An Estimator for Matching Size in Low Arboricity Graphs with Two Applications

Hossein Jowhari

Department of Computer Science and Statistics, Faculty of Mathematics, K. N. Toosi University of Technology, Tehran, Iran

Abstract

In this paper, we present a new degree-based estimator for the size of maximum matching in bounded arboricity graphs. When the arboricity of the graph is bounded by α, the estimator gives a $\alpha + 2$ factor approximation of the matching size. For planar graphs, we show the estimator does better and returns a 3.5 approximation of the matching size. Using this estimator, we get new results for approximating the matching size of planar graphs in the streaming and distributed models of computation. In particular, in the vertex-arrival streams, we get a randomized $O(\frac{\sqrt{n}}{\varepsilon^2}\log n)$ space algorithm for approximating the matching size of a planar graph on n vertices within $(3.5+\varepsilon)$ factor. Similarly, we get a simultaneous protocol in the vertex-partition model for approximating the matching size within $(3.5+\varepsilon)$ factor using $O(\frac{n^{2/3}}{\varepsilon^2}\log n)$ communication from each player. In comparison with the previous estimators, the estimator in this paper does not need to know the arboricity of the input graph and improves the approximation factor in the case of planar graphs.

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

Keywords and phrases Data Stream Algorithms, Maximum Matching, Planar Graphs

Digital Object Identifier 10.4230/LIPIcs.APPROX/RANDOM.2021.10

Category APPROX

Related Version *Previous Version*: `https://arxiv.org/pdf/2011.11706.pdf`

Funding *Hossein Jowhari*: This work is supported by the Iranian Institute for Research in Fundamental Sciences (IPM), Project Number 98050014.

1 Introduction

A matching in a graph $G = (V, E)$ is a subset of edges $M \subseteq E$ where no two edges in M share an endpoint. A maximum matching of G has the maximum number of edges among all possible matchings. Let $m(G)$ denote the *matching size* of G, in other words the size of a maximum matching in G. In this paper, we present algorithms for approximating $m(G)$ in the sublinear models of computation. In particular, our results are for the vertex-arrival stream model (also known as the adjacency list streams). In the vertex-arrival model, the input stream is an arbitrary ordering of vertex set V. Additionally, followed by each $u \in V$ in the stream, the algorithm also gets the list of the neighbors of u. This is in contrast with the *edge-arrival* version where the input stream is an arbitrary ordering of the edge set E.

The problem of estimating $m(G)$ or computing an approximate maximum matching of G in the data stream model has been studied in several works [14, 13, 10, 6, 17]. Here we focus on algorithms for graphs with bounded arboricity. A graph $G = (V, E)$ has arboricity bounded by α if the edge set E can be partitioned into at most α forests. A well-known fact (known as the Nash-William theorem [19]) states that a graph has arboricity α, if and only if every induced subgraph on t vertices has at most $\alpha(t-1)$ number of edges. Graphs with low arboricity cover a wide range of graphs such as graphs with constant degree, planar graphs, and graphs with small tree-widths. In particular planar graphs have arboricity bounded by 3.

A simple reduction from counting distinct elements implies that the exact computation of $m(G)$ in the data stream model has $\Omega(n)$ space complexity even for trees and randomized algorithms (see [1] for the lower bound on distinct elements problem.) This negative result has inspired a growing interest in finding estimators for $m(G)$ that take sublinear space to compute. Specially when the input graph G has low arboricity, it has been shown by Esfandiari *et al.* [9], and the subsequent works [16, 7, 18, 4], that it is possible to compute $m(G)$ approximately in $o(n)$ space by only checking the degree information and the immediate local neighborhood of the vertices (and the edges). This line of research has led to the invent of several *degree-based* estimators for the matching size. In this paper, we design another degree-based estimator for $m(G)$ in low arboricity graphs that has certain advantages in comparison with the previous works and leads to new algorithmic results. Before describing our estimator we briefly review the previous ideas.

1.1 Previous Works

In the following we assume G has arboricity bounded by α. Also, unless explicitly stated, all the algorithms mentioned below are randomized.

Shallow edges, high degree vertices

Esfandiari *et al.* [9] were first to observe that one can approximately characterize the matching size of low arboricity graphs based on the degree information of the vertices and the local neighborhood of the edges. Let H denote the set of vertices with degree more than $h = 2\alpha + 3$ and let F denote the set of edges with both endpoints having degree at most h. Esfandiari *et al.* have shown that $m(G) \le |H| + |F| \le (5\alpha + 9)m(G)$. Based on this estimator, the authors in [9] have designed a $\tilde{O}(\varepsilon^{-2}\alpha n^{2/3})$ space algorithm for approximating $m(G)$ within $5\alpha + 9 + \varepsilon$ factor in the edge-arrival model.

Fractional matchings

By establishing an interesting connection with fractional matchings and the Edmonds Polytope theorem, Mcgregor and Vorotnikova [16] have shown that the following quantity approximates $m(G)$ within $(\alpha + 2)$ factor.

$$(\alpha + 1) \sum_{(u,v)\in E} \min\{\frac{1}{\deg(u)}, \frac{1}{deg(v)}, \frac{1}{\alpha + 1}\}.$$

Based on this estimator, the authors in [16] have given a $\tilde{O}(\varepsilon^{-2}n^{2/3})$ space streaming algorithm (in the edge-arrival model) that approximate $m(G)$ within $\alpha + 2 + \varepsilon$ factor. Also in the same work, another degree-based estimator is given that returns a $\frac{(\alpha+2)^2}{2}$ factor approximation of $m(G)$. A notable property of this estimator is that it can be implemented *deterministically* in the vertex-arrival model using only $O(\log n)$ bits of space.

