Graph Problems in the Semi-Streaming Model

Siddhant Chaudhary

CMI, December 2021

Why the Semi-Streaming Model

- Input: a massive graph $G = (V, E)$, presented as a *stream* of edges.
- Graph problems are difficult in the *streaming model* (polylog space) due to memory limitation.
- The *semi-streaming model* is an interesting area: here, algorithms may use $O(n \cdot \text{polylog } n)$ space.
- Multiple passes of the input stream allowed.
- Authors have studied some classical graph problems in this model.
- Today we will see the *unweighted bipartite matching problem* in this model.

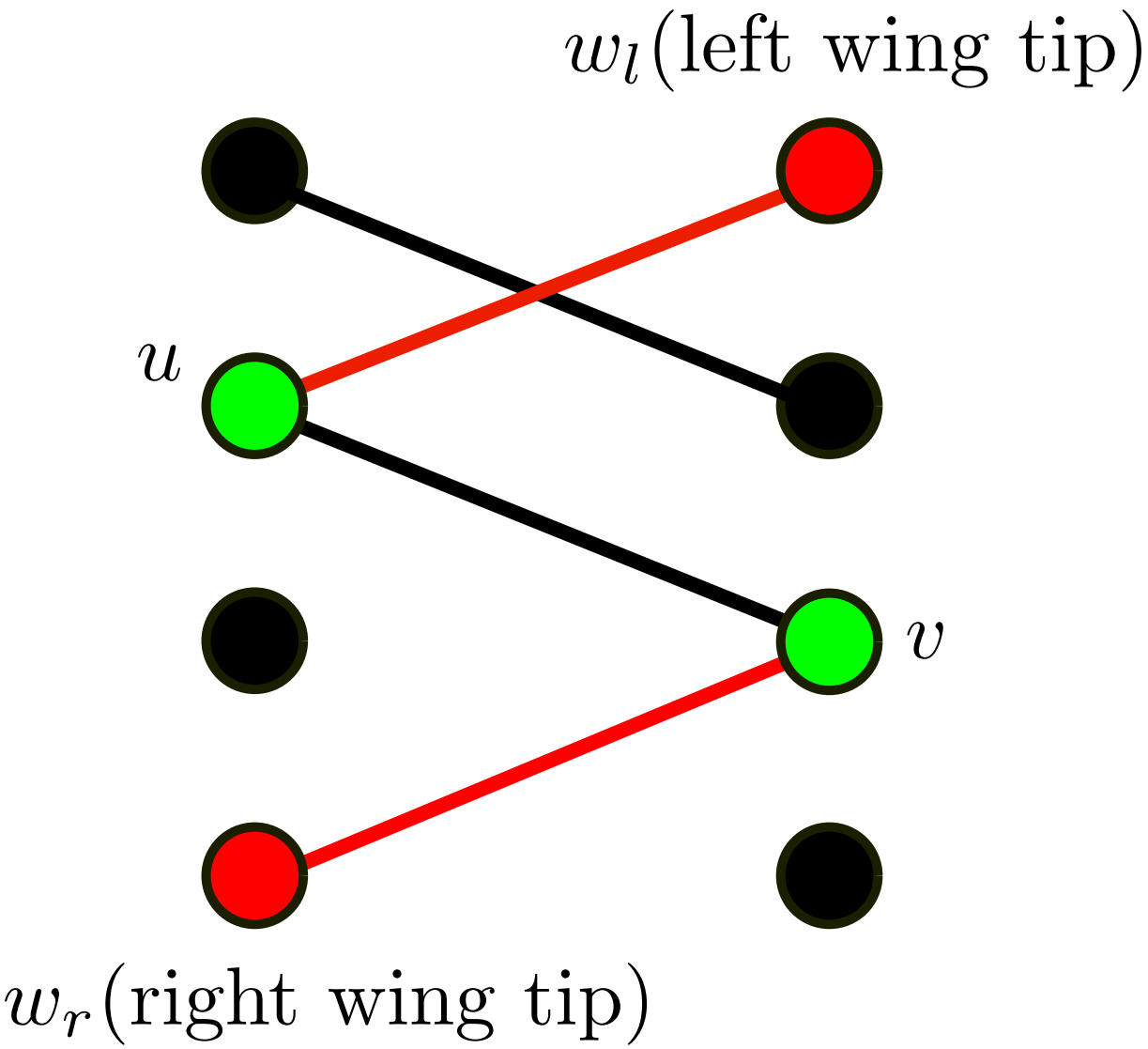
Some notation and preliminaries

- $V = \{v_1, ..., v_n\}$ and $E = \{e_1, ..., e_m\}$.
- A sequence of edges $e_{i_1}, ..., e_{i_m}$ where $e_{i_j} \in E$ and $i_1, ..., i_m \in S_m$. This is called a *graph stream*.
- A *semi-streaming algorithm* computes over a graph stream and uses $O(n \cdot \text{polylog } n)$ space.

Graph Matchings: Some Useful Ideas

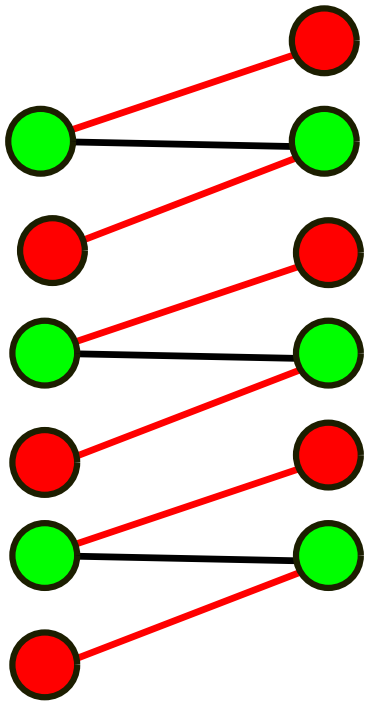
- Let M be a matching.
- A vertex $v \in V$ is said to be *free* w.r.t M if no edge of M is incident on v.
- Given a bipartite graph $G = (L \cup R, E)$ and a matching $M \subseteq E$, a *length-3 augmenting path* for an edge $e = (u, v) \in M$ is defined as follows:
