Towards Tighter Space Bounds for Counting Triangles and Other Substructures in Graph Streams[*][†]

Suman K. Bera[1] **and Amit Chakrabarti**[2]

1 Department of Computer Science, Dartmouth College, Hanover, USA
2 Department of Computer Science, Dartmouth College, Hanover, USA

Abstract

We revisit the much-studied problem of space-efficiently estimating the number of triangles in a graph stream, and extensions of this problem to counting fixed-sized cliques and cycles. For the important special case of counting triangles, we give a 4-pass, $(1 \pm \varepsilon)$-approximate, randomized algorithm using $\widetilde{O}(\varepsilon^{-2}\, m^{3/2}/T)$ space, where m is the number of edges and T is a promised lower bound on the number of triangles. This matches the space bound of a recent algorithm (McGregor et al., PODS 2016), with an arguably simpler and more general technique. We give an improved multi-pass lower bound of $\Omega(\min\{m^{3/2}/T, m/\sqrt{T}\})$, applicable at essentially all densities $\Omega(n) \leqslant m \leqslant O(n^2)$. We prove other multi-pass lower bounds in terms of various structural parameters of the input graph. Together, our results resolve a couple of open questions raised in recent work (Braverman et al., ICALP 2013).

Our presentation emphasizes more general frameworks, for both upper and lower bounds. We give a sampling algorithm for counting arbitrary subgraphs and then improve it via combinatorial means in the special cases of counting odd cliques and odd cycles. Our results show that these problems are considerably easier in the cash-register streaming model than in the turnstile model, where previous work had focused (Manjunath et al., ESA 2011; Kane et al., ICALP 2012). We use Turán graphs and related gadgets to derive lower bounds for counting cliques and cycles, with triangle-counting lower bounds following as a corollary.

1998 ACM Subject Classification F.2.2 Nonnumerical Algorithms and Problems, G.2.2 Graph Theory

Keywords and phrases data streaming, graph algorithms, triangles, subgraph counting, lower bounds

Digital Object Identifier 10.4230/LIPIcs.STACS.2017.11

1 Introduction

Algorithms for analyzing large graphs have been the topic of two decades of intense research. The theory of such algorithms encompasses two large disciplines: *streaming* algorithms [25, 14], where the input graph is presented as a stream of edges that must be read sequentially, in one or more passes, using space sublinear in the total input size; and *property-testing* algorithms [16, 17], where the input graph may be randomly accessed and the goal is to decide whether it satisfies some property or is far from doing so, while reading a sublinear fraction of the input. This paper is concerned with streaming algorithms.

[*] In this extended abstract, several proofs are either shortened or omitted, and a few theorems and lemmas are stated in not-quite-full detail. We refer the interested reader to the full version of this paper.

[†] This work was supported in part by the NSF under Award 1650992.

34th Symposium on Theoretical Aspects of Computer Science (STACS 2017).
Editors: Heribert Vollmer and Brigitte Vallée; Article No. 11; pp. 11:1–11:14
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Table 1 Results from prior work. See the discussion at the start of Section 1.1.

Problem	Space	Remarks	Source
Triangle counting (TRI-CNT)	$\widetilde{O}\big(mn/T\big)^2$	–	[3]
	$\widetilde{O}\big(m\Delta^2/T\big)$	Δ = maximum degree	[19]
	$\widetilde{O}\big(mn/T\big)$	n known a priori	[8]
	$\widetilde{O}\big(m^3/T^2\big)$	turnstile	[21]
	$\widetilde{O}\big(m\Delta/T\big)$	Δ = maximum degree	[28]
	$\widetilde{O}\big(m\gamma/T\big)$	γ = tangle coefficient (Section 2)	[28]
	$\widetilde{O}\big(mJ/T + m/\sqrt{T}\big)$	J = max # triangles containing an edge	[27]
	$C + \widetilde{O}\big(P_2/T\big)$	C = vertex cover, P_2 = # of 2-paths	[15]
	$\widetilde{O}\big(m/\sqrt{T}\big)$	dependence on ε is $1/\varepsilon^{2.5}$	[11]
	$\widetilde{O}\big(m^{3/2}/T\big)$	multi-pass	[24]
	$\widetilde{O}\big(m/\sqrt{T}\big)$	multi-pass	[24]
	$\Omega\big(n^2\big)$	one pass, $T = 1$	[3]
	$\Omega\big(n/T\big)$	multi-pass, $T < n$	[19]
	$\Omega\big(m\big)$	one pass, $m \in [c_1 n, c_2 n^2]$, $T < n$	[7]
	$\Omega\big(m/T\big)$	multi-pass	[7]
	$\Omega\big(m^3/T^2\big)$	one pass, optimal	[22]
	$\Omega\big(m/T^{2/3}\big)$	multi-pass	[11]
	$\Omega\big(m/\sqrt{T}\big)$	multi-pass, for $m = \Theta(n\sqrt{T})$	[11]
Clique counting (CLQ-CNT$_h$)	$\widetilde{O}\big(m^{h(h-1)/2}/T^2\big)$	turnstile	[21]
	$\widetilde{O}\big(\eta(h)/T\big)$	$\eta(h) = \max\{m^\alpha \Delta^{h-2\alpha} : \alpha \in \{1, \lfloor h/2 \rfloor\}\}$	[28]
Cycle counting (CYC-CNT$_h$)	$\widetilde{O}\big(m^h/T^2\big)$	turnstile	[23]

Specifically, this paper is about the SUBGRAPH-COUNTING problem, which asks for an (approximate) count of the number of occurrences of a particular constant-sized subgraph, H, in an input graph, G, which has n vertices and m edges. We denote this problem SUB-CNT$_H$. After giving a basic algorithm for SUB-CNT$_H$, we provide improvements (in terms of space usage) for special classes of subgraphs, namely, when H is either an h-clique $\mathcal{K}_h$ (the CLIQUE-COUNTING problem, or CLQ-CNT$_h$) or an h-cycle $\mathcal{C}_h$ (the CYCLE-COUNTING problem, or CYC-CNT$_h$). We also give upper and lower bounds, several of them optimal, for the very important special case of TRIANGLE-COUNTING (henceforth, TRI-CNT), when H is a triangle.

The number of triangles in a graph is a basic parameter of interest for a variety of reasons, including social network analysis [26] and spam and fraud detection [4]; see [12, 31] for a more thorough discussion of applications. The TRI-CNT problem has been the focus of a remarkably large number of papers. Nevertheless, the definitive upper and lower bounds on its space complexity have not yet been obtained, leaving us with several mutually incomparable results (Table 1). Notably, distinguishing a triangle-free graph from a graph containing one or more triangles requires $\Omega(n^2)$ space in general [3], ruling out unconditional sublinear-space solutions. Therefore, nontrivial bounds must either assume some structural guarantees on the input graph or provide space guarantees that depend on some structural parameter.

1.1 Our Results and Comparison with Prior Work

To set the context for our results, we summarize the salient related results from prior work (most of which are about TRI-CNT) in Table 1. The ubiquitous parameter "T" represents a

Table 2 Our main results. The $\widetilde{O}$-notation hides $1/\varepsilon^2$ and $\log m$ factors; h is considered a constant. The upper bounds use 4 passes for counting odd cliques and odd cycles (including triangles) and 2 passes in all other cases. The notations Δ, γ, ρ, and β are as discussed at the start of Section 1.1.

