Approximating Semi-matchings in Streaming and in Two-party Communication

ICALP 2013

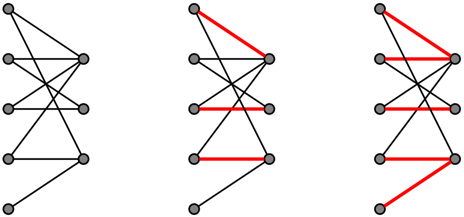
Christian Konrad and Adi Rosén

LIAFA
University Paris Diderot

July 9, 2013

Semi-matchings

Unweighted bipartite graph $G = (A, B, E)$ with $n = |A|$

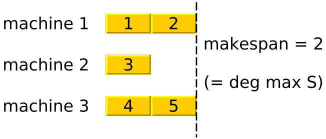


G Matching Semi-matching

Definition:

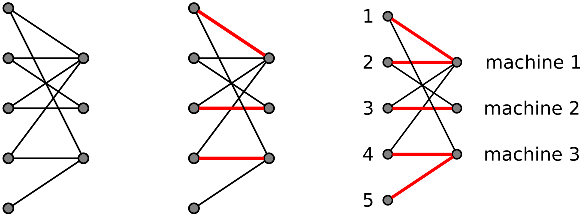
A *Semi-matching* S is a subset of edges $S \subseteq E$ s.t. $\forall a \in A : \deg_S(a) = 1$

Scheduling: Equivalent to scheduling a set of unit length jobs on identical machines with assignment constraints



Semi-matchings

Unweighted bipartite graph $G = (A, B, E)$ with $n = |A|$

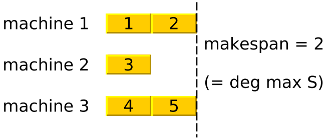


G Matching Semi-matching

Definition:

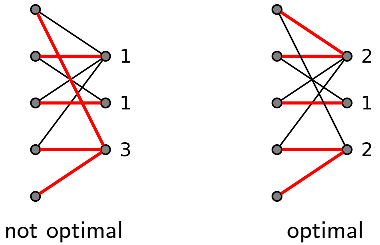
A *Semi-matching* S is a subset of edges $S \subseteq E$ s.t. $\forall a \in A : \deg_S(a) = 1$

Scheduling: Equivalent to scheduling a set of unit length jobs on identical machines with assignment constraints



Optimality of a Semi-matching

Semi-matching S^* is *optimal* if $\forall$ semi-matchings S : $\deg\max S^* \leq \deg\max S$

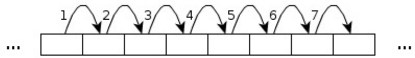


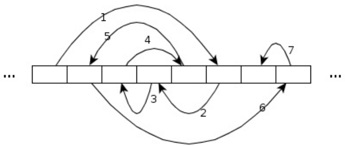
Stronger Notion of Optimality [Harvey et al. WADS 2003]

- Absence of degree-minimizing paths $\iff$
- Minimization of convex cost functions: $\sum_{b \in B} f(\deg_S(b))$

Algorithm: $\mathrm{O}(\sqrt{n}m \log n)$ time [Fakcharoenphol et al. ICALP 2010]

sequential access **random access**





Streaming

- **Objective:** compute some function $f(x_1, \ldots, x_n)$ given only sequential access

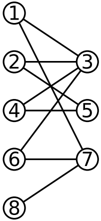
  How much RAM is required for the computation of f?

- **Streaming Complexity:**
  - Number of passes p, usually $\in \mathrm{O}(1)$
  - Memory space (sublinear in the input size)

Computing Semi-matchings in the Streaming Model

$G = (A, B, E)$ bipartite, $n = |A| = |B|$, $m = |E|$

Graph Stream: sequence of edges, any order
$(4, 5), (8, 7), (1, 3), (1, 7), \dots$



Semi-matching Problem in Streaming:
Compute approximate Semi-matching in one pass using space $o(n^2)$

Notion of Approximation: S is a c-approximation if:

