Maximum Matchings in Dynamic Graph Streams and the Simultaneous Communication Model

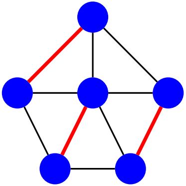
Sepehr Assadi

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

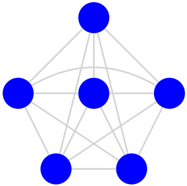
Dynamic Graph Streams

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

Stream:

Edge-frequency vector:

$$\vec{f} = \left[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\right]$$



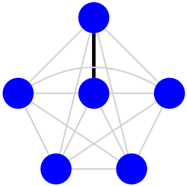
Dynamic Graph Streams

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

Stream: $+e_1$

Edge-frequency vector:

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



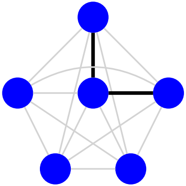
Dynamic Graph Streams

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

Stream: $+e_1, +e_7$

Edge-frequency vector:

$$\vec{f} = \left[1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0\right]$$



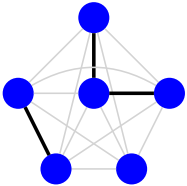
Dynamic Graph Streams

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

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

Edge-frequency vector:

$$\vec{f} = \left[1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0\right]$$



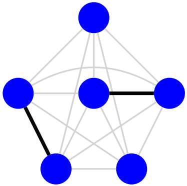
Dynamic Graph Streams

- 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:

$$\vec{f} = \left[0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0\right]$$



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:

$$\begin{bmatrix} & & M & & \end{bmatrix}_{s \times n^2} \cdot \begin{bmatrix} \\ \overrightarrow{f} \\ \\ \end{bmatrix}_{n^2} = \begin{bmatrix} \\ \end{bmatrix}_s$$

Algorithm:

1. Maintain a linear sketch of the input graph during the stream.
   - When edge e_i is updated: $M \cdot (\overrightarrow{f} \pm \overrightarrow{e_i}) = M \cdot \overrightarrow{f} \pm M \cdot \overrightarrow{e_i}$
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^{ϵ}-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^{ϵ}*-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^{ϵ}*-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^{ϵ}-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^{ϵ}-approximate maximum matching.

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

Recent Related Work

Two recent results obtained independently and concurrently:

	Upper bound	Lower bound
[Kon15]	$\tilde{O}(n^{2-2\epsilon})$	$\tilde{\Omega}(n^{3/2-4\epsilon})$
[CCE+16]	$\tilde{O}(n^{2-3\epsilon})$ $(\epsilon \leq 1/2)$	-
This work	$\tilde{O}(n^{2-3\epsilon})$ $(\epsilon \leq 1/2)$ $\tilde{O}(n^{1-\epsilon})$ $(\epsilon > 1/2)$	$\tilde{\Omega}(n^{2-3\epsilon})$

Lower Bound for n^ϵ-Approximation

Theorem (Lower bound)

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

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

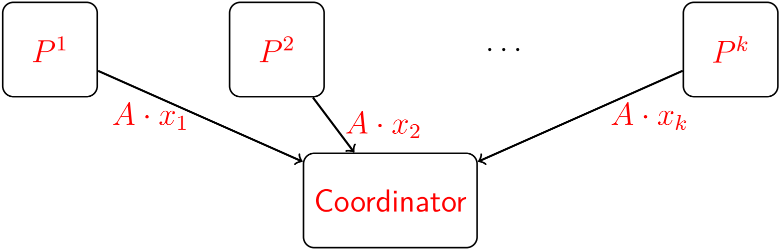
Our Lower Bound Approach

We prove the lower bound using simultaneous communication complexity:

- The input graph is edge partitioned between k players $P^1, \ldots, P^k$.
- There exists another party called the coordinator, with no input.
- Players 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 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 $O(s)$.



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

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

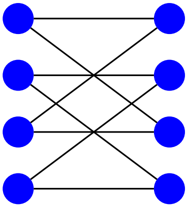
Ruzsa-Szemerédi Graphs

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.

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



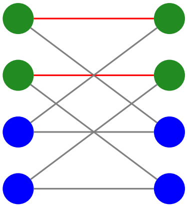
Ruzsa-Szemerédi Graphs

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.

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



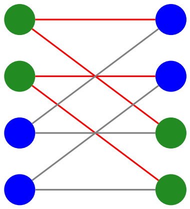
Ruzsa-Szemerédi Graphs

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.