  1. It is a quadruple (w_l, u, v, w_r).
  2. $(w_l, u) \in E$.
  3. $(v, w_r) \in E$.
  4. w_l, w_r are *free*.

Length-3 Augmenting Path



Simultaneous Augmenting Paths

- For a matching M in a bipartite graph, a set of *simultaneously augmentable length-3 augmenting paths* is a set of vertex disjoint length-3 augmenting paths.



Maximal Objects

- A matching M is said to be *maximal* if it is not a proper subset of some other matching.
- Similarly, one can define *maximal* sets of the following form.
  1. A maximal set of vertex disjoint left wings/right wings.
  2. A maximal set of simultaneously augmentable length-3 augmenting paths.
- On the other hand, a *maximum* matching/set of vertex disjoint left wings/right wings/set of simultaneously augmentable length-3 augmenting paths is a such a set with maximum possible cardinality.

Today's Algorithm

- Given a bipartite graph G, we want to approximate the cardinality of a maximum matching in G.
- $0 < \epsilon < \frac{1}{3}$.
- A semi-streaming algorithm to find a $\frac{2}{3} - \epsilon$ approximation of a maximum matching in G.
- Algorithm will use $O(n \log n)$ space.
- Total passes over the stream: $O(\log(1/\epsilon)/\epsilon)$.

