Streaming and Communication Complexity of Clique Approximation

Magnús M. Halldórsson[1,⋆], Xiaoming Sun[2,⋆⋆],
Mario Szegedy[3,⋆⋆⋆], and Chengu Wang[4,†]

[1] ICE-TCS, School of Computer Science, Reykjavik University, Iceland
[2] Institute of Computing Technology, Chinese Academy of Sciences
[3] Department of Computer Science, Rutgers University, New Jersey
[4] Institute for Interdisciplinary Information Sciences, Tsinghua University, Beijing

Abstract. We consider the classic clique (or, equivalently, the independent set) problem in two settings. In the streaming model, edges are given one by one in an adversarial order, and the algorithm aims to output a good approximation under space restrictions. In the communication complexity setting, two players, each holds a graph on n vertices, and they wish to use a limited amount of communication to distinguish between the cases when the union of the two graphs has a low or a high clique number. The settings are related in that the communication complexity gives a lower bound on the space complexity of streaming algorithms.

We give several results that illustrate different tradeoffs between clique separability and the required communication/space complexity under randomization. The main result is a lower bound of $\Omega(\frac{n^2}{r^2 \log^2 n})$-space for any r-approximate randomized streaming algorithm for maximum clique. A simple random sampling argument shows that this is tight up to a logarithmic factor. For the case when $r = o(\log n)$, we present another lower bound of $\Omega(\frac{n^2}{r^4})$. In particular, it implies that any constant approximation randomized streaming algorithm requires $\Omega(n^2)$ space, even if the algorithm runs in exponential time. Finally, we give a third lower bound that holds for the extremal case of $s - 1$ vs. $\mathcal{R}(s) - 1$, where $\mathcal{R}(s)$ is the s-th Ramsey number. This is the extremal setting of clique numbers that can be separated. The proofs involve some novel combinatorial structures and sophisticated combinatorial constructions.

1 Introduction

Streaming for cliques. In the *streaming model* for graph problems, edges are presented sequentially in the form of a data stream, and the objective is to

[⋆] Research partially supported by Icelandic Research Fund grant 90032021.

[⋆⋆] Research partially supported by the National Natural Science Foundation of China Grant 61170062, 61061130540, and the National Basic Research Program of China Grant 2011CBA00300, 2011CBA00301.

[⋆⋆⋆] Research partially supported by NSF grant CCF-0832787.

[†] Research partially supported by the National Basic Research Program of China Grant 2011CBA00300, 2011CBA00301, the National Natural Science Foundation of China Grant 61033001, 61061130540, 61073174.

A. Czumaj et al. (Eds.): ICALP 2012, Part I, LNCS 7391, pp. 449–460, 2012.

compute a good near-optimal solution using working space significantly less than the size of the data stream. The motivation for streaming comes from practical applications of managing massive data sets such as, e.g., real-time network traffic, on-line auctions, and telephone call records. These data sets are huge and arrive at a very high rate, making it impossible to store more than a small part of the input.

We consider the space requirements of finding or approximating the maximum clique in a graph, or equivalently the maximum independent set. We assume that the graph is given as a stream of edges, where the algorithm can view the stream several times. In this paper, we assume the algorithm only views the stream a constant number of times. More generally, we treat the following *gap* problem: given a graph G and numbers U and L with $L \leq U$, decide whether G contains a U-clique, or contains no $(L+1)$-clique. When the clique number is greater than L or less than U, the algorithm can answer arbitrarily. Here, U and L can be functions of the order n of the input graph.

Several graph problems have been considered in the streaming setting, including bipartite matching (weighted and unweighted cases) [14], diameter and shortest paths [14,15], min-cut [1], and graph spanners [15]. Except for certain counting problems, such as counting triangles [5], cycles [24], $K_{3,3}$ bipartite cliques [9] and small graph minors [8], these use $n \cdot \mathit{polylog}(n)$ space.

Limited attention has been given to streaming algorithms for NP-hard problems; exceptions include Max-Cut [1,27] and certain clustering problems (e.g., [19]). In [17], the independent set problem in graphs and hypergraphs was considered, but with the primary focus on the fine-grained space requirements of matching the Turán bound on sparse (hyper)graphs. Some additional upper bounds are given in [23], but with a focus on general hypergraphs. We are not aware of any lower bounds for the space complexity of computing any classical NP-hard graph parameter like clique number (except for max-cut [27]).

The Max-Clique problem, and its sister the independent set problem, is one of the central problems in optimization, and graph theory. For instance, the algorithm textbook of Kleinberg and Tardos uses variations of the independent set problem as a common theme for the whole book. It has long been one of the cornerstones of complexity theory, including monotone circuit complexity [3], decision tree complexity [7], fixed-parameter intractability [10], and interactive proofs and approximation hardness [18]. The current best intractability bound for Max-Clique is $n/2^{(\log n)^{3/4+\epsilon}}$ [21], and the best approximation result is $O(n(\log\log n)^2/\log^3 n)$ [13].