Problem	Upper bounds	Lower bounds		Remarks
	multi-pass	one pass	multi-pass	
TRI-CNT	$\widetilde{O}\left(m^{3/2}/T\right)$	–	$\Omega\left(\min\left\{m^{3/2}/T,\, m/\sqrt{T}\right\}\right)$	optimal
			$\Omega\left(m\Delta/T\right)$	optimal
			$\Omega\left(m\gamma/T\right)$	optimal
			$\Omega\left(m/\rho\right)$	optimal
CLQ-CNT$_h$	$\widetilde{O}(m^{h/2}/T)$	$\Omega(m^h/T^2)$	$\Omega\left(\min\left\{m^{h/2}/T,\, m/T^{1/(h-1)}\right\}\right)$	
CYC-CNT$_h$	$\widetilde{O}(m^{h/2}/T)$	$\Omega(m^{h/2}/T)$	$\Omega\left(m^{h/2}/T\right)$	even h
	$\widetilde{O}(m^{h/2}/T)$	$\Omega(m^h/T^2)$	$\Omega\left(\min\left\{m^{h/2}/T,\, m/T^{1/(h-1)}\right\}\right)$	odd h
SUB-CNT$_H$	$\widetilde{O}(m^{\beta(H)}/T)$	–	–	

guaranteed lower bound on the number of copies of the target subgraph (triangle, $\mathcal{K}_h$, or $\mathcal{C}_h$) in the input graph G. Some of the results involve additional graph parameters (definitions in Section 2): the maximum degree $\Delta = \Delta(G)$, the triangle density $\rho = \rho(G)$, the tangle coefficient $\gamma = \gamma(G)$ and, for the problem SUB-CNT$_H$, the edge cover number $\beta = \beta(H)$.

In reading Table 1, note that all upper bounds represent randomized algorithms that provide $(1 \pm \varepsilon)$-multiplicative approximations with high constant probability, and work in one pass, except as noted. Their space usage involves a factor of $\log m$ and an ε-dependent factor that, with one exception, equals $1/\varepsilon^2$. We omit these factors for clarity, hiding them into an $\widetilde{O}(\cdot)$ notation. Lower bounds are proven by studying the communication complexity of distinguishing between "high" and "low" values of T. As can be seen, there are not many multi-pass lower bounds in prior work and none that explain the form of any of the upper bounds. Our lower bounds in this paper serve to fill this explanatory gap. Meanwhile, our upper bounds demonstrate the power of using a small constant number of passes, rather than one pass, for these problems. Table 2 summarizes our main contributions.

Our first upper bound is for SUB-CNT$_H$. We give a 2-pass algorithm based on (implicitly) sampling a suitable set of vertices and then counting copies of H induced by this set. Since the edge cover number, $\beta(H)$, equals $\lceil h/2 \rceil$ when H is either $\mathcal{K}_h$ or $\mathcal{C}_h$, our general upper bound for SUB-CNT$_H$ already implies the claimed upper bounds for CLQ-CNT$_h$ and CYC-CNT$_h$ for even h. To improve these bounds in the case of odd h – in particular, TRI-CNT – we bring in some combinatorial ideas from Eden et al. [13], who were interested in a query complexity version of TRI-CNT; an overview of these ideas appears at the start of Section 3.2. We believe that the resulting algorithms are conceptually novel in a streaming context.

Braverman et al. [7] introduced the parameter *triangle density*, ρ, defining it to be the number of vertices that belong to some triangle in G. They conjectured a lower bound of $\Omega(m/\rho)$ for multi-pass TRIANGLE-COUNTING. Our Theorem 4.12 settles this conjecture positively. Further, they posed the problem of designing a multi-pass algorithm for TRIANGLE-COUNTING with space complexity depending only on m and T (presumably they meant one that beats the one-pass upper bound of Kane et al. [21]). We address this in Algorithm 2. Pavan et al. [28] introduced the *tangle coefficient*, γ; see Section 2. Our results in Theorem 4.13 and Corollary 4.14 show that each of their one-pass upper bounds – namely, $\widetilde{O}(m\gamma/T)$ and $O(m\Delta/T)$ – is matched by a multi-pass lower bound.

Jha et al. [18] gave an $O(m/\sqrt{T})$-space algorithm for a variant of TRI-CNT where the error guarantee is additive; they also showed how to multiplicatively estimate a related quantity called the *clustering coefficient*. While related, these results are not directly comparable to ours, or to the ones in Table 1. Very recently, McGregor et al. [24] extended some ideas from this work of Jha et al., obtaining an $\widetilde{O}(m/\sqrt{T})$-space algorithm for TRI-CNT. In the same paper, they also gave an $\widetilde{O}(m^{3/2}/T)$-space 4-pass[1] algorithm for TRI-CNT, which matches the space bound of our algorithm for TRI-CNT (Corollary 3.9). However, their algorithm relies on a more complex primitive of fast ℓ_p sampling and is not immediately generalizable to counting larger subgraphs. Our algorithm, which uses only basic sampling, is arguably simpler, solves a more general problem, and is space-optimal for counting cliques and odd cycles in some parameter regimes.

On the lower bound side, at a high level we proceed along the expected lines of reducing from the INDEX and SET-DISJOINTNESS communication problems, for one-pass and multi-pass bounds, respectively. The meat of the work is in designing appropriate gadgets, mostly based on Turán graphs, for the reductions. The most closely-related work is by Cormode and Jowhari [11], who give multi-pass $\Omega(m/T^{2/3})$ and $\Omega(m/\sqrt{T})$ lower bounds[2] for TRI-CNT. Their constructions imply these lower bounds for specific settings of the edge-density and number-of-triangles parameters. Our own lower bounds for TRI-CNT (Corollary 4.11) apply at all edge densities between $m = \Theta(n)$ and $m = \Theta(n^2)$ and at all triangle counts between $T = 1$ and $T = m^{3/2-\delta}$. Moreover, our bounds generalize to CLQ-CNT and CYC-CNT.

2 Preliminaries

Throughout this paper, our input graph will be $G = (V, E)$, a simple undirected graph with $|V| = n$ and $|E| = m$. For a vertex $v \in V$, N_v denotes its set of neighbors and $d_v = |N_v|$ denotes its degree. We put $\Delta = \max_{v \in V} d_v$. The input graph is presented as a stream of edges $(e_1, e_2, \ldots, e_m)$ in some adversarial order. Each edge in E appears exactly once and the stream only builds up the graph: there are no edge deletions. This is sometimes called the cash-register streaming model.

For an edge $e \in E$, slightly abusing notation, N_e denotes the set of edges in E that are adjacent to e. Let $N_e^>$ be the set of edges in N_e that come after e in the streaming order; thus $N_e^> \subseteq N_e$. Let $\mathcal{T}$ denote the set of triangles in G. We now define some special graph parameters that are useful in the context of subgraph counting algorithms: the first two were introduced in the context of the TRI-CNT problem.

▶ **Definition 2.1.** The *triangle density* [7] $\rho = \rho(G)$ is the number of vertices that belong to some triangle in G. With $T = |\mathcal{T}|$, we have $\Theta(T^{1/3}) \leqslant \rho \leqslant 3T$, where the lower and upper bounds correspond to a clique and T vertex-disjoint triangles, respectively.

▶ **Definition 2.2.** Suppose $T = |\mathcal{T}| > 0$. For $\tau \in \mathcal{T}$, let $e(\tau)$ be the edge in τ appearing earliest in the stream order. The *tangle coefficient* [28] of the stream presentation of G, denoted $\gamma = \gamma(G)$, is defined as $\gamma = (1/T) \sum_{\tau \in \mathcal{T}} |N_{e(\tau)}^>|$. Clearly, $\gamma \leqslant 2\Delta$, since $|N_{e(\tau)}^>| \leqslant 2\Delta$.