α-Last edges

Cormode *et al.* [7] (later revised by Mcgregor and Vorotnikova [18]) have designed an improved estimator for $m(G)$ that also depends on a given ordering of the edges. Given a stream of edges $S = e_1, \ldots, e_m$, let $E_\alpha(S)$ denote a subset of edges where $(u, v) \in E_\alpha(S)$ iff the vertices u and v both appear at most α times in S after the edge (u, v). It is shown that $m(G) \le |E_\alpha(S)| \le (\alpha + 2)m(G)$. Moreover a $O(\frac{1}{\varepsilon^2} \log^2 n)$ space streaming algorithm is shown that approximates $|E_\alpha(S)|$ within $1 + \varepsilon$ factor in the edge-arrival model.

1.2 The estimator in this paper

The new estimator is purely based on the degree of the vertices in the graph without any need to have a pre-knowledge of α. To estimate the matching size, we count the number of what we call *locally superior* vertices in the graph. Consider the following definition.

▶ **Definition 1.** *In graph $G = (V, E)$, we call $u \in V$ a locally superior vertex if u has a neighbor v such that $deg(u) \geq deg(v)$. We let $\ell(G)$ denote the number of locally superior vertices in G.*

We show when the arboricity of G is bounded by α, $\ell(G)$ approximates $m(G)$ within $(\alpha + 2)$ factor (Lemma 2.) This repeats the same bound obtained by the estimators in [16] and [7], however for planar graphs, we prove the approximation factor is at most 3.5 which beats the previous bounds (Lemma 5). This result is the main technical contribution of the paper. As an evidence on the improved approximation quality, consider the 4-regular planar graph on 9 vertices. Both of the estimators in [16] and [7], report 18 as the estimation for $m(G)$ while the exact answer is 4. It follows their approximation factor is at least 4.5.

As a first application of Lemma 5, we obtain a randomized $O(\frac{\sqrt{n}}{\varepsilon^2} \log n)$ space streaming algorithm for approximating $m(G)$ within $(3.5 + \varepsilon)$ factor in the vertex-arrival model. In terms of approximation factor, this improves over existing sub-linear algorithms [16, 18].

As another application of our estimator, we get a sublinear simultaneous protocol in the *vertex-partition* model for approximating $m(G)$ when G is planar. In this model, the vertex set V is partitioned into t subsets $V_1, \ldots, V_t$ where each subset is given to a player. Additionally the i-th player knows the edges on V_i. The players do not communicate with each other. They only send one message to a *referee* whom at the end computes an approximation of the matching size. (The referee does not get any part of the input.) We assume the referee and the players have a shared source of randomness. Within this setting, we design a protocol that approximates $m(G)$ within $3.5 + \varepsilon$ factor using $O(\frac{n^{2/3}}{\varepsilon^2} \log n)$ communication from each player. Note that for $t > 3$ and $t = o(n^{1/3})$, this result is non-trivial. The best previous result, implicit in the works of [5, 16] computes a $5 + \varepsilon$ factor approximation using $\tilde{O}(n^{4/5})$ communication from each player. Also we should mention that, using the the estimator in [16], one can get a deterministic simultaneous protocol where each player sends only $O(\log n)$ bits to the referee. However this protocol computes a 12.5 factor approximation of $m(G)$.

1.3 Related Works

Kapralov *et al.* [13] have given an estimator for $m(G)$ when G is a general graph. Their estimator gives an approximation of $m(G)$ by looking at the degree information of the vertices in a series of nested subgraphs of G. The main challenge is implementing the estimator in sublinear space. Based on this estimator, Kapralov *et al.* has shown one can obtain a poly$(\log n)$ approximation of $m(G)$ in poly$(\log n)$ space assuming the input edge stream is randomly ordered. There are subsequent works with improved results and analysis [14]. As far as we know, there is no similar result for arbitrarily ordered streams.

There is a large body of works that addresses the problem of finding a matching of near optimal size using $O(n.\text{poly}(\log n))$ space. This falls within the category of semi-streaming model. See the recent works [12, 3] for the latest results on this.

Maximum matching has also been studied within the context of distributed local algorithms [15] and massively parallel computations [2]. The main objective of these works is to find a large matching of near optimal size in a distributed manner using small number of communication rounds. See the works [8, 11] for related results on graphs with bounded arboricity.

2 Graph properties

In the following proofs, we let $M \subseteq E$ denote a maximum matching in graph G. When the underlying graph is clear from the context, for the vertex set S, we use $N(S)$ to denote the neighbors of the vertices in S excluding S itself. For vertex u, we simply use $N(x)$ to denote the neighbors of u. The vertex v is a neighbor of the edge (x, y) if v is adjacent with x or y. When x is paired with y in the matching M, abusing the notation, we define $M(x) = y$.

▶ **Lemma 2.** *Let $G = (V, E)$ be a graph with arboricity α. We have*

$$m(G) \leq \ell(G) \leq (\alpha + 2)m(G).$$

Proof. The left hand side of the inequality is easy to show. For every edge in E, at least one of the endpoints is locally superior. Since edges in M are disjoint, at least $|M|$ number of endpoints must be locally superior. This proves $m(G) \leq \ell(G)$.

To show the right hand side, we use a charging argument. Let L denote the locally superior vertices in G. Our goal is to show an upper bound on $|L|$ in terms of $|M|$ and α. Let $X \subseteq L$ be the set of locally superior vertices that are NOT endpoints of a matching edge. The challenge is to prove an upper on $|X|$.