Communication Complexity. Communication complexity, introduced by Yao [26], is a powerful tool to solve a variety of problems in areas as disparate as VLSI design, decision trees, data structures, and circuit complexity [22]. It is a game between two parties, Alice and Bob, with unlimited computing power, that want to compute the value of a function $f : X \times Y \mapsto \{0, 1\}$. Alice only knows $x \in X$, while Bob only knows $y \in Y$. To perform the computation, they are allowed to send messages to each other in order to converge on a shared output $P(x, y)$. In a randomized protocol, Alice and Bob toss coins, and the messages can depend on the coin flips. We say a randomized (deterministic) protocol P computes f if $\Pr[P(x,y)=f(x,y)] \geq 2/3$ ($P(x,y)=f(x,y)$) for any input x,y, and define the randomized (deterministic) *communication complexity* $R_{1/3}(f)$ ($D(f)$) to be the number of bits communicated for the worst input under the best randomized (deterministic) protocol computing f, respectively. Here, 1/3 refers to the error rate. Since a deterministic protocol is a randomized protocol, $D(f) \geq R_{1/3}(f)$.

Our Results. We give several constructions that imply communication lower bounds for clique separation, resulting in equivalent lower bounds for the space complexity of streaming algorithms. The constructions differ in their range of parameters U and L, as well as the strengths of the lower bounds.

The results are summarized in the table. $R_{1/3}(\textsc{Clique-Gap}(U,L))$ denotes the randomized communication complexity to determine whether the clique number of the union of two graphs is at least U or at most L, and $\mathcal{R}(s)$ refers the s-th diagonal Ramsey number (see Sec. 2 for formal definitions).

Table 1. A summary of our results

U	L	$R_{1/3}(\textsc{Clique-Gap}(U,L))$
r	$10 \cdot 2^{1/\epsilon} \log n$	$\Omega(n^2/r^2)$
r	s	$O(n^2/(r/s)^2)$
r	$2\sqrt{r}-1$	$\Omega(n^2/r^2)$
$r=\mathcal{R}(s)-1$	$s-1$	$\Omega\left(\max\left(n/r, \frac{n^2}{r^3 \exp(10\sqrt{\log r \log \frac{n}{2r^2}})}\right)\right)$
$r=\mathcal{R}(s)$	$s-1$	$O(1)$

The first two results in the table match up to a logarithmic factor. Thus, except for the case of very small or very large cliques, this gives a fairly precise characterization of what cliques can be separated. For smaller cliques, the bounds are still open to a large extent. The third result shows that any constant approximation requires quadratic space, which is a supplement to the first result when the clique number is a constant. Finally, the last two bounds give a sharp threshold within which we can separate cliques: constant space suffices below the threshold, while non-trivial and even superlinear space is necessary above the threshold.

We note that our results hold equally for the Max Independent Set problem. While the optimization and approximation of cliques and independent sets are equivalent in general graphs, the streaming problems are not identical since the stream is formed by edges and not non-edges. This distinction disappears in the communication problem, as well as in the sampling-based upper bounds.

The clique problem appears at first to be strongly related to the previously studied problem of counting triangles [5,6], and in fact, the known hardness of detecting triangles and short cycles in stream [5,15] yields a starting point for proving hardness of clique computation. Nevertheless, while a large clique implies many triangles, the converse is not true (viz. complete 3-partite graphs). Indeed, different arguments are needed for the clique problem.

While our hardness results involve reductions to the prototypical problem of set disjointness, our proofs involve some novel connections between Ramsey theory and additive combinatorics. Obtaining superlinear constant-pass lower bounds on graph problems via disjointness is often hampered by dependencies between edges. The use of designs and random partitition to get around this here may be useful for proving such lower bounds for other graph problems in the semi-streaming model.

Outline of the Paper. We define the problems and notation formally in Section 2, and introduce our methodology in Section 3. The bulk of the paper is in Section 4, where we give several different space-approximation tradeoffs for the clique problem. Some upper bounds are given in Section 5. Some proofs of lemmas have been deferred to the full version.

2 Problem Definitions

A clique in a graph is a subset of mutually adjacent vertices. The Max-Clique problem is that of finding a clique of approximately maximum size. Let $\omega(G)$ denote the clique number of graph G. Let n denote the number of vertices of the graph input to Max-Clique. A t-subgraph refers to a subgraph induced by t vertices. The Ramsey number $\mathcal{R}(r)$ is the smallest n so that for any graph G of size n, either G or its complement, $\overline{G}$, has a r-clique. By the classic results of [11,12], $\mathcal{R}(r) = 2^{\theta(r)}$, and in particular $\sqrt{2}^r < \mathcal{R}(r) < 4^r$.

Let $[n] = \{1, 2, \ldots, n\}$. An edge stream is formally defined to be a sequence $\langle a_1, a_2, ..., a_m\rangle$, where $a_j \in \binom{[n]}{2}$, inducing the undirected graph $G = (V, E)$ on n vertices with $V = [n]$ and $E = \{a_j : j \in [m]\}$. Each edge may appear more than once. Only in Sec. 4.1 do we need to allow edges to appear more than once (specifically, twice), and only when $r > \sqrt{n}$.

