Maximum Matchings in Dynamic Graph Streams and the Simultaneous Communication Model

Sepehr Assadi

University of Pennsylvania

Joint work with Sanjeev Khanna (Penn), Yang Li (Penn), and Grigory Yaroslavtsev (Penn)
Matchings in Graphs

- **Matching**: A collection of vertex-disjoint edges.

- **Maximum Matching problem**: Find a matching with a largest number of edges.
Matchings in Graphs

Maximum matching is a fundamental problem with many applications.

In this talk, we focus on matchings in two related models:

- Dynamic graph streams
- Simultaneous communication model
Dynamic Graph Streams

- The input graph is presented as a sequence of edge insertions and deletions.

Stream:

Edge-frequency vector:

$$\mathbf{f} = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$$
Dynamic Graph Streams

- The input graph is presented as a sequence of edge insertions and deletions.

Stream: \(+e_1\)

Edge-frequency vector:

\[
\vec{f} = [1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
\]
Dynamic Graph Streams

- The input graph is presented as a sequence of edge insertions and deletions.

Stream: $+e_1, +e_7$

Edge-frequency vector:

$$\vec{f} = [1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0]$$
Dynamic Graph Streams

- The input graph is presented as a sequence of edge insertions and deletions.

Stream: $+e_1, +e_7, +e_{11}$

Edge-frequency vector:

$\vec{f} = [1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0]$
Dynamic Graph Streams

- The input graph is presented as a sequence of edge insertions and deletions.

Stream: \( +e_1, +e_7, +e_{11}, -e_1 \)

Edge-frequency vector:

\[
\overrightarrow{f} = [0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0]
\]
Dynamic Graph Streams

- The input graph is presented as a sequence of edge insertions and deletions.
- Algorithm makes a single pass over the entire input but has only a small space to store information about the input as it passes by.
- At the end of the sequence, the algorithm outputs a solution using the stored information.
Dynamic Graph Streams

Main technique: **Linear sketching**

Linear sketch:

\[
\begin{bmatrix}
M
\end{bmatrix}_{s \times n^2} \cdot \begin{bmatrix}
\vec{f}
\end{bmatrix}_{n^2} = \begin{bmatrix}
\end{bmatrix}_{s}
\]

Algorithm:

1. **Maintain a linear sketch** of the input graph during the stream.
   - When edge \( e_i \) is updated: \( M \cdot (\vec{f} \pm \vec{e}_i) = M \cdot \vec{f} \pm M \cdot \vec{e}_i \)

2. Solve the problem using the sketch at the end.

Dynamic graph stream algorithms and linear sketches are (essentially) equivalent [LNW14, AHW15].
Dynamic Graph Streams

Many different results:

- Connectivity, edge connectivity, minimum spanning tree, spectral sparsification, triangle counting, densest subgraph, ... 
- Most of them have essentially the same space requirement in both insertion-only streams and dynamic graph streams.

An important missing problem is the maximum matching problem.
Matching in Graph Streams

Insertion-only streams:

- Exact computation requires $\Omega(n^2)$ space [FKM$^+$05].
- 2-approximation in $O(n)$ space is trivial but no better than 2-approximation in $o(n^2)$ space is known.
- Beating $\frac{e}{e-1}$-approximation requires $n^{1+\Omega(1/\log \log n)}$ space [Kap13, GKK12].
- Lots and lots of other results: [McG05] [FKM$^+$05] [EKS09] [ELMS11] [GKK12] [KMM12] [Zel12] [AGM12] [AG13b] [Kap13] [GO13] [KKS14] [CS14] [EHL$^+$15] [AG13a] . . .

Dynamic graph streams:

- Prior to our work, no non-trivial results were known for single-pass algorithms.
Our Results in Dynamic Graph Streams

We provide a complete resolution of matchings in dynamic graph streams:

**Theorem (Upper bound)**

For any $0 \leq \epsilon \leq 1/2$, space of $\tilde{O}(n^{2-3\epsilon})$ is sufficient for computing an $n^\epsilon$-approximate maximum matching.
Our Results in Dynamic Graph Streams

We provide a complete resolution of matchings in dynamic graph streams:

**Theorem (Upper bound)**

For any $0 \leq \epsilon \leq 1/2$, space of $\tilde{O}(n^{2-3\epsilon})$ is sufficient for computing an $n^\epsilon$-approximate maximum matching.