Maximal Objects as Approximations

Theorem

Any *maximal matching* is a $\frac{1}{2}-$approximation to a *maximum matching*, i.e

$$M_{\text{maximum}} \leq 2M_{\text{maximal}}$$

Theorem

A *maximal set* of simultaneously augmentable length-3 augmenting paths is a $\frac{1}{3}$-approximation to a *maximum* such set, i.e

$$AP_{\text{maximum}} \leq 3AP_{\text{maximal}}$$

Approximation using augmenting paths

Theorem

1. Let M be a maximal matching for a bipartite graph.
2. Let X be a maximum-sized set of simultaneously augmentable length-3 augmenting paths for M. Let $\alpha = |X|/|M|$.
3. Let OPT be a maximum matching.
4. Then,

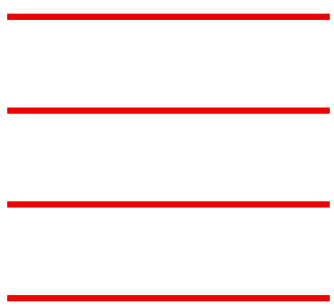
$$|M|(1+\alpha) = |M| + |X| \geq \frac{2}{3}|\text{OPT}|$$

Proof (contd.)

- Consider the symmetric difference $\mathsf{OPT} \Delta M$, and consider the graph induced by this symmetric difference.
- Let C be a connected component of this graph. We claim that

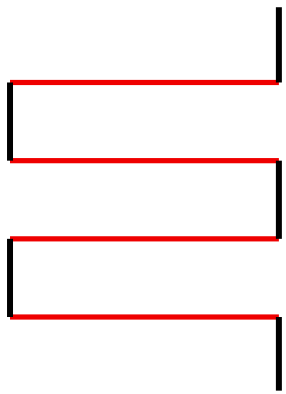
$$\mathsf{OPT}_C \leq M_C + 1 \tag{1}$$

- To prove this, first draw all the edges of C in M_C.



(contd.)

- Now, all other edges of C are edges in OPT_C (and they are not in M_C). Also, recall that no two edges of OPT_C intersect (because OPT_C is a matching).
- This means that *atmost two edges of OPT_C are incident to an edge in M_C.* (if more than two edges are incident to a single edge, two edges of OPT_C will have to intersect).
- So for the connected component C to be indeed connected, our graph must have the following structure (black edges belong to OPT_C, and red ones are in M_C).



(contd.)

- Clearly, $\text{OPT}_C \leq M_C + 1$, and this proves (1). Infact, this also proves that this inequality holds for every connected component in the original graph as well (and not just the symmetric difference). From now on, we will only talk about the original graph.
- There is no connected component which has no edge of M (since M is maximal). So, $M_C \geq 1$ for all connected components C.
- Let S_1 be the set of all connected components with $M_C = 1$ and $\text{OPT}_C = 2$. Let S_2 be the set of all other components. So, for all $C \in S_2$, either $M_C = \text{OPT}_C = 1$ or $M_C \geq 2$.
- Clearly, $|S_1| \leq |X|$ (by the definition of X). Moreover, by inequality (1), $3M_C \geq 2\text{OPT}_C$ for all $C \in S_2$.

(contd.)

- We can complete the proof using the following series of inequalities.

$$
\begin{aligned}
2|\mathsf{OPT}| &= 2\sum_{C \in S_1} |\mathsf{OPT}|_C + 2\sum_{C \in S_2} |\mathsf{OPT}|_C \\
&\leq 4|S_1| + \sum_{C \in S_2} 3|M_C| \\
&\leq 3|S_1| + 3|S_1| + 3\sum_{C \in S_2} |M_C| \\
&\leq 3|X| + 3\sum_{C \in S_1} |M_C| + 3\sum_{C \in S_2} |M_C| \\
&= 3|X| + 3|M|
\end{aligned}
$$

Finding Simultaneous Augmenting Paths

1. Input: $G = (L \cup R, E)$. A matching M for G and a parameter $0 < \delta < 1$.
2. In one pass, find a *maximal set* of disjoint left wings. If number of wings found $\leq \delta M$, terminate.
3. In a second pass, for the edges in M for which left wings were found, find a maximal set of disjoint right wings (At this point, we've found a bunch of disjoint augmenting paths).
4. Remember and *ignore* those vertices which:
    1. are endpoints of a matched edge which got a left wing.
    2. are the wing tips of a matching edge which got both wings.
5. Repeat from step 2.