Set disjointness, denoted Disj, is a communication complexity problem where Alice and Bob hold two subsets, x and y, of $[N]$, respectively, and they want to determine whether the intersection of their subsets is empty. Improving a result in [4], Kalyanasundaram and Schnitger [20] proved that $R_{1/3}(\text{Disj}) = \Omega(N)$.

The clique gap problem is the communication complexity problem for clique approximation, where Alice and Bob hold two subgraphs $G_A = \langle V_n, E_A\rangle$ and $G_B = \langle V_n, E_B\rangle$ and they want to approximately determine the clique number of the combined graph $G_A \cup G_B = \langle V_n, E_A \cup E_B\rangle$. We define the value of the function Clique-Gap(U, L) to be 1 if $\omega(G) \geq U$, 0 if $\omega(G) \leq L$, and arbitrary (0 or 1) otherwise.

The communication complexity of a decision problem is closely related to the space complexity of the problem, in that the former gives a lower bound for the latter. Namely, for any decision problem Π, it holds that $\text{space}_{1/3}(\Pi) \geq R_{1/3}(\Pi)$, where $\text{space}_{1/3}(\Pi)$ denotes the space complexity of a randomized streaming algorithm that answers correctly with at least 2/3-probability on any instance of Π. This holds, up to constant factors, even if we allow the streaming algorithm passes through the input constant times.

3 Our Methodology

Reduction from the set disjointness problem is generally the method of choice for proving communication complexity lower bounds for graph problems. Yet, to come up with reductions with near-optimal parameters to Clique-Gap(U, L) involves a number of combinatorial challenges.

Our starting point was the following reduction from the set disjointness problem with parameter $N = (n/4)^2$ to Clique-Gap$(4, 2)$:

For any input of set disjointness problem, where Alice holds $x \in \{0,1\}^{(n/4)^2}$ and Bob holds $y \in \{0,1\}^{(n/4)^2}$, we construct an input for the clique problem as follows. We denote the vertices by $\{v_{i,j} | i = 1, 2, 3, 4; j = 1, 2, 3, \cdots, n/4\}$. Alice has edges $(v_{1,j}, v_{3,j'})$ and $(v_{2,j}, v_{4,j'})$ if $x[j, j'] = 1$. Bob has edges $(v_{1,j}, v_{4,j'})$ and $(v_{2,j}, v_{3,j'})$ if $y[j, j'] = 1$. Finally, both of them have the edges $(v_{1,j}, v_{2,j})$ and $(v_{3,j}, v_{4,j})$, for $j = 1, 2, ..., n/4$. In this construction, the graph has a 4-clique if x intersects with y, and the clique number is only 2 if x doesn't intersect with y.

The above construction can be viewed as an extension of constructions from [5,15] on detecting triangles in streams. This argument can, however, not be extended further: proving an $\Omega(n^2)$ lower bound for Clique-Gap$(5, 2)$ is impossible because of the counting version of the Szemerédi's Regularity Lemma [25]. We will detail the reason and give a weaker lower bound for Clique-Gap$(5, 2)$ in Sec. 4.2. This obstacle shows that some non-trivial combinatorics lies beneath our problem. We overcome this and other obstacles for different U, L pairs by applying different arguments, and by exploiting properties of the worst case distribution for the set disjointness problem. Along the way, we create some interesting combinatorial structures, such as the one in Lemma 1, which we could not find elsewhere in the literature.

4 Lower Bounds

We reduce the set disjointness problem to the approximate clique determination problem, thereby obtaining lower bounds on space for streaming algorithms approximating cliques. We give several constructions that apply to different combinations of the parameters U and L.

The structure of the arguments is as follows. Given an instance (x, y) of Disj, we form a graph $\tilde{G}$ that is a packing of "gadgets", or clique subgraphs, each corresponding to a single bitpair of the vectors x and y. Some of the edges of each gadget are reserved for Alice, and the remaining edges for Bob. The actual graphs G_A and G_B handed to Alice and Bob are subgraphs of $\tilde{G}$, where Alice (Bob) receives her (his) edges of gadget i only if the corresponding bit x_i (y_i) is set, respectively. This ensures that if $x_i = y_i = 1$ – the case of a positive set intersection instance – then the corresponding gadget is a clique, yielding a positive answer to the clique separation problem. The main issue is to ensure that for negative instances, the clique size of the whole graph $G_A \cup G_B$ remains small.

We present three constructions. The first gives optimal space lower bounds, up to logarithmic factor, for all but very large clique numbers. The second yields weaker lower bounds, but holds for sub-logarithmic values of L. The third one gives optimal $\Omega(n^2)$-space lower bound for the case of constant clique sizes.

4.1 r vs. $\log n$

Theorem 1. *For $0 < \epsilon < 1$, $r = n^{1-\epsilon}$ and $s = 100 \cdot 2^{2/\epsilon} \log n$, it holds that $R_{1/3}(\textsc{Clique-Gap}(r,s)) = \Omega(n^{2\epsilon})$. Thus, for some constant c, any randomized streaming algorithm for* Max-Clique *with approximation ratio $\frac{c \cdot r}{\log n}$ requires $\Omega(n^2/r^2)$ space (when $r = O(n^{1-\epsilon})$).*

We reduce Disj to Clique-Gap(r,s) in such a way that positive instances will have clique-size r, while negative instances will be like the Erdös-Renyi random graphs $G_{n,p}$, and thus have clique-size $s = O(\log_{1/p} n)$ (we shall specify p later).