▶ **Definition 2.3.** An edge cover of a graph is a set of edges that covers all the vertices. For a graph H, its *edge cover number*, denoted $\beta(H)$, is the cardinality of its smallest edge cover.

[1] McGregor et al. state their result as a 3-pass algorithm in a nonstandard model where vertex degrees can be queried for free. In this model, our algorithm for TRI-CNT would also use only 3 passes.

[2] The Cormode–Jowhari lower bound of $\Omega(m/\sqrt{T})$, does not contradict our upper bound of $\widetilde{O}(m^{3/2}/T)$ because their lower bound holds only at $m = \Theta(n\sqrt{T})$ and $T \leqslant n^2$ triangles.

We fix a total ordering on the vertices of G according to their degrees. For vertices u and v, let $u \prec v$ if either $d_u < d_v$, or $d_u = d_v$ and $u < v$ lexicographically. We record an important observation made in Eden et al. [13].

▶ **Fact 2.4** (Eden et al. [13]). *For each $u \in V$, $|\{v \in N_u : u \prec v\}| \leqslant \sqrt{2m}$.*

Let Q be some nonnegative function of an input stream that we wish to estimate. Let $\varepsilon, \delta \in [0,1]$ be certain parameters. If an algorithm $\mathcal{A}$ produces an estimate $\hat{Q}$ for Q such that $\Pr[\hat{Q} \in (1 \pm \varepsilon)Q] \geqslant 1 - \delta$, then $\mathcal{A}$ is called an (ε, δ)-*estimator* for Q. Our algorithms will follow the common strategy of designing an unbiased "basic estimator" for Q – i.e., a random variable with expectation Q – and bounding its variance. We note the following widely-used lemma that combines several such basic estimators (computed in parallel) into an (ε, δ)-estimator.

▶ **Lemma 2.5** (Median-of-Means Improvement [2, 9]). *Let X be the distribution of an unbiased estimator for a real quantity Q. Let $\{X_{ij}\}_{i \in [t], j \in [k]}$ be a collection of i.i.d. copies of X, where $t = c \log(1/\delta)$ and $k = 3\operatorname{Var}[X]/(\varepsilon^2 \mathbb{E}[X]^2)$, for a certain universal positive constant c. Let $Z = \mathit{median}_{i \in [t]}\left(\frac{1}{k}\sum_{j=1}^{k} X_{ij}\right)$. Then $\Pr[Z \in (1 \pm \varepsilon)Q] \geqslant 1 - \delta$.*

3 Algorithms for Counting Subgraphs

In this section we present multi-pass algorithms for SUB-CNT$_H$, the problem of estimating the number of occurrences of a fixed subgraph H of constant order. We first consider general H and give a 2-pass algorithm. When specialized to $\mathcal{K}_h$ and $\mathcal{C}_h$, this algorithm uses $\widetilde{O}(m^{\lceil h/2 \rceil}/T)$ space. Later, for the case of odd $h = 2\ell + 1$, we introduce additional combinatorial ideas to improve the exponent of m from $\ell + 1$ to $\ell + \frac{1}{2}$, at the cost of two additional passes. In particular, this gives us an $\widetilde{O}(m^{3/2}/T)$-space TRI-CNT algorithm.

3.1 A Sampling-Based Algorithm for Arbitrary Subgraphs

▶ **Theorem 3.1.** *Let H be a graph of constant order whose edge cover number is β. There is an $(\varepsilon, 1/3)$-estimator for* SUB-CNT$_H$ *that uses two passes and $\widetilde{O}(S)$ space, provided $S = \Omega(m^\beta/T)$, where T is the number of distinct occurrences of H in the input graph.*

▶ Remark. The above bound could instead have been stated as $\widetilde{O}(m^\beta/T)$ with T being a promised lower bound on the number of distinct occurrences of H. Similar remarks apply to our other upper bounds. We remind the reader that $\widetilde{O}(\cdot)$ hides $1/\varepsilon^2$ and $\log m$ factors.

Proof. Let $V(H)$ and $E(H)$ be the vertex set and edge set of H, respectively. Let ξ be the number of *lists* of distinct edges of H that form minimum-sized edge covers of H.[3] Note that β and ξ are constants, independent of the input graph G. Therefore the following algorithm, which reads a stream of the m edges of G and computes an estimator X, uses $\widetilde{O}(1)$ space.

The analysis of Algorithm 1 is handled by the next two lemmas, which show that X is an unbiased estimator and that its variance can be controlled. Let $H_1, H_2, \ldots, H_T$ be the occurrences of H in G. Let T_i be an indicator random variable to denote whether H_i is detected in Pass 2, at Line 4. Then $X = \frac{m^\beta}{\xi}\sum_{i=1}^{T} T_i$.

▶ **Lemma 3.2.** *For each i, $\mathbb{E}[T_i] = \xi/m^\beta$. Thus, $\mathbb{E}[X] = T$.*

[3] E.g., if H is $\mathcal{C}_4$, with edges a, b, c, d in cyclic order, these lists are (a,c), (b,d), (c,a), and (d,b); so $\xi = 4$.

Algorithm 1 Basic estimator for SUB-CNT$_H$.

```
    Pass 1:
1:    select $\beta$ edges $\{e_i = \{u_i, v_i\}\}_{i=1}^{\beta}$ independently and u.a.r., using reservoir sampling
2:    if $\{u_1, v_1, \ldots, u_\beta, v_\beta\}$ does not have exactly $|V(H)|$ distinct vertices then
3:        $X \leftarrow 0$; abort
    Pass 2:
4:    $c \leftarrow$ number of distinct copies of $H$ on $\{u_1, v_1, \ldots, u_\beta, v_\beta\}$ that contain each of $e_1, \ldots, e_\beta$
5:    $X \leftarrow cm^\beta/\xi$
```

▶ **Lemma 3.3.** $\mathrm{Var}[X] = O(m^\beta T)$.

Proof. By Lemma 3.2,

$$\mathrm{Var}[X] \leqslant \mathbb{E}\left[X^2\right] = \frac{m^{2\beta}}{\xi^2}\left(\sum_{i=1}^{T}\mathbb{E}\left[T_i^2\right] + \sum_{i\neq j}\mathbb{E}\left[T_iT_j\right]\right) = \frac{m^{2\beta}}{\xi^2}\left(\frac{T\xi}{m^\beta} + \sum_{i\neq j}\mathbb{E}\left[T_iT_j\right]\right) . \quad (1)$$

The term $\mathbb{E}[T_iT_j]$, with $i \neq j$, is nonzero iff H_i and H_j can be simultaneously detected at Line 4. Examining the logic of the algorithm, we see that this can happen only if $V(H_i) = V(H_j)$ and there is a set of β edges that forms a minimum edge cover of both H_i and H_j. When these conditions hold, we shall say that H_i and H_j are *neighbors*. Since H is a constant-order graph, each H_i has $O(1)$ many neighbors: a crude bound, but one that suffices for our purposes.