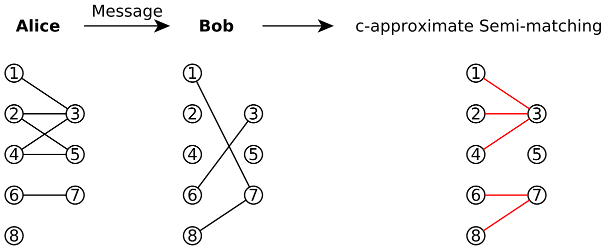
$$\deg \max S \leq c \deg \max S^*$$

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

Semi-streaming Model: $\mathrm{O}(n \operatorname{polylog} n)$ space

Question: **Approximation factor/space trade-off?**

Model: Edge set is split between Alice and Bob

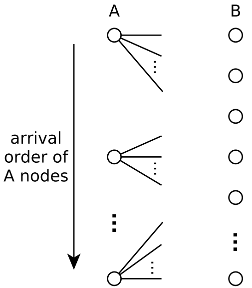


Question: **Approximation factor/message size trade-off?**

Connection to Streaming:
Lower bound on message size for communication protocol is lower bound for space of any streaming algorithm

Online Model

- A vertices arrive with incident edges
- Match incoming A vertex irrevocably to a B vertex



Greedy Algorithm

[Azar, Naor, Rom, SODA 1992]

Match incoming A vertex to B vertex that currently has the minimal degree

$\rightarrow (\lceil \log(n) \rceil + 1)$-competitive (tight)

Streaming with Vertex Arrival Order

- Edges are sorted with respect to their incident A vertex
- Previous Greedy algorithm is a $(\lceil \log(n) \rceil + 1)$-approx. semi-streaming algorithm (using $\tilde{O}(n)$ space) if input stream is in vertex arrival order

Results: Streaming Algorithms

Adversarial Order: no assumption on order of input stream

- **1-pass:** $\forall 0 \leq \epsilon \leq 1$: $\tilde{\mathrm{O}}(n^{1+\epsilon})$ space , $\mathrm{O}(n^{(1-\epsilon)/2})$ approximation

  $\rightarrow \tilde{\mathrm{O}}(n)$ space, $\mathrm{O}(\sqrt{n})$ approximation

- $\log(n)$-**pass:** $\tilde{\mathrm{O}}(n)$ space, $\mathrm{O}(\log n)$ approximation

Vertex Arrival Order: edges sorted with respect to incident A vertex

- **1-pass:** $\tilde{\mathrm{O}}(n)$ space, $\mathrm{O}(\log n)$ approximation **[Azar, Naor, Rom, SODA 1992]**

One-way Two-party Communication Lower Bound

- Deterministic protocols that communicate edges: $\forall \epsilon > 0$: $\mathrm{O}(n^{\frac{1}{(1+\epsilon)c+1}})$ approximation requires cn edges

Results: Streaming Algorithms

Adversarial Order: no assumption on order of input stream

- **1-pass:** $\forall 0 \leq \epsilon \leq 1 : \tilde{\mathrm{O}}(n^{1+\epsilon})$ space , $\mathrm{O}(n^{(1-\epsilon)/2})$ approximation

  $\rightarrow \tilde{\mathrm{O}}(n)$ **space,** $\mathrm{O}(\sqrt{n})$ **approximation**

- **log(n)-pass:** $\tilde{\mathrm{O}}(n)$ space, $\mathrm{O}(\log n)$ approximation

Vertex Arrival Order: edges sorted with respect to incident A vertex

- **1-pass:** $\tilde{\mathrm{O}}(n)$ space, $\mathrm{O}(\log n)$ approximation **[Azar, Naor, Rom, SODA 1992]**

One-way Two-party Communication Lower Bound

- Deterministic protocols that communicate edges: $\forall \epsilon > 0 : \mathrm{O}(n^{\frac{1}{(1+\epsilon)c+1}})$ approximation requires cn edges

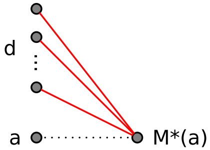
$\rightarrow$ *1-pass $\mathrm{O}(\sqrt{n})$-approx. tight among deterministic alg. that store n edges*
$\rightarrow$ *Vertex arrival order helps*

One-pass Streaming Algorithm

Suppose that $G = (A, B, E)$ contains a perfect matching M^*

Strategy: Run two Greedy algorithms in parallel (Goal: $2d$-Approx.)