The vertices in X do not contribute to the maximum matching. However all the vertices in $N(X)$ must be endpoints of matching edges (otherwise M would not be maximal.) For the same reason, there cannot be an edge between the vertices in X. To prove an upper bound on $|X|$, in the first step, we assign a subset of vertices in X to edges in M in a way any target edge gets at most $\alpha - 1$ locally superior vertices. We do the assignments in the following way.

The Assignment Procedure

If we find a $y \in N(X)$ with at most $\alpha - 1$ neighbors in X, we assign all the neighbors of y in X to the matching edge $(y, M(y))$. We repeat this process, every time picking a vertex in $N(X)$ with less than α neighbors in X and do the assignment that we just described, until we cannot find such a vertex in $N(X)$. Note that when we assign a locally superior vertex x, we remove the edges on x before continuing the procedure.

Here we emphasize the fact that if y has a neighbor $x \in X$, then $M(y)$ cannot have neighbors in $X \setminus \{x\}$ (otherwise it would create an augmenting path and contradict with the optimality of M.)

Let $X_1 \subseteq X$ be the assigned locally superior vertices and $M_1 \subseteq M$ be the used matching edges in the assignment procedure. We have

$$|X_1| \leq (\alpha - 1)|M_1|. \tag{1}$$

Let $X_2 = X \setminus X_1$ be the unassigned vertices in X. Now we try to prove an upper bound on $|X_2|$. For this, we need to make a few observations.

▶ **Observation 3.** *Let $Y_2 = N(X_2)$. The pair y and $M(y)$ cannot be both in Y_2.*

Proof. Suppose y and $M(y)$ are both in $N(X_2)$. Let B and C be the neighbors of y and $M(y)$ in X_2 respectively. If $|B \cup C| > 1$, then one can find an augmenting path of length 3 (with respect to M.) A contradiction.

On the other hand, if $|B \cup C| = 1$, then y and $M(y)$ have only a shared neighbor $x \in X_2$ which means the edge $e = (y, M(y))$ should have been used by the assignment procedure and as result $x \in X_1$. Another contradiction. ◀

▶ **Observation 4.** *Every vertex $x \in X_2$ has degree at least $\alpha + 1$.*

Proof. Consider $x \in X_2$. Suppose, for the sake of contradiction, $\deg(x)$ is k where $k \leq \alpha$. Since x is a locally superior vertex, there must be a $y \in N(x)$ with degree at most k in G. We know that y is an endpoint of a matching edge. In the assignments procedure, whenever we used an edge $e \in M$ all the neighbors of its endpoints (in X) were assigned. Since x is not assigned yet, it means the edge $(y, M(y))$ has not been used. Consequently y must have at least α neighbors in X_2. Counting the edge $(y, M(y))$, we should have $\deg(y) \geq \alpha + 1$. A contradiction. ◀

Let $G' = (X_2 \cup Y_2, E')$ be a bipartite graph where E' is the set of edges between X_2 and Y_2. From Observation 4, we have

$$(\alpha + 1)|X_2| \leq |E'|. \tag{2}$$

Since G' is a subgraph of G, its arboricity is bounded by α. As result,

$$|E'| \leq \alpha(|X_2| + |Y_2|). \tag{3}$$

Recall that Y_2 are endpoints of matching edges. Let M_2 be those matching edges. Observation 3 implies that $|Y_2| = |M_2|$. As result, combining (2) and (3), we get the following.

$$|X_2| \leq \alpha|Y_2| = \alpha|M_2|. \tag{4}$$

To prove an upper bound on $|L|$, we also need to count the locally superior vertices that are endpoints of matching edges. Let $Z = L \setminus X$. We have $|Z| \leq 2|M|$. Summing up, we get

$$\begin{aligned} |L| &= |X_1| + |X_2| + |Z| \\ &\leq (\alpha - 1)|M_1| + \alpha|M_2| + 2|M| \\ &= \alpha(|M_1| + |M_2|) + 2|M| - |M_1| \\ &\leq (\alpha + 2)|M| - |M_1| \\ &\leq (\alpha + 2)|M| \end{aligned}$$

This proves the lemma. ◀

▶ **Lemma 5.** *Let $G = (V, E)$ be a planar graph. We have $\ell(G) \leq 3.5m(G)$.*

Proof. For planar graphs, similar to what we did in the proof of Lemma 2, we first try to assign some of the vertices in X to the matching edges using a simple assignment procedure. (Recall that X is the set of vertices in L that are not endpoints of edges in M.)

The Assignment Procedure

Let $Y_1 = \emptyset$. If we find a $y \in N(X)$ with only 1 neighbor $x \in X$, we assign x to the matching edge $(y, M(y))$. Also we add y to Y_1. We continue the procedure until we cannot find such a vertex in $N(X)$. Note that when we assign a locally superior vertex x, we remove the edges on x.

Let $X_1 \subseteq X$ be the assigned locally superior vertices and $M_1 \subseteq M$ be the used matching edges in the assignment procedure. Note that $|Y_1| = |M_1|$. We have

$$|X_1| \leq |M_1|. \tag{5}$$

Let $X_2 = X \setminus X_1$. Using a similar argument that we used for proving Observation 4, we can show every vertex in X_2 has degree at least 3. Also letting $Y_2 = N(X_2)$, we observe that $y \in Y_2$ and $M(y)$ cannot be both in Y_2 as we noticed in the Observation 3. Let $M_2 \subseteq M$ be the matching edges with one endpoint in Y_2. We have $|Y_2| = |M_2|$.

