Matching in randomly ordered Streams
Workshop on Algorithms for Data Streams 2012

Christian Konrad
christian.konrad@liafa.jussieu.fr
joint work with
Frédéric Magniez and Claire Mathieu

LIAFA
University Paris Diderot
Graph Streams and Bipartite Matching

\[ G = (A, B, E) \text{ bipartite}, \ n = |A| = |B|, \ m = |E| \]

**Graph stream**: sequence of edges, any order
\[ \pi = (3, 2), (7, 6), (1, 2), (7, 8), \ldots (5, 6) \]

**Bipartite Matching in Streaming**: perform one pass, compute large matching using *little* space

**Memory considerations**: [Feigenbaum, Kannan, Mcgregor, Suri, Zhang, SODA 2005]
deciding basic graph properties such as bipartiteness and connectivity requires \( \Omega(n) \) space

*Semi-Streaming Model*: \( O(n \polylog n) \) space

**From now on:**

\( M^* \): fixed maximum matching (matching of maximal size)
Simplification: graph has perfect matching (all vertices matched)
**Input sequence:** No assumption on the order

**Upper Bound:** $\frac{1}{2}$-approximation, Greedy Algorithm
- start with empty matching, insert incoming edge if possible
- **Example:** $\pi = (2, 3), (1, 2), (3, 4)$

```
1 --- 2 --- 3 --- 4
```

Greedy($\pi$) = \{(2, 3)\} \quad $M^*$ = \{(1, 2), (3, 4)\}

- **Maximal matchings:** cannot be enlarged by simply adding an edge
  - Maximal matchings are of size at least $\frac{1}{2} |M^*|$
  - Greedy computes a maximal matching $\rightarrow \frac{1}{2}$ approximation

**Lower Bound:** [Kapralov, 2012] $1 - \frac{1}{e} \approx 0.63$
Adversarial Arrival Order

**Input sequence:** No assumption on the order

**Upper Bound:** $\frac{1}{2}$-approximation, Greedy Algorithm
- start with empty matching, insert incoming edge if possible
- **Example:** $\pi = (2, 3), (1, 2), (3, 4)$

![Graph representation of example](image)

Greedy($\pi$) = \{(2, 3)\}  \hspace{1cm} M^* = \{(1, 2), (3, 4)\}

- **Maximal matchings:** cannot be enlarged by simply adding an edge
  - Maximal matchings are of size at least $\frac{1}{2}|M^*|$
  - Greedy computes a maximal matching $\rightarrow \frac{1}{2}$ approximation

**Lower Bound:** [Kapralov, 2012] $1 - \frac{1}{e} \approx 0.63$

**Open question:** Can we break $\frac{1}{2}$ in one pass?
Adversarial Arrival Order

**Input sequence:** No assumption on the order

**Upper Bound:** $\frac{1}{2}$-approximation, Greedy Algorithm
- start with empty matching, insert incoming edge if possible
- **Example:** $\pi = (2, 3), (1, 2), (3, 4)$

\[ \begin{array}{c}
1 \quad 2 \quad 3 \quad 4 \\
\end{array} \]

Greedy($\pi$) = \{(2, 3)\} \quad M^* = \{(1, 2), (3, 4)\}

- **Maximal matchings:** cannot be enlarged by simply adding an edge
  - Maximal matchings are of size at least $\frac{1}{2} |M^*|$
  - Greedy computes a maximal matching $\rightarrow \frac{1}{2}$ approximation

**Lower Bound:** [Kapralov, 2012] $1 - \frac{1}{e} \approx 0.63$

**Open question:** Can we break $\frac{1}{2}$ in one pass?

**Two passes:** [Konrad, Magniez, Mathieu, APPROX 2012] $\frac{1}{2} + 0.019$ approx.
**Input sequence:** Edges sorted with respect to incident $A$ node

**Upper Bounds:** $1 - \frac{1}{e}$ approximation

- [Karp, Vazirani, Vazirani, STOC 1990]
  - Online Algorithm: upon arrival of a node with its edges, match node irrevocably
  - Rank $B$ nodes randomly, match $A$ node to free $B$ node with highest rank

- [Goel, Kapralov, Khanna, SODA 2012]
  - Deterministic Algorithm achieving same approximation

**Lower Bound:** [Kapralov, 2012] $1 - \frac{1}{e}$
**Input sequence:** Edges come in in uniform random order

**Upper Bound:** [Konrad, Magniez, Mathieu, APPROX 2012]  
\[ \frac{1}{2} + 0.005 \text{ approximation in expectation} \]

- Random Arrival Order allows to break \( \frac{1}{2} \)
- *randomized Greedy Algorithm*

**Analysis of Greedy Matching Algorithms:**

- **Another Greedy Algorithm:** choose randomly some vertex, and then randomly an incident edge
  
  - [Aronson, Dyer, Frieze, Suen, 1995] \( \frac{1}{2} + 0.0000025 \) approximation
  
  - [Poloczek, Szegedy, FOCS 2012] \( \frac{1}{2} + 0.0039 \) approximation
Some Intuition: Hard Instance for Greedy

\[ G = (A, B, E), \quad |A| = |B| = N \]