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



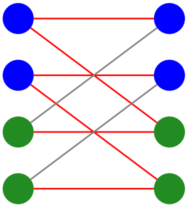
Ruzsa-Szemerédi Graphs

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.

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



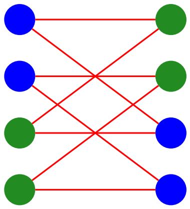
Ruzsa-Szemerédi Graphs

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.

1. 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])

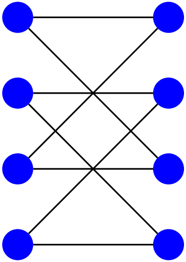
There exists an (r, t)-RS graph on N vertices and $\Omega(N^2)$ edges with $t = N^{1+o(1)}$ induced matchings of size $r = N^{1-o(1)}$.

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

- Parameters:

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

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



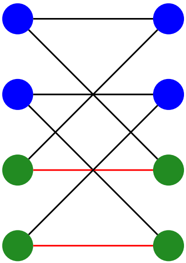
Local view of P^i

$n^{2-3\epsilon-o(1)}$ 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.



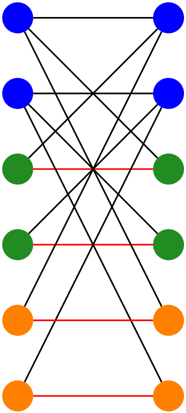
Special matching of P^i

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

- Parameters:

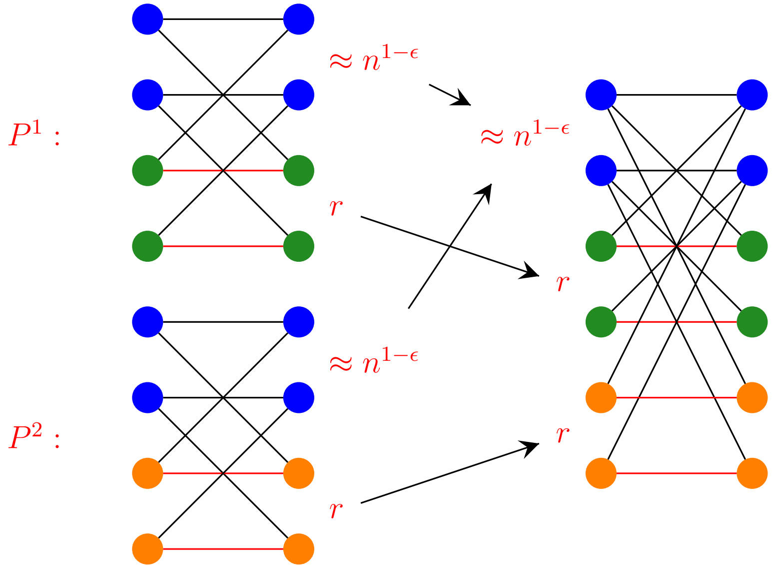
$$k \approx n^{\epsilon}, r = n^{1-\epsilon-o(1)}, t = n^{1-\epsilon}$$

- 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

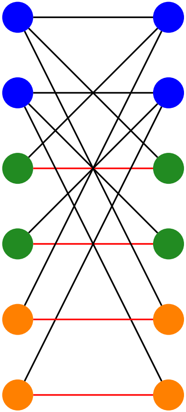


$n^{2-3\epsilon-o(1)}$ Lower Bound - Analysis

Parameters:

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

- Special matchings are necessary for any large matching.



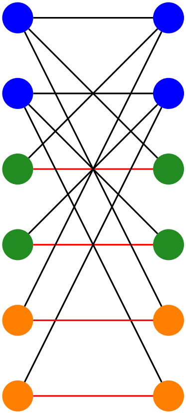
Global view

$n^{2-3\epsilon-o(1)}$ 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.



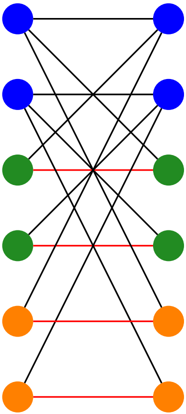
Global view

$n^{2-3\epsilon-o(1)}$ 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

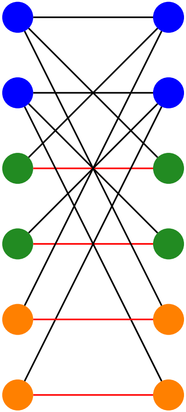
$n^{2-3\epsilon-o(1)}$ Lower Bound - Analysis

Parameters:

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

- 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.