Now consider the bipartite graph $G' = (X_2 \cup Y_2, E')$ where E' is the set of edges between X_2 and Y_2. Every planar bipartite graph with n vertices has at most $2n - 4$ edges [1]. Since G' is a bipartite planar graph, it follows,

$$3|X_2| \leq |E'| < 2(|X_2| + |Y_2|) = 2(|X_2| + |M_2|). \tag{6}$$

This shows $|X_2| < 2|M_2|$. Letting $Z = L \setminus X$ and $M_3 = M \setminus (M_1 \cup M_2)$, we get

$$|L| = |X_1| + |X_2| + |Z| \leq |M_1| + 2|M_2| + 2|M| \leq 3|M| + |M_2| - |M_3|. \tag{7}$$

This already proves $|L|$ is bounded by $4|M|$. To prove the bound claimed in the lemma, we also show that $|L| \leq 3|M| + |M_1| + |M_3|$. Combined with the inequality (7), this proves the lemma.

Let $Y = Y_1 \cup Y_2$. Note that Y are one side of the matching edges in $M_1 \cup M_2$. Let $Y' = \{M(y) \mid y \in Y\}$. We use a special subset of Y', named Y'' which is defined as follows. We let Y'' denote the locally superior vertices in Y' that have degree 2 or they are adjacent with both endpoints of an edge in M_3. We make the following observation regarding the vertices in Y''.

▶ **Observation 6.** *We can assign each vertex $y' \in Y''$ to a distinct $e \in Y_1 \cup M_3$ where e has no neighbor in $Y' \setminus \{y'\}$.*

Proof. Consider $y' \in Y''$. If y' is adjacent with both endpoints of an edge $e = (z, z') \in M_3$, we assign y' to e (when there are multiple edges with this condition we pick one of them arbitrarily.) Note that z and z' cannot have neighbors in Y' other than y' because otherwise it would create an augmenting path.

Now suppose y' has degree 2. Since y' is a locally superior vertex, it must have a neighbor z of degree at most 2. The neighbor z cannot be in $Y_2 \cup X_2$ because the vertices in $Y_2 \cup X_2$ have degree at least 3. We distinguish between two cases.

- $M(y') \in Y_2$. In this case, z cannot be in Y_1 either because the vertices in Y_1 are already of degree 2 without y'. Also $z \notin X_1$ because otherwise it would create an augmenting path. The only possibility is that z is an endpoint of a matching edge in M_3. We assign y' to the matching edge $(z, z') \in M_3$. Note that z' cannot have a neighbor in $Y' \setminus \{y'\}$ because it would create an augmenting path.
- $M(y') \in Y_1$. Here z could be in X_1. If this is the case, then $M(y')$ cannot have a neighbor in $Y' \setminus \{y'\}$ because it would create an augmenting path. In this case, we assign y' to $M(y')$. If $z = M(y')$, then again we assign y' to $M(y')$. The only remaining possibility is that z an endpoint of a matching edge in M_3 which we handle it similar to the previous case. ◀

Now, assume we assign the vertices in Y'' to the elements in $Y_1 \cup M_3$ according to the above observation. Let $Y'_1 \subseteq Y_1$ and $M'_3 \subseteq M_3$ be the vertices and edges that were used in the assignment. Let Y''' be the remaining locally superior vertices in Y'. Namely, $Y''' = (L \cap Y') \setminus Y''$. Before making the final point, we observe that only one endpoint of the

[1] For a short proof of this, combine the Euler's formula $|V| - |E| + |F| = 2$ with the inequality $2|E| \geq 4|F|$ caused by each face having at least 4 sides (since there are no odd cycles) and we get $|E| \leq 2|V| - 4$.

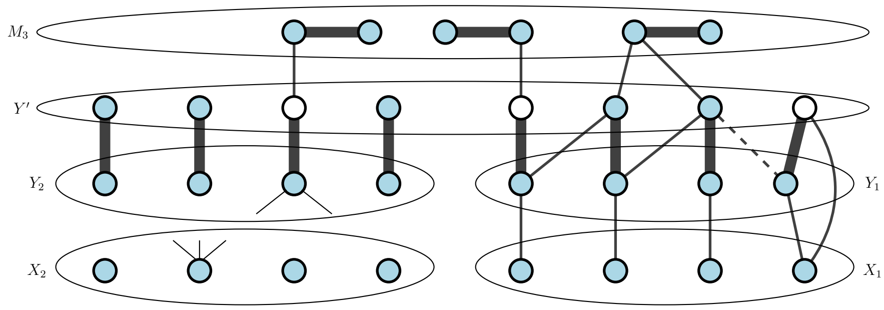

Figure 1 A demonstration of the construction in the proof of lemmas 2 and 5. Thick edges represent matching edges. The unfilled vertices belong to the set Y''.

edges in M_3 are adjacent with vertices in Y'''. Let Y_3 be the endpoint of edges in $M_3 \setminus M_3'$ that have neighbors in Y'''. Consider the bipartite graph $G''(V'', E'')$ where

$$V'' = (X_2 \cup Y''') \cup \big(Y_2 \cup (Y_1 \setminus Y_1') \cup Y_3\big)$$

and E'' is the set of edges between X_2 and Y_2, and the edges between Y''' and $Y_2 \cup (Y_1 \setminus Y_1') \cup Y_3$.