Thus, in the double summation in (1), only $O(T)$ terms are nonzero. For each nonzero term, we have $\mathbb{E}\left[T_iT_j\right] = \Pr\left[T_i = 1 \wedge T_j = 1\right] \leqslant \Pr[T_i = 1] = \frac{\xi}{m^\beta}$. Plugging this into (1), we obtain our required estimate. ◀

To complete the proof of Theorem 3.1, we invoke Lemma 2.5 on the unbiased estimator X and use the above bound on $\mathrm{Var}[X]$. ◀

Let us specialize the above result to the problems CLQ-CNT$_h$ and CYC-CNT$_h$. We have $\beta(\mathcal{K}_h) = \beta(\mathcal{C}_h) = \lceil h/2 \rceil$. Therefore, Theorem 3.1 gives us a 2-pass estimator that uses $\widetilde{O}(S)$ space, provided $S = \Omega(m^{\lceil h/2 \rceil}/T)$, where T is the number of h-cliques or h-cycles (as appropriate) in the input graph. We shall later show, in Theorem 4.2, that this bound is optimal for CLQ-CNT$_h$ when h is even.

3.2 An Improved Algorithm for Odd Cliques and Odd Cycles

We now present an algorithm for CLQ-CNT$_{2\ell+1}$, for constant ℓ, improving the space bound from $\widetilde{O}(m^{\ell+1}/T)$, as implied by Theorem 3.1, to $\widetilde{O}(m^{\ell+\frac{1}{2}}/T)$. As before, all space bounds are for an $(\varepsilon, 1/3)$-estimator.

Our algorithm builds on ideas from Pavan et al. [28] and Eden et al. [13]. The former paper gives a streaming algorithm for estimating the number of triangles T in a graph. The idea is to sample an edge uniformly at random from the stream, using reservoir sampling; then sample one more edge uniformly at random from the neighborhood of previously chosen edge; and finally, check whether these two edges are closed by any edge in the "future stream" to form a triangle. This leads to an unbiased estimator for T with variance bounded by $O(m\Delta T)$. This leads to an $\widetilde{O}(m\Delta/T)$-space $(\varepsilon, 1/3)$-estimator, with the caveat that prior knowledge of Δ is required. We build on their algorithm by improving the bound on the variance of the unbiased estimator to $O(m^{3/2}T)$. This gives an $\widetilde{O}(m^{3/2}/T)$ space estimator for TRI-CNT as well as removes the dependency on Δ. We reduce the variance of our estimator by repeating the neighborhood sampling step for edges whose endpoints have "large" degree.

The challenge now is to reduce the number of triangles that an edge participates in. For this, we use an idea of vertex ordering from Eden et al. [13], who tackled triangle counting in a property testing model. They fix a total ordering on the vertices of G according to their degrees. For vertices u and v, let $u \prec v$ if either $d_u < d_v$, or $d_u = d_v$ and $u < v$ lexicographically. Now let $\tau = \{v_1, v_2, v_3\}$ be a triangle in G with $v_1 \prec v_2 \prec v_3$. Then τ can be associated with (e_1, v_3) where $e_1 = \{v_1, v_2\}$. Observe that each triangle τ in G is uniquely associated with a distinct edge. Let the number of triangles associated with edge e be T_e. Clearly, $\sum_{e \in E(G)} T_e = T$. From Fact 2.4, it follows that $T_e \leqslant \sqrt{2m}$. Since each edge is associated with not-too-many triangles, we get a "strong" upper bound on the variance. In fact the idea of vertex ordering has been proved to be useful for counting triangles in other (offline) settings as well [30]. By invoking a result from Chiba et al. [10], we show that in spite of such repetition the space usage for our estimator remains constant in expectation. We in fact show that we can generalize this idea for larger cliques.

Now we formally describe our estimator for CLQ-CNT. We fix a total ordering of $V(G)$ as described above. Let $\tau = \{v_1, v_2, \ldots, v_{2\ell}, v_{2\ell+1}\}$ induce a $\mathcal{K}_{2\ell+1}$ in G with $v_1 \prec v_2 \prec \ldots \prec v_{2\ell} \prec v_{2\ell+1}$. We associate τ with $(e_1, e_2, \ldots, e_\ell, v_{2\ell+1})$ where $e_i = \{v_{2i-1}, v_{2i}\}$. Observe that each $\mathcal{K}_{2\ell+1}$ in G is uniquely associated with ℓ distinct edges. Let the number of $(2\ell+1)$-cliques associated with $(e_1, e_2, \ldots, e_\ell)$ be $T_{(e_1,e_2,\ldots,e_\ell)}$. Then, we have the following simple lemma.

▶ **Lemma 3.4.** *Each $T_{(e_1,e_2,\ldots,e_\ell)} \leqslant \sqrt{2m}$. Further, $\sum_{(e_1,e_2,\ldots,e_\ell) \in E(G)^\ell} T_{(e_1,e_2,\ldots,e_\ell)} = T$.*

We shall also need the following combinatorial lemma, from Chiba and Nishizeki [10].

▶ **Lemma 3.5** (Based on Lemmas 1(a) and 2 of [10]). *Let $G = (V, E)$ be a graph with n vertices and m edges such that $m = \Omega(n)$. Then $\sum_{\{u,v\} \in E} \min\{d_u, d_v\} = O\left(m^{3/2}\right)$.*

Algorithm 2 computes our basic estimator X.

▶ **Theorem 3.6.** *Suppose the input graph G contains T copies of $\mathcal{K}_{2\ell+1}$. Then Algorithm 2 leads to an $\widetilde{O}(S)$-space $(\varepsilon, 1/3)$-estimator for T when $S = \Omega(m^{\ell+\frac{1}{2}}/T)$.*[4]

Proof. Let $\mathcal{E}_{(e_1,e_2,\ldots,e_\ell)}$ be the event that edges $e_1 = \{u_1, v_1\}, e_2 = \{u_2, v_2\}, \ldots, e_\ell = \{u_\ell, v_\ell\}$ are sampled at Line 1 and the algorithm does not abort in Pass 1. WLOG assume $u_i \prec v_i$ for all $i \in [\ell]$. In the next two lemmas, we shall show that X is an unbiased estimator and its variance can be controlled. Then we shall analyze the space usage of Algorithm 2.

▶ **Lemma 3.7.** *For each $\mathcal{E}_{(e_1,e_2,\ldots,e_\ell)}$, $\mathbb{E}[Z_k \mid \mathcal{E}_{(e_1,e_2,\ldots,e_\ell)}] = T_{(e_1,e_2,\ldots,e_\ell)}$. Thus, $\mathbb{E}[X] = T$.*

▶ **Lemma 3.8.** $\mathrm{Var}[X] = O(m^{\ell+1/2}T)$.

Proof. As in the previous lemma, $\mathbb{E}\left[Z_k^2 \mid \mathcal{E}_{(e_1,e_2,\ldots,e_\ell)}\right] = d_{u_1} T_{(e_1,e_2,\ldots,e_\ell)}$. Next, note that Z_{k_1} and Z_{k_2} are independent for $k_1 \neq k_2$, even after conditioning on $\mathcal{E}_{(e_1,e_2,\ldots,e_\ell)}$. Therefore

$$\mathbb{E}[Z_{k_1} Z_{k_2} \mid \mathcal{E}_{(e_1,e_2,\ldots,e_\ell)}] = \mathbb{E}[Z_{k_1} \mid \mathcal{E}_{(e_1,e_2,\ldots,e_\ell)}] \cdot \mathbb{E}[Z_{k_2} \mid \mathcal{E}_{(e_1,e_2,\ldots,e_\ell)}] = T^2_{(e_1,e_2,\ldots,e_\ell)} . \quad (2)$$

[4] Please see the remark after Theorem 3.1 for an alternate interpretation of the space bound.