We construct optimal reductions (up to a factor of $\log n$) from the set-disjointness when $r = O(n^{1-\epsilon})$. At the heart of the reduction, there is a combinatorial lemma:

Lemma 1. *For every $n > 2^{2/\epsilon}$ and every $r < n/2$, there is a set system $\mathcal{C}$ on $[n]$ with n^2/r^2 sets of size r each, such that each pair of distinct points is covered by at most d sets from $\mathcal{C}$, where $d = \lceil 2/\epsilon \rceil - 2$.*

Proof. Let P be the largest prime with the property that $rP \leq n$. Then, $rP > n/2$, by Bertrand's postulate. We identify $[P]$ with GF_P performing all arithmetic modulo P. We also identify $[r]$ with an arbitrary subset of GF_P^d, and assume that there is an injective mapping $f : [r] \mapsto GF_P^d$ because $P^d \geq (\frac{n}{2r})^d > r$. For $(x,y) \in GF_P^2$ we define the set

$$C_{x,y} = \{(a_1, a_2, \ldots, a_d, a) \mid a = a_d x^d + \ldots + a_1 x - y \text{ and } (a_1, a_2, \ldots, a_d) \in f([r])\}.$$

Notice that $C_{x,y}$ has size exactly r, since given x and y the values of $a_1, a_2, \ldots, a_d$ determine the value of a. In particular, this implies that for two distinct points that $C_{x,y}$ covers, the first d coordinates are always different. Consider now two distinct points $(a_1, a_2, \ldots, a_d, a)$ and $(b_1, b_2, \ldots, b_d, b)$. If they are covered by the same $C_{x,y}$, we get that $a_d x^d + \ldots + a_1 x - y = a$ and $b_d x^d + \ldots + b_1 x - y = b$, implying that

$$(a_d - b_d)x^d + \ldots + (a_1 - b_1)x = a - b. \tag{1}$$

Notice that $C_{x,y}$ and $C_{x,y'}$ are disjoint whenever $y \neq y'$. Thus, if $C_{x,y}$ and $C_{x',y'}$ intersect in a point, and $(x,y) \neq (x',y')$, then it is necessary that $x \neq x'$. Thus, in particular, if there are $(x_1, y_1), \ldots, (x_{d+1}, y_{d+1})$ such that C_{x_i,y_i} cover the same two points, then $x_1, \ldots, x_{d+1}$ are all distinct, and Eqn. 1 holds for all $x_1, \ldots, x_{d+1}$. Since by our earlier remark $(a_1, a_2, \ldots, a_d) \neq (b_1, b_2, \ldots, b_d)$, we get a contradiction by discovering that a degree d polynomial (namely $(a_d - b_d)x^d + \ldots + (a_1 - b_1)x - a + b$) has $d+1$ roots.

Our reduction from the set disjointness problem of size $N = n^2/r^2$ will be the following. First, we define N cliques (gadgets) of size r on n nodes, as shown in the above lemma. Let us denote the i^{th} clique by C_i $(1 \leq i \leq N)$. We then associate each edge of C_i to Alice or Bob with probability 1/2 independently. We call the set of edges associated this way to Alice and Bob C_i^A and C_i^B, respectively. Note that it is possible that the same edge of the graph is associated to both Alice and Bob, since an edge may occur in up to d different C_is.

The graph G_A given to Alice consists of the edges in the union of those C_i^As, for which the bit x_i in the set disjointness problem is set to 1. Similarly, we give to Bob the graph G_B, which is the union of those C_i^Bs, for which the bit y_i is set to 1. Clearly, if $x_i = y_i = 1$ then the combined graph $G_A \cup G_B$ will contain all of C_i, and thus have clique size at least r.

We argue now that in the negative case, we can embed the resulting graph in an Erdös-Renyi random graph $G_{n,p}$ with edge probability $p = 1 - 1/2^d$.

Lemma 2. *For any negative instance (x, y) of* Disj *on s bits ($x \cap y = \emptyset$), let q_1 be the probability that the graph $G_A \cup G_B$, generated by the above described randomized map of (x, y), contains an s-clique. Let q_2 be the probability that an Erdös-Renyi random graph, where each edge is drawn with probability $1 - 1/2^d$, contains an s-clique. Then $q_1 < q_2$.*

Proof. For an edge e in the graph $G_A \cup G_B$ generated by (x, y), we consider the set of cliques $\mathcal{C}_e = \{C_i | e \in C_i \text{ and } i \in x \cup y\}$. In the method we described above, we choose e in each clique in $\mathcal{C}_e$ with probability 1/2 independently, and e appears in $G_A \cup G_B$ if e is chosen in any clique in $\mathcal{C}_e$. Thus, the probability that $G_A \cup G_B$ contains e is $1 - 1/2^{|\mathcal{C}_e|} \leq 1 - 1/2^d$, because $|\mathcal{C}_e| \leq |\{C_i | e \in C_i\}| \leq d$ by Lemma 1. However, in the Erdös-Renyi random graph, each edge is chosen with probability $1 - 1/2^d$. Therefore, $G_A \cup G_B$ has sparser edges, and it has an s-clique with less probability.