Relying on the facts that G'' is a planar bipartite graph, Y''' is composed of vertices with degree at least 3, and the edges on Y''' are all in E'', we have

$$3|X_2| + 3|Y'''| \leq |E''| \leq 2(|X_2| + |Y_2| + |Y_1 \setminus Y_1'| + |Y'''| + |Y_3|).$$

It follows,

$$\begin{aligned} |X_2| + |Y'''| &\leq 2(|Y_2| + |Y_1 \setminus Y_1'| + |Y_3|) \\ &\leq 2(|M_2| + |M_1| - |Y_1'| + |M_3| - |M_3'|) \\ &= 2(|M| - |Y_1'| - |M_3'|) \end{aligned}$$

Since $|Y''| = |Y_1'| + |M_3'|$, we get

$$|X_2| + |Y'''| \leq 2|M| - 2|Y''| \tag{8}$$

Let Z_1, Z_2 and Z_3 denote the locally superior vertices that are endpoints of matching edges in M_1, M_2 and M_3 respectively. From the definition of Y'' and Y''', we have

$$|Z_1| + |Z_2| \leq |M_1| + |M_2| + |Y''| + |Y'''| \tag{9}$$

From (8) and (9), we get

$$\begin{aligned} |L| &= |X_1| + |X_2| + |Z_1| + |Z_2| + |Z_3| \\ &\leq |M_1| + |X_2| + (|M_1| + |M_2| + |Y''| + |Y'''|) + 2|M_3| \\ &= 2|M_1| + (|X_2| + |Y'''|) + |M_2| + |Y''| + 2|M_3| \\ &\leq 2|M_1| + |M_2| + 2|M| - |Y''| + 2|M_3| \\ &= 3|M| + |M_1| + |M_3| - |Y''| \\ &\leq 3|M| + |M_1| + |M_3| \end{aligned}$$

This finishes the proof of the lemma. ◀

3 Algorithms

We first present a high-level sampling-based estimator for $\ell(G)$. Then we show how this estimator can be implemented in the streaming and distributed settings using small space and communication. For our streaming result, we use a combination of the estimator for $\ell(G)$ and the greedy maximal matching algorithm. For the simultaneous protocol, we use the estimator for $\ell(G)$ in combination with the edge-sampling primitive in [5] and an estimator in [16].

The high-level estimator (described in Algorithm 1) samples a subset of vertices $S \subseteq V$ and computes the locally superior vertices in S. The quantity $\ell(G)$ is estimated from the scaled ratio of the locally superior vertices in the sample set.

Algorithm 1 The high-level description of the estimator for $\ell(G)$.

Run the following estimator $r = \lceil \frac{8}{\varepsilon^2} \rceil$ number of times in parallel. In the end, report the average of the outcomes.

1. Sample s vertices (uniformly at random) from V without replacement.
2. Let S be the set of sampled vertices.
3. Compute S' where S' is the set of locally superior vertices in S.
4. Return $\frac{n}{s}|S'|$ as an estimation for $\ell(G)$.

▶ **Lemma 7.** *Assuming $s \geq \frac{n}{\ell(G)}$, the high-level estimator in Algoirthm 1 returns a $1+\varepsilon$ factor approximation of $\ell(G)$ with probability at least $7/8$.*

Proof. Fix a parallel repetition of the algorithm and let X denote the outcome of the associated estimator. Assuming an arbitrary ordering on the locally superior vertices, let X_i denote the random variable associated with i-th locally superior vertex. We define $X_i = 1$ if the i-th locally superior vertex has been sampled, otherwise $X_i = 0$. We have $X = \frac{n}{s}\sum_{i=1}^{\ell(G)} X_i$. Since $Pr(X_i = 1) = \frac{s}{n}$, we get $E[X] = \ell(G)$. Further we have

$$\begin{aligned}
E[X^2] = \frac{n^2}{s^2} E\Big[\sum_{i,j}^{\ell(G)} X_i X_j\Big] &= \frac{n^2}{s^2}\Big[\sum_{i}^{\ell(G)} E[X_i^2] + \sum_{i\neq j}^{\ell(G)} E[X_i X_j]\Big] \\
&= \frac{n^2}{s^2}\Big[\frac{s}{n}\ell(G) + \binom{\ell(G)}{2}\frac{s(s-1)}{n(n-1)}\Big] \\
&= \frac{n}{s}\ell(G) + \binom{\ell(G)}{2}\frac{n(s-1)}{s(n-1)} \\
&< \frac{n}{s}\ell(G) + \ell^2(G)
\end{aligned}$$

Consequently, $Var[X] = E[X^2] - E^2[X] < \frac{n}{s}\ell(G)$.

Let Y be the average of the outcomes of r parallel and independent repetitions of the basic estimator. We have $E[Y] = \ell(G)$ and $Var[Y] < \frac{n}{sr}\ell(G)$. Using the Chebyshev's inequality,

$$Pr(|Y - E[Y]| \geq \varepsilon E[Y]) \leq \frac{Var[Y]}{\varepsilon^2 E^2[X]} < \frac{n/s}{r\varepsilon^2 \ell(G)}.$$

Setting $r = \frac{8}{\varepsilon^2}$ and $s \geq \frac{n}{\ell(G)}$, the above probability will be less than $1/8$. ◀

3.1 The streaming algorithm

We first note that we can implement the high-level estimator of Algorithm 1 in the vertex-arrival stream model using $O(\frac{s}{\varepsilon^2}\log n)$ space. Consider a single repetition of the estimator. The sampled set S is selected in the beginning of the algorithm (before the stream.) This can be done using a reservoir sampling strategy [20] in $O(|S|\log n)$ space. To decide if $u \in S$ is locally superior or not, we just need to store $\deg(u)$ and the minimum degree of the neighbors that are visited so far. Note that when processing a vertex $v \in V$ and its neighbors, we know if v is a neighbor of u or not. Consequently, checking if u is a locally superior or not takes $O(\log n)$ bits of space. Therefore the whole space needed to implement a single repetition is $O(s\log n)$ bits.