Algorithm 2 Basic estimator for CLQ-CNT$_{2\ell+1}$.

```
Pass 1:
    select $\ell$ edges $\{e_i = \{u_i, v_i\}\}_{i=1}^{\ell}$ independently and u.a.r., using reservoir sampling
    if $\{u_1, v_1, \ldots, u_\ell, v_\ell\}$ does not have exactly $2\ell$ distinct vertices then
        $X \leftarrow 0$;  abort
Pass 2:
    count $d_{u_i}$ and $d_{v_i}$ for all $i \in [\ell]$
Pass 3:
    $r \leftarrow \lceil \min\{d_{u_1}, d_{v_1}\} / \sqrt{m} \rceil$
    by renaming vertices if needed, ensure that $u_i \prec v_i$ for all $i \in [\ell]$
    for $k \leftarrow 1$ to $r$ do
        $Z_k \leftarrow 0$;  select a vertex $w_k$ from $N_{u_1}$ u.a.r., using reservoir sampling
Pass 4:
    compute $d_{w_1}, \ldots, d_{w_r}$
    for $k \leftarrow 1$ to $r$ do
        if $(e_1, \ldots, e_\ell, w_k)$ forms a $\mathcal{K}_{2\ell+1}$ and $u_1 \prec v_1 \prec u_2 \prec \ldots \prec u_\ell \prec v_\ell \prec w_k$ then
            $Z_k \leftarrow d_{u_1}$
    $Y \leftarrow (1/r)\sum_{k=1}^{r} Z_k$
    $X \leftarrow m^\ell Y$
```

Now, using Lemma 3.7,

$$\begin{aligned}\mathbb{E}\left[Y^2 \mid \mathcal{E}_{(e_1,e_2,\ldots,e_\ell)}\right] &= \frac{1}{r^2}\mathbb{E}\left[\left(\sum_{k=1}^{r} Z_k\right)^2 \,\middle|\, \mathcal{E}_{(e_1,e_2,\ldots,e_\ell)}\right] \\ &= \frac{1}{r^2}\sum_{k=1}^{r}\mathbb{E}\left[Z_k^2 \mid \mathcal{E}_{(e_1,e_2,\ldots,e_\ell)}\right] + \frac{1}{r^2}\sum_{k_1\neq k_2}\mathbb{E}\left[Z_{k_1}Z_{k_2} \mid \mathcal{E}_{(e_1,e_2,\ldots,e_\ell)}\right] \\ &\leqslant \frac{d_{u_1}}{r}T_{(e_1,e_2,\ldots,e_\ell)} + T^2_{(e_1,e_2,\ldots,e_\ell)} = \sqrt{m}\,T_{(e_1,e_2,\ldots,e_\ell)} + T^2_{(e_1,e_2,\ldots,e_\ell)}\,.\end{aligned}$$

Therefore,

$$\begin{aligned}\operatorname{Var}[X] = m^{2\ell}\operatorname{Var}[Y] \leqslant m^{2\ell}\mathbb{E}[Y^2] &= m^{2\ell}\cdot\frac{1}{m^\ell}\sum_{(e_1,e_2,\ldots,e_\ell)\in E^\ell}\mathbb{E}\left[Y^2 \mid \mathcal{E}_{(e_1,e_2,\ldots,e_\ell)}\right] \\ &\leqslant m^{\ell+1/2}\sum_{(e_1,\ldots,e_\ell)\in E^\ell}T_{(e_1,\ldots,e_\ell)} + m^\ell\sum_{(e_1,\ldots,e_\ell)\in E^\ell}T^2_{(e_1,\ldots,e_\ell)} = (1+\sqrt{2})\,m^{\ell+1/2}T\,,\end{aligned}$$

where the final step uses both parts of Lemma 3.4. ◀

We return to the proof of Theorem 3.6. Invoking Lemma 2.5, we get an $(\varepsilon, 1/3)$-approximation algorithm for T with space $\widetilde{O}(m^{\ell+\frac{1}{2}}\cdot B/T)$ bits, where B is the space used by Algorithm 2. To estimate B, note that Algorithm 2 keeps $r = \lceil \min\{d_{u_1}, d_{v_1}\} / \sqrt{m} \rceil$ many neighboring vertices of u_1. Recall that the edge $\{u_1, v_1\}$ is chosen uniformly at random. Hence the expected space usage is

$$\frac{1}{m}\sum_{\{u,v\}\in E(G)}\left\lceil\frac{\min\{d_u, d_v\}}{\sqrt{m}}\right\rceil \leqslant 1 + \frac{1}{m}\sum_{\{u,v\}\in E(G)}\frac{\min\{d_u, d_v\}}{\sqrt{m}} = O(1)\,,$$

where the final step uses Lemma 3.5.

We can easily turn this expected space bound – call it B_0 – into a worst-case space bound that holds w.h.p. We simply abort the algorithm if we find that r would exceed $10B_0$ (say). By Markov's inequality, this ensures that $B \leqslant 10B_0 = O(1)$, with probability at least $\frac{9}{10}$.

Thus, the overall space usage of our final estimator is $\widetilde{O}(m^{\ell+\frac{1}{2}}/T)$, as required. ◀

Given the importance of the problem of counting triangles, it is worthwhile to record the following immediate corollary of Theorem 3.6.

▶ **Corollary 3.9.** *For a graph with n vertices, m edges, and T triangles, the* TRI-CNT *problem admits a four-pass $\widetilde{O}(S)$-space $(\varepsilon, 1/3)$-estimator when $S = \Omega\big(m^{3/2}/T\big)$.*

A similar improvement is possible for counting odd cycles of constant order. Details, including a pseudocode description of the relevant algorithm, appear in the full paper.

▶ **Theorem 3.10.** *For a graph with m edges, and T copies of $\mathcal{C}_{2\ell+1}$, the* CYC-CNT$_{2\ell+1}$ *problem admits a four-pass $\widetilde{O}(S)$-space $(\varepsilon, 1/3)$-estimator when $S = \Omega\left(m^{\ell+\frac{1}{2}}/T\right)$.*

4 Lower Bounds

The remainder of this paper is concerned with lower bounds. Unless otherwise stated, all the lower bounds are for randomized algorithms with success probabilities at least $2/3$. We first prove multi-pass lower bounds for CLQ-CNT and CYC-CNT. The latter turns out to require different constructions for the cases of odd cycles and even cycles. In particular, this makes the lower bound for CYC-CNT$_h$ more restrictive when h is even. Next, we give single pass lower bounds for CLQ-CNT and CYC-CNT: our lower bound for even cycles is weaker than that for odd cycles. Finally, we focus on TRI-CNT and establish tight space lower bounds in terms of special structural parameters of the input graph, these parameters being ones studied in previous works on triangle counting.

For some intuition on why even cycles are harder to deal with, we recall an important result from extremal graph theory [6, 5]: for each integer $\ell \geqslant 2$, $\mathrm{ex}(n, \mathcal{C}_{2\ell}) = O(\ell n^{1+1/\ell})$, where $\mathrm{ex}(n, H) = \max\{m : \exists \text{ graph on } n \text{ vertices and } m \text{ edges that does not contain } H\}$. Our lower bounds for CLQ-CNT$_h$ and CYC-CNT$_{2\ell+1}$ depend on constructing dense $\mathcal{K}_h$-free and $\mathcal{C}_{2\ell+1}$-free graphs (respectively). However, in view of the above bound, dense $\mathcal{C}_{2\ell}$-free graphs do not exist, necessitating weaker constructions.

