2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)

Near-Quadratic Lower Bounds for Two-Pass Graph Streaming Algorithms*

Sepehr Assadi
Department of Computer Science
Rutgers University
Piscataway, NJ, USA
sepehr.assadi@rutgers.edu

Ran Raz
Department of Computer Science
Princeton University
Princeton, NJ, USA
ran.raz.mail@gmail.com

***Abstract*—We prove that any two-pass graph streaming algorithm for the s-t reachability problem in n-vertex directed graphs requires near-quadratic space of $n^{2-o(1)}$ bits. As a corollary, we also obtain near-quadratic space lower bounds for several other fundamental problems including maximum bipartite matching and (approximate) shortest path in undirected graphs.**

Our results collectively imply that a wide range of graph problems admit essentially no non-trivial streaming algorithm even when two passes over the input is allowed. Prior to our work, such impossibility results were only known for single-pass streaming algorithms, and the best two-pass lower bounds only ruled out $o(n^{7/6})$ space algorithms, leaving open a large gap between (trivial) upper bounds and lower bounds.

***Keywords*-Graph streaming; communication complexity; s-t reachability, multi pass streaming lower bounds**

I. Introduction

Graph streaming algorithms process the input graph with n known vertices by making one or a few passes over the sequence of its unknown edges (given in an arbitrary order) and using a limited memory (much smaller than the input size which is $O(n^2)$ for a graph problem). In recent years, graph streaming algorithms and lower bounds for numerous problems have been studied extensively. In particular, we now have a relatively clear picture of the powers and limitations of *single-pass* algorithms. With a rather gross oversimplification, this can be stated as follows:

- The *exact* variant of most graph problems of interest are intractable: There are $\Omega(n^2)$ space lower bounds for maximum matching and minimum vertex cover [1], [2], (directed) reachability and topological sorting [1], [3], [4], shortest path and diameter [1], [5], minimum or maximum cut [6], maximal independent set [7], [8], dominating set [9], [10], and many others.
- On the other hand, *approximate* variants of many graph problems are tractable: There are $O(n \cdot \text{polylog}(n))$ space algorithms (often referred to as *semi-streaming* algorithms) for approximate (weighted) matching and vertex cover [1], [11]–[13], spanner computation and approximation for distance problems [5], [14]–[16], cut or spectral sparsifiers and approximation for cut problems [17]–[20], large independents sets [8], [21], graph coloring [7], [22], and approximate dominating set [9], [10], among others[1].

Recent years have also witnessed a surge of interest in designing *multi-pass* graph streaming algorithms (see, e.g. [1], [4], [23]–[36]); see, e.g., [1], [37] for discussions on practical applications of multi-pass streaming algorithms in particular in obtaining I/O-efficiency. These results suggest that allowing even just one more pass over the input greatly enhances the capability of the algorithms. For instance, while computing the *exact* global or s-t minimum cut in undirected graphs requires $\Omega(n^2)$ space in a single pass [6], perhaps surprisingly, one can solve both problems in only *two* passes with $\widetilde{O}(n)$ and $\widetilde{O}(n^{5/3})$ space, respectively [38] (see also [39] for an $O(\log n)$-pass algorithm for weighted minimum cut). Qualitatively similar separations are known for numerous other problems such as triangle counting [33], [40] (with two passes), approximate matching [2], [23], [26], [35] (with $O(1)$ passes), maximal independent set [7], [8], [41] (with $O(\log\log n)$ passes), approximate dominating set [10], [42], [43] (with $O(\log n)$ passes), and exact shortest path [5], [36] (with $O(\sqrt{n})$ passes).

Despite this tremendous progress, the general picture for the abilities and limitations of multi-pass algorithms is not so clear even when we focus on *two-pass* algorithms. What other problems beside minimum cut admit non-trivial two-pass streaming algorithms? For instance, can we obtain similar results for directed versions of these problems? What about closely related problems such as maximum bipartite matching or not-so-related problems such as shortest path? Currently, none of these problems admit any non-trivial two-pass streaming algorithm, while known lower bounds only rule out algorithms with $o(n^{7/6})$ space [4], [5], [44] leaving a considerable gap between upper and lower bounds (see [45] for a discussion on the current landscape of multi-pass graph streaming lower bounds).

*A full version of the paper including all technical proofs is available on arXiv: https://arxiv.org/abs/2009.01161.

[1]It should be noted that, in contrast, determining the *best* approximation ratio possible for many of these problems have remained elusive and is an active area of research.

2575-8454/20/$31.00 ©2020 IEEE
DOI 10.1109/FOCS46700.2020.00040

A. Our Contributions

We present near-quadratic space lower bounds for *two-pass* streaming algorithms for several fundamental graph problems including reachability, bipartite matching, and shortest path.

Reachability and related problems in directed graphs: We prove the following lower bound for the reachability problem in directed graphs.

Result 1 (Formalized in Theorem 4)**.** *Any two-pass streaming algorithm (deterministic or randomized) that given an n-vertex directed graph $G = (V, E)$ with two designated vertices $s, t \in V$ can determine whether or not s can reach t in G requires $\Omega(\frac{n^2}{2^{\Theta(\sqrt{\log n})}})$ space.*

The reachability problem is one of the earliest problems studied in the graph streaming model [3]. Previously, Henzinger *et al.* [3] and Feigenbaum *et al.* [5] proved an $\Omega(n^2)$ space lower bound for this problem for single-pass algorithms, and Guruswami and Onak [44] gave an $\widetilde{\Omega}_p(n^{1+1/(2p+2)})$ lower bound for p-pass algorithms which translates to $\widetilde{\Omega}(n^{7/6})$ space for two-pass algorithms; this lower bound was recently extended to random-order streams by Chakrabarti *et al.* [4]. Note that the *undirected* version of this problem has a simple $O(n)$ space algorithm in one pass by maintaining a spanning forest of the input graph (see, e.g. [1]).

Using standard reductions, our results in this part can be extended to several other related problems on directed graphs such as estimating number of vertices reachable from a given source or approximating minimum feedback arc set, studied in [3] and [4], respectively.

Matching and cut problems: We have the following lower bound for bipartite matching.

Result 2 (Formalized in Theorem 5)**.** *Any two-pass streaming algorithm (deterministic or randomized) that given an n-vertex undirected bipartite graph $G = (L \sqcup R, E)$ can determine whether or not G has a perfect matching requires $\Omega(\frac{n^2}{2^{\Theta(\sqrt{\log n})}})$ space.*

Maximum matching problem is arguably the most studied problem in the graph streaming model. However, the main focus on this problem so far has been on approximation algorithms and not much is known for exact computation of this problem, beside that it can be done in $\widetilde{O}(k^2)$ space in a single pass where k is size of the maximum matching [46] (for the perfect matching problem, this gives an $O(n^2)$ space algorithm which is the same as storing the entire input). Previously, Feigenbaum *et al.* [1] and Chitnis *et al.* [47] proved an $\Omega(n^2)$ space lower bound for single-pass algorithms for this problem and Guruswami and Onak [44] extended the lower bound to $\widetilde{\Omega}_p(n^{1+1/(2p+2)})$ for p-pass algorithms.

Both the perfect matching problem and the s-t reachability problem are simpler versions of the s-t minimum cut problem in directed graphs. As such, our lower bounds imply that even though the s-t minimum cut problem can be solved in undirected graphs in $\widetilde{O}(n^{5/3})$ space and two passes [38], its directed version requires $n^{2-o(1)}$ space in two passes (for any multiplicative approximation). Previously, Assadi *et al.* [45] proved a lower bound of $\Omega(n^2/p^5)$ for p-pass algorithms for the *weighted* s-t minimum cut problem (with exponential-in-p weights); for the unweighted problem, the previous best lower bound was still $\widetilde{\Omega}_p(n^{1+1/(2p+2)})$.

Shortest path problem: Finally, we also prove a lower bound for the shortest path problem.

Result 3 (Formalized in Theorem 6)**.** *Any two-pass streaming algorithm (deterministic or randomized) that given an undirected graph $G = (V, E)$ and two designated vertices $s, t \in V$, can output the length of the shortest s-t-path in G requires $\Omega(\frac{n^2}{2^{\Theta(\sqrt{\log n})}})$ space. The lower bound continues to hold even for approximation algorithms with approximation ratio better than $9/7$.*

Shortest path problem have also been extensively studied in graph streaming literature. For single-pass streams, the focus has been on maintaining *spanners* (subgraphs that preserve pairwise distances approximately) which allow for obtaining algorithms with different space-approximation tradeoffs [5], [14]–[16] (starting from 2-approximation in $O(n^{3/2})$ space to $O(\log n)$ approximation in $O(n)$ space), which are known to be almost tight [5]. For multi-pass algorithms, $\widetilde{O}(n)$ space algorithms are known for $(1+\varepsilon)$-approximation with $\text{poly}(\log n, \frac{1}{\varepsilon})$ passes [28], [48], and exact algorithms with $O(\sqrt{n})$ passes [36]. On the lower bound front, an $\Omega(n^2)$ space lower bound is known for single-pass algorithms [5] and $\widetilde{\Omega}_p(n^{1+1/2p+2})$ for p-pass algorithms [44]) (for exact answer or even some small approximation $\approx (2p+4)/(2p+2)$); a stronger lower bound of $\Omega(n^{1+1/2p})$ was proven earlier in [5] for algorithms that need to *find* the shortest path itself.