Conclusion: Assuming each player sends only $o(n^{2-3\epsilon})$ bits, the coordinator cannot output a large enough matching.



Global view

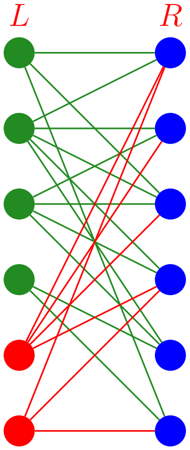
Matching in Dynamic Graph Streams: Wrap-up

Space of $\tilde{O}(n^{2-3\epsilon})$ is both sufficient and necessary for computing an n^{ϵ}-approximate maximum matching in dynamic graph streams.

Back to Simultaneous Communication Model

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 α-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 α-approximation for any $\sqrt{k} \leq \alpha \leq k$.

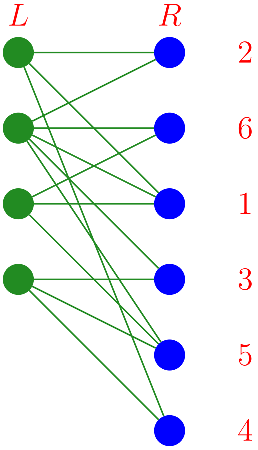
Theorem (Lower bound)

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

$\sqrt{k}$-Approximation in Vertex-Partition Model

Each player:

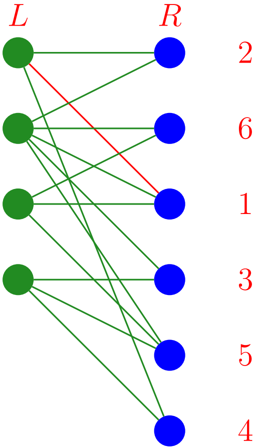
1. Picks independently a random permutation π of vertices in R.



$\sqrt{k}$-Approximation in Vertex-Partition Model

Each player:

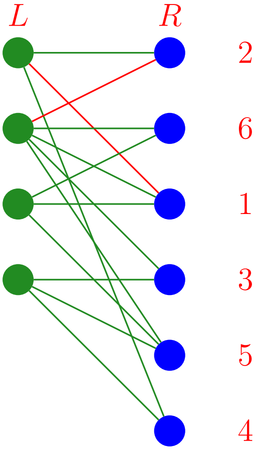
1. Picks independently a random permutation π of vertices in R.
2. Computes a maximal matching using ordering of π.



$\sqrt{k}$-Approximation in Vertex-Partition Model

Each player:

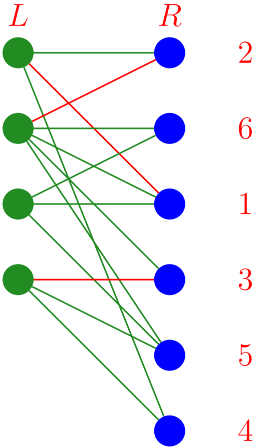
1. Picks independently a random permutation π of vertices in R.
2. Computes a maximal matching using ordering of π.



$\sqrt{k}$-Approximation in Vertex-Partition Model

Each player:

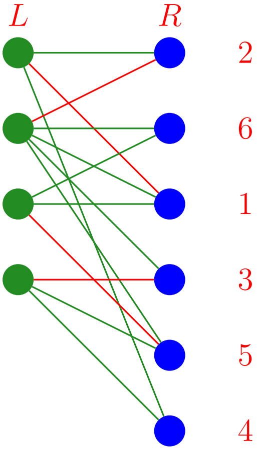
1. Picks independently a random permutation π of vertices in R.
2. Computes a maximal matching using ordering of π.



$\sqrt{k}$-Approximation in Vertex-Partition Model

Each player:

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

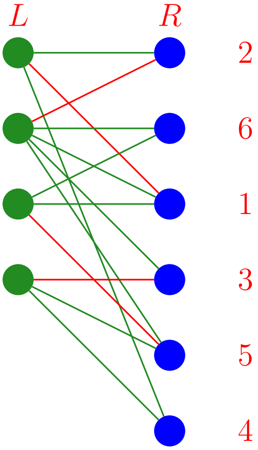


$\sqrt{k}$-Approximation in Vertex-Partition Model

Each player:

1. Picks independently a random permutation π of vertices in R.
2. Computes a maximal matching using ordering of π.
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?