The streaming algorithm runs two threads in parallel. In one thread it runs the streaming implementation of Algorithm 1 after setting $s = \lceil \sqrt{n}\, \rceil$. In the other thread, it runs a greedy algorithm to find a maximal matching in the input graph. We stop the greedy algorithm whenever the size of the discovered matching F exceeds $\sqrt{n}$. In the end, if $|F| < \sqrt{n}$, we output $|F|$ as an approximation for $m(G)$, otherwise we report the outcome of the first thread.

Note that if $|F| < \sqrt{n}$, F is a maximal matching in G. Hence $|F| \geq \frac{1}{2}m(G)$. Assume $|F| \geq \sqrt{n}$. In this case the algorithm outputs the result of first thread. In this case, by Lemma 2, we know $\ell(G) \geq \sqrt{n}$. Consequently, by Lemma 7, the first thread returns a $1 + O(\varepsilon)$ approximation of $\ell(G)$ and hence it returns a $3.5 + O(\varepsilon)$ approximation of $m(G)$. Since the greedy algorithm takes at most $O(\sqrt{n})$ space, the space complexity of the algorithm is dominated by the space usage of the first thread. We get the following result.

▶ **Theorem 8.** *Let G be a planar graph. There is a randomized streaming algorithm in the vertex-arrival model that returns a $3.5 + \epsilon$ factor approximation of $m(G)$ using $O(\frac{\sqrt{n}}{\epsilon^2})$ space.*

3.2 A simultaneous communication protocol

In this section we describe a communication protocol for approximating $m(G)$ in the vertex-partition model. Recall that in this model the vertex set V is partitioned into t subsets $V_1, \ldots, V_t$ where the subset V_i is given to the i-th player. Additionally the i-th player knows the edges on the vertices in V_i. The players do not communicate with each other. They only send one message to a referee whom at the end computes an approximation of the matching size. Also we emphasize the assumption that the referee and the players have a shared source of randomness.

To describe the simultaneous protocol, we consider two cases separately: (a) when the matching size is low; to be precise, when it is smaller than some fixed value $k = n^{1/3}$, and (b) when the matching size is high, *i.e.* at least $\Omega(k)$. For each case, we describe a separate solution. The overall protocol will be the parallel run of these two solutions along with a sub-protocol to distinguish between the cases.

Graphs with large matching size

In the case when matching size is large, similar to what was done in the streaming model, we run an implementation of Algorithm 1 in the given simultaneous model. To see how this is implemented, in the simultaneous model all the players (including the referee) know the sampled set S. This results from access to the shared randomness. For each $u \in S$, the players send the minimum degree of the neighbors of u in his input to the referee. The player that owns u, also sends $\deg(u)$ to the referee. Having received this information, the referee can decide if u is a locally superior vertex or not. As result, we can implement Algorithm 1 in the simultaneous model using a protocol with $O(\frac{s}{\varepsilon^2}\log n)$ message size.

Graphs with small matching size

In the case where the matching size is small, we use the edge-sampling method of [5]. Here we review their basic sampling primitive in its general form. Given a graph $G(V, E)$, let $c : V \to [b]$ be a totally random function that assigns each vertex in V a random number (color) in $[b] = \{1, \ldots, b\}$. The set $\text{Sample}_{b,d,1}$ is a random subset of E picked in the following way. Given a subset $K \subseteq [b]$ of size $d \in \{1, 2\}$, let E_K be the edges of G where the color of their endpoints matches K. For example when $K = \{3, 4\}$, the set $E_{\{3,4\}}$ contains all edges (u, v) such that $\{c(u), c(v)\} = \{3, 4\}$. For all $K \subseteq [b]$ of size d, the set $\text{Sample}_{b,d,1}$ picks a random edge from E_K. Finally, the random set $\text{Sample}_{b,d,r}$ is the union of r independent instances of $\text{Sample}_{b,d,1}$. We have the following lemma from [5] (see Theorems 4 in the reference.)

▶ **Lemma 9.** *Let $G = (V, E)$ be a graph. Assuming $m(G) \le k$, with probability $1 - 1/\text{poly}(k)$, the random set $Sample_{100k,2,O(\log k)}$ contains a matching of size $m(G)$.*

Note that, in the simultaneous vertex-partition model, the referee can obtain an instance of $\text{Sample}_{b,d,1}$ via a protocol with $O(b^d \log n)$ message size. To see this, using the shared randomness, the players pick the random function $c : V \to [b]$. Let $E^{(i)}$ be the subset of edges owned by the i-th player. We have $E = \bigcup_{i=1}^{t} E^{(i)}$. To pick a random edge from E_K for a given $K \subseteq [b]$, the i-th player randomly picks an edge $e \in E_K \cap E^{(i)}$ and sends it along with $|E_K \cap E^{(i)}|$ to the referee. After receiving this information from all the players, the referee can generate a random element of E_K. Since there are $O(b^d)$ different d-subsets of $[b]$, the size of the message from a player to the referee is bounded by $O(b^d \log n)$ bits. Consequently, the referee can produce a rightful instance of $\text{Sample}_{b,d,r}$ using $O(rb^d \log n)$ communication from each player.