Our results show that a wide range of graph problems including directed reachability, cut and matching, and shortest path problems, admit essentially no non-trivial two-pass streaming algorithms: modulo the $n^{o(1)}$-term in our bounds, the best one could do to is to simply store the entire stream in $O(n^2)$ space and solve the problem at the end using any offline algorithm.

B. Our Techniques

We prove our main lower bound for the s-t reachability problem; the other lower bounds then follow easily from this using standard ideas.

It helps to start the discussion with the lower bounds in [4], [5], [44]. These lower bounds work with *random graphs* wherein s can reach $\Theta(\sqrt{n})$ random vertices S and

t is *independently* reachable from $\Theta(\sqrt{n})$ random vertices T; thus, by Birthday Paradox, s-t reachability can have either answer with constant probability. One then shows that to determine this, the algorithm needs to "find" S and T explicitly. The final part is then to use ideas from *pointer chasing* problems [49]–[53] to prove a lower bound for this task. The particular space-pass tradeoff is then determined as follows: (i) as a streaming algorithm can find the p-hop neighborhood of s and t in p passes (by BFS), S and T need to be $(p+1)$-hop away from s and t; (ii) as we are working with random graphs, to achieve the bound of $O(\sqrt{n})$ on size of S and T, we need the degree of the graph to be $O(n^{1/2(p+1)})$, leading to an $O(n^{1+1/(2p+2)})$ space lower bound for p-pass algorithms. We note that the *limit* of these approaches based on random graphs seem to be $\widetilde{O}(n^{3/2})$; see [4, Section 5.2].

Our lower bound takes a different route and works with "more structured" graphs. We start with proving a *single-pass* streaming lower bound for an "algorithmically easier" variant of the reachability problem. In this problem, we are promised that s can reach a *unique* vertex $s^\star$ chosen uniformly at random from a set U of $n^{1-o(1)}$ vertices and the goal is to "find" this vertex. Previous lower bounds [4], [5], [44] already imply that if our goal was to determine the identity of $s^\star$ *exactly*, we need $\Omega(n^2)$ space. In this paper, we prove a stronger lower bound that an $n^{2-o(1)}$-space single-pass algorithm essentially cannot even change the distribution of $s^\star$ from uniform over U. The proof of this part is based on information theoretic arguments that rely on "embedding" multiple instances of the set intersection problem (see Section III) inside a *Ruzsa-Szemerédi* (RS) graph (see Section II-B), and proving a new lower bound for the set intersection problem.

We remark that our new lower bound for set intersection is related to the recent lower bound of [45] with a subtle technical difference that is explained in Section III and in more details in the full version. We also note that RS graphs have been used extensively for proving graph streaming lower bounds [2], [8], [26], [54]–[56] starting from [2], but this is their first application to the s-t reachability problem.

In the next part of the argument, we construct a family of graphs in which the s-t reachability is determined by existence of a single edge $(s^\star, t^\star)$ in the graph, where $s^\star$ is the unique vertex reachable from s in a large set U and $t^\star$ is the unique vertex that can reach t in a large set W (see Figure 2 for an illustration). By exploiting our lower bound in the first part, we show that a $n^{2-o(1)}$-space algorithm cannot properly "find" the pairs $s^\star$ and $t^\star$ in the first pass. We then argue that this forces the algorithm to effectively "store" all the edges between U and W in the second pass to determine if $(s^\star, t^\star)$ is an edge of the graph, leading to an $n^{2-o(1)}$ space lower bound.

Remark (**More than two passes?**). *The intermediate "simpler" problem we considered in our proofs (part one above) is only hard in one pass (see Section IV) and thus our lower bound proof does not directly go beyond two passes. However, it appears that our techniques can be extended to multi-pass algorithms to prove lower bounds of the type $n^{1+\Omega(1/p)}$ space for p-pass algorithms which are slightly better in terms of dependence on p in the exponent compared to [4], [5], [44]. Nevertheless, as unlike the case for two-pass algorithms, it is no longer clear whether such bounds are the "right" answer to the problems at hand, we opted to not pursue this direction in this paper.*

II. Preliminaries

Notation: For any integer $t \geq 1$, we use $[t] := \{1, \ldots, t\}$. For any k-tuple $X = (X_1, \ldots, X_k)$ and integer $i \in [k]$, we define $X^{<i} := (X_1, \ldots, X_{i-1})$.

Throughout the paper, we use 'sans serif' letters to denote random variables (e.g., A) , and the corresponding normal letters to denote their values (e.g. A). For brevity and to avoid the clutter in notation, in conditioning terms which involve assignments to random variables, we directly use the value of the random variable (with the same letter), e.g., write $\mathsf{B} \mid A$ instead of $\mathsf{B} \mid \mathsf{A} = A$.

For random variables A, B, we use $\mathbb{H}(\mathsf{A})$ and $\mathbb{I}(\mathsf{A} ; \mathsf{B}) := \mathbb{H}(\mathsf{A}) - \mathbb{H}(\mathsf{A} \mid \mathsf{B})$ to denote the Shannon entropy and mutual information, respectively. Moreover, for two distributions μ, ν on the same support, $\|\mu - \nu\|_{\text{tvd}}$ denotes the total variation distance, and $\mathbb{D}(\mu \mid\mid \nu)$ is the KL-divergence. A summary of basic information theory facts that we use in our proofs appear in the full version.

A. Communication Complexity and Information Complexity

We work with the two-party communication model of Yao [57]. See the excellent textbook by Kushilevitz and Nisan [58] for an overview of communication complexity.

Let $P : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$ be a relation. Alice receives an input $X \in \mathcal{X}$ and Bob receives $Y \in \mathcal{Y}$, where (X, Y) are chosen from a distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$. We allow players to have access to both public and private randomness. They communicate with each other by exchanging messages according to some *protocol* π. Each message in π depends only on the private input and random bits of the player sending the message, the already communicated messages, and the public randomness. At the end, one of the players outputs an answer Z such that $Z \in P(X, Y)$. For any protocol π, we use $\Pi := \Pi(X, Y)$ to denote the messages *and* the public randomness used by π on the input (X, Y).

We now define two measures of "cost" of a protocol.

Definition II.1 (Communication cost)**.** *The* communication cost *of a protocol π, denoted by $\mathsf{CC}(\pi)$, is the* worst-case *length of the messages communicated between Alice and Bob in the protocol.*

Definition II.2 (Information cost)**.** *The* information cost *of a protocol* π*, when the inputs* (X, Y) *are drawn from a distribution* $\mathcal{D}$*, is* $\mathsf{IC}_{\mathcal{D}}(\pi) := \mathbb{I}(\mathsf{\Pi} ; \mathsf{X} \mid \mathsf{Y}) + \mathbb{I}(\mathsf{\Pi} ; \mathsf{Y} \mid \mathsf{X})$.

The internal information cost (introduced by [59]; see also [59]–[63]) measures the average amount of information each player learns about the input of the other player by observing the transcript of the protocol. As each bit of communication cannot reveal more than one bit of information, the internal information cost of a protocol lower bounds its communication cost [62].

Communication complexity and streaming: There is a standard connection between the communication cost of any protocol π for a communication problem $P(X, Y)$ and the space of any streaming algorithm that can solve $P(X, Y)$ on a stream $X \circ Y$ (see Proposition VI.1); we use this connection to establish our streaming lower bounds.

B. Ruzsa-Szemerédi Graphs

A graph $G^{\mathsf{RS}} = (V, E)$ is a called an (r, t)-Ruzsa-Szemerédi (RS) graph iff its edge-set E can be partitioned into t *induced matchings* $M_1^{\mathsf{RS}}, \ldots, M_t^{\mathsf{RS}}$, each of size r. We further define an (r, t)-RS *digraph* as a directed bipartite graph $G^{\mathsf{RS}} = (L, R, E)$ obtained by directing every edge of a bipartite (r, t)-RS graph from L to R.

We use the original construction of RS graphs due to Ruzsa and Szemerédi [64] based on the existence of large sets of integers with no 3-term arithmetic progression, proven by Behrend [65]. We note that there are multiple other constructions with different parameters (see, e.g. [2], [66]–[68] and references therein) but the following construction works best for our purpose.

Proposition II.3 ([64])**.** *For infinitely many integers* N*, there are* (r, t)*-RS digraphs with* N *vertices on each side of the bipartition and parameters* $r = \frac{N}{e^{\Theta(\sqrt{\log N})}}$ *and* $t = N/3$*.*

III. A New Lower Bound for the Set Intersection Problem

One key ingredient of our paper is a new lower bound for the set intersection problem, defined formally as follows.