1. $S_1 \leftarrow \varnothing$
   for all edges ab in stream **do**
       if $\deg_{S_1}(a) = 0$ **and** $\deg_{S_1}(b) < d$ **then**
           $S_1 \leftarrow S_1 \cup \{ab\}$
   return S_1



2. **for all** $a \in A$: store arbitrary k incident edges to a in set S_2

Analysis

- **Lemma:** $|S_1| \geq \frac{d}{d+1} n$ (at most $\frac{1}{d+1} n$ A vertices unmatched in S_1)
- **Lemma:** If $k = \frac{n}{d(d+1)}$ then unmatched vertices can be matched with maximal degree d using edges of S_2

$\rightarrow$ $2d$-approximation using space $\tilde{O}(nk) = \tilde{O}(n \cdot \frac{n}{d(d+1)})$

Theorem: $\forall \epsilon \geq 0$: one-pass, $O(n^{(1-\epsilon)/2})$-approximation, $\tilde{O}(n^{1+\epsilon})$ space

Alice —Message→ **Bob** ——→ c-approximate Semi-matching

$G_1 = (A, B, E_1)$ $\quad$ $G_2 = (A, B, E_2)$

Upper Bounds

- Alice sends n edges: $\mathrm{O}(\sqrt{n})$ approximation
- Alice sends $2n$ edges: $\mathrm{O}(n^{1/3})$ approximation
- **Conjecture:** Alice sends cn edges: $\mathrm{O}(n^{1/(c+1)})$ approximation

Lower Bounds

- For protocols communicating edges: $\forall \epsilon > 0 : \mathrm{O}(n^{\frac{1}{(1+\epsilon)c+1}})$ approximation requires cn edges
- For any protocol: $\forall \epsilon > 0 : \mathrm{O}(n^{\frac{1}{(1+\epsilon)c+1}})$ approximation requires cn bits

Alice $\xrightarrow{\text{Message}}$ **Bob** $\longrightarrow$ c-approximate Semi-matching

$G_1 = (A, B, E_1)$ $\quad$ $G_2 = (A, B, E_2)$

Strategy: Alice sends a *c-semi-matching skeleton* $S \subseteq E_1$:

$$\forall A' \subseteq A : \deg \max \text{semi}(A', B, S) \leq c \deg \max \text{semi}(A', B, E_1)$$

Lemma: Alice sends c-semi-matching skeleton
$\rightarrow$ Bob outputs $(c+1)$-approximation

Upper Bounds for Skeletons:

- n edges: $\mathrm{O}(\sqrt{n})$-semi-matching skeleton (tight up to a constant)
- $2n$ edges: $\mathrm{O}(n^{1/3})$-semi-matching skeleton (tight up to a constant)
- **Conjecture:** cn edges: $\mathrm{O}(n^{1/(c+1)})$-semi-matching skeleton

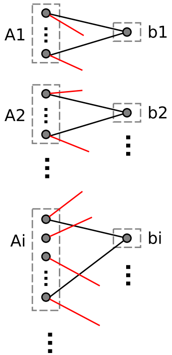
Lower Bound for Skeletons: $\forall \epsilon > 0$: an $\mathrm{O}(n^{\frac{1}{(1+\epsilon)c+1}})$-semi-matching skeleton requires cn edges

Semi-matching Skeletons (2)

Semi-matching Skeleton with n Edges:
Optimal semi-matching is a $(\sqrt{n})$-semi-matching skeleton

Semi-matching Skeleton with $2n$ Edges:

$$\begin{aligned} S &= \text{semi}(A, B, E_1) \\ A_i &= \Gamma_S(b_i) \\ S' &= S \cup \bigcup_i \text{semi}(A_i, B, E_1) \end{aligned}$$



S' is a $(2n^{1/3})$-semi-matching skeleton

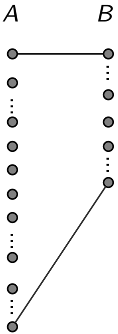
Conjecture:
There is an $\mathrm{O}(n^{1/(c+1)})$-semi-matching skeleton with cn edges

Lower Bound: $\forall \epsilon > 0$: an $\mathrm{O}(n^{\frac{1}{(1+\epsilon)c+1}})$-semi-matching skeleton requires cn edges

