

(5) Phase II.B starts after Phase II.A and continues to the end of the stream:

- **a.** $M \leftarrow M_H$. Upon arrival of each edge e in Phase II.B, we iteratively take an arbitrary augmenting path P for M of length up to five using the edges in $M \cup T \cup \{e\}$ and let $M \leftarrow M \oplus P$. We repeat this process until no more augmenting paths of length up to five exist in $M \cup T \cup \{e\}$; we then continue to the next edge of the stream in Phase II.B.

(6) Finally, we return a maximum matching of $M \cup H \cup U$.

Analysis of Algorithm 2

It is straightforward to verify that Algorithm 2 uses $O(|H| + |U| + nb)$ space which by Lemma 5 is $O(n \log n)$ w.h.p. Here we analyze the approximation ratio of the algorithm.

Let $M^\star$ be an arbitrary maximum matching of $G_{\geq \varepsilon m}$. Fixing an arbitrary maximum matching of G, each of its edges appears in $G_{\geq \varepsilon m}$ with probability $(1-\varepsilon)$, thus $\mathbf{E}|M^\star| \geq (1-\varepsilon)\mu(G)$. Now so long as $\mu(G) \geq 20\log(n)\varepsilon^{-2}$ and $\varepsilon < 1/2$ (which we can assume to hold as discussed in Section 2.1), we can prove a high probability lower bound on the size of $M^\star$ via a Chernoff bound on negatively associated random variables. See, e.g., [5, Lemma 2.2] for the proof of the following:

▶ **Observation 12.** *If $\mu(G) \geq 20\log(n)\varepsilon^{-2}$ and $\varepsilon < 1/2$, $\mathbf{Pr}[|M^\star| \geq (1-2\varepsilon)\mu(G)] \geq 1 - n^{-5}$.*

From now on, we condition on $G_{<\varepsilon m}$ which fixes subgraph H and matching $M^\star$. We only assume that $G_{<\varepsilon m}$ is chosen such that the high probability event of Observation 12 holds.

▶ **Assumption 13.** $|M^\star| \geq (1-2\varepsilon)\mu(G)$.

Other than Assumption 13, we do not need any other assumption on how $G_{<\varepsilon m}$ is chosen for the rest of the analysis of the approximation ratio.[3] By conditioning on the outcome of Phase I, the only randomization that will be left, is the order with which the edges of $G_{\geq \varepsilon m}$ arrive in the stream. For brevity, we do not explicitly write the conditioning on $G_{<\varepsilon m}$ for the rest of the section, but it should be noted that **all random statements are conditioned on the outcome of Phase I**.

Let $\mathcal{P}$ be the set of all augmenting paths of M_H in $S := M^\star \Delta M_H$ with length at most five. Note that since we regard H (and thus M_H) as given, the set $\mathcal{P}$ is deterministic (as it only depends on M_H and $M^\star$ and not on the order of edges in $G_{\geq \varepsilon m}$).

▶ **Observation 14.** *We have $|\mathcal{P}| \geq |M^\star| - \frac{4}{3} \cdot \mu(H)$.*

We use $G_{\text{II.A}}$ to denote the subgraph of G that arrives in Phase II.A and use $G_{\text{II.B}}$ to denote the subgraph of G that arrives in Phase II.B.

▶ **Definition 15.** *We say an augmenting path $P \in \mathcal{P}$ is "lucky" if the following holds:*

1. *If $P = \langle e_1 \rangle$ then $e_1 \in G_{II.B}$.*
2. *If $P = \langle e_1, e_2, e_3 \rangle$ then $e_1, e_3 \in G_{II.A}$.*
3. *If $P = \langle e_1, e_2, e_3, e_4, e_5 \rangle$ then $e_1, e_5 \in G_{II.A}$ and $e_3 \in G_{II.B}$.*

We denote the set of lucky augmenting paths in $\mathcal{P}$ by $\mathcal{P}_L$.

Note that the subset $\mathcal{P}_L$ of $\mathcal{P}$ is now random since it depends on the order of edges in $G_{\geq \varepsilon m}$. Lemma 16 below proves that a relatively large fraction of augmenting paths in $\mathcal{P}$ will turn out to be lucky with high probability. The proof is straightforward; see [2].