Problem 1 (set-intersection)**.** *The* set-intersection *problem is a two-player communication problem in which Alice and Bob are given sets* A *and* B *from* $[m]$*, respectively, with the promise that there exists a* ***unique*** *element* $e^\star$ *such that* $\{e^\star\} = A \cap B$*. The goal is to find the* ***target element*** $e^\star$ *using back and forth communication (i.e., in the two-way communication model).*

The set-intersection problem is closely related to the well-known *set disjointness* problem. It is in fact straightforward to prove an $\Omega(m)$ lower bound on the communication complexity of set-intersection using a simple reduction from the set disjointness problem. However, in this paper, we are interested in an algorithmically simpler variant of this problem which we define below.

Definition III.1. *Let* $\mathcal{D}$ *be a distribution of inputs* (A, B) *for* set-intersection *(known to both players). A protocol* π ***internal*** ε***-solves*** set-intersection *over* $\mathcal{D}$ *iff* <u>*at least*</u> *one of the following holds:*

$$\underset{\mathsf{\Pi},\mathsf{A}}{\mathbb{E}} \|\mathrm{dist}(\mathsf{e}^\star \mid \Pi, A) - \mathrm{dist}(\mathsf{e}^\star \mid A)\|_{\mathrm{tvd}} \geq \varepsilon \quad \text{or}$$
$$\underset{\mathsf{\Pi},\mathsf{B}}{\mathbb{E}} \|\mathrm{dist}(\mathsf{e}^\star \mid \Pi, B) - \mathrm{dist}(\mathsf{e}^\star \mid B)\|_{\mathrm{tvd}} \geq \varepsilon,$$

where all variables are defined with respect to the distribution $\mathcal{D}$ *and the internal randomness of* π *(recall that* Π *includes the transcript and the public randomness).*

Definition III.1 basically states that a protocol can internal ε-solve the set-intersection problem iff the transcript of the protocol can change the distribution of the target element $e^\star$ *from the perspective of Alice or Bob* by at least ε in the total variation distance on average.

Our definition is inspired but different from ε-solving in [45] (which we call *external* ε-solving to avoid ambiguity) which required the transcript to change the distribution of the target element by ε *from the perspective of an external observer* (who only sees the transcript but not the inputs of players). More formally, external ε-solving of set-intersection over a distribution μ, as defined in [45], requires the protocol π to have the following property (compare this with Definition III.1),

$$\underset{\mathsf{\Pi}}{\mathbb{E}} \|\mathrm{dist}(\mathsf{e}^\star \mid \Pi) - \mathrm{dist}(\mathsf{e}^\star)\|_{\mathrm{tvd}} \geq \varepsilon.$$

The previous work in [45] has shown that there is a distribution μ such that any protocol that external ε-solves set-intersection over μ has information cost $\Omega(\varepsilon^2 \cdot m)$. This however does *not* imply a lower bound for the internal ε-solving problem. This is because, *in principle*, these two tasks can be different. For instance, (i) a protocol that reveals the entire set of Alice, changes the distribution of target for Bob dramatically but not so much for an external observer; or (ii) a protocol that reveals all the elements that are neither in A nor in B changes the distribution of the target for an external observer by a lot but does not change the distribution for either of the players at all.

We prove the following lower bound on the information cost of internal ε-solving of set-intersection.

Theorem 1. *There is a distribution* $\mathcal{D}_{\mathsf{SI}}$ *for* set-intersection *over the universe* $[m]$ *such that:*

1) *For any* A *or* B *sampled from* $\mathcal{D}_{\mathsf{SI}}$*, both* $\mathrm{dist}(\mathsf{e}^\star \mid A)$ *and* $\mathrm{dist}(\mathsf{e}^\star \mid B)$ *are uniform distributions on* A *and* B*, each of size* $m/4$*, respectively.*
2) *For any* $\varepsilon \in (0, 1)$*, any protocol* π *that internal* ε*-solves the* set-intersection *problem over the distribution* $\mathcal{D}_{\mathsf{SI}}$ *(Definition III.1) has internal information cost* $\mathsf{IC}_{\mathcal{D}_{\mathsf{SI}}}(\pi) = \Omega(\varepsilon^2 \cdot m)$*.*

IV. The Unique-Reach Communication Problem

We now start with our main lower bounds. Define the following two-player communication problem.

Problem 2 (unique-reach)**.** *The* unique-reach *problem is defined as follows. Consider a digraph $G = (V, E)$ on n vertices where $V := \{s\} \sqcup V_1 \sqcup V_2 \sqcup V_3$ and any edge $(u, v) \in E$ is directed from s to V_1 or some V_i to V_{i+1} for $i \in [2]$ (we refer to each V_i as a* layer*). We are promised that there is a **unique** vertex $s^\star$ in the layer V_3 reachable from s.*

*Alice is given edges in E from V_1 to V_2, denoted by E_A, and Bob is given the remainder of the edges in E, denoted by E_B (the partitioning of vertices of V is known to both players). The goal for the players is to **find** $s^\star$ by Alice sending a single message to Bob (i.e., in the one-way communication model).*

It is easy to prove a lower bound of $\Omega(n^2)$ on the one-way communication complexity of unique-reach using a reduction from the Index problem. It is also easy to see that this problem can be solved with $O(n \log n)$ bits of communication, if we allow Bob to send a single message to Alice: By the uniqueness promise on $s^\star$, no vertex with out-degree more than one in V_2 should be reachable from s and thus Bob can communicate all the remaining edges in E_B to Alice.

Nevertheless, in this paper, we are interested in an algorithmically simpler variant of this problem similar-in-spirit to ε-solving for set-intersection (Definition III.1).

Definition IV.1. *Let $\mathcal{D}$ be any distribution of valid inputs $G = (V, E_A \sqcup E_B)$ for* unique-reach *(known to both players). We say that a protocol π **internal ε-solves*** unique-reach *over $\mathcal{D}$ iff:*

$$\mathop{\mathbb{E}}_{\Pi, \mathsf{E}_B} \|\text{dist}(\mathsf{s}^\star \mid \Pi, E_B) - \text{dist}(\mathsf{s}^\star \mid E_B)\|_{\text{tvd}} \geq \varepsilon, \quad (1)$$

where all variables are defined with respect to the distribution $\mathcal{D}$ and the internal randomness of π (recall that Π includes the transcript and the public randomness).

Definition IV.1 basically states that a protocol can internal ε-solve the problem iff the message sent from Alice can change the distribution of the unique vertex $s^\star$ from the perspective of Bob by at least ε in the total variation distance (in expectation over Alice's message and Bob's input).

Our main theorem in this section is the following.

Theorem 2. *There is a distribution $\mathcal{D}_{\mathsf{UR}}$ for* unique-reach *and an integer $b := \frac{n}{2^{\Theta(\sqrt{\log n})}}$ with the following properties:*

1) *For any E_B sampled from $\mathcal{D}_{\mathsf{UR}}$, $\text{dist}(\mathsf{s}^\star \mid E_B)$ is a uniform distribution over a subset $V_3^\star$ of b vertices in the layer V_3 of the input graph;*
2) *for any $\varepsilon \in (0, 1)$, any one-way protocol π that internal ε-solves* unique-reach *over the distribution $\mathcal{D}_{\mathsf{UR}}$ (Definition IV.1) has communication cost $\mathsf{CC}(\pi) = \Omega(\varepsilon^2 \cdot n \cdot b)$.*

Proof of Theorem 2 is by a reduction from our Theorem 1 using a combinatorial construction based on Ruzsa-Szemerédi graphs (see Section II-B). In the following section, we first present our distribution $\mathcal{D}_{\mathsf{UR}}$ and then in the subsequent section prove the desired lower bound.

A. *Distribution $\mathcal{D}_{\mathsf{UR}}$ in Theorem 2*

To continue, we need to set up some notation. Let $G^{\mathsf{RS}} = (L, R, E)$ be an (r, t)-RS digraph with induced matchings $M_1^{\mathsf{RS}}, \ldots, M_t^{\mathsf{RS}}$ as defined in Section II-B. For each induced matching M_i^{RS}, we assume an arbitrary ordering of edges $e_{i,1}, \ldots, e_{i,r}$ in the matching and for each $j \in [r]$ denote $e_{ij} := (u_{ij}, v_{ij})$ for $u_{ij} \in L$ and $v_{ij} \in R$; moreover, we let $L(M_i^{\mathsf{RS}}) := \{u_{i1}, \ldots, u_{ir}\}$ and $R(M_i^{\mathsf{RS}}) := \{v_{i1}, \ldots, v_{ir}\}$. Based on these, we have the following definition:

- For any matching M_i^{RS} and any set $S \subseteq [r]$, we define $M_i^{\mathsf{RS}}|S$ as the matching in G^{RS} consisting of the edges $e_{ij} \in M_i^{\mathsf{RS}}$ for all $j \in S$.