Suppose that Alice is allowed to send a set M of n edges

Hard Instances

Alice: edge set E_1 of a complete bipartite graph $K_{n,\sqrt{n}}$

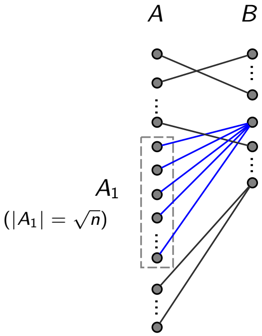


Deterministic Communication Lower Bound

Suppose that Alice is allowed to send a set M of n edges

Hard Instances

Alice: edge set E_1 of a complete bipartite graph $K_{n,\sqrt{n}}$



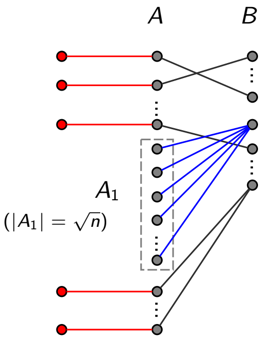
Alice sends n edges M to Bob

Lemma: $\exists A_1 \subset A$ with $|A_1| = \sqrt{n}$ and $|\Gamma_M(A_1)| = 1$ (counting argument)

Suppose that Alice is allowed to send a set M of n edges

Hard Instances

Alice: edge set E_1 of a complete bipartite graph $K_{n,\sqrt{n}}$



Alice sends n edges M to Bob

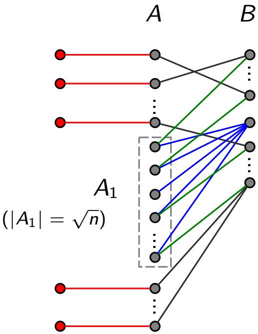
Lemma: $\exists A_1 \subset A$ with $|A_1| = \sqrt{n}$ and $|\Gamma_M(A_1)| = 1$ (counting argument)

Bob: E_2: edges connecting each $a \in A \setminus A_1$ to a new vertex

Suppose that Alice is allowed to send a set M of n edges

Hard Instances

Alice: edge set E_1 of a complete bipartite graph $K_{n,\sqrt{n}}$



Graph has perfect matching (■ + ■):

$$\text{deg max } S^* = 1$$

Output of Bob (■ + ■):

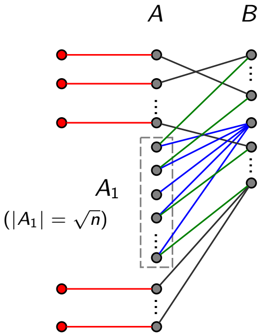
$$\text{deg max semi}(A, B, M \cup E_2) = \sqrt{n}$$

n **edges** $\rightarrow \Omega(\sqrt{n})$**-approximation**

Suppose that Alice is allowed to send a set M of n edges

Hard Instances

Alice: edge set E_1 of a complete bipartite graph $K_{n,\sqrt{n}}$



Graph has perfect matching (■ + ■):

$$\text{deg max } S^* = 1$$

Output of Bob (■ + ■):

$$\text{deg max semi}(A, B, M \cup E_2) = \sqrt{n}$$

n **edges** $\rightarrow \Omega(\sqrt{n})$**-approximation**

Extensions

- cn edges
- edges to bits

Conclusion

Summary:

- Deterministic 1-pass $\tilde{O}(n^{1+\epsilon})$ space streaming algorithms with approximation factor $O(n^{\frac{1-\epsilon}{2}})$
  $\tilde{O}(n)$ space $\rightarrow O(\sqrt{n})$ approximation
- Two optimal deterministic one-way two-party protocols with approximation factors $O(\sqrt{n})$ and $O(n^{1/3})$
- LB on one-way two-party communcation: computing an $O(n^{\frac{1}{(1+\epsilon)c+1}})$ approximation deterministically requires sending cn edges

Open Problems:

- Is $O(\sqrt{n})$-approximation tight for 1-pass streaming with space $\tilde{O}(n)$?
- Does randomization help?

Conclusion

Summary:

- Deterministic 1-pass $\tilde{O}(n^{1+\epsilon})$ space streaming algorithms with approximation factor $O(n^{\frac{1-\epsilon}{2}})$
  $\tilde{O}(n)$ space $\rightarrow$ $O(\sqrt{n})$ approximation
- Two optimal deterministic one-way two-party protocols with approximation factors $O(\sqrt{n})$ and $O(n^{1/3})$
- LB on one-way two-party communcation: computing an $O(n^{\frac{1}{(1+\epsilon)c+1}})$ approximation deterministically requires sending cn edges

Open Problems:

- Is $O(\sqrt{n})$-approximation tight for 1-pass streaming with space $\tilde{O}(n)$?
- Does randomization help?

Thank you for your attention