How to distinguish between the cases?

For this task, we use a degree-based estimator by Mcgregor and Vorotnikova [16] described in the following lemma.

▶ **Lemma 10.** *Let $G = (V, E)$ be a planar graph. Let $A'(G) = \sum_{u \in V} \min\{\deg(u)/2, 4 - \deg(u)/2\}$. We have*

$$m(G) \le A'(G) \le 12.5\, m(G).$$

It is easy to see that, in the simultaneous vertex-partition model, we can implement this estimator with $O(\log n)$ bits communication from each player.

The final protocol

Let $k = \lceil n^{1/3} \rceil$. We run the following threads in parallel.

1. A protocol that implements the high-level estimator (Algorithm 1) with $s = \lceil 12.5n/k \rceil$ as its input parameter according to the discussions above. Let z_1 be the output of this protocol.
2. A protocol to compute an instance of $\text{Sample}_{b,d,r}$ for $b = 100k$ and $d = 2$ and $r = O(\log k)$. Let z_2 be the size of maximum matching in the sampled set.
3. A protocol to compute $A'(G)$. Let z_3 be the output of this thread.

In the end, if $z_3 \ge \frac{k}{12.5}$, the referee outputs z_1 as an approximation for $m(G)$, otherwise the referee reports z_2 as the final answer.

▶ **Theorem 11.** *Let G be a planar graph on n vertices. The above simultaneous protocol, with probability $3/4$, returns a $3.5 + O(\varepsilon)$ approximation of $m(G)$ where each player sends $O(\frac{n^{2/3}}{\varepsilon^2})$ bits to the referee.*

Proof. First we note that by choosing the constants large enough, we can assume the thread (2) errs with probability at most $1/8$. If $z_3 \geq \frac{k}{12.5}$, then we know $m(G) \geq \frac{k}{12.5}$. This follows from Lemma 10. Consequently by Lemma 2, we have $\ell(G) \geq \frac{k}{12.5}$. Therefore from Lemma 7, we have $|z_1 - \ell(G)| \leq \varepsilon\ell(G)$ with probability at least $7/8$. It follows from Lemma 5 that $(1-\varepsilon)m(G) \leq z_1 \leq (3.5 + 3.5\varepsilon)m(G)$.

On the other hand, if $z_3 < \frac{k}{12.5}$, by Lemma 10 we know that $m(G)$ must be less than k. Having this, from Lemma 9, with probability at least $7/8$, we get $z_2 = m(G)$. In this case the protocol computes the exact matching size of the graph.

The message size of each player is dominated by the cost of the first thread which is $O(n^{2/3}\varepsilon^{-2}\log n)$. The total error probability is bounded by $1/4$. This finishes the proof. ◀

4 Conclusion

In this paper we presented a degree-based estimator for the size of maximum matching in planar graphs. We showed our estimator gives a 3.5 factor approximation of the matching size. This improves the approximation factor of the previous degree-based estimators. We do not have tight examples for our analysis. In fact, we conjecture that $\ell(G)$ approximates $m(G)$ within 3 factor when G is planar.

Using our estimator, we obtained an improved sublinear space algorithm for estimating the matching size in the vertex-arrival streams. We also showed a more efficient simultaneous protocol for estimating the matching size in planar graphs. Unfortunately, the new estimator, in spite of its simplicity, does not immediately lead to one-pass sublinear algorithm in the edge-arrival model. To decide if a vertex is locally superior, we need to know its neighbors and learn their degrees which becomes burdensome in one pass. However, given an extra pass over the stream the same space bound and approximation factor is achievable for the edge-arrival streams as well. It would be interesting to do this without the extra pass.

References

1. Noga Alon, Yossi Matias, and Mario Szegedy. The space complexity of approximating the frequency moments. *J. Comput. Syst. Sci.*, 58(1):137–147, 1999. `doi:10.1006/jcss.1997.1545`.
2. Soheil Behnezhad, MohammadTaghi Hajiaghayi, and David G. Harris. Exponentially faster massively parallel maximal matching. In David Zuckerman, editor, *60th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2019, Baltimore, Maryland, USA, November 9-12, 2019*, pages 1637–1649. IEEE Computer Society, 2019. `doi:10.1109/FOCS.2019.00096`.
3. Aaron Bernstein. Improved bounds for matching in random-order streams. In Artur Czumaj, Anuj Dawar, and Emanuela Merelli, editors, *47th International Colloquium on Automata, Languages, and Programming, ICALP 2020, July 8-11, 2020, Saarbrücken, Germany (Virtual Conference)*, volume 168 of *LIPIcs*, pages 12:1–12:13. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020. `doi:10.4230/LIPIcs.ICALP.2020.12`.
4. Marc Bury, Elena Grigorescu, Andrew McGregor, Morteza Monemizadeh, Chris Schwiegelshohn, Sofya Vorotnikova, and Samson Zhou. Structural results on matching estimation with applications to streaming. *Algorithmica*, 81(1):367–392, 2019. `doi:10.1007/s00453-018-0449-y`.
5. Rajesh Chitnis, Graham Cormode, Hossein Esfandiari, MohammadTaghi Hajiaghayi, Andrew McGregor, Morteza Monemizadeh, and Sofya Vorotnikova. Kernelization via sampling with

applications to finding matchings and related problems in dynamic graph streams. In Robert Krauthgamer, editor, *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2016, Arlington, VA, USA, January 10-12, 2016*, pages 1326–1344. SIAM, 2016. doi:10.1137/1.9781611974331.ch92.