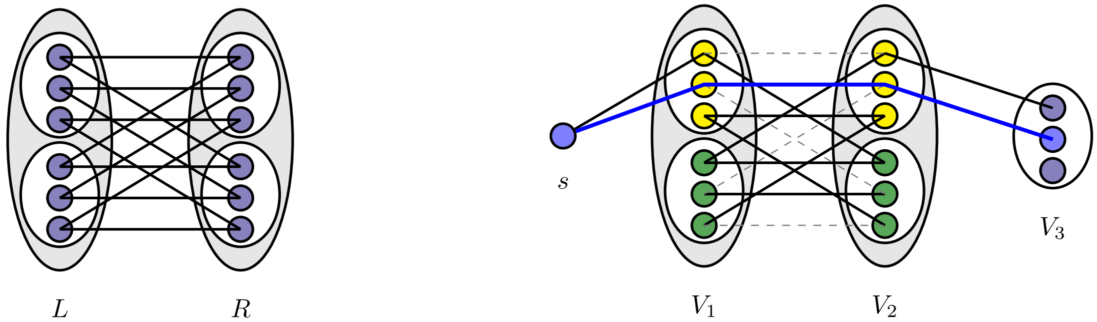
We are now ready to define our distribution. See Figure 1 for an illustration.

Distribution $\mathcal{D}_{\mathsf{UR}}$. An input distribution on graphs $G = (\{s\} \sqcup V_1 \sqcup V_2 \sqcup V_3, E_A \sqcup E_B)$.

1) Let $G^{\mathsf{RS}} = (L, R, E^{\mathsf{RS}})$ be a *fixed* (r, t)-RS digraph on $2N$ vertices from Proposition II.3 with parameters $r = \frac{N}{2^{\Theta(\sqrt{\log N})}}$, and $t = \frac{N}{3}$. We note that this graph is known to both players.
2) Let $V_1 = L = \{u_1, \ldots, u_N\}$, $V_2 = R = \{v_1, \ldots, v_N\}$, and V_3 be r new vertices $\{w_1, \ldots, w_r\}$.
3) Sample t *independent* instances $(S_1, T_1), \ldots, (S_t, T_t)$ of set-intersection on the universe $[r]$ from the distribution $\mathcal{D}_{\mathsf{SI}}$ in Theorem 1.
4) The input E_A to Alice is $E_A := (M_1^{\mathsf{RS}}|S_1) \cup \ldots \cup (M_t^{\mathsf{RS}}|S_t)$.
5) Sample $i^\star \in [t]$ *uniformly at random*.
6) The input E_B to Bob is the set of edges $(s, u_{i^\star j})$ for $j \in T_{i^\star}$ and $(v_{i^\star j}, w_j)$ for $j \in T_{i^\star}$.

Observation IV.2. *Several observations are in order:*

1) *For any $G \sim \mathcal{D}_{\mathsf{UR}}$, there is a unique vertex $s^\star$ reachable from s in V_3. Moreover, $s^\star = w_{e^\star}$ where $e^\star \in [r]$ is the unique element in the intersection of $S_{i^\star}$ and $T_{i^\star}$.*
2) *For any $E_B \sim \mathcal{D}_{\mathsf{UR}}$, $\text{dist}(\mathsf{s}^\star \mid E_B)$ is uniform over vertices $w_j \in V_3$ for $j \in T_{i^\star}$.*



(a) A fixed $(3, 4)$-RS digraph in the distribution.

(b) The graph G of $\mathcal{D}_{\mathsf{UR}}$; dashed edges no longer belong to the graph, and yellow vertices are incident on $M^{\mathsf{RS}}_{i^\star}$.

Figure 1: An illustration of the input distribution $\mathcal{D}_{\mathsf{UR}}$. Here, directions of all edges are from left to right and hence omitted. The marked vertex (blue) in V_3 denotes the unique vertex $s^\star$ in this example along with the path connecting s to $s^\star$.

3) *In $\mathcal{D}_{\mathsf{UR}}$, the index $i^\star \in [t]$ is* independent *of the sets $(S_1, T_1), \ldots, (S_t, T_t)$. Moreover, the pairs $(S_1, T_1), \ldots, (S_t, T_t)$ are* mutually independent.
4) *The input E_A to Alice in $\mathcal{D}_{\mathsf{UR}}$ is uniquely determined by $S_1, \ldots, S_t$, and the input E_B to Bob is determined by $i^\star$ and $T_{i^\star}$.*

B. Proof Sketch of Theorem 2

Let π_{UR} be any one-way protocol that internal ε-solves unique-reach on the distribution $\mathcal{D}_{\mathsf{UR}}$. We will prove that $\mathsf{CC}(\pi_{\mathsf{UR}}) = \Omega(\varepsilon^2 \cdot r \cdot t)$ which proves Theorem 2. The argument relies on the following two claims: (i) internal ε-solving of unique-reach on $\mathcal{D}_{\mathsf{UR}}$ is equivalent to internal ε-solving of set-intersection on $\mathcal{D}_{\mathsf{SI}}$ for the pair $(S_{i^\star}, T_{i^\star})$; and (ii) the information revealed by π_{UR} about the instance $(S_{i^\star}, T_{i^\star})$ is *at least t times smaller* than $\mathsf{CC}(\pi_{\mathsf{UR}})$. Having both these steps, we can then invoke Theorem 1 to conclude the proof.

We shall emphasize that this is *not* an immediate reduction from Theorem 1 as we are aiming to gain an *additional* factor of t in the information cost lower bound for π_{UR} compared to the lower bound for set-intersection. This part crucially relies on the fact that π_{UR} is a one-way protocol and that index $i^\star \in [t]$ in the distribution is independent of Alice's input (and thus her message).

We now present the formal proof. Consider the following protocol π_{SI} for set-intersection on the distribution $\mathcal{D}_{\mathsf{SI}}$ using π_{UR} as a subroutine.

Protocol π_{SI}: Given an instance $(A, B) \sim \mathcal{D}_{\mathsf{SI}}$ on universe $[r]$, Alice and Bob do as follows:

1) Alice and Bob sample $i^\star \in [t]$ using public randomness.
2) Alice sets $S_{i^\star} = A$ and samples the remaining sets S_i for $i \neq i^\star \in [t]$ independently from $\mathcal{D}_{\mathsf{SI}}$ using private randomness (this is doable by part (iii) of Observation IV.2). This allows Alice to generate the set E_A of edges for π_{UR} as in $\mathcal{D}_{\mathsf{UR}}$ (by part (iv) of Observation IV.2).
3) Bob sets $T_{i^\star} = B$ and creates the set of edges E_B for π_{UR} as in $\mathcal{D}_{\mathsf{UR}}$ (again doable by part (iv) of Observation IV.2 as Bob also knows $i^\star$).
4) The players then run the protocol π_{UR} on the input (E_A, E_B) with Alice sending the message in π_{UR} to Bob.

The first step of the proof is the following claim.

Claim IV.3. *π_{SI} internal ε-solves* set-intersection *on $\mathcal{D}_{\mathsf{SI}}$.*

We can also bound the internal information cost of π_{SI} which allows us to apply Theorem 1 and conclude the proof. The proof of this lemma is by a direct-sum style argument. We note that these arguments (based on information theory tools) are by now mostly standard in the literature.

Lemma IV.4. $\mathsf{IC}_{\mathcal{D}_{\mathsf{SI}}}(\pi_{\mathsf{SI}}) \leq \frac{1}{t} \cdot \mathsf{CC}(\pi_{\mathsf{UR}})$.

The proofs of Claim IV.3 and Lemma IV.4 are deferred to the full version.

We now conclude the proof of Theorem 2. By Claim IV.3, π_{SI} internal ε-solves set-intersection and thus by Theorem 1, we have $\mathsf{IC}_{\mathcal{D}_{\mathsf{SI}}}(\pi_{\mathsf{SI}}) = \Omega(\varepsilon^2 \cdot r)$. Plugging in this bound in Lemma IV.4, we obtain that

$$\mathsf{CC}(\pi_{\mathsf{UR}}) = \Omega(\varepsilon^2 \cdot r \cdot t) = \Omega(\varepsilon^2 \cdot \frac{n^2}{2^{\Theta(\sqrt{\log n})}}),$$

as the number of vertices n in the graph is $O(N)$. Setting $b = r/4 = \frac{n}{2^{\Theta(\sqrt{\log n})}}$ concludes the proof of Theorem 2.

C. The Inverse Unique-Reach Problem

In addition to the unique-reach problem, we also need another (almost identical) variant of this problem which we call the *inverse* of the unique-reach problem, denoted by $\overleftarrow{\text{unique-reach}}$. This problem is basically what one would naturally expect if we *reverse* the direction of all edges in an instance of unique-reach and ask for finding the unique vertex that can now reach the end-vertex t (corresponding to s). Formally, we define this problem as follows.

