Advanced Topics in Theoretical Computer Science

University of BRISTOL

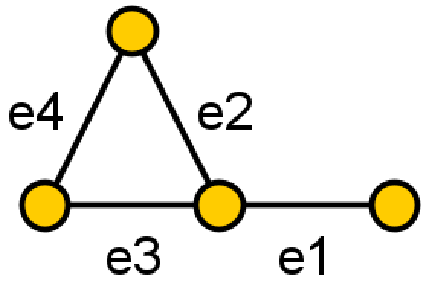
Lecture 13

Matching in Insertion-deletion Streams

Insertion-deletion Streams

Edge-arrival insertion-only Model:

- Stream consists of sequence of edges of a graph

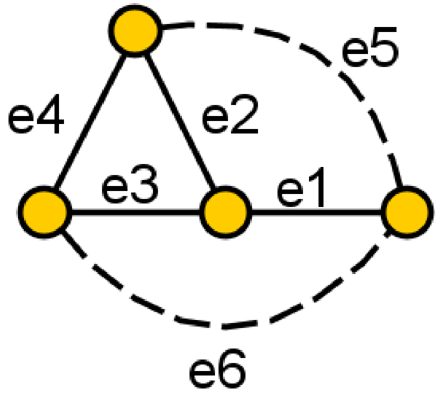


$$S = e_2 e_4 e_3 e_1$$

- All graph algorithms seen so far consider this model

Insertion-deletion (or dynamic) Model: (since 2013)

- Sequence of edge insertions and deletions
- Only inserted edge can be deleted



$$S = e_4 e_3 e_5 \overline{e_5} e_2 e_6 \overline{e_2} e_2 e_1 \overline{e_6}$$

How to Process Insertion-deletion Streams?

Strategies:

- Greedy Algorithm for Matchings clearly fails…
- Even worse: Algorithms that *deterministically* store a set of edges may completely fail since stored edges could later be deleted…

Instead:

- Randomization is crucial
- Linear sketches, in particular, l_0-sampling!

Recap on l_0-sampling

Turnstile stream:

- Stream describes vector $f \in \{-m, \dots, m\}^n$ by updates to its coordinates ($m \in \mathbb{N}$)
- Initially, $f = (0, 0, \dots, 0)$
- Each item in the stream is an update (j, c), meaning $f_j \leftarrow f_j + c$ ($c \in \{-1, 1\}$)

l_0-sampling: [Jowhari, Sağlam, Tardos, 2011**]**

There is a turnstile streaming algorithm with space $O\left(\log^2 n \log \frac{1}{\delta}\right)$ that outputs a uniform random coordinate among the non-zero coordinates of f. It succeeds with proba. $1 - \delta$.

Example: $\boldsymbol{f = (2, -4, 0, 0, 1, 0)}$

Then, the l_0-sampler outputs 1, 2, or 5 each with probability $\frac{1}{3}$ (with success prob. $1 - \delta$).

Insertion-deletion Streams are Turnstile Streams

Insertion-deletion Graph Streams are Turnstile Streams:

- Insertion-deletion graph stream describes vector $f \epsilon \{0,1\}^{\binom{n}{2}}$ (or $f \epsilon [m]^{\binom{n}{2}}$ if multi-edges are allowed, for some integer m, $[m] = \{0, 1, 2, \dots, m\}$)
- l_0-sampling thus corresponds to sampling an edge from the input graph

Other Applications of l_0-sampling in Graph Streams:

By considering substreams of the input stream we can sample from…

- The set of edges incident to a specific vertex
- A random edge in a specific induced subgraph
- …

Error Probability in l_0-sampling

l_0-sampling: [Jowhari, Sağlam, Tardos, 2011**]**

There is a turnstile streaming algorithm with space $O\left(\log^2 n \log\frac{1}{\delta}\right)$ that outputs a uniform random coordinate among the non-zero coordinates of f. It succeeds with proba. $1 - \delta$.