Kook Jin Ahn and Sudipto Guha.
Access to data and number of iterations: Dual primal algorithms for maximum matching under resource constraints.
CoRR, abs/1307.4359, 2013.

Kook Jin Ahn and Sudipto Guha.
Linear programming in the semi-streaming model with application to the maximum matching problem.
Inf. Comput., 222:59–79, 2013.

Kook Jin Ahn, Sudipto Guha, and Andrew McGregor.
Graph sketches: sparsification, spanners, and subgraphs.
In *Proceedings of the 31st ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, PODS*, pages 5–14, 2012.

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

Personal communication, 2015.

Noga Alon, Ankur Moitra, and Benny Sudakov.
Nearly complete graphs decomposable into large induced matchings and their applications.
In *Proceedings of the 44th Symposium on Theory of Computing Conference, STOC*, pages 1079–1090, 2012.

Rajesh Chitnis, Graham Cormode, Hossein Esfandiari, MohammadTaghi Hajiaghayi, Andrew McGregor, Morteza Monemizadeh, and Sofya Vorotnikova.
Kernelization via sampling with applications to finding matchings and related problems in dynamic graph streams.
In *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2016, Arlington, VA, USA, January 10-12, 2016*, pages 1326–1344, 2016.

Michael Crouch and Daniel S. Stubbs.

Improved streaming algorithms for weighted matching, via unweighted matching.
In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM*, pages 96–104, 2014.

Shahar Dobzinski, Noam Nisan, and Sigal Oren.
Economic efficiency requires interaction.
In *Symposium on Theory of Computing, STOC*, pages 233–242, 2014.

Hossein Esfandiari, Mohammad Taghi Hajiaghayi, Vahid Liaghat, Morteza Monemizadeh, and Krzysztof Onak.
Streaming algorithms for estimating the matching size in planar graphs and beyond.
In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 1217–1233, 2015.

Sebastian Eggert, Lasse Kliemann, and Anand Srivastav.

Bipartite graph matchings in the semi-streaming model.
In *Algorithms - ESA 2009, 17th Annual European Symposium*, pages 492–503, 2009.

Leah Epstein, Asaf Levin, Julián Mestre, and Danny Segev.
Improved approximation guarantees for weighted matching in the semi-streaming model.
SIAM J. Discrete Math., 25(3):1251–1265, 2011.

Joan Feigenbaum, Sampath Kannan, Andrew McGregor, Siddharth Suri, and Jian Zhang.
On graph problems in a semi-streaming model.
Theor. Comput. Sci., 348(2-3):207–216, 2005.

Ashish Goel, Michael Kapralov, and Sanjeev Khanna.
On the communication and streaming complexity of maximum bipartite matching.

In *Proceedings of the Twenty-third Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 468–485. SIAM, 2012.

Venkatesan Guruswami and Krzysztof Onak.
Superlinear lower bounds for multipass graph processing.
In *Proceedings of the 28th Conference on Computational Complexity, CCC*, pages 287–298, 2013.

Michael Kapralov.
Better bounds for matchings in the streaming model.
In *Proceedings of the Twenty-Fourth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 1679–1697, 2013.

Michael Kapralov, Sanjeev Khanna, and Madhu Sudan.
Approximating matching size from random streams.
In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 734–751, 2014.

Christian Konrad, Frédéric Magniez, and Claire Mathieu.
Maximum matching in semi-streaming with few passes.
In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques - 15th International Workshop, APPROX, and 16th International Workshop, RANDOM*, pages 231–242, 2012.

Christian Konrad.
Maximum matching in turnstile streams.
In *Algorithms - ESA 2015 - 23rd Annual European Symposium*, pages 840–852, 2015.

Yi Li, Huy L. Nguyen, and David P. Woodruff.
Turnstile streaming algorithms might as well be linear sketches.
In *Symposium on Theory of Computing, STOC*, pages 174–183, 2014.

Andrew McGregor.
Finding graph matchings in data streams.

In *Approximation, Randomization and Combinatorial Optimization, Algorithms and Techniques, 8th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems, APPROX and 9th InternationalWorkshop on Randomization and Computation, RANDOM*, pages 170–181, 2005.

Mariano Zelke.
Weighted matching in the semi-streaming model.
Algorithmica, 62(1-2):1–20, 2012.