▶ **Lemma 16.** *It holds that* $\mathbf{Pr}\left(|\mathcal{P}_L| \leq \gamma^2(1-\gamma)|\mathcal{P}| - \sqrt{15\mu(G)\ln n}\right) \leq 2n^{-5}$.

Next, observe that in Phase II.B of Algorithm 2 where we iteratively discover augmenting paths, we do not have the whole subgraph $G_{\text{II.A}}$ and have stored only a subgraph T of $G_{\text{II.A}}$ in the memory. In addition, when finding augmenting paths we use only the current edge e of $G_{\text{II.B}}$ in Algorithm 2. Therefore, not all lucky paths are actually discoverable by Algorithm 2. This motivates our next definition for "discoverable paths".

[3] We note, however, that the randomization in $G_{<\varepsilon m}$ is crucial for arguing that the algorithm uses $O(n \log n)$ space. Here, however, we are only analyzing the approximation ratio.

▶ **Definition 17.** *We say an augmenting path P (not necessarily in $\mathcal{P}$) for M_H is "discoverable" if $|P| \leq 5$, all edges of P are in $M_H \cup T \cup G_{II.B}$, and P has ≤ 1 edge in $G_{II.B}$.*

The next lemma proves there are many vertex-disjoint discoverable augmenting paths, by relating them to the number of lucky augmenting paths $|\mathcal{P}_L|$.

▶ **Lemma 18.** *There is a set $\mathcal{Q}$ of vertex-disjoint discoverable augmenting paths of M_H with*

$$|\mathcal{Q}| \geq \frac{1}{2b+3}\left(|\mathcal{P}_L| - \frac{4}{b}\cdot\mu(H)\right).$$

Observe that $\mathcal{Q}$ is only a set of vertex-disjoint discoverable augmenting paths. However, since Algorithm 2 applies augmenting paths greedily and in an arbitrary order, the set of applied augmenting paths may be very different from $\mathcal{Q}$. The next claim shows that we can nonetheless relate the number of augmenting paths that Algorithm 2 applies to the size of $\mathcal{Q}$.

▷ Claim 19. Let $\mathcal{Q}$ be as in Lemma 18. Algorithm 2 applies at least $|\mathcal{Q}|/6$ augmenting paths in Phase II.B. In other words, $|M| \geq \mu(H) + \frac{1}{6}|\mathcal{Q}|$.

Next, we show that the output of Algorithm 2 is strictly larger than $\mu(H)$.

▶ **Lemma 20.** *There is an absolute constant $\varepsilon_0' > 0$ such that for any $\varepsilon < 0.01$, if $\mu(H) \leq 0.68\mu(G)$ then with probability $1 - 1/\text{poly}(n)$, we have $|M| \geq \mu(H) + \varepsilon_0' \cdot \mu(G)$.*

Proof. We have

$$|M| \overset{\text{Claim 19}}{\geq} \mu(H) + \frac{1}{6}|\mathcal{Q}| \overset{\text{Lemma 18}}{\geq} \mu(H) + \frac{|\mathcal{P}_L| - \frac{4}{b}\mu(H)}{6(2b+3)}. \tag{3}$$

On the other hand, by Lemma 16 we know that with $1 - 1/\text{poly}(n)$ probability,

$$\begin{aligned}
|\mathcal{P}_L| &> \gamma^2(1-\gamma)|\mathcal{P}| - \sqrt{15\mu(G)\ln n} && \text{(By Lemma 16)}\\
&= \frac{4}{27}|\mathcal{P}| - \sqrt{15\mu(G)\ln n} && \text{(Since } \gamma = 2/3\text{)}\\
&\geq \frac{4}{27}\left(|M^\star| - \frac{4}{3}\mu(H)\right) - \sqrt{15\mu(G)\ln n} && \text{(By Observation 14)}\\
&\geq \frac{4}{27}\left((1-2\varepsilon)\mu(G) - \frac{4}{3}\mu(H)\right) - \sqrt{15\mu(G)\ln n} && \text{(By Assumption 13)}\\
&> 0.0108\mu(G) - \sqrt{15\mu(G)\ln n} && (\varepsilon < 0.01 \text{ and } \mu(H) \leq 0.68\mu(G))\\
&> 0.01\mu(G). && \text{(Since } \mu(G) > c\log n \text{ for any desirably large constant } c.\text{)}
\end{aligned}$$