6 Rajesh Hemant Chitnis, Graham Cormode, Mohammad Taghi Hajiaghayi, and Morteza Monemizadeh. Parameterized streaming: Maximal matching and vertex cover. In Piotr Indyk, editor, *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2015, San Diego, CA, USA, January 4-6, 2015*, pages 1234–1251. SIAM, 2015. doi:10.1137/1.9781611973730.82.

7 Graham Cormode, Hossein Jowhari, Morteza Monemizadeh, and S. Muthukrishnan. The sparse awakens: Streaming algorithms for matching size estimation in sparse graphs. In *25th Annual European Symposium on Algorithms, ESA 2017, September 4-6, 2017, Vienna, Austria*, pages 29:1–29:15, 2017. doi:10.4230/LIPIcs.ESA.2017.29.

8 Andrzej Czygrinow, Michal Hanckowiak, and Edyta Szymanska. Fast distributed approximation algorithm for the maximum matching problem in bounded arboricity graphs. In Yingfei Dong, Ding-Zhu Du, and Oscar H. Ibarra, editors, *Algorithms and Computation, 20th International Symposium, ISAAC 2009, Honolulu, Hawaii, USA, December 16-18, 2009. Proceedings*, volume 5878 of *Lecture Notes in Computer Science*, pages 668–678. Springer, 2009. doi:10.1007/978-3-642-10631-6_68.

9 Hossein Esfandiari, Mohammad Taghi Hajiaghayi, Vahid Liaghat, Morteza Monemizadeh, and Krzysztof Onak. Streaming algorithms for estimating the matching size in planar graphs and beyond. In Piotr Indyk, editor, *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2015, San Diego, CA, USA, January 4-6, 2015*, pages 1217–1233. SIAM, 2015. doi:10.1137/1.9781611973730.81.

10 Hossein Esfandiari, MohammadTaghi Hajiaghayi, and Morteza Monemizadeh. Finding large matchings in semi-streaming. In Carlotta Domeniconi, Francesco Gullo, Francesco Bonchi, Josep Domingo-Ferrer, Ricardo Baeza-Yates, Zhi-Hua Zhou, and Xindong Wu, editors, *IEEE International Conference on Data Mining Workshops, ICDM Workshops 2016, December 12-15, 2016, Barcelona, Spain*, pages 608–614. IEEE Computer Society, 2016. doi:10.1109/ICDMW.2016.0092.

11 Mohsen Ghaffari, Christoph Grunau, and Ce Jin. Improved MPC algorithms for mis, matching, and coloring on trees and beyond. In Hagit Attiya, editor, *34th International Symposium on Distributed Computing, DISC 2020, October 12-16, 2020, Virtual Conference*, volume 179 of *LIPIcs*, pages 34:1–34:18. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020. doi:10.4230/LIPIcs.DISC.2020.34.

12 Mohsen Ghaffari and David Wajc. Simplified and space-optimal semi-streaming (2+epsilon)-approximate matching. In Jeremy T. Fineman and Michael Mitzenmacher, editors, *2nd Symposium on Simplicity in Algorithms, SOSA 2019, January 8-9, 2019, San Diego, CA, USA*, volume 69 of *OASICS*, pages 13:1–13:8. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019. doi:10.4230/OASIcs.SOSA.2019.13.

13 Michael Kapralov, Sanjeev Khanna, and Madhu Sudan. Approximating matching size from random streams. In Chandra Chekuri, editor, *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2014, Portland, Oregon, USA, January 5-7, 2014*, pages 734–751. SIAM, 2014. doi:10.1137/1.9781611973402.55.

14 Michael Kapralov, Slobodan Mitrovic, Ashkan Norouzi-Fard, and Jakab Tardos. Space efficient approximation to maximum matching size from uniform edge samples. In Shuchi Chawla, editor, *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms, SODA 2020, Salt Lake City, UT, USA, January 5-8, 2020*, pages 1753–1772. SIAM, 2020. doi:10.1137/1.9781611975994.107.

15 Zvi Lotker, Boaz Patt-Shamir, and Seth Pettie. Improved distributed approximate matching. *J. ACM*, 62(5):38:1–38:17, 2015. doi:10.1145/2786753.

16 A. McGregor and S. Vorotnikova. Planar matching in streams revisited. In *Proceedings of the 19th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems (APPROX)*, 2016.

17 Andrew McGregor. Finding graph matchings in data streams. In Chandra Chekuri, Klaus Jansen, José D. P. Rolim, and Luca Trevisan, editors, *Approximation, Randomization and Combinatorial Optimization, Algorithms and Techniques, 8th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems, APPROX 2005 and 9th InternationalWorkshop on Randomization and Computation, RANDOM 2005, Berkeley, CA, USA, August 22-24, 2005, Proceedings*, volume 3624 of *Lecture Notes in Computer Science*, pages 170–181. Springer, 2005. doi:10.1007/11538462_15.

18 Andrew McGregor and Sofya Vorotnikova. A simple, space-efficient, streaming algorithm for matchings in low arboricity graphs. In *1st Symposium on Simplicity in Algorithms, SOSA 2018, January 7-10, 2018, New Orleans, LA, USA*, pages 14:1–14:4, 2018. doi:10.4230/OASIcs.SOSA.2018.14.

19 C. Nash-Williams. Decomposition of finite graphs into forests. *J. London Math. Soc.*, 39(12), 1964.

20 Jeffrey Scott Vitter. Random sampling with a reservoir. *ACM Trans. Math. Softw.*, 11(1):37–57, 1985. doi:10.1145/3147.3165.