*Proof (**Proof of Theorem 1**).* Given instance x, y to Disj, we form and hand the graphs G_A and G_B to Alice and Bob, as expressed above. On positive instance, when $x_i = y_i = 1$, for some bit i, the corresponding subgraph in $G_A \cup G_B$ is an r-clique. On negative instances, $G_A \cup G_B$ is sparser than the Erdös-Renyi random graph $G_{n,p}$, with $p = 1 - 1/2^d$. As shown by Grimmett and McDiarmid [16], $\omega(G_{n,p}) \leq 2 \log n / \log(1/p) + o(\log n) \leq 2^{d+1} \log n + o(\log n)$, with high probability. The theorem now follows.

4.2 $\mathcal{R}(s) - 1$ vs. $s - 1$

When proving an $\Omega(n^2)$ lower bound for Clique-Gap$(5, 2)$, the $s = 3$ case of Clique-Gap$(\mathcal{R}(s) - 1, s - 1)$, we run into obstacles if we use the approach for Clique-Gap$(4, 2)$. To do so, we must pack $\Theta(n^2)$ 5-clique gadgets in a graph on n vertices. We then need to partition the $\binom{5}{2}$ edges into two parts, one for Alice and the other for Bob, such that each part has no triangles. In fact, the partition is unique up to a permutation, and it does not contain "hard-wired" edges like the gadget in the proof of Clique-Gap$(4, 2)$ does. Furthermore, we require more properties of the packing: all the gadgets are edge-disjoint and each triangle must lie fully within one gadget. The Triangle Removal Lemma, which can be proven from Szemerédi's Regularity Lemma [25], states that we can remove $o(n^2)$ edges from a graph containing $o(n^3)$ triangles to make it triangle-free. If we take one triangle from each gadget, these $\Theta(n^2)$ triangles are edge-disjoint and $o(n^2)$ edges do not suffice to destroy them all. Therefore, we cannot pack $\Theta(n^2)$ gadgets in a graph of size n.

Instead, we can prove the following result, using a different packing requirement.

Theorem 2. *For any r, $R_{1/3}(\textsc{Clique-Gap}(r, s-1)) = \Omega\left(\frac{n^2}{r^3 \exp(10\sqrt{\log r \log \frac{n}{2r^2}})}\right)$, where $r = \mathcal{R}(s) - 1$.*

For instance, this gives a $n^2/\exp(O(\sqrt{\log n})) = n^{2-o(1)}$ lower bound for Clique-Gap$(5, 2)$. That is the best we can hope for in the sense that Clique-Gap$(6, 2)$ has a trivial upper bound, as we shall see in Section 5.

We shall use the following combinatorial structure and theorem of Alon and Shapira.

Definition 1 (h-Sum-Free). *[2] A set $X \subseteq [n]$ is called h-sum-free if for every three positive integers $a, b, c \leq h$ such that $a + b = c$, if $x, y, z \in X$ satisfy the equation $ax + by = cz$, then $x = y = z$. That is, whenever $a + b = c$, and $a, b, c \leq h$, the only solution to the equation that uses values from X, is one of the $|X|$ trivial solutions.*

Theorem 3. *[2] For every positive integer n, there exists an h-sum-free subset $X \subseteq [n]$ of size at least $|X| \geq \frac{n}{e^{10\sqrt{\log h \log n}}} \doteq g(n, h)$.*

We say that a set system $\mathcal{C} = \{C_i\}_i$ is *edge-disjoint* if any pair of points is contained in at most a single set, and that it is *triangle-free* if whenever $u, v \in C_i$, $v, w \in C_j$ and $w, u \in C_k$, for some $C_i, C_j, C_k \in \mathcal{C}$, then $C_i = C_j = C_k$.

Lemma 3. *For any n, there is an edge-disjoint triangle-free set system on $[n]$ with $g(n/(2r^2), r) \cdot n/r = \Omega(n^2/(r^3 \exp(10\sqrt{\log r \log n/(2r^2)})))$ sets of size r each.*

Proof. We first pick an r-sum-free set $Z \subseteq [\frac{n}{2r^2}]$ such that

$$|Z| = m \geq g(n/(2r^2), r) = \frac{n/(2r^2)}{\exp(10\sqrt{\log r \log \frac{n}{2r^2}})}.$$

Suppose $Z = \{z_1, \ldots, z_m\}$. For $i \in [m]$, let $S_i = (z_i r + 1) \cdot [r] = \{(z_i r + 1)a : a \in [r]\}$. We denote the set shift j from S_i by $S_i^{(j)}$, namely we define $S_i^{(j)} = S_i + jr$, for $i \in [m]$, $j \in [n/(2r)]$. Finally, we define the set family $\mathcal{C} = \{S_i^{(j)} | i \in [m],\ j \in [n/(2r)]\}$, and let $\tilde{G} = ([n], E)$, where $E = \{(u, v) | \exists S \in \mathcal{C}, u, v \in S\}$. It is clear that for each $S \in \mathcal{C}$, the subgraph on S induces an r-clique in $\tilde{G}$.