Replacing this high probability lower bound for $|\mathcal{P}_L|$ into (3) we get that w.h.p.,

$$\begin{aligned}
|M| &\geq \mu(H) + \frac{0.01\mu(G) - \frac{4}{b}\mu(H)}{6(2b+3)}\\
&> \mu(H) + 10^{-7}\mu(G). && \text{(Replacing } b = 500 \text{ and noting } \mu(H) \leq 0.68\mu(G)\text{.)}
\end{aligned}$$

This completes the proof. ◀

Lemma 20 states that the output of Algorithm 2 is strictly larger than $\mu(H)$. Moreover, the guarantee of Section 3 implies that $\mu(H)$ must be at least (almost) $\frac{2}{3}\mu(G)$ (or otherwise $\mu(H \cup U)$ is strictly larger than $2/3\mu(G)$). The combination of these implies the output of Algorithm 2 is at least $(2/3 + \Omega(1))\mu(G)$ proving Theorem 2.

References

1 Sepehr Assadi, MohammadHossein Bateni, Aaron Bernstein, Vahab S. Mirrokni, and Cliff Stein. Coresets meet EDCS: algorithms for matching and vertex cover on massive graphs. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2019, San Diego, California, USA, January 6-9, 2019*, pages 1616–1635, 2019.

2 Sepehr Assadi and Soheil Behnezhad. Beating two-thirds for random-order streaming matching. *CoRR*, abs/2102.07011, 2021. `arXiv:2102.07011`.

3 Sepehr Assadi and Aaron Bernstein. Towards a unified theory of sparsification for matching problems. In *2nd Symposium on Simplicity in Algorithms, SOSA@SODA 2019, January 8-9, 2019 - San Diego, CA, USA*, pages 11:1–11:20, 2019.

4 Claude Berge. *The theory of graphs*. Courier Corporation, 1962.

5 Aaron Bernstein. Improved bounds for matching in random-order streams. In *47th International Colloquium on Automata, Languages, and Programming, ICALP 2020, July 8-11, 2020, Saarbrücken, Germany (Virtual Conference)*, pages 12:1–12:13, 2020.

6 Aaron Bernstein and Cliff Stein. Fully dynamic matching in bipartite graphs. In *Automata, Languages, and Programming - 42nd International Colloquium, ICALP 2015, July 6-10, 2015, Proceedings, Part I*, pages 167–179, 2015.

7 Aaron Bernstein and Cliff Stein. Faster fully dynamic matchings with small approximation ratios. In *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2016, January 10-12, 2016*, pages 692–711, 2016.

8 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, 2020.

9 Joan Feigenbaum, Sampath Kannan, Andrew McGregor, Siddharth Suri, and Jian Zhang. On graph problems in a semi-streaming model. *Theor. Comput. Sci.*, 348(2-3):207–216, 2005.

10 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, PODC 2019, Toronto, ON, Canada, July 29 - August 2, 2019*, pages 491–500, 2019.

11 Ashish Goel, Michael Kapralov, and Sanjeev Khanna. On the communication and streaming complexity of maximum bipartite matching. In *Proceedings of the Twenty-third Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '12, pages 468–485. SIAM, 2012. URL: `http://dl.acm.org/citation.cfm?id=2095116.2095157`.

12 Philip Hall. On representatives of subsets. *Journal of the London Mathematical Society*, 1(1):26–30, 1935.

13 Michael Kapralov. Better bounds for matchings in the streaming model. In *Proceedings of the Twenty-Fourth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2013, New Orleans, Louisiana, USA, January 6-8, 2013*, pages 1679–1697, 2013. `doi:10.1137/1.9781611973105.121`.

14 Michael Kapralov. Space lower bounds for approximating maximum matching in the edge arrival model. In *Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2021*, 2021.

15 Christian Konrad. A simple augmentation method for matchings with applications to streaming algorithms. In *43rd International Symposium on Mathematical Foundations of Computer Science, MFCS 2018, August 27-31, 2018, Liverpool, UK*, pages 74:1–74:16, 2018.

16 Christian Konrad, Frédéric Magniez, and Claire Mathieu. Maximum matching in semi-streaming with few passes. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques - 15th International Workshop, APPROX 2012, and 16th International Workshop, RANDOM 2012, Cambridge, MA, USA, August 15-17, 2012. Proceedings*, pages 231–242, 2012.

17 William T Tutte. The factorization of linear graphs. *Journal of the London Mathematical Society*, 1(2):107–111, 1947.