In $\overleftarrow{\textsf{unique-reach}}$, we have a digraph $\overleftarrow{G} = (U, \overleftarrow{E})$ on n vertices where $U := U_3 \sqcup U_2 \sqcup U_1 \sqcup \{t\}$, all edges of the graph are directed from U_1 to t or some U_{i+1} to U_i for $i \in [2]$, and we are promised that there is a *unique* vertex $t^\star$ in U_3 that can reach t. The goal is to find this vertex $t^\star$, or rather, internal ε-solve it exactly as in Definition IV.1. As before, the edges between U_2 and U_1, denoted by $\overleftarrow{E}_A$, are given to Alice, and the remaining edges, denoted by $\overleftarrow{E}_B$, are given to Bob. The communication is also one-way from Alice to Bob.

We also define a hard input distribution for $\overleftarrow{\textsf{unique-reach}}$, named $\overleftarrow{\mathcal{D}}_{\textsf{UR}}$, in exact analogy with $\mathcal{D}_{\textsf{UR}}$ for unique-reach: $\overleftarrow{\mathcal{D}}_{\textsf{UR}}$ is a distribution over graphs $\overleftarrow{G} = (U_3 \sqcup U_2 \sqcup U_1 \sqcup \{t\}, \overleftarrow{E}_A \sqcup \overleftarrow{E}_B)$, obtained by sampling a graph $G = (\{s\} \sqcup V_1 \sqcup V_2 \sqcup V_3, E_A \sqcup E_B)$ from $\mathcal{D}_{\textsf{UR}}$, setting $U_3 = V_3$, $U_2 = V_2$, $U_1 = V_1$, and $t = s$, and *reversing* the direction of all edges in E_A and E_B to obtain $\overleftarrow{E}_A$ and $\overleftarrow{E}_B$.

V. The st-Reachability Communication Problem

We now define the main two-player communication problem (the setting of this problem is rather non-standard in terms of the communication model).

Problem 3 (st-reachability)**.** *Consider a digraph $G = (V, E)$ with two designated vertices s, t and $E := E_1 \sqcup E_2 \sqcup E_3$. The goal is to determine whether or not s can reach t in G.*

Initially, Alice receives E_1 and Bob receives E_2 (the vertices s, t are known to both players). Next, Alice and Bob will have one round of communication by Alice sending a message Π_{A1} to Bob and Bob responding back with a message Π_{B1}. At this point, the edges E_3 are revealed to **both** *players. Finally, Alice is allowed to send yet another message Π_{A2} to Bob (which this time depends on E_3 as well) and Bob outputs the answer (also a function of E_3).*

The following theorem is the main result of our paper.

Theorem 3. *For any $\varepsilon \in (n^{-1/2}, 1/2)$, any communication protocol for* st-reachability *that succeeds with probability at least $\frac{1}{2} + \varepsilon$ requires $\Omega(\varepsilon^2 \cdot \frac{n^2}{2^{\Theta(\sqrt{\log n})}})$ bits of communication.*

We note that the $n^{-1/2}$ lower bound on ε in Theorem 3 is not sacrosanct and any term which is $\omega\left(\frac{\log n}{b}\right)$ still works where $b = \frac{n}{2^{\Theta(\sqrt{\log n})}}$ is the parameter in Theorem 2.

A. A Hard Distribution for st-reachability

Recall the distributions $\mathcal{D}_{\textsf{UR}}$, $\overleftarrow{\mathcal{D}}_{\textsf{UR}}$ from Section IV. We will use them to define our distribution for st-reachability. See Figure 2 for an illustration.

Distribution $\mathcal{D}_{\textsf{ST}}$. A hard input distribution for the st-reachability problem.

1) Let $V := \{s\} \sqcup V_1 \sqcup V_2 \sqcup V_3 \sqcup U_3 \sqcup U_2 \sqcup U_1 \sqcup \{t\}$ – each V_i or U_i is called a *layer* of G (this partitioning is known to both players).
2) Sample the graph $G_1 := (V_3 \sqcup U_3, E_1)$ by picking each edge $(v, u) \in V_3 \times U_3$ independently and with probability half.
3) Sample the following two graphs *independently*:
   a) $H := (\{s\} \sqcup V_1 \sqcup V_2 \sqcup V_3, E_A \sqcup E_B)$ sampled from the distribution $\mathcal{D}_{\textsf{UR}}$;
   b) $\overleftarrow{H} := (U_3 \sqcup U_2 \sqcup U_1 \sqcup \{t\}, \overleftarrow{E}_A \sqcup \overleftarrow{E}_B)$ sampled from the distribution $\overleftarrow{\mathcal{D}}_{\textsf{UR}}$.
4) The initial input to Alice and Bob are, respectively, E_1 and $E_2 := E_A \cup \overleftarrow{E}_A$, and the input revealed to both players in the second round is $E_3 := E_B \cup \overleftarrow{E}_B$.

To avoid potential confusion, we should note right away that Bob in the distribution $\mathcal{D}_{\textsf{ST}}$ is receiving the input of Alice in $\mathcal{D}_{\textsf{UR}}$ and $\overleftarrow{\mathcal{D}}_{\textsf{UR}}$.

Observation V.1. *The following two remarks are in order:*

1) *The distributions of E_1, H, and $\overleftarrow{H}$ are mutually independent in $\mathcal{D}_{\textsf{ST}}$.*
2) *s can reach t in G iff the edge $(s^\star, t^\star) \in E_1$.*
   (proof: the only vertex in V_3 reachable from s is $s^\star$ and the only vertex in U_3 that reaches t is $t^\star$, thus the only potential s-t path is $s \rightsquigarrow s^\star \to t^\star \rightsquigarrow t$.)

B. Setup and Notation

Let $\pi_{\textsf{ST}}$ be any *deterministic* protocol for st-reachability over the distribution $\mathcal{D}_{\textsf{ST}}$ with

$$\textsf{CC}(\pi_{\textsf{ST}}) = o(\varepsilon^2 \cdot b^2), \quad (2)$$

where $b := \frac{n}{2^{\Theta(\sqrt{\log n})}}$ is the parameter in Theorem 2 for instances of $\mathcal{D}_{\textsf{UR}}$ and $\overleftarrow{\mathcal{D}}_{\textsf{UR}}$. We will prove that the probability that $\pi_{\textsf{ST}}$ outputs the correct answer to st-reachability is $\frac{1}{2} + o(\varepsilon)$, hence proving Theorem 3 for deterministic protocols. The results for randomized protocols follows immediately from this and an averaging argument (i.e., the easy direction of Yao's minimax principle [69]).

To facilitate our proofs, the following notation would be useful. For brevity, we use

$$\Pi := (\Pi_{A1}, \Pi_{B1}, \Pi_{A2}),$$
$$Z_1 := (\Pi_{A1}, \Pi_{B1}, E_3),$$
$$Z_2 := (\Pi, E_3, s^\star, t^\star).$$

We also use $O \in \{0, 1\}$ to denote the output Bob at the end of the protocol.

For any pair of vertices $v, u \in V_3 \times U_3$, we use the notation $E_1(v, u) \in \{0, 1\}$ to denote whether or not the edge $(v, u) \in$

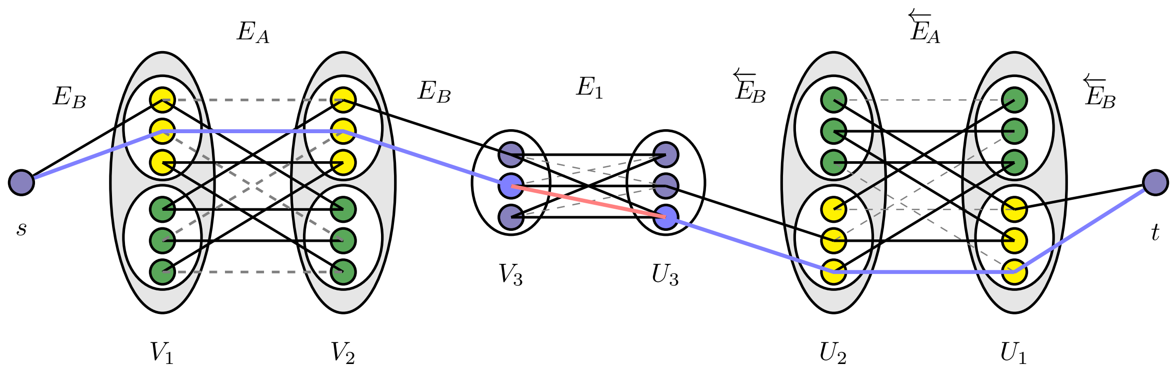


Figure 2: An illustration of the input distribution $\mathcal{D}_{\mathsf{ST}}$. Here, the directions of all edges are from left to right and hence omitted. The vertices $s^\star \in V_3$ and $t^\star \in U_3$ are marked blue and the potential edge $(s^\star, t^\star)$ is marked red–existence or non-existence of this edge uniquely determines whether or not s can reach t in G.