**Theorem (Lower bound)**

For any $\epsilon \geq 0$, space of $\tilde{\Omega}(n^{2-3\epsilon})$ is necessary for computing an $n^\epsilon$-approximate maximum matching.
Our Results in Dynamic Graph Streams

We provide a complete resolution of matchings in dynamic graph streams:

**Theorem (Upper bound)**

For any $0 \leq \epsilon \leq 1/2$, space of $\tilde{O}(n^{2-3\epsilon})$ is sufficient for computing an $n^\epsilon$-approximate maximum matching.

**Theorem (Lower bound)**

For any $\epsilon \geq 0$, space of $\tilde{\Omega}(n^{2-3\epsilon})$ is necessary for computing an $n^\epsilon$-approximate maximum matching.

For $\epsilon > 1/2$, $\tilde{O}(n^{1-\epsilon})$ space suffices for an $n^\epsilon$-approximation.
**Recent Related Work**

Two recent results obtained independently and concurrently:

<table>
<thead>
<tr>
<th></th>
<th>Upper bound</th>
<th>Lower bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Kon15]</td>
<td>$\tilde{O}(n^{2-2\epsilon})$</td>
<td>$\tilde{\Omega}(n^{3/2-4\epsilon})$</td>
</tr>
<tr>
<td>[CCE+16]</td>
<td>$\tilde{O}(n^{2-3\epsilon})$ $(\epsilon \leq 1/2)$</td>
<td>-</td>
</tr>
<tr>
<td>This work</td>
<td>$\tilde{O}(n^{2-3\epsilon})$ $(\epsilon \leq 1/2)$</td>
<td>$\tilde{\Omega}(n^{2-3\epsilon})$</td>
</tr>
</tbody>
</table>
Lower Bound for $n^\epsilon$-Approximation

**Theorem (Lower bound)**

For any $\epsilon \geq 0$, space of $\tilde{\Omega}(n^{2-3\epsilon})$ is necessary for computing an $n^\epsilon$-approximate maximum matching.

- We prove the lower bound for linear sketches.
- Combined with the work of [AHW15], this provides a tight lower bound for all dynamic graph stream algorithms.
Our Lower Bound Approach

We prove the lower bound using \textit{simultaneous communication complexity}:

- The input graph is \textit{edge partitioned} between $k$ players $P^1, \ldots, P^k$.
- There exists another party called the \textit{coordinator}, with no input.
- Players \textit{simultaneously} send a message to the coordinator and the coordinator outputs the final matching.
- Communication measure: maximum \# of bits send by any player.
- Players have access to \textit{public randomness}.
Connection to Linear Sketches

If there exists a randomized linear sketch $A$ of size $s$ for a problem $P$, then the randomized simultaneous communication complexity of $P$ is at most $O(s)$.

$$A \cdot x_1$$
$$A \cdot x_2$$
$$A \cdot x_k$$

Coordinator

$$A \cdot x = A \cdot (x_1 + \ldots + x_k)$$

Hence, a communication lower bound in this model implies an identical space lower bound for linear sketching algorithms.
Ruzsa-Szemerédi Graphs

We prove our lower bound using a construction based on Ruzsa-Szemerédi graphs.

**Definition ((r, t)-RS graphs)**

A graph $G(V, E)$ whose edges can be partitioned into $t$ induced matchings of size $r$ each.

Example. A $(2, 4)$-RS graph on 8 vertices:
Ruzsa-Szemerédi Graphs

We prove our lower bound using a construction based on Ruzsa-Szemerédi graphs.

**Definition** \((r, t)-\text{RS graphs}\)

A graph \(G(V, E)\) whose edges can be partitioned into \(t\) induced matchings of size \(r\) each.

\[1\] **Example.** A \((2, 4)\)-RS graph on 8 vertices:
Ruzsa-Szemerédi Graphs

We prove our lower bound using a construction based on Ruzsa-Szemerédi graphs.

Definition ($((r, t))$-RS graphs)

A graph $G(V, E)$ whose edges can be partitioned into $t$ induced matchings of size $r$ each.

Example. A $(2, 4)$-RS graph on 8 vertices:
Ruzsa-Szemerédi Graphs

We prove our lower bound using a construction based on Ruzsa-Szemerédi graphs.