The lemma follows from the following two claims.

Claim. $\mathcal{C}$ is edge-disjoint, i.e., any $S_{i_1}^{(j_1)}$ and $S_{i_2}^{(j_2)}$ intersect in at most one element if $(i_1, j_1) \neq (i_2, j_2)$.

Proof. Suppose they have two common elements u and v. From $u, v \in S_{i_1}^{(j_1)}$, by definition we have $u = (z_{i_1} r + 1)b_1 + j_1 r$ and $v = (z_{i_1} r + 1)c_1 + j_1 r$, for some $b_1, c_1 \in [r]$. Similarly, there are $b_2, c_2 \in [r]$, such that $u = (z_{i_2} r + 1)b_2 + j_2 r$, and $v = (z_{i_2} r + 1)c_2 + j_2 r$. So we have

$$u = (z_{i_1} r+1)b_1+j_1 r = (z_{i_2} r+1)b_2+j_2 r \text{ , and } v = (z_{i_1} r+1)c_1+j_1 r = (z_{i_2} r+1)c_2+j_2 r. \tag{2}$$

Modulo r, we have $b_1 = b_2$ and $c_1 = c_2$ (because $|b_i|, |c_i| < r$). We denote $b = b_1 = b_2$ and $c = c_1 = c_2$. By computing $u - v$, $(z_{i_1} r+1)(b-c) = (z_{i_2} r+1)(b-c)$. Now, $b \neq c$ because $u \neq v$. So, $z_{i_1} = z_{i_2}$, then we have $i_1 = i_2$. By (2) then, $j_1 = j_2$. Therefore, $(i_1, j_1) = (i_2, j_2)$, which is a contradiction.

Claim. $\mathcal{C}$ is triangle-free, i.e., for any distinct u, v, w, if $v, w \in S_{i_1}^{(j_1)}$, $w, u \in S_{i_2}^{(j_2)}$, and $u, v \in S_{i_3}^{(j_3)}$, then $(i_1, j_1) = (i_2, j_2) = (i_3, j_3)$.

*Proof (**Proof of Theorem 2**).* We reduce the set disjointness problem with $N = t \cdot q$ bits to
Clique-Gap$(r, s-1)$, where $t = \frac{n/(2r^2)}{\exp(10\sqrt{\log r \log(n/(2r^2))})} = g(n/(2r^2), r)$ and $q = n/(2r)$.

By the definition of Ramsey number, for each $S_i^{(j)}$, there exists a subgraph $Q_i^{(j)}$ of the clique on $S_i^{(j)}$, such that neither $Q_i^{(j)}$ nor $\overline{Q_i^{(j)}}$ has a clique of size s.

Given a Disj instance $x, y \subseteq [t] \times [q]$, we consider each $S_i^{(j)}$ as a gadget and construct a clique separation instance, in which we give Alice $G_A = \bigcup_{(i,j)\in x} Q_i^{(j)}$, and give Bob $G_B = \bigcup_{(i,j)\in y} \overline{Q_i^{(j)}}$. We are going to prove that $G_A \cup G_B$ has an r-clique if $x \cap y \neq \emptyset$, and has no s-clique if $x \cap y = \emptyset$.

On positive Disj instances, when $x_{i,j} = y_{i,j} = 1$, the corresponding gadget $S_i^{(j)}$ induces an r-clique in $G_A \cup G_B$. On negative Disj instance, for each (i, j), each subgraph induced by $S_i^{(j)}$ in $G_A \cup G_B$ is one of three possibilities: $Q_i^{(j)}$, $\overline{Q_i^{(j)}}$ or empty. By construction, none of these contain a $(2 \log r)$-clique, so if $G_A \cup G_B$ contains one, there exists a triangle (u, v, w) which is not in any $S \in \mathcal{C}$. This contradicts the triangle-freeness property of $\mathcal{C}$.

Therefore, if we have a protocol of the Clique-Gap$(r, s-1)$ problem, we have a protocol of set disjointness problem with the same communication complexity. Hence, Clique-Gap$(r, s-1)$ problem has communication complexity lower of $\Omega(N) = \Omega(t \cdot q)$.

For larger values of r (e.g., $r = n/\text{polylog}(n)$), a naive packing gives better bounds: Simply combine $\lfloor n/r \rfloor$ *vertex*-disjoint r-cliques. This yields an edge-disjoint triangle-free set system with n/r sets of size r each.

Theorem 4. *For any n and any s, $R_{1/3}(\textsc{Clique-Gap}(r, s-1)) = \Omega(n/r)$, where $r = \mathcal{R}(s) - 1$.*

4.3 r^2 vs. $2r-1$

We now focus on graphs of constant clique number, for which we obtain optimal quadratic space lower bounds.