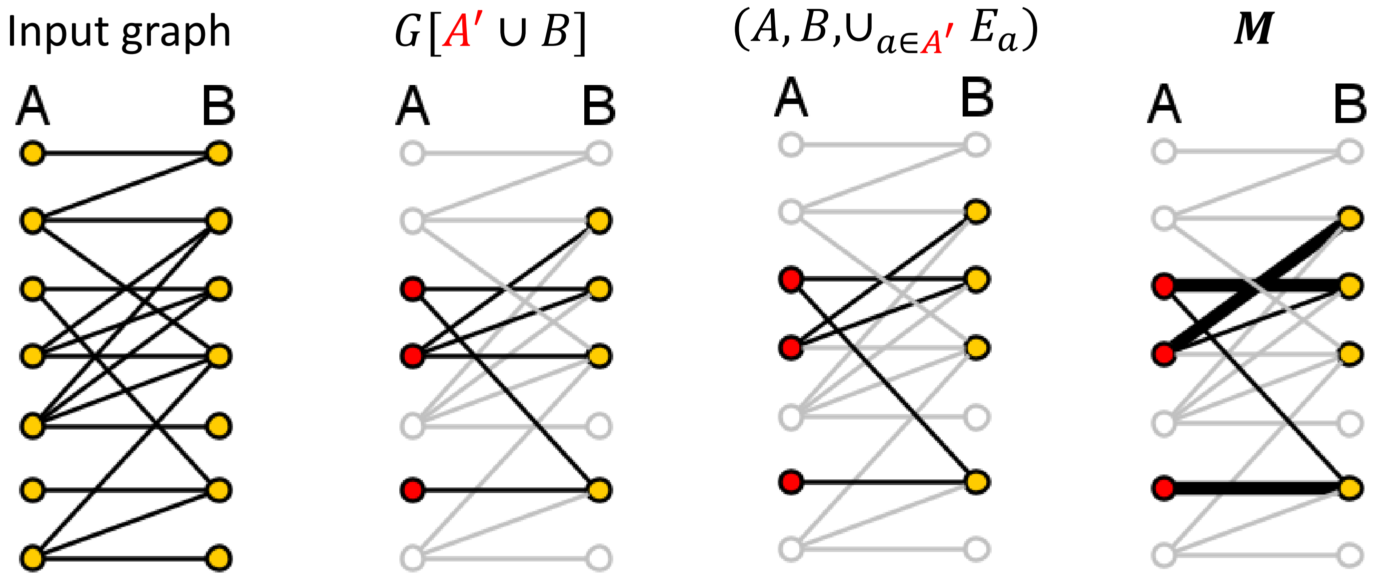
Example:

- Suppose we wish to run an l_0-sampler for each vertex $v \in V$ in order to sample one incident edge to v. Further, our algorithm should be successful with probability $\geq \frac{99}{100}$
- Observe that we are running n l_0-samplers
- We will choose $\delta = \frac{1}{100n}$ in all l_0-samplers since: (union bound)
  Pr[at least one sampler errs] $\leq n \cdot$ Pr[one sampler errs] $= n \cdot \delta = n \cdot \frac{1}{100n} = \frac{1}{100}$
- Each sampler thus requires space $O(\log^2 n \log\frac{1}{\delta}) = O(\log^3 n)$.
- $O(n \log^3 n)$ total space which is semi-streaming space!

Offline Matching Algorithm for Bipartite Graphs

Input: Bipartite Graph $G = (A, B, E)$ with $|A| = |B| = n$ and integer parameter k

1. Sample uniform random subset $A' \subseteq A$ of size k
2. For each $a \in A'$, select arbitrary $\min\{\deg(a), k\}$ edges incident to a
   Let E_a denote this subset
3. Compute a maximum matching $\boldsymbol{M}$ in the graph $(A, B, \cup_{a \in A'} E_a)$
4. Return $\boldsymbol{M}$



Analysis

Lemma. Suppose that G contains a perfect matching (all vertices are matched). Then the matching algorithm on previous slide has an approximation factor of $\frac{k}{2n}$.

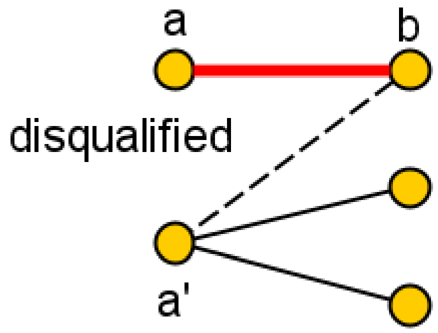
Proof.

- Let M^* be a perfect matching in G
- Let $A_1' \subseteq A'$ be the subset of vertices a such that $|E_a| = \deg(a)$, let $A_2' = A' \setminus A_1'$
- First, suppose that $|A_1'| \geq |A_2'|$ (which implies that $\mathrm{k} \geq |A_1'| \geq \frac{\mathrm{k}}{2}$). Observe that for every $a \in A_1'$, the optimal edge incident to a in M^* is retained in E_a. We hence stored at least $\frac{k}{2}$ edges from M^* and will thus be able to report a matching of at least that size.
- The approximation factor is thus $\frac{\mathrm{k}}{2} / \mathrm{n} = \frac{k}{2n}$

Analysis (2)

- Next, suppose that $|A'_2| > |A'_1|$ (which implies $\mathrm{k} \geq |A'_2| \geq \frac{\mathrm{k}}{2}$). Observe that for each $a \in A'_2$, we have $|E_a| = k$. Consider the graph $H = (A'_2, B, \cup_{a \in A'_2} E_a)$. Then the degree of every A'_2 vertex is k.
- We will show that H contains a matching of size $|A'_2|$:

- Consider the matching $\boldsymbol{M'}$ produced by running Greedy on an arbitrary sequence of the edges of H. Suppose an edge ab is inserted into $\boldsymbol{M'}$. Then, for every $a' \neq a$, this disqualifies at most one edge incident to a' (i.e., the potential edge $a'b$) from being added to M'.
- Since the degree of every $a' \in A'_2$ is k and $|A'_2| \leq k$, we are able to insert an edge incident to every vertex in A'_2.
- Since $|A'_2| \geq \frac{\mathrm{k}}{2}$, the approximation factor is at least $\frac{\mathrm{k}}{2}/\mathrm{n} = \frac{k}{2n}$.