4.1 Multi-Pass Lower Bounds

The source of these lower bounds is the following promise version of the SET-DISJOINTNESS problem that is of central importance in communication complexity theory. Alice and Bob have two N-bit strings x and y respectively, each with exactly R ones. They want to decide whether there exists an index $i \in [N]$ such that $x_i = 1 = y_i$. Let this problem be denoted as DISJ$_N^R$. The randomized communication complexity $\mathrm{R}(\text{DISJ}_N^R) = \Omega(R)$ for all $R \leqslant N/2$ [20, 29].

Our reductions from DISJ$_N^R$ naturally lead to lower bounds on a clique *detection* problem, where the goal is to distinguish input graphs G that have no h-cliques from ones that have "many" h-cliques. Note that this provides a legitimate counterpart to our upper bounds, all of which will safely report "0" on input graphs that are h-clique-free.

▶ **Definition 4.1** (Clique and cycle detection problems). Consider two graph families: $\mathcal{G}_1$ consisting of h-clique-free graphs with n vertices and m edges; $\mathcal{G}_2$ consisting of graphs with

n vertices, m edges, and at least T h-cliques. Given a streamed input graph $G \in \mathcal{G}_1 \cup \mathcal{G}_2$, $\textsc{clq-detect}_h(p,n,m,T)$ is the problem of deciding whether $G \in \mathcal{G}_1$ or $G \in \mathcal{G}_2$ with success probability at least $2/3$, using at most p passes over the edge stream. Analogously, we define the cycle detection problem $\textsc{cyc-detect}_h(p,n,m,T)$.

We shall prove the following two theorems about clique detection.

▶ **Theorem 4.2.** *For each constant h and constant p, solving $\textsc{clq-detect}_h(p,n,m,T)$ requires $\Omega\left(m^{h/2}/T\right)$ bits of space, provided $T = \Omega(m^{(h-1)/2})$. Furthermore, this holds for any $m = \Theta(n^r)$, $1 \leqslant r \leqslant 2$, and any T with $\Omega(m^{(h-1)/2}) \leqslant T \leqslant m^{(h-1)/2+\delta}$, where δ is an arbitrary constant in $[0, 1/2)$.*

▶ **Theorem 4.3.** *For each constant h, constant p, and for any T with $1 \leqslant T \leqslant m^{(h-1)/2}$, solving $\textsc{clq-detect}_h(p,n,m,T)$ requires $\Omega\left(m/T^{1/(h-1)}\right)$ bits of space.*

▶ Remark. Our lower bounds admit the possibility that there may exist more efficient algorithms when the number of cliques is relatively small in the graph.

We note that McGregor et al. [24] have given two algorithms that match our lower bounds when the subgraph of interest is a triangle (3-clique).

We present a detailed proof of Theorem 4.2 via reductions from $\textsc{disj}_N^{N/3}$; for the proof of Theorem 4.3, see the full paper. Turán graphs play a central role. Recall that a Turán graph is a complete multipartite graph where the blocks (of the vertex partition) are as close as possible to being equal in size. Let $T(n,t)$ denote an n-vertex t-partite Turán graph: it has $t - (n \bmod t)$ blocks with $\lfloor n/t \rfloor$ vertices each and another $(n \bmod t)$ blocks with $\lceil n/t \rceil$ vertices each. As is well known, $T(n,t)$ is the densest n-vertex graph that is $\mathcal{K}_{t+1}$-free.

Proof of Theorem 4.2. We reduce $\textsc{disj}_N^{N/3}$ to $\textsc{clq-detect}_h(p,n,m,T)$ by constructing a certain graph that has some fixed edges, some edges depending on Alice's input, x, and some edges depending on Bob's input, y.

Let $H = (V_H, E_H)$ be a copy of the Turán graph $T(b(h-1), h-1)$, with B_j denoting the jth block in V_H. By construction, $|B_j| = b$ for all $j \in [h-1]$, and

$$E_H = \bigcup_{i \neq j} \{\{u,v\} : \ u \in B_i, \ v \in B_j\} \ .$$

To this fixed graph H, we add N additional blocks of vertices, denoted $V_1, \ldots, V_N$, with each $|V_i| = d$. Then Alice and Bob add edge sets E_{Alice} and E_{Bob} respectively, defined as follows

$$E_{\text{Alice}} = \bigcup_{\substack{i:x_i=1\,, \\ j \in [h-2]}} \{\{u,v\} : \ u \in V_i, \ v \in B_j\} \ , \quad E_{\text{Bob}} = \bigcup_{j:y_j=1} \{\{u,v\} : \ u \in V_j, \ v \in B_{h-1}\} \ .$$

In words, for each index i with $x_i = 1$, Alice constructs a complete bipartite subgraph between V_i and B_j for all $j \in [h-2]$. Similarly, for each index j with $y_j = 1$, Bob creates a complete bipartite subgraph between V_j and B_{h-1}. Let the final resulting graph be denoted $G_{\text{clique}} = (V_{\text{clique}}, E_{\text{clique}})$ where $V_{\text{clique}} = V_H \cup (V_1 \cup \ldots \cup V_N)$, and $E_{\text{clique}} = E_H \cup E_{\text{Alice}} \cup E_{\text{Bob}}$.

▶ **Lemma 4.4.** *The graph G_{clique} is $\mathcal{K}_h$-free iff x and y are disjoint.*

Proof. Observe that graphs $G_A = (V_{\text{clique}}, \ E_H \cup E_{\text{Alice}})$ and $G_B = (V_{\text{clique}}, \ E_H \cup E_{\text{Bob}})$ are both $\mathcal{K}_h$-free. Thus, any h-clique in G_{clique} must of the form $\{v_1, \ldots, v_h\}$ where $v_1 \in B_1, \ldots, v_{h-1} \in B_{h-1}$, and $v_h \in V_i$ for some $i \in [N]$. But this implies that V_i is connected to B_i for all $i \in [h-1]$. Hence, $x_i = y_i = 1$. ◀

Next, we record some important parameters of G_{clique}. First, $|E_H| = \binom{h-1}{2} b^2 = \Theta(b^2)$. Put $n = |V_{\text{clique}}|$, $m = |E_{\text{clique}}|$. Let T_H denote the number of h-cliques in the graph. Recall that, as an instance of $\text{DISJ}_N^{N/3}$, each of x and y has exactly $N/3$ ones. Thus,

$$n = |V_H| + \sum_{i=1}^{N} |V_i| = \Theta(b) + Nd\,, \quad m = |E_H| + |E_{\text{Alice}}| + |E_{\text{Bob}}| = \Theta(b^2) + \Theta(Nbd)\,,$$
$$T_H = 0\,, \quad \text{if } x \cap y = \varnothing\,; \quad T_H \geqslant b^{h-1} d\,, \quad \text{otherwise}\,.$$

Setting $b = N$ and $d = 1$, we get $n = \Theta(N)$, $m = \Theta(N^2)$, and $T_H \geqslant N^{h-1}$ if x and y are not disjoint. Suppose that there is an algorithm $\mathcal{A}$ that solves $\text{CLQ-DETECT}_h(p, n, m, T)$ with $T = N^{h-1}$ in only $o(m^{h/2}/T)$ space, for some constant p. Then there exists a communication protocol with cost $o(R)$ that solves $\text{DISJ}_N^{N/3}$. This gives us the main result of Theorem 4.2.