Theorem 5. *For any number $r \geq 18$, $R_{1/3}(\textsc{Clique-Gap}(r^2, 2r-1)) = \Omega(n^2/r^4)$. Thus, any randomized ρ-approximation streaming algorithm for* Max-Clique *requires $\Omega(n^2/\rho^4)$ space.*

We construct a gadget $H = \langle V_H, E_H \rangle$ on r^2 vertices corresponding to a single bit in Disj. We shall ensure that H is clique if the corresponding bits of both Alice and Bob are both 1, and that H contains no $2r$-clique otherwise. The vertex set V_H consists of r groups, r vertices each: $V_H = \{u_{i,j} | i, j \in [r]\}$. We color all the $\binom{r^2}{2}$ edges with three colors: A (Alice), B (Bob), and C (Common).

We say that a triplet $u, v, w \in V_H$ is a *colorful triangle* if all three mutual edges are differently colored. The proof of the following lemma is based on the probabilistic method.

Lemma 4. *For large r, there exists a coloring of E_H satisfying*

1. *Edge $\{u_{i,j}, u_{i',j'}\}$ is with color C if and only if $i = i'$ and $j \neq j'$.*
2. *Any $2r$-subgraph of H contains a colorful triangle.*

Let P be a prime in the range $[n/(2r^2), n/r^2]$. We reduce the Clique-Gap problem of size n from Disj problem of size $N = P^2$ by packing P^2 gadgets in a graph of size n, where each gadget is of size r^2. We isolate the remaining $n - r^2 P$ vertices, and focus on the $r^2 P$ vertices $\{v_{i,j,k} | i, j \in [r], k \in [P]\}$. On these vertices, the edges are given by $E_G^C = \{\{v_{i,j,k}, v_{i,j',k}\} | \ i, j, j' \in [r], k \in [P], j \neq j'\}$, $E_G^A = \{\{v_{i,j,(s+ti) \bmod P}, v_{i',j',(s+ti') \bmod P}\} | i, i', j, j' \in [r], i \neq i'$ and $\{u_{i,j}, u_{i',j'}\}$ is with color A and $x_{s,t} = 1\}$, and $E_G^B = \{\{v_{i,j,(s+ti) \bmod P}, v_{i',j',(s+ti') \bmod P}\} | \ i, i', j, j' \in [r], i \neq i'$ and $\{u_{i,j}, u_{i',j'}\}$ is with color B and $y_{s,t} = 1\}$.

Alice is given the edges in $E_G^A \cup E_G^C$, and Bob given the edges in $E_G^B \cup E_G^C$.

Lemma 5. *If* Disj$(x, y) = 1$*, then $\omega(G) = r^2$.*

Lemma 6. *If* Disj$(x, y) = 0$*, then $\omega(G) < 2r$.*

Proof (Proof of Theorem 5). We reduce Disj problem of size $N = P^2$ to the Clique-Gap$(r^2, 2r-1)$ problem. Since $\mathcal{R}(\textsc{Disj}) = \Omega(P^2)$, any randomized protocol to separate graphs with r^2-cliques from those with only $(2r-1)$-cliques requires $\Omega(n^2/r^4)$ communication.

5 Upper Bounds

The following simple random sampling argument shows that the lower bound of Thm. 1 is within a logarithmic factor of optimal.

Theorem 6. *There is a randomized streaming algorithm for* $\textsc{Clique-Gap}(r, r/\rho)$ *that uses* $O((n/\rho)^2)$ *space (for* $\rho = O(n/\sqrt{\log n})$*). Thus,*
$R_{1/3}(\textsc{Clique-Gap}(r, r/\rho)) \leq \text{space}_{1/3}(\textsc{Clique-Gap}(r, r/\rho) = O((n/\rho)^2)$.

Proof. Assuming that the vertices are numbered $0, 1, \ldots, n-1$, we initially choose a random number h from $[n]$. This specifies a set S consisting of the n/ρ vertices numbered h through $h+n/\rho-1 \pmod n$. In processing the stream, we only store edges between pairs of vertices in S and afterwards output the maximum clique within S. The probability that any given vertex falls within S is $(n/\rho)/n = 1/\rho$. Thus, by linearity of expectation, the expected number of vertices within any r-clique that fall inside S is r/ρ.

Finally, we cannot expect to get a non-trivial lower bound on the separation of $(s-1)$-cliques vs. $\mathcal{R}(s)$-cliques using communication complexity. Namely, by the definition of Ramsey numbers, any 2-coloring of an $\mathcal{R}(s)$-clique – or a splitting of the clique edges between Alice and Bob – leaves a monochromatic s-clique. Thus, at least one of Alice and Bob can detect a s-clique, without any communication. The gap in Thm. 2 is therefore best possible, even though the space lower bound is not optimal. In fact, we get a sharp transition for $\textsc{Clique-Gap}(U, L)$ in terms of the values of U and L for which non-trivial communication is needed.

Theorem 7. *There is a deterministic communication protocol for* $\textsc{Clique-Gap}(\mathcal{R}(s), s-1)$ *that uses* $O(1)$*-bits. That is,* $D(\textsc{Clique-Gap}(\mathcal{R}(s), s-1)) = O(1)$.