E_1. For a fixed choice of $E_3 = E_B \cup \overleftarrow{E}_B$ in $\mathcal{D}_{\mathsf{ST}}$, we use $V_3^\star$ and $U_3^\star$ to denote the sets from which $s^\star$ and $t^\star$ are chosen uniformly at random from conditioned on E_B and $\overleftarrow{E}_B$, respectively (see part (i) of Theorem 2). We also define:

$$E_1(V_3^\star, U_3^\star) := \{E_1(v_i, u_i) \mid (v_i, u_i) \in V_3^\star \times U_3^\star\}.$$

We further assume a fixed arbitrary ordering of pairs $v, u \in V_3 \times U_3$ and define:

$$E_1^{<(v,u)} := E_1(v_1, u_1), E_1(v_2, u_2), \dots$$

for all pairs $(v_i, u_i) \in E_1(V_3^\star, U_3^\star)$ that appear before (v, u) in this ordering (note that we ignore the other edges of E_1 that are not in $E_1(V_3^\star, U_3^\star)$ here).

Crucial Independence Properties

The following independence properties are crucial for our proofs. They are all based on the rectangle property of communication protocols and part (i) of Observation V.1.

$$\mathsf{\Pi}_{A2} \perp \mathsf{s}^\star, \mathsf{t}^\star \mid \mathsf{Z}_1 \tag{3}$$
$$\mathsf{E}_1 \perp \mathsf{s}^\star, \mathsf{t}^\star \mid \mathsf{Z}_1, \mathsf{\Pi}_{A2} \tag{4}$$
$$\mathsf{E}_2 \perp \mathsf{E}_1(\mathsf{s}^\star, \mathsf{t}^\star) \mid \mathsf{Z}_1, \mathsf{Z}_2. \tag{5}$$

The proofs appear in full version.

C. Part One: The First Round of Communication

In the following lemma, we prove that after the first round of the protocol, the (joint) distribution of $(s^\star, t^\star)$ conditioned on $Z_1 = (\Pi_{A1}, \Pi_{B1}, E_3)$ is almost the same as if we only conditioned on E_3. This is basically through a reduction from Theorem 2 considering $s^\star, t^\star$ are distributed (originally) according to $\mathcal{D}_{\mathsf{UR}}$ and $\overleftarrow{\mathcal{D}}_{\mathsf{UR}}$ and the public information E_3 provides the input of Bob in the instances of $\mathsf{unique\text{-}reach}$ and $\overleftarrow{\mathsf{unique\text{-}reach}}$ in this reduction.

Lemma V.2.

$$\mathop{\mathbb{E}}_{\mathsf{Z}_1} \|\mathrm{dist}(\mathsf{s}^\star, \mathsf{t}^\star \mid Z_1) - \mathrm{dist}(\mathsf{s}^\star, \mathsf{t}^\star \mid E_3)\|_{\mathrm{tvd}} = o(\varepsilon).$$

D. Part Two: The Second Round of Communication

Lemma V.2 implies that the extra information Z_1 available to Alice at the beginning of the second round does not change the distribution of $(s^\star, t^\star)$ by much. We use this to show that the message of Alice in the second round does not change the distribution of $E_1(s^\star, t^\star) \in \{0, 1\}$ by much.

Lemma V.3.

$$\mathop{\mathbb{E}}_{\mathsf{Z}_1, \mathsf{Z}_2} \|\mathrm{dist}(\mathsf{E}_1(s^\star, t^\star) \mid Z_1, Z_2) - \mathrm{dist}(\mathsf{E}_1(s^\star, t^\star))\|_{\mathrm{tvd}} = o(\varepsilon).$$

E. Concluding the Proof of Theorem 3

We are now ready to conclude the proof of Theorem 3. Lemma V.3 implies that conditioning on Z_1, Z_2 does not change the distribution of $E_1(s^\star, t^\star)$ by much. By the independence property of Eq (5), we know that this continues to hold even if we further condition on the input of Bob, i.e., E_2. We use this to prove that the probability that π_{ST} outputs the correct answer is almost the same as random guessing.

Claim V.4. $\Pr(\pi_{\mathsf{ST}} \text{ outputs the correct answer}) = \frac{1}{2} + o(\varepsilon)$.

To conclude, we have shown that for any deterministic protocol π_{ST} with $\mathsf{CC}(\pi_{\mathsf{ST}}) = o(\varepsilon^2 \cdot b^2)$, the probability that π_{ST} outputs the correct answer over the distribution $\mathcal{D}_{\mathsf{ST}}$ is only $\frac{1}{2} + o(\varepsilon)$. This can be extended directly to randomized protocols as by an averaging argument, we can always fix the randomness of any randomized protocol π_{ST} on the distribution $\mathcal{D}_{\mathsf{ST}}$ to obtain a deterministic protocol with the same error guarantee. Noting that $b = \frac{n}{2^{\Theta(\sqrt{\log n})}}$ concludes the proof of Theorem 3.

VI. Graph Streaming Lower Bounds

We now obtain our graph streaming lower bounds by reductions from the $\mathsf{st\text{-}reachability}$ communication problem defined in Section V. The first step of all these reductions is to show that one can simulate any two-pass graph streaming algorithm on graphs $G = (V, E)$ using a protocol in the setting of the $\mathsf{st\text{-}reachability}$ problem. The proof is via a standard simulation.

Proposition VI.1. *Any two-pass S-space streaming algorithm $\mathcal{A}$ on graphs $G = (V, E_1 \sqcup E_2 \sqcup E_3)$ of* $\mathsf{st\text{-}reachability}$ *can be simulated exactly by a communication protocol $\pi_{\mathcal{A}}$ with $\mathsf{CC}(\pi_{\mathcal{A}}) = O(S)$ and the communication-pattern restrictions of the* $\mathsf{st\text{-}reachability}$ *problem.*

A. Directed Reachability

We obtain the following theorem for the directed reachability problem.

Theorem 4 (Formalization of Result 1)**.** *Any streaming algorithm that makes two passes over the edges of any n-vertex directed graph $G = (V, E)$ with two designated vertices $s, t \in V$ and outputs whether or not s can reach t in G with probability at least $2/3$ requires $\Omega(\frac{n^2}{2^{\Theta(\sqrt{\log n})}})$ space.*

Theorem 4 follows immediately from Proposition VI.1 and our lower bound in Theorem 3.

We also present some standard extension of this lower bound to other problems related to the directed reachability problem.

- **Estimating number of vertices reachable from a source:** Consider any instance of the problem in Theorem 4 and connect t to $2n$ *new vertices*. In the new graph, if s can reach t, then it can also reach at least $2n$ other vertices, while if s does not reach t, it can reach at most n other vertices. Hence, the lower bound in Theorem 4 extends to this problem as well which was studied (in a similar format) in [3].
- **Testing if G is acyclic or not:** Recall that the hard distribution of graphs in Theorem 3 and hence Theorem 4 is supported on *acyclic* graphs. If in these graphs, we connect t to s directly, then the graph remains acyclic iff s cannot reach t. Hence, the lower bound in Theorem 4 extends to this problem as well.
- **Approximating minimum feedback arc set:** The lower bound for acyclicity implies the same bounds for any (multiplicative) approximation algorithm of minimum feedback arc set (the minimum number of edges to be deleted to make a graph acyclic) studied in [4].

B. Bipartite Perfect Matching

We obtain the following theorem for the bipartite perfect matching problem using a standard reduction.

Theorem 5 (Formalization of Result 2)**.** *Any streaming algorithm that makes two passes over the edges of any n-vertex undirected bipartite graph $G = (L \sqcup R, E)$ and outputs whether or not G has a perfect matching with probability at least $2/3$ requires $\Omega(\frac{n^2}{2^{\Theta(\sqrt{\log n})}})$ space.*

C. Single-Source Shortest Path

Finally, we have the following theorem for the shortest path problem, again using a standard reduction.

Theorem 6 (Formalization of Result 3)**.** *Any streaming algorithm that makes two passes over the edges of any n-vertex undirected graph $G = (V, E)$ with two designated vertices $s, t \in V$ and outputs the length of the shortest s-t path in G with probability at least $2/3$ requires $\Omega(\frac{n^2}{2^{\Theta(\sqrt{\log n})}})$ space.*

The lower bound continues to hold even if the algorithm is allowed to output an estimate which, with probability at least $2/3$, is as large as the length of the shortest s-t path and strictly smaller than $9/7$ times the length of the shortest s-t path.

Acknowledgment

Sepehr Assadi done part of this project while at Princeton University and was supported in part by the Simons Collaboration on Algorithms and Geometry. Ran Raz was partially supported by the Simons Collaboration on Algorithms and Geometry, by a Simons Investigator Award, and by the National Science Foundation grants No. CCF-171477 and CCF-2007462.