**Definition** \( ((r, t)\text{-RS graphs}) \)

A graph \( G(V, E) \) whose edges can be partitioned into \( t \) induced matchings of size \( r \) each.

Example. A \( (2, 4)\)-RS graph on 8 vertices:
Ruzsa-Szemerédi Graphs

We prove our lower bound using a construction based on Ruzsa-Szemerédi graphs.

**Definition \((r, t)\)-RS graphs**

A graph \(G(V, E)\) whose edges can be partitioned into \(t\) induced matchings of size \(r\) each.

Example. A \((2, 4)\)-RS graph on 8 vertices:
Ruzsa-Szemerédi Graphs

How dense a graph with many large induced matching can be?

**Theorem ([AMS12])**

There exists an $(r, t)$-RS graph on $N$ vertices and $\Omega(N^2)$ edges with $t = N^{1+o(1)}$ induced matchings of size $r = N^{1-o(1)}$. 
$n^{2-3\epsilon-o(1)}$ Lower Bound - Distribution

- Parameters:
  \[ k \approx n^\epsilon, \quad r = n^{1-\epsilon-o(1)}, \quad t = n^{1-\epsilon} \]

- Each of the $k$ players is given an $(r, t)$-RS graph on $n^{1-\epsilon}$ vertices.

Local view of $P^i$
$n^{2-3\epsilon-o(1)}$ Lower Bound - Distribution

- Parameters:
  \[ k \approx n^\epsilon, \quad r = n^{1-\epsilon-o(1)}, \quad t = n^{1-\epsilon} \]

- Each of the $k$ players is given an $(r, t)$-RS graph on $n^{1-\epsilon}$ vertices.

- One induced matching (red edges) of each player’s graph is special.
$n^{2-3\epsilon-o(1)}$ Lower Bound - Distribution

- Parameters:
  
  $$k \approx n^\epsilon, r = n^{1-\epsilon-o(1)}, t = n^{1-\epsilon}$$

- Each of the $k$ players is given an $(r, t)$-RS graph on $n^{1-\epsilon}$ vertices.

- One induced matching (red edges) of each player’s graph is special.

- Across the players, vertices in the special matchings are unique, while other vertices are shared.

Global view
$n^{2-3\epsilon-o(1)}$ Lower Bound - Distribution

\[ P_1 : \approx n^{1-\epsilon} \]

\[ P_2 : \approx n^{1-\epsilon} \]
Lower Bound - Analysis

Parameters:

\[ k \approx n^\epsilon, \, r = n^{1-\epsilon-o(1)}, \, t = n^{1-\epsilon} \]

- Special matchings are necessary for any large matching.
$n^{2 - 3\epsilon - o(1)}$ Lower Bound - Analysis

Parameters:

$$k \approx n^\epsilon, r = n^{1-\epsilon-o(1)}, t = n^{1-\epsilon}$$

- Special matchings are necessary for any large matching.
- To obtain $o(n^{2-3\epsilon})$ communication, the players have to compress their graph by an $\Omega(n^\epsilon)$ factor.
$n^{2-3\epsilon-o(1)}$ Lower Bound - Analysis

Parameters:

$$k \approx n^\epsilon, r = n^{1-\epsilon-o(1)}, t = n^{1-\epsilon}$$

- Special matchings are **necessary** for any large matching.
- To obtain $o(n^{2-3\epsilon})$ communication, the players have to compress their graph by an $\Omega(n^\epsilon)$ factor.
- Players are **oblivious** to the identity of their special matching.

Global view
\( n^{2-3\epsilon-o(1)} \) Lower Bound - Analysis

Parameters:

\[
k \approx n^\epsilon, \quad r = n^{1-\epsilon-o(1)}, \quad t = n^{1-\epsilon}
\]

- Special matchings are necessary for any large matching.
- To obtain \( o(n^{2-3\epsilon}) \) communication, the players have to compress their graph by an \( \Omega(n^\epsilon) \) factor.
- Players are oblivious to the identity of their special matching.

Conclusion: Assuming each player sends only \( o(n^{2-3\epsilon}) \) bits, the coordinator cannot output a large enough matching.
Space of $\tilde{O}(n^{2-3\epsilon})$ is both sufficient and necessary for computing an $n^\epsilon$-approximate maximum matching in dynamic graph streams.
Vertex-partition model:

1. The input graph is a bipartite graph $G(L, R, E)$.
2. Each player receives a subset of vertices in $L$ with all neighboring edges.
Vertex-Partition Model

The special case of this problem where $k = n$ has been recently studied in [DNO14]:

- $O(\sqrt{n})$ approximation can be obtained with $\tilde{O}(n)$ total communication and this bound is also tight.
Our Results in Vertex-Partition Model

**Theorem (Upper bound)**

There exists a $\sqrt{k}$-approximation algorithm with total communication of $\tilde{O}(n)$.