The proof so far applies to a graph with $m = \Theta(n^2)$. To generalize it to arbitrary m and T, assume $m = \Theta(n^r)$, and $T = m^{(h-1)/2+\delta}$ for some fixed r, δ such that $1 \leqslant r \leqslant 2$, and $0 \leqslant \delta < 1/2$. We modify the construction of G_{clique} as follows (note that $\delta = 1/2$ gives the maximum possible number of h-cliques in a graph with m edges). We set $b = N^q$ and $d = N^{q-1}$ where $q = 1/\left(\frac{r}{2} - \delta r\right)$. Now mark $b^{r/2}$ vertices in each block B_i and $d^{\frac{qr-2}{2(q-1)}}$ vertices in each set V_i as *active* vertices. Then we only add edges between active vertices of each block. In the modified G_{clique}, we have

$$n = \Theta(N^q)\,, \quad m = \Theta(b^r) + \Theta(N b^{\frac{r}{2}} d^{\frac{qr-2}{2(q-1)}}) = \Theta(N^{qr})\,,$$
$$T_H = 0\,, \quad \text{if } x \cap y = \varnothing\,; \quad T_H \geqslant b^{(h-1)\frac{r}{2}} d^{\frac{qr-2}{2(q-1)}} = \Theta\left(N^{\frac{hqr}{2}-1}\right)\,, \quad \text{otherwise}\,.$$

Plugging in $q = 1/\left(\frac{r}{2} - \delta r\right)$, we get $T_H = \Theta\left(m^{(h-1)/2+\delta}\right)$ when x and y are not disjoint. The lower bound of $\Omega(N)$ for $\text{DISJ}_N^{N/3}$ implies a lower bound of $\Omega\left(m^{h/2}/T\right)$ for $\text{CLQ-DETECT}_h(p, n, m, T)$ with $T = m^{(h-1)/2+\delta}$. ◀

In the full paper, we prove the following lower bounds for $\text{CYC-DETECT}_h(p, n, m, T)$. The first two, for odd h, are analogous to Theorem 4.2 and Theorem 4.3, except that Turán graphs are replaced with an appropriate dense gadget. The third lower bound, for even h, uses a different, "sparse" gadget, leading to a more restrictive lower bound.

▶ **Theorem 4.5.** *For each odd constant h and constant p, solving $\text{CYC-DETECT}_h(p, n, m, T)$ requires $\Omega\left(m^{h/2}/T\right)$ bits of space, provided $T = \Omega(m^{(h-1)/2})$. Furthermore, this holds for any $m = \Theta(n^r)$, $1 \leqslant r \leqslant 2$, and any T with $\Omega(m^{(h-1)/2}) \leqslant T \leqslant m^{(h-1)/2+\delta}$, where δ is an arbitrary constant in $[0, 1/2)$.*

▶ **Theorem 4.6.** *For each odd constant h, constant p, and for any T with $1 \leqslant T \leqslant m^{(h-1)/2}$, solving $\text{CYC-DETECT}_h(p, n, m, T)$ requires $\Omega\left(m/T^{1/(h-1)}\right)$ bits of space.*

▶ **Theorem 4.7.** *For each even constant h and constant p, there is a family of instances with $m = \Theta(n)$ and $T = \Theta(m^{(h-2)/2})$, such that solving $\text{CYC-DETECT}_h(p, n, m, T)$ requires $\Omega\left(m^{h/2}/T\right)$ bits of space.*

4.2 Single Pass Lower Bounds

We also obtain one-pass streaming lower bounds for the special subgraph counting problems studied in the previous section. Proofs appear in the full paper. These bounds use reductions from the INDEX_N communication problem: Alice has a N-bit string, x, and Bob has an index $z \in [N]$. The goal is to output the bit x_z. The one-way randomized communication complexity $\mathrm{R}^{\rightarrow}(\textsc{index}_N) = \Omega(N)$ [1].

Lower bounds for CLQ-CNT and CYC-CNT are obtained by studying the corresponding detection problems CLQ-DETECT and CYC-DETECT. As before, $\textsc{cyc-detect}_h$ is treated differently for odd h and even h. In each of these theorems, h is a constant.

Theorem 4.8 and Theorem 4.10 have the weakness that they apply only at carefully chosen parameter settings, à la Theorem 4.7. A more thorough treatment of these theorems is deferred to the full paper.

▶ **Theorem 4.8.** *Solving* $\textsc{clq-detect}_h(1, n, m, T)$ *requires* $\Omega(m^{h-\varepsilon}/T^2)$ *space for every small constant* $\varepsilon > 0$.

▶ **Theorem 4.9.** *For odd* h*, solving* $\textsc{cyc-detect}_h(1, n, m, T)$ *requires* $\Omega(m^h/T^2)$ *space.*

▶ **Theorem 4.10.** *For even* h*,* $\textsc{cyc-detect}_h(1, n, m, T)$ *requires* $\Omega\left(m^{h/2}/T\right)$ *space.*

4.3 Special Lower Bounds for Triangle Counting

Finally, we present some tight multi-pass space lower bounds for TRI-CNT in terms of graph structural parameters introduced in previous works. Analogous to Definition 4.1, we define $\textsc{tri-detect}(p, n, m, T)$, where the goal is to distinguish between graphs from "no triangles" family and "at least T triangles" family. $\textsc{tri-detect-density}(p, n, m, T, \rho)$, $\textsc{tri-detect-tangle}(p, n, m, T, \gamma)$, and $\textsc{tri-detect-degree}(p, n, m, T, \Delta)$ are variants of this problem where the triangle density ρ, the tangle coefficient γ, and maximum degree Δ (respectively) are supplied as parameters.

In each of the following theorems, the number of passes, p, is a constant.

As a direct consequence of Theorem 4.2 and Theorem 4.3, we have the following basic lower bound.

▶ **Corollary 4.11.** *Solving* $\textsc{tri-detect}(p, n, m, T)$ *requires* $\Omega\left(\min\left\{m/T^{2/3}, m/\sqrt{T}\right\}\right)$ *space.*

We can prove the following lower bounds for other variants of TRI-DETECT by reductions from $\textsc{disj}_N^R$ using suitable gadgets. Details appear in the full paper.

▶ **Theorem 4.12.** *Solving* $\textsc{tri-detect-density}(p, n, m, T, \rho)$ *requires* $\Omega(m/\rho)$ *space.*

▶ **Theorem 4.13.** *Solving* $\textsc{tri-detect-tangle}(p, n, m, T, \gamma)$ *requires* $\Omega(m\gamma/T)$ *space.*

▶ **Theorem 4.14.** *Solving* $\textsc{tri-detect-degree}(p, n, m, T, \Delta)$ *requires* $\Omega(m\Delta/T)$ *space.*

5 Concluding Remarks

In this paper, we have made several advances in our understanding of the space complexity of subgraph counting problems. Nevertheless, a number of key problems remain open and we end by highlighting some significant ones.

- Consider the data streaming problems $\textsc{clq-cnt}_h$ (for arbitrary constant h) and $\textsc{cyc-cnt}_h$ (for odd constant h), using a constant number of passes. In each case, we have given a space lower bound of $\Omega\big(\min\left\{m^{h/2}/T,\ m/T^{1/(h-1)}\right\}\big)$ and an upper bound of $\widetilde{O}(m^{h/2}/T)$. Suppose that T, the actual number of cliques or cycles (as applicable) in the input graph, is relatively small: to be precise, suppose that $T \leqslant m^{(h-1)/2}$. In this regime, there is a gap between the upper and lower bounds, as discussed after Theorem 4.3. Can we design a constant-pass algorithm using $\widetilde{O}\big(m/T^{1/(h-1)}\big)$ space?