Analysis:

- Perfect matching \(|M^*| = N\)
- Greedy: \(|E_\pi \text{ Greedy}(\pi)| = \frac{1}{2}N\)
Some Intuition: Hard Instance for Greedy

\( G = (A, B, E), |A| = |B| = N \)

**Analysis:**
- Perfect matching \(|M^*| = N\)
- Greedy: \(|E_\pi \text{Greedy}(\pi)| = \frac{1}{2}N\)

**Structure of small maximal matchings (\(\approx \frac{1}{2}\)-approximations):**

Almost all edges form 3-augmenting paths with optimal edges
Three passes Streaming Algorithm

Maximal matching $M_0$: Greedy

Left wings $M_1$: Greedy between $A(M_0)$ and $B(M_0)$

Right wings $M_2$: Greedy between $•$ and $A(M_0)$

Augment $M_0$ with $M_1$ and $M_2$

Can we implement this strategy with less passes?

Difficulty: highly linear approach

$M_1$ depends on $M_0$, $M_2$ depends on $M_1$ and $M_0$
Maximal matching $M_0$: Greedy
Three passes Streaming Algorithm

1. Maximal matching $M_0$: Greedy
2. Left wings $M_1$: Greedy between $A(M_0)$ and $B \setminus B(M_0)$
Three passes Streaming Algorithm

- Maximal matching $M_0$: Greedy
- Left wings $M_1$: Greedy between $A(M_0)$ and $B \setminus B(M_0)$

![Diagram of the algorithm](image-url)
Three passes Streaming Algorithm

Maximal matching $M_0$: Greedy

Left wings $M_1$: Greedy between $A(M_0)$ and $B \setminus B(M_0)$

Right wings $M_2$: Greedy between $\bullet$ and $A \setminus A(M_0)$
Three passes Streaming Algorithm

1. Maximal matching $M_0$: Greedy
2. Left wings $M_1$: Greedy between $A(M_0)$ and $B \setminus B(M_0)$
3. Right wings $M_2$: Greedy between $\bullet$ and $A \setminus A(M_0)$
4. Augment $M_0$ with $M_1$ and $M_2$
Three passes Streaming Algorithm

Maximal matching $M_0$: Greedy

Left wings $M_1$: Greedy between $A(M_0)$ and $B \setminus B(M_0)$

Right wings $M_2$: Greedy between $\bullet$ and $A \setminus A(M_0)$

Augment $M_0$ with $M_1$ and $M_2$

Can we implement this strategy with less passes?

Difficulty: highly linear approach

$M_1$ depends on $M_0$, $M_2$ depends on $M_1$ and $M_0$
**Idea:** split stream into 3 parts, run on each part a pass

**Crucial properties:**
- $M_0$: maximal matching
- Sufficiently many edges for augmentation in second part of the stream (guaranteed by random order assumption)
New Property of Greedy

**Lemma:** If Greedy performs badly then Greedy converges quickly

If $|\mathbb{E}_\pi \text{Greedy}(\pi)| = (\frac{1}{2} + \epsilon)|M^*|$ then

$$|\mathbb{E}_\pi \text{Greedy}(\pi[1, \alpha m])| = \left(\frac{1}{2} - \left(\frac{1}{\alpha} - 2\epsilon\right)\right)|M^*|$$

**Corollary:** ($\alpha = \frac{1}{2}$) $|\mathbb{E}_\pi \text{Greedy}(\pi[1, \frac{1}{2} m])| \geq \frac{1}{2}|M^*|$

**Some Intuition:**

- Greedy performs badly: it misses almost all optimal edges
- Random order assumption: many optimal edges arrive *early*
- Early optimal edges blocked: many non-optimal edges taken early

**Blocks:** $[0, 0.43m], [0.43m, 0.76m], [0.76m, m]$

$\rightarrow \frac{1}{2} + 0.005$ approximation in expectation for random order
Two-pass Algorithm for adversarial order:

- **First pass:** $M_0$ and $M_1$ (Greedy matching + left wings)
- **Second pass:** $M_2$ (right wings)

**Difficulty:** $M_1$ depends strongly on $M_0$:

$$M_1 = \text{Greedy between } A(M_0) \text{ and } B \setminus B(M_0)$$
Another new property of Greedy

Matching subsets of $B$:

- **Lemma**: $\pi$ any input sequence, $B' \subset B$ uniform random sample such that $\forall b \in B : P[b \in B'] = p$. Then:
  \[ \mathbb{E}_{B'} |\text{Greedy}(\pi, G|_{A \times B'})| \geq \frac{p}{1+p}|M^*| \]

- **Intuition**:
  - Graph with perfect matching $M^*$, $B' \subset B$
  - Potential $\phi$: perfect edges in $G|_{A \times B'}$
  - $\mathbb{E}\phi_0 = |M^*|p$
Another new property of Greedy

Matching subsets of $B$:

- **Lemma**: $\pi$ any input sequence, $B' \subset B$ uniform random sample such that $\forall b \in B : P[b \in B'] = p$. Then:

  $$\mathbb{E}_{B'} |\text{Greedy}(\pi, G_{A \times B'})| \geq \frac{p}{1+p}|M^*|$$