References

[1] J. Feigenbaum, S. Kannan, A. McGregor, S. Suri, and J. Zhang, "On graph problems in a semi-streaming model," *Theor. Comput. Sci.*, vol. 348, no. 2-3, pp. 207–216, 2005. [Online]. Available: http://dx.doi.org/10.1016/j.tcs.2005.09.013 1, 2

[2] A. Goel, M. Kapralov, and S. Khanna, "On the communication and streaming complexity of maximum bipartite matching," in *Proceedings of the Twenty-third Annual ACM-SIAM Symposium on Discrete Algorithms*, ser. SODA '12. SIAM, 2012, pp. 468–485. [Online]. Available: http://dl.acm.org/citation.cfm?id=2095116.2095157 1, 3, 4

[3] M. R. Henzinger, P. Raghavan, and S. Rajagopalan, "Computing on data streams," in *External Memory Algorithms, Proceedings of a DIMACS Workshop, New Brunswick, New Jersey, USA, May 20-22, 1998*, 1998, pp. 107–118. 1, 2, 9

[4] A. Chakrabarti, P. Ghosh, A. McGregor, and S. Vorotnikova, "Vertex ordering problems in directed graph streams," in *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms, SODA 2020, Salt Lake City, UT, USA, January 5-8, 2020*, 2020, pp. 1786–1802. 1, 2, 3, 9

[5] J. Feigenbaum, S. Kannan, A. McGregor, S. Suri, and J. Zhang, "Graph distances in the data-stream model," *SIAM J. Comput.*, vol. 38, no. 5, pp. 1709–1727, 2008. 1, 2, 3

[6] M. Zelke, "Intractability of min- and max-cut in streaming graphs," *Inf. Process. Lett.*, vol. 111, no. 3, pp. 145–150, 2011. 1

[7] S. Assadi, Y. Chen, and S. Khanna, "Sublinear algorithms for $(\Delta + 1)$ vertex coloring," in *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2019, San Diego, California, USA, January 6-9, 2019*, 2019, pp. 767–786. 1

[8] G. Cormode, J. Dark, and C. Konrad, "Independent sets in vertex-arrival streams," in *46th International Colloquium on Automata, Languages, and Programming, ICALP 2019, July 9-12, 2019, Patras, Greece*, 2019, pp. 45:1–45:14. 1, 3

[9] Y. Emek and A. Rosén, "Semi-streaming set cover - (extended abstract)," in *Automata, Languages, and Programming - 41st International Colloquium, ICALP 2014, Copenhagen, Denmark, July 8-11, 2014, Proceedings, Part I*, 2014, pp. 453–464. 1

[10] S. Assadi, S. Khanna, and Y. Li, "Tight bounds for single-pass streaming complexity of the set cover problem," in *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016*, 2016, pp. 698–711. 1

[11] L. Epstein, A. Levin, J. Mestre, and D. Segev, "Improved approximation guarantees for weighted matching in the semi-streaming model," *SIAM J. Discrete Math.*, vol. 25, no. 3, pp. 1251–1265, 2011. 1

[12] M. Crouch and D. S. Stubbs, "Improved streaming algorithms for weighted matching, via unweighted matching," in *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2014, September 4-6, 2014*, 2014, pp. 96–104. 1

[13] A. Paz and G. Schwartzman, "A $(2 + \varepsilon)$-approximation for maximum weight matching in the semi-streaming model," in *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2017, Barcelona, Spain, Hotel Porta Fira, January 16-19*, 2017, pp. 2153–2161. 1

[14] M. Elkin and J. Zhang, "Efficient algorithms for constructing (1+, varepsilon;, beta)-spanners in the distributed and streaming models," in *Proceedings of the Twenty-Third Annual ACM Symposium on Principles of Distributed Computing, PODC 2004, St. John's, Newfoundland, Canada, July 25-28, 2004*, 2004, pp. 160–168. 1, 2

[15] S. Baswana, "Streaming algorithm for graph spanners - single pass and constant processing time per edge," *Inf. Process. Lett.*, vol. 106, no. 3, pp. 110–114, 2008. 1, 2

[16] M. Elkin, "Streaming and fully dynamic centralized algorithms for constructing and maintaining sparse spanners," *ACM Trans. Algorithms*, vol. 7, no. 2, pp. 20:1–20:17, 2011. 1, 2

[17] K. J. Ahn and S. Guha, "Graph sparsification in the semi-streaming model," in *Automata, Languages and Programming, 36th Internatilonal Colloquium, ICALP 2009, Rhodes, Greece, July 5-12, 2009, Proceedings, Part II*, 2009, pp. 328–338. 1

[18] K. J. Ahn, S. Guha, and A. McGregor, "Spectral sparsification in dynamic graph streams," in *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques - 16th International Workshop, APPROX 2013, and 17th International Workshop, RANDOM 2013, Berkeley, CA, USA, August 21-23, 2013. Proceedings*, 2013, pp. 1–10. 1

[19] J. A. Kelner and A. Levin, "Spectral sparsification in the semi-streaming setting," in *28th International Symposium on Theoretical Aspects of Computer Science, STACS 2011, March 10-12, 2011, Dortmund, Germany*, 2011, pp. 440–451. 1

[20] M. Kapralov, Y. T. Lee, C. Musco, C. Musco, and A. Sidford, "Single pass spectral sparsification in dynamic streams," in *55th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2014, Philadelphia, PA, USA, October 18-21, 2014*, 2014, pp. 561–570. [Online]. Available: http://dx.doi.org/10.1109/FOCS.2014.66 1

[21] B. V. Halldórsson, M. M. Halldórsson, E. Losievskaja, and M. Szegedy, "Streaming algorithms for independent sets," in *Automata, Languages and Programming, 37th International Colloquium, ICALP 2010, Bordeaux, France, July 6-10, 2010, Proceedings, Part I*, 2010, pp. 641–652. 1

[22] S. K. Bera, A. Chakrabarti, and P. Ghosh, "Graph coloring via degeneracy in streaming and other space-conscious models," in *47th International Colloquium on Automata, Languages, and Programming, ICALP 2020, July 8-11, 2020, Saarbrücken, Germany (Virtual Conference)*, 2020, pp. 11:1–11:21. 1

[23] A. 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 2005 and 9th InternationalWorkshop on Randomization and Computation, RANDOM 2005, Berkeley, CA, USA, August 22-24, 2005, Proceedings*, 2005, pp. 170–181. [Online]. Available: http://dx.doi.org/10.1007/11538462_15 1

[24] S. Eggert, L. Kliemann, and A. Srivastav, "Bipartite graph matchings in the semi-streaming model," in *Algorithms - ESA 2009, 17th Annual European Symposium, September 7-9, 2009. Proceedings*, 2009, pp. 492–503. 1

[25] C. Konrad, F. Magniez, and C. Mathieu, "Maximum matching in semi-streaming with few passes," in *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques - 15th International Workshop, APPROX 2012, and 16th International Workshop, RANDOM 2012, Cambridge, MA, USA, August 15-17, 2012. Proceedings*, 2012, pp. 231–242. 1

[26] M. Kapralov, "Better bounds for matchings in the streaming model," in *Proceedings of the Twenty-Fourth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2013, New Orleans, Louisiana, USA, January 6-8, 2013*, 2013, pp. 1679–1697. [Online]. Available: http://dx.doi.org/10.1137/1.9781611973105.121 1, 3

[27] S. Kale and S. Tirodkar, “Maximum matching in two, three, and a few more passes over graph streams,” in *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2017, August 16-18, 2017, Berkeley, CA, USA*, 2017, pp. 15:1–15:21. 1

[28] R. Becker, A. Karrenbauer, S. Krinninger, and C. Lenzen, “Near-optimal approximate shortest paths and transshipment in distributed and streaming models,” in *31st International Symposium on Distributed Computing, DISC 2017, October 16-20, 2017, Vienna, Austria*, 2017, pp. 7:1–7:16. 1, 2

[29] K. J. Ahn, S. Guha, and A. McGregor, “Graph sketches: sparsification, spanners, and subgraphs,” in *Proceedings of the 31st ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, PODS 2012, Scottsdale, AZ, USA, May 20-24, 2012*, 2012, pp. 5–14. [Online]. Available: http://doi.acm.org/10.1145/2213556.2213560 1

[30] A. D. Sarma, S. Gollapudi, and R. Panigrahy, “Estimating pagerank on graph streams,” *J. ACM*, vol. 58, no. 3, pp. 13:1–13:19, 2011. 1

[31] M. Kapralov and D. P. Woodruff, “Spanners and sparsifiers in dynamic streams,” in *ACM Symposium on Principles of Distributed Computing, PODC ’14, Paris, France, July 15-18, 2014*, 2014, pp. 272–281. 1

[32] A. McGregor, S. Vorotnikova, and H. T. Vu, “Better algorithms for counting triangles in data streams,” in *Proceedings of the 35th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems, PODS 2016, San Francisco, CA, USA, June 26 - July 01, 2016*, 2016, pp. 401–411. 1

[33] G. Cormode and H. Jowhari, “A second look at counting triangles in graph streams (corrected),” *Theor. Comput. Sci.*, vol. 683, pp. 22–30, 2017. 1

[34] S. K. Bera and A. Chakrabarti, “Towards tighter space bounds for counting triangles and other substructures in graph streams,” in *34th Symposium on Theoretical Aspects of Computer Science, STACS 2017, March 8-11, 2017, Hannover, Germany*, 2017, pp. 11:1–11:14. 1