Generalizes to tight bounds for $\alpha$-approximation for any $\sqrt{k} \leq \alpha \leq k$. 
Our Results in Vertex-Partition Model

Theorem (Upper bound)

There exists a $\sqrt{k}$-approximation algorithm with total communication of $\tilde{O}(n)$.

Generalizes to tight bounds for $\alpha$-approximation for any $\sqrt{k} \leq \alpha \leq k$.

Theorem (Lower bound)

Any $o(\sqrt{k})$-approximation algorithm requires $n^{1+\Omega(1/\log \log n)}$ total communication.
Each player:

1. Picks independently a random permutation $\pi$ of vertices in $R$. 

\[ \begin{array}{c}
L & R \\
\text{1} & \text{2} \\
\text{6} & \\
\text{1} & \text{6} \\
\text{3} & \\
\text{5} & \\
\text{4} & \\
\end{array} \]
Each player:

1. Picks independently a random permutation $\pi$ of vertices in $R$.
2. Computes a maximal matching using ordering of $\pi$. 

\[ L \quad R \]

\[
\begin{aligned}
&\text{2} \\
&\text{6} \\
&\text{1} \\
&\text{3} \\
&\text{5} \\
&\text{4} \\
\end{aligned}
\]
Each player:

1. Picks independently a random permutation $\pi$ of vertices in $R$.
2. Computes a maximal matching using ordering of $\pi$.
$\sqrt{k}$-Approximation in Vertex-Partition Model

Each player:

1. Picks independently a random permutation $\pi$ of vertices in $R$.

2. Computes a maximal matching using ordering of $\pi$. 

---

Sepehr Assadi (Penn)

Symposium on Discrete Algorithms
Each player:

1. Picks independently a random permutation $\pi$ of vertices in $R$.
2. Computes a maximal matching using ordering of $\pi$. 

```
L  R
1  2
6  1
3  3
5  4
```
\( \sqrt{k} \)-Approximation in Vertex-Partition Model

Each player:
1. Picks independently a random permutation \( \pi \) of vertices in \( R \).
2. Computes a maximal matching using ordering of \( \pi \).
3. Sends the maximal matching to the coordinator.

Coordinator: Computes a maximum matching of received edges.
Conclusion and Open Problems

Streaming model:

- **Tight** bounds for matchings in dynamic graph streams.
- **Open question:** Can we improve the trivial $2$-approximation algorithm for matchings in insertion-only streams?

Communication model:

- $\sqrt{k}$-approximation algorithm for matchings with simultaneous communication complexity of $\tilde{O}(n)$ and no $o(\sqrt{k})$-approximation in $\tilde{O}(n)$ communication.
- **Open question:** Can we achieve better than $\sqrt{k}$ approximation in $\tilde{O}(n)$ communication if we allow constant rounds of interaction?
Questions?


Yuqing Ai, Wei Hu, and David P. Woodruff. Additive error norm approximation and new characterizations in turnstile streams.


Michael Crouch and Daniel S. Stubbs.
Improved streaming algorithms for weighted matching, via unweighted matching.


Hossein Esfandiari, Mohammad Taghi Hajiaghayi, Vahid Liaghat, Morteza Monemizadeh, and Krzysztof Onak. Streaming algorithms for estimating the matching size in planar graphs and beyond.

Sepehr Assadi (Penn)
Bipartite graph matchings in the semi-streaming model.


Ashish Goel, Michael Kapralov, and Sanjeev Khanna. On the communication and streaming complexity of maximum bipartite matching.
Venkatesan Guruswami and Krzysztof Onak.
Superlinear lower bounds for multipass graph processing.

Michael Kapralov.
Better bounds for matchings in the streaming model.

Michael Kapralov, Sanjeev Khanna, and Madhu Sudan.
Approximating matching size from random streams.


Andrew McGregor. Finding graph matchings in data streams.
Mariano Zelke.
Weighted matching in the semi-streaming model.