- **Intuition**:
  
  - Graph with perfect matching $M^*$, $B' \subset B$
  - Potential $\phi$: perfect edges in $G_{A \times B'}$
    $$\mathbb{E} \phi_0 = |M^*|p$$
  - Consider edge $a'b$ incident to node $b \in B'$
    
    - **Bad case**: $\Delta \phi = 2$ if $b' \in B'$
    - **Good case**: $\Delta \phi = 1$ if $b' \notin B'$
    $$\mathbb{E} \Delta \phi = p \cdot 2 + (1 - p) \cdot 1 = 1 + p$$
Another new property of Greedy

**Matching subsets of** $B$:

- **Lemma:** For any input sequence, $B' \subseteq B$ uniform random sample such that $\forall b \in B : P[b \in B'] = p$. Then:
  \[
  \mathbb{E}_{B'} |\text{Greedy}(\pi, G|_{A \times B'})| \geq \frac{p}{1+p} |M^*|
  \]

- **Intuition:**
  - Graph with perfect matching $M^*$, $B' \subseteq B$
  - Potential $\phi$: perfect edges in $G|_{A \times B'}$
    \[
    \mathbb{E} \phi_0 = |M^*| p
    \]
  - Consider edge $a' b$ incident to node $b \in B'$
    - **Bad case:** $\Delta \phi = 2$ if $b' \in B'$
    - **Good case:** $\Delta \phi = 1$ if $b' \notin B'$
    \[
    \mathbb{E} \Delta \phi = p \cdot 2 + (1 - p) \cdot 1 = 1 + p
    \]
  - Matching size: \# rounds until potential $= 0$

\[
\frac{\mathbb{E} \phi_0}{\mathbb{E} \Delta \phi} = \frac{p}{1+p} |M^*|
\]
Two passes Algorithm

1. Sample \( A' \subset A \) such that \( \Pr[a \in A'] = 0 \).
2. In one pass:
   - \( M_0 = \text{Greedy}(A, B) \)
   - \( M_1 = \text{Greedy}(A', B) \)
3. In one pass: find left wings \( M_2 \) for • nodes (Greedy matching)
4. Augment \( M_0 \) by \( M_1 \cup M_2 \rightarrow \approx 1.2 + 0.019 \) approximation
Two passes Algorithm

Sample $A' \subset A$ such that $Pr[a \in A'] = 0.1 \forall a \in A$
Two passes Algorithm

1. Sample $A' \subset A$ such that $\Pr[a \in A'] = 0.1 \forall a \in A$

2. in one pass: $M_0 = \text{Greedy}(A, B)$ and $M_1 = \text{Greedy}(A', B)$
Two passes Algorithm

1. Sample $A' \subset A$ such that $\Pr[a \in A'] = 0.1 \forall a \in A$
2. In one pass: $M_0 = \text{Greedy}(A, B)$ and $M_1 = \text{Greedy}(A', B)$
Two passes Algorithm

1. Sample $A' \subset A$ such that $\Pr[a \in A'] = 0.1 \forall a \in A$
2. in one pass: $M_0 = \text{Greedy}(A, B)$ and $M_1 = \text{Greedy}(A', B)$
3. in one pass: find left wings $M_2$ for $\bullet$ nodes (Greedy matching)
Two passes Algorithm

1. Sample $A' \subset A$ such that $\Pr[a \in A'] = 0.1 \forall a \in A$
2. in one pass: $M_0 = \text{Greedy}(A, B)$ and $M_1 = \text{Greedy}(A', B)$
3. in one pass: find left wings $M_2$ for $\bullet$ nodes (Greedy matching)
4. augment $M_0$ by $M_1 \cup M_2 \rightarrow \frac{1}{2} + 0.019$ approximation
### Bipartite Matching:

<table>
<thead>
<tr>
<th>Order</th>
<th>Passes</th>
<th>Upper Bound Approx.</th>
<th>Lower Bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adversarial</td>
<td>1 pass</td>
<td>$1/2$</td>
<td>$1 - 1/e$</td>
</tr>
<tr>
<td>Adversarial</td>
<td>2 passes</td>
<td>$1/2 + 0.019$</td>
<td>-</td>
</tr>
<tr>
<td>Vertex Arrival</td>
<td>1 pass</td>
<td>$1 - 1/e$</td>
<td>$1 - 1/e$</td>
</tr>
<tr>
<td>Random</td>
<td>1 pass</td>
<td>$1/2 + 0.005$</td>
<td>-</td>
</tr>
</tbody>
</table>

### Remarks:
- Deterministic 2-passes version for adversarial order
- No upper bounds require randomization
- Presented algorithms extends to general graphs

### Open Problem: Extended vertex arrival order
Extended vertex arrival order

**Input sequence:** Vertices come in together with incident edges to already present vertices

Model captures vertex arrival model:

$B$ vertices arrive first, then $A$ vertices

Can $\frac{1}{2}$ be broken in this model?
Thank you for your attention.

Questions?