[35] B. Gamlath, S. Kale, S. Mitrovic, and O. Svensson, “Weighted matchings via unweighted augmentations,” in *Proceedings of the 2019 ACM Symposium on Principles of Distributed Computing, PODC 2019, Toronto, ON, Canada, July 29 - August 2, 2019*, 2019, pp. 491–500. 1

[36] Y. Chang, M. Farach-Colton, T. Hsu, and M. Tsai, “Streaming complexity of spanning tree computation,” in *37th International Symposium on Theoretical Aspects of Computer Science, STACS 2020, March 10-13, 2020, Montpellier, France*, 2020, pp. 34:1–34:19. 1, 2

[37] A. McGregor, “Graph stream algorithms: a survey,” *SIGMOD Record*, vol. 43, no. 1, pp. 9–20, 2014. [Online]. Available: http://doi.acm.org/10.1145/2627692.2627694 1

[38] A. Rubinstein, T. Schramm, and S. M. Weinberg, “Computing exact minimum cuts without knowing the graph,” in *9th Innovations in Theoretical Computer Science Conference, ITCS 2018, January 11-14, 2018, Cambridge, MA, USA*, 2018, pp. 39:1–39:16. 1, 2

[39] S. Mukhopadhyay and D. Nanongkai, “Weighted min-cut: sequential, cut-query, and streaming algorithms,” in *Procceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing, STOC 2020, Chicago, IL, USA, June 22-26, 2020*, 2020, pp. 496–509. 1

[40] L. Bulteau, V. Froese, K. Kutzkov, and R. Pagh, “Triangle counting in dynamic graph streams,” *Algorithmica*, vol. 76, no. 1, pp. 259–278, 2016. 1

[41] M. Ghaffari, T. Gouleakis, C. Konrad, S. Mitrovic, and R. Rubinfeld, “Improved massively parallel computation algorithms for mis, matching, and vertex cover,” in *Proceedings of the 2018 ACM Symposium on Principles of Distributed Computing, PODC 2018, July 23-27, 2018*, 2018, pp. 129–138. 1

[42] A. Chakrabarti and A. Wirth, “Incidence geometries and the pass complexity of semi-streaming set cover,” in *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2016, Arlington, VA, USA, January 10-12, 2016*, 2016, pp. 1365–1373. 1

[43] S. Har-Peled, P. Indyk, S. Mahabadi, and A. Vakilian, “Towards tight bounds for the streaming set cover problem,” in *Proceedings of the 35th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems, PODS 2016, San Francisco, CA, USA, June 26 - July 01, 2016*, 2016, pp. 371–383. 1

[44] V. Guruswami and K. Onak, “Superlinear lower bounds for multipass graph processing,” in *Proceedings of the 28th Conference on Computational Complexity, CCC 2013, K.lo Alto, California, USA, 5-7 June, 2013*, 2013, pp. 287–298. 1, 2, 3

[45] S. Assadi, Y. Chen, and S. Khanna, “Polynomial pass lower bounds for graph streaming algorithms,” in *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, STOC 2019, Phoenix, AZ, USA, June 23-26, 2019.*, 2019, pp. 265–276. 1, 2, 3, 4

[46] R. Chitnis, G. Cormode, H. Esfandiari, M. Hajiaghayi, A. McGregor, M. Monemizadeh, and S. 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, January 10-12, 2016*, 2016, pp. 1326–1344. 2

[47] R. H. Chitnis, G. Cormode, M. T. Hajiaghayi, and M. Monemizadeh, “Parameterized streaming: Maximal matching and vertex cover,” in *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2015, San Diego, CA, USA, January 4-6, 2015*, 2015, pp. 1234–1251. 2

[48] M. Henzinger, S. Krinninger, and D. Nanongkai, “A deterministic almost-tight distributed algorithm for approximating single-source shortest paths,” in *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016*, 2016, pp. 489–498. 2

[49] C. H. Papadimitriou and M. Sipser, "Communication complexity," *J. Comput. Syst. Sci.*, vol. 28, no. 2, pp. 260–269, 1984. 3

[50] N. Nisan and A. Wigderson, "Rounds in communication complexity revisited," in *Proceedings of the 23rd Annual ACM Symposium on Theory of Computing, May 5-8, 1991, New Orleans, Louisiana, USA*, 1991, pp. 419–429. 3

[51] S. Ponzio, J. Radhakrishnan, and S. Venkatesh, "The communication complexity of pointer chasing: Applications of entropy and sampling," in *Proceedings of the Thirty-First Annual ACM Symposium on Theory of Computing, May 1-4, 1999, Atlanta, Georgia, USA*, 1999, pp. 602–611. 3

[52] A. Chakrabarti, G. Cormode, and A. McGregor, "Robust lower bounds for communication and stream computation," in *Proceedings of the 40th Annual ACM Symposium on Theory of Computing, May 17-20, 2008*, 2008, pp. 641–650. 3

[53] A. Yehudayoff, "Pointer chasing via triangular discrimination," *Electronic Colloquium on Computational Complexity (ECCC)*, vol. 23, p. 151, 2016. 3

[54] C. Konrad, "Maximum matching in turnstile streams," in *Algorithms - ESA 2015 - 23rd Annual European Symposium, September 14-16, 2015, Proceedings*, 2015, pp. 840–852. 3

[55] S. Assadi, S. Khanna, Y. Li, and G. Yaroslavtsev, "Maximum matchings in dynamic graph streams and the simultaneous communication model," in *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2016, Arlington, VA, USA, January 10-12, 2016*, 2016, pp. 1345–1364. 3

[56] S. Assadi, S. Khanna, and Y. Li, "On estimating maximum matching size in graph streams," in *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2017, Barcelona, Spain, Hotel Porta Fira, January 16-19*, 2017, pp. 1723–1742. 3

[57] A. C. Yao, "Some complexity questions related to distributive computing (preliminary report)," in *Proceedings of the 11h Annual ACM Symposium on Theory of Computing, April 30 - May 2, 1979, Atlanta, Georgia, USA*, 1979, pp. 209–213. 3

[58] E. Kushilevitz and N. Nisan, *Communication complexity*. Cambridge University Press, 1997. 3

[59] B. Barak, M. Braverman, X. Chen, and A. Rao, "How to compress interactive communication," in *Proceedings of the 42nd ACM Symposium on Theory of Computing, STOC 2010, 5-8 June 2010*, 2010, pp. 67–76. 4

[60] A. Chakrabarti, Y. Shi, A. Wirth, and A. C. Yao, "Informational complexity and the direct sum problem for simultaneous message complexity," in *42nd Annual Symposium on Foundations of Computer Science, FOCS 2001, 14-17 October 2001*, 2001, pp. 270–278. 4

[61] Z. Bar-Yossef, T. S. Jayram, R. Kumar, and D. Sivakumar, "An information statistics approach to data stream and communication complexity," in *43rd Symposium on Foundations of Computer Science (FOCS 2002), 16-19 November 2002, Proceedings*, 2002, pp. 209–218. 4

[62] M. Braverman and A. Rao, "Information equals amortized communication," in *IEEE 52nd Annual Symposium on Foundations of Computer Science, FOCS 2011, October 22-25, 2011*, 2011, pp. 748–757. 4

[63] M. Braverman, F. Ellen, R. Oshman, T. Pitassi, and V. Vaikuntanathan, "A tight bound for set disjointness in the message-passing model," in *54th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2013, 26-29 October, 2013, Berkeley, CA, USA*, 2013, pp. 668–677. 4

[64] I. Z. Ruzsa and E. Szemerédi, "Triple systems with no six points carrying three triangles," *Combinatorics (Keszthely, 1976), Coll. Math. Soc. J. Bolyai*, vol. 18, pp. 939–945, 1978. 4

[65] F. A. Behrend, "On sets of integers which contain no three terms in arithmetical progression," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 32, no. 12, p. 331, 1946. 4

[66] E. Fischer, E. Lehman, I. Newman, S. Raskhodnikova, R. Rubinfeld, and A. Samorodnitsky, "Monotonicity testing over general poset domains," in *Proceedings on 34th Annual ACM Symposium on Theory of Computing, May 19-21, 2002, Montréal, Québec, Canada*, 2002, pp. 474–483. 4

[67] N. Alon, A. Moitra, and B. Sudakov, "Nearly complete graphs decomposable into large induced matchings and their applications," in *Proceedings of the 44th Symposium on Theory of Computing Conference, STOC 2012, New York, NY, USA, May 19 - 22, 2012*, 2012, pp. 1079–1090. 4

[68] J. Fox, H. Huang, and B. Sudakov, "On graphs decomposable into induced matchings of linear sizes," *arXiv preprint arXiv:1512.07852*, 2015. 4

[69] A. C. Yao, "Lower bounds by probabilistic arguments (extended abstract)," in *24th Annual Symposium on Foundations of Computer Science, Tucson, Arizona, USA, 7-9 November 1983*, 1983, pp. 420–428. 7