References

1. Ahn, K.J., Guha, S.: Graph Sparsification in the Semi-streaming Model. In: Albers, S., Marchetti-Spaccamela, A., Matias, Y., Nikoletseas, S., Thomas, W. (eds.) ICALP 2009. LNCS, vol. 5556, pp. 328–338. Springer, Heidelberg (2009)
2. Alon, N., Shapira, A.: A characterization of easily testable induced subgraphs. In: SODA 2004, pp. 942–951. SIAM (2004)
3. Alon, N., Boppana, R.B.: The monotone circuit complexity of boolean functions. Combinatorica 7(1), 1–22 (1987)
4. Babai, L., Frankl, P., Simon, J.: Complexity classes in communication complexity theory. In: FOCS 1986, pp. 337–347. IEEE Computer Society (1986)
5. Bar-Yossef, Z., Kumar, R., Sivakumar, D.: Reductions in streaming algorithms, with an application to counting triangles in graphs. In: SODA 2002, pp. 623–632 (2002)
6. Becchetti, L., Boldi, P., Castillo, C., Gionis, A.: Efficient semi-streaming algorithms for local triangle counting in massive graphs. In: KDD 2008, pp. 16–24 (2008)

7. Bollobás, B.: Complete subgraphs are elusive. Journal of Combinatorial Theory, Series B 21(1), 1–7 (1976)
8. Bordino, I., Donato, D., Gionis, A., Leonardi, S.: Mining Large Networks with Subgraph Counting. In: 8th IEEE International Conference on Data Mining, pp. 737–742. IEEE Computer Society (2008)
9. Buriol, L.S., Frahling, G., Leonardi, S., Sohler, C.: Estimating Clustering Indexes in Data Streams. In: Arge, L., Hoffmann, M., Welzl, E. (eds.) ESA 2007. LNCS, vol. 4698, pp. 618–632. Springer, Heidelberg (2007)
10. Downey, R.G., Fellows, M.R.: Fixed-parameter tractability and completeness. II. On completeness for W[1]. Theoretical Computer Science 141(1-2), 109–131 (1995)
11. Erdős, P.: Some remarks on the theory of graphs. Bull. Amer. Math. Soc. 53, 292–294 (1947)
12. Erdős, P., Szekeres, G.: A combinatorial problem in geometry. Compositio. Math. 2, 463–470 (1935)
13. Feige, U.: Approximating maximum clique by removing subgraphs. SIAM J. Discrete Math. 18(2), 219–225 (2004)
14. Feigenbaum, J., Kannan, S., McGregor, A., Suri, S., Zhang, J.: On graph problems in a semi-streaming model. Theoretical Computer Science 348(2), 207–216 (2005)
15. Feigenbaum, J., Kannan, S., McGregor, A., Suri, S., Zhang, J.: Graph distances in the data-stream model. SIAM J. Comput. 38(5), 1709–1727 (2008)
16. Grimmett, G.R., McDiarmid, C.J.H.: On colouring random graphs. Mathematical Proceedings of the Cambridge Philosophical Society 77, 313–324 (1975)
17. Halldórsson, B.V., Halldórsson, M.M., Losievskaja, E., Szegedy, M.: Streaming Algorithms for Independent Sets. In: Abramsky, S., Gavoille, C., Kirchner, C., Meyer auf der Heide, F., Spirakis, P.G. (eds.) ICALP 2010. LNCS, vol. 6198, pp. 641–652. Springer, Heidelberg (2010)
18. Håstad, J.: Clique is hard to approximate within $n^{1-\epsilon}$. Acta Mathematica 182, 105–142 (1999)
19. Indyk, P., Price, E.: K-median clustering, model-based compressive sensing, and sparse recovery for earth mover distance. In: STOC 2011, pp. 627–636 (2011)
20. Kalyanasundaram, B., Schnitger, G.: The probabilistic communication complexity of set intersection. SIAM J. Discrete Math. 5(4), 545–557 (1992)
21. Khot, S., Ponnuswami, A.K.: Better Inapproximability Results for maxClique, Chromatic Number and Min-3Lin-Deletion. In: Bugliesi, M., Preneel, B., Sassone, V., Wegener, I. (eds.) ICALP 2006. LNCS, vol. 4051, pp. 226–237. Springer, Heidelberg (2006)
22. Kushilevitz, E., Nisan, N.: Communication Complexity. Cambridge Univ. Pr. (1997)
23. Losievskaja, E.: Approximation Algorithms for Independent Set Problems on Hypergraphs. PhD thesis. Reykjavik University (January 2010)
24. Manjunath, M., Mehlhorn, K., Panagiotou, K., Sun, H.: Approximate Counting of Cycles in Streams. In: Demetrescu, C., Halldórsson, M.M. (eds.) ESA 2011. LNCS, vol. 6942, pp. 677–688. Springer, Heidelberg (2011)
25. Ruzsa, I., Szemerédi, E.: Triple systems with no six points carrying three triangles. Combinatorics (Keszthely, 1976), Coll. Math. Soc. J. Bolyai 18, 939–945 (1976)
26. Yao, A.C.C.: Some complexity questions related to distributive computing (preliminary report). In: STOC 1979, pp. 209–213. ACM (1979)
27. Zelke, M.: Intractability of min- and max-cut in streaming graphs. Inf. Process. Lett. 111(3), 145–150 (2011)