A claim on the previous algorithm

Theorem

Let $\alpha = |X|/|M|$ be as before. The algorithm in the previous slide finds a *maximal* set of simultaneously augmentable length-3 augmenting paths of size $\geq (\alpha|M| - 2\delta|M|)/3$. Also, it uses atmost $3/\delta$ phases.

- Proof uses the two approximation theorems we saw in the previous slides (maximal matchings and maximal sets of simultaneously augmentable paths).

Main Algorithm

1. Input: $G = (L \cup R, E)$. Accuracy parameter $0 < \epsilon < 1/3$.
2. In one pass, find a maximal matching and a bipartition of G.
3. For $\lceil \log(6\epsilon)/\log(8/9) \rceil$ steps do the following.
    1. Run the previous algorithm (to find simultaneous augmenting paths) with G, M and $\delta = \epsilon/(2 - 3\epsilon)$.
    2. For each edge augmenting path (w_l, u, v, w_r) found, remove (u, v) from M and add (w_l, u) and (v, w_r) to M (i.e augmenting all the augmenting paths found).

Main Theorem

Theorem

- Let $0 < \epsilon < 1/3$.
- Let G be a bipartite graph.
- The above previous algorithm computes a $2/3 - \epsilon$ approximation to a maximum matching of G.
- It takes $O(\log(1/\epsilon)/\epsilon)$ passes of the edge stream.
- Takes $O(n \log n)$ space.

Proof

- Space complexity is easy. Need the following.
  1. Finding a bipartition: $O(n \log n)$ space.
  2. Part of the bipartition to which a vertex belongs: $O(n)$ space.
  3. Storing the matching at each step: $O(n \log n)$ space (matching has size $\leq n$).
  4. Some other auxilliary data: $O(n \log n)$ space.
- Computing the number of passes is easy: there are $k = \lceil \log(6\epsilon)/\log(8/9) \rceil$ steps using the subroutine.
- In each step, subroutine takes the following number of passes.

$$\frac{3}{\delta} = \frac{(6 - 9\epsilon)}{\epsilon}$$

- Just multiply k with above to get the bound.

(contd.)

- In the ith phase, suppose M_i is the matching found. Let X_i be a *maximum* set of simultaneously augmentable length-3 augmenting paths for M_i. So, M_0 is a maximal matching. Claim: each M_i is a maximal matching.
- Let $\alpha_i = |X_i|/|M_i|$.
- Let OPT be a maximum matching for G.
- Define a sequence s_i as follows.

$$s_i = \frac{|M_i|}{|\text{OPT}|} \qquad \forall i$$

- The proof can be broken down to two cases.

(contd.)

- Suppose $\alpha_i \leq \frac{3\epsilon}{2-3\epsilon}$ for some phase i. We claim that M_i is already a $\frac{2}{3} - \epsilon$-approximation to OPT.
  1. By a previous slide, we know that
  $$|M_i|(1+\alpha_i) = |M_i| + |X_i| \geq \frac{2}{3}|\text{OPT}|$$
  2. This implies that
  $$\begin{aligned} |M_i| &\geq \frac{2}{3}\frac{1}{1+\alpha_i}|\text{OPT}| \\ &\geq \frac{2}{3}\left(\frac{1}{1+\frac{3\epsilon}{2-3\epsilon}}\right)|\text{OPT}| \\ &= \left(\frac{2}{3}-\epsilon\right)|\text{OPT}| \end{aligned}$$

(contd.)

- So we can assume that $\alpha_i > \frac{3\epsilon}{2-3\epsilon}$ for all phases i.
- In this case, it can be shown (using the previous mentioned results) that

$$s_{i+1} \geq \frac{8}{9} \cdot s_i + \frac{2}{27}$$

$$s_0 \geq \frac{1}{2} \qquad (M_0 \text{ is maximal.})$$

- Unfold this recurrence to get

$$s_i \geq \frac{2}{3} - \frac{1}{6}\left(\frac{8}{9}\right)^i$$

(contd.)

- By our choice $k = \lceil \log(6\epsilon)/\log(8/9) \rceil$,

$$s_k \geq \frac{2}{3} - \epsilon$$

proving the correctness.