□

Turning the Algorithm into a Streaming Algorithm

Input: Bipartite Graph $G = (A, B, E)$ with $|A| = |B| = n$ and integer parameter k

1. Sample uniform random subset $A' \subseteq A$ of size k
2. For each $a \in A'$, select arbitrary $\min\{\deg(a), k\}$ edges incident to a
   Let E_a denote this subset
3. Compute a maximum matching $\boldsymbol{M}$ in the graph $(A, B, \cup_{a \in A'} E_a)$
4. Return $\boldsymbol{M}$

- Step 1 before processing the stream (pre-processing step)
- Step 3 after processing the stream (post-processing step)

How can we implement step 2 in the insertion-deletion streaming model?

Remaining Task to Solve

Task: For a given vertex $a \in A'$, compute arbitrary $\min\{\deg(a), k\}$ edges while processing the stream

$\boldsymbol{l_0}$-sampling: (on substream of edges incident to a)

- Returns a uniform random edge incident to a
- Strategy: Run enough l_0-samplers to yield $\min\{\deg(a), k\}$ different edges with very high probability
- *Exercise*: If we run $10\, k \log n$ l_0-samplers, then the probability that we do not obtain $\min\{\deg(a), k\}$ different edges is at most $\frac{1}{n^5}$.

Final Insertion-deletion Streaming Algorithm

Insertion-deletion Streaming Algorithm:

Input: Bipartite Graph $G = (A, B, E)$ with $|A| = |B| = n$ and integer parameter k

1. Sample uniform random subset $A' \subseteq A$ of size k
2. While processing the stream: For each $a \in A'$, maintain $10\, k \log n$ l_0-samplers on substream of edges incident to a
3. Compute a maximum matching $\boldsymbol{M}$ among all the edges sampled
4. Return $\boldsymbol{M}$

Space Complexity: # samplers × space complexity of samplers

$$O\left(k \cdot 10\, k \log n \cdot \log^2 n \log \frac{1}{\delta}\right) = O\left(k^2 \log^3 n \cdot \log \frac{1}{\delta}\right)$$

Error Probability and how to chose δ?

Error probability:

- Error is introduced by the l_0-samplers and the possibility that sampling $10\ k \log n$ uniform random edges incident to a vertex a does not yield $\min\{\deg(a), k\}$ different edges
- By the union bound, we can sum up all these error probabilities to bound the total error probability of the algorithm
- $\boldsymbol{k}$ **times:** The probability that $10\ k \log n$ uniform random edges do not yield enough different edges for a specific vertex is at most $1/n^5$.
- $\mathbf{10}\ \boldsymbol{k^2} \mathbf{\log} \boldsymbol{n}$ **times:** l_0-sampler fails with probability δ.

Total error probability: (union bound)

$$k \cdot \frac{1}{n^5} + 10k^2 \log n \cdot \delta \leq \frac{1}{\mathrm{n}^4} + 10\mathrm{n}^3\delta$$

We select $\delta = \frac{1}{10n^5}$. Then total error at most $\frac{1}{n}$.

Final Theorem

Theorem. There is an insertion-deletion streaming algorithm for Maximum Matching with approximation ratio $\frac{k}{2n}$ and space $O(k^2 \log^4 n)$ that fails with probability at most $\frac{1}{n}$.

Corollary. (by setting $\boldsymbol{k = \sqrt{n}}$) There is an insertion-deletion semi-streaming algorithm with approximation ratio $\frac{1}{2\sqrt{n}}$ that fails with probability at most $\frac{1}{n}$.

Summary

- The algorithm presented here is due to [Konrad 2015]
- It is known today that the best insertion-deletion semi-streaming algorithm for maximum matching has approximation ratio $\frac{1}{\Theta(n^{\frac{1}{3}})}$. The algorithm is due to [Assadi et al. 2016] and the optimality proof is due to [Dark, Konrad 2020]

[Konrad, 2015] Christian Konrad: Maximum Matching in Turnstile Streams. ESA 2015: 840-852

[Assadi et al. 2016] Sepehr Assadi, Sanjeev Khanna, Yang Li, Grigory Yaroslavtsev: Maximum Matchings in Dynamic Graph Streams and the Simultaneous Communication Model. SODA 2016: 1345-1364

[Dark, Konrad 2020] Jacques Dark, Christian Konrad: Optimal Lower Bounds for Matching and Vertex Cover in Dynamic Graph Streams. Computational Complexity Conference 2020: 30:1-30:14