- We have proved a one-pass lower bound of $\Omega(m^h/T^2)$ for CLQ-DETECT$_h$. The best known one-pass upper bound for CLQ-CNT$_h$ is $\widetilde{O}\big(m^{h(h-1)/2}/T^2\big)$ [21]. Bridging this gap remains an open problem. The situation for cycle counting is better: the upper bound of $\widetilde{O}\big(m^h/T^2\big)$ for CYC-CNT$_h$ [23] matches our lower bound up to a logarithmic factor, when h is odd.
- Can one improve the one-pass and multi-pass lower bounds for CYC-CNT$_h$ for even h to match those for odd h? Since it is impossible to construct a "dense" graph without creating even cycles, one may hope that there exist more efficient algorithms for counting even cycles. It would be very interesting to settle the problem either way.

References

1 Farid Ablayev. Lower bounds for one-way probabilistic communication complexity and their application to space complexity. *Theor. Comput. Sci.*, 175(2):139–159, 1996.

2 Noga Alon, Yossi Matias, and Mario Szegedy. The space complexity of approximating the frequency moments. *J. Comput. Syst. Sci.*, 58(1):137–147, 1999. Preliminary version in *Proc. 28th Annual ACM Symposium on the Theory of Computing*, pages 20–29, 1996.

3 Ziv Bar-Yossef, Ravi Kumar, and D. Sivakumar. Reductions in streaming algorithms, with an application to counting triangles in graphs. In *Proc. 13th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 623–632, 2002.

4 Luca Becchetti, Paolo Boldi, Carlos Castillo, and Aristides Gionis. Efficient semi-streaming algorithms for local triangle counting in massive graphs. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Las Vegas, Nevada, USA, August 24-27, 2008*, pages 16–24, 2008.

5 Béla Bollobás. *Extremal Graph Theory*. Academic Press, New York, NY, 1978.

6 John A. Bondy and Miklós Simonovits. Cycles of even length in graphs. *Journal of Combinatorial Theory, Series B*, pages 97–105, 1974.

7 Vladimir Braverman, Rafail Ostrovsky, and Dan Vilenchik. How hard is counting triangles in the streaming model? In *Proc. 40th International Colloquium on Automata, Languages and Programming*, pages 244–254, 2013.

8 Luciana S. Buriol, Gereon Frahling, Stefano Leonardi, Alberto Marchetti-Spaccamela, and Christian Sohler. Counting triangles in data streams. In *Proc. 25th ACM Symposium on Principles of Database Systems*, pages 253–262, 2006.

9 Amit Chakrabarti. CS49: Data Stream Algorithms Lecture Notes, Fall 2011. URL: `http://www.cs.dartmouth.edu/~ac/Teach/data-streams-lecnotes.pdf`.

10 Norishige Chiba and Takao Nishizeki. Arboricity and subgraph listing algorithms. *SIAM J. Comput.*, 14(1):210–223, 1985.

11 Graham Cormode and Hossein Jowhari. A second look at counting triangles in graph streams. *Theoretical Computer Science*, 552:44–51, 2014.

12 David Easley and Jon Kleinberg. *Networks, crowds, and markets: Reasoning about a highly connected world*. Cambridge University Press, 2012.

13 Talya Eden, Amit Levi, Dana Ron, and C. Seshadhri. Approximately counting triangles in sublinear time. In *Proc. 56th Annual IEEE Symposium on Foundations of Computer Science*, pages 614–633, 2015.

14 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. Preliminary version in *Proc. 31st International Colloquium on Automata, Languages and Programming*, pages 531–543, 2004.

15 David Garcıa-Soriano and Konstantin Kutzkov. Triangle counting in streamed graphs via small vertex covers. *Tc*, 2:3, 2014.

16 Oded Goldreich. A brief introduction to property testing. In *Studies in Complexity and Cryptography. Miscellanea on the Interplay between Randomness and Computation – In Collaboration with Lidor Avigad, Mihir Bellare, Zvika Brakerski, Shafi Goldwasser, Shai Halevi, Tali Kaufman, Leonid Levin, Noam Nisan, Dana Ron, Madhu Sudan, Luca Trevisan, Salil Vadhan, Avi Wigderson, David Zuckerman*, pages 465–469. Springer, 2011. doi:10.1007/978-3-642-22670-0_31.

17 Oded Goldreich. Introduction to testing graph properties. In *Studies in Complexity and Cryptography. Miscellanea on the Interplay between Randomness and Computation – In Collaboration with Lidor Avigad, Mihir Bellare, Zvika Brakerski, Shafi Goldwasser, Shai Halevi, Tali Kaufman, Leonid Levin, Noam Nisan, Dana Ron, Madhu Sudan, Luca Trevisan, Salil Vadhan, Avi Wigderson, David Zuckerman*, pages 470–506. Springer, 2011. doi:10.1007/978-3-642-22670-0_32.

18 Madhav Jha, C. Seshadhri, and Ali Pinar. A space efficient streaming algorithm for triangle counting using the birthday paradox. In *Proc. 19th Annual SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 589–597, 2013.

19 Hossein Jowhari and Mohammad Ghodsi. New streaming algorithms for counting triangles in graphs. In *Computing and Combinatorics*, pages 710–716. Springer, 2005.

20 Bala Kalyanasundaram and Georg Schintger. The probabilistic communication complexity of set intersection. *SIAM J. Disc. Math.*, 5(4):545–557, 1992.

21 Daniel M Kane, Kurt Mehlhorn, Thomas Sauerwald, and He Sun. Counting arbitrary subgraphs in data streams. In *Proc. 39th International Colloquium on Automata, Languages and Programming*, pages 598–609, 2012.

22 Konstantin Kutzkov and Rasmus Pagh. Triangle counting in dynamic graph streams. In *Proc. 14th Scandinavian Symposium and Workshops on Algorithm Theory*, pages 306–318, 2014.

23 Madhusudan Manjunath, Kurt Mehlhorn, Konstantinos Panagiotou, and He Sun. Approximate counting of cycles in streams. In *Proc. 19th Annual European Symposium on Algorithms*, pages 677–688, 2011.

24 Andrew McGregor, Sofya Vorotnikova, and Hoa 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*, pages 401–411, 2016.

25 S. Muthukrishnan. Data streams: Algorithms and applications. *Found. Trends Theor. Comput. Sci.*, 1(2):117–236, 2005. doi:10.1561/0400000002.

26 Mark E. J. Newman. The structure and function of complex networks. *SIAM Review*, 45(2):167–256, 2003. doi:10.1137/S003614450342480.

27 Rasmus Pagh and Charalampos E. Tsourakakis. Colorful triangle counting and a mapreduce implementation. *Information Processing Letters*, 112(7):277–281, 2012.

28 Aduri Pavan, Kanat Tangwongsan, Srikanta Tirthapura, and Kun-Lung Wu. Counting and sampling triangles from a graph stream. *Proceedings of the VLDB Endowment*, 6(14):1870–1881, 2013.

29 Alexander A. Razborov. On the distributional complexity of disjointness. *Theor. Comput. Sci.*, 106(2):385–390, 1992.

30 Siddharth Suri and Sergei Vassilvitskii. Counting triangles and the curse of the last reducer. In *Proceedings of the 20th international conference on World wide web*, pages 607–614, 2011.

31 Stanley Wasserman and Katherine Faust. *Social network analysis: Methods and applications*, volume 8. Cambridge university press, 1994.