





Query Lower Bounds for Correlation Clustering Under Memory Constraints

Sumegha Garg  

Rutgers University, New Brunswick, NJ, USA

Songhua He  

Rutgers University, New Brunswick, NJ, USA

Periklis A. Papakonstantinou  

Rutgers University, New Brunswick, NJ, USA

Abstract

This work initiates the study of memory–query tradeoffs for graph problems, with a focus on correlation clustering. Correlation clustering asks for a partition of the vertices that minimizes disagreements: non-edges inside clusters plus edges across clusters. Our first result is a tight query lower bound: to output a partition whose cost approximates the optimum up to an additive error of εn^2 , any algorithm requires $\Omega(n/\varepsilon^2)$ adjacency-matrix queries. Under memory constraints, we show that even for the seemingly easier task of approximating the optimal clustering cost (without producing a partition), any algorithm in the random query model must make $\gg n/\varepsilon^2$ adjacency-matrix queries. Finally, we prove the first general graph model query lower bound for correlation clustering, where algorithms are allowed adjacency-matrix, neighbor, and degree queries. The latter two bounds are not yet tight, leaving room for sharper results.

2012 ACM Subject Classification Theory of computation → Lower bounds and information complexity

Keywords and phrases correlation clustering, query-space complexity, information theory

Digital Object Identifier 10.4230/LIPIcs.ITCS.2026.67

Funding *Songhua He*: The author was partially supported by Karthik C. S. through the National Science Foundation under Grant CCF-2443697.

Acknowledgements The authors would like to thank anonymous reviewers for their valuable suggestions. Songhua He would also like to thank Karthik C. S. for generously providing partial research fellowship.

1 Introduction

How much information must an algorithm gather from a massive graph to solve a fundamental task like clustering, and how does this cost increase when the algorithm is constrained by limited memory? This work investigates these questions for the problem of Correlation Clustering (CC) – a cornerstone of machine learning and network analysis. We establish the precise information-theoretic requirements of solving CC in sublinear time (i.e., in terms of query complexity) and, importantly, initiate the study of query-space tradeoffs for graph problems.

Given an input graph for the correlation clustering problem, we interpret edges as pairs of “similar” (+) items and non-edges as pairs of “dissimilar” (–) items. The goal is to partition the vertices into clusters to minimize the total number of “disagreements”: the number of similar pairs that are cut apart plus the number of dissimilar pairs placed in the same cluster. This problem is closely related to other classic cut problems, such as Max-Cut and Minimum Bisection, and our first two hardness results also apply directly to these.



© Sumegha Garg, Songhua He, and Periklis A. Papakonstantinou;
licensed under Creative Commons License CC-BY 4.0

17th Innovations in Theoretical Computer Science Conference (ITCS 2026).

Editor: Shubhangi Saraf, Article No. 67; pp. 67:1–67:24

Leibniz International Proceedings in Informatics



LIPIC Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Correlation clustering has been extensively studied in the literature in various models and settings. The problem is known to be APX-hard [16]. After many years of effort, an exciting line of works has improved the approximation factor significantly from $O(1)$ to 1.437 [6, 16, 1, 17, 22, 21, 23, 15, 14, 13]. For sublinear-time algorithms, the best-known approximation ratio is improved to the same as the best ratio 1.437 of polynomial-time algorithms [20, 5, 23, 13]. Efficient algorithms and lower bounds for this problem have also been studied in the Massively Parallel Computation (MPC) setting [7, 15, 23], dynamic setting [9], streaming model [10, 20, 5, 4, 8, 35, 12], and the query model [11] with unbounded time. In this work, we study feasibility of approximate correlation clustering (with additive error) in various query models and under memory constraints.

Our work makes three primary contributions towards understanding the query complexity of correlation clustering. Throughout, an additive error of εn^2 means the algorithm's solution cost may be at most εn^2 worse than the true optimum. We remark that our main conceptual and technical contribution is the memory-query tradeoff for approximating the optimal clustering cost (in the random query model). However, to put things in proper context, we will present this as our second result in the introduction.

A tight query bound in the standard model

First, we consider the classic setting. An algorithm with unlimited memory that can adaptively query any pair of vertices to see if an edge exists. This is the adjacency-matrix query model. We show the following, asymptotically tight, query lower bound for finding a nearly optimal clustering.

Theorem 1 (Informal)¹

Every randomized algorithm that finds a clustering within an additive error of $O(\varepsilon n^2)$ must make $\Omega(n/\varepsilon^2)$ adaptive adjacency-matrix queries.

This result improves a previous $\Omega(n/\varepsilon)$ lower bound given in [11] and perfectly matches the known upper bound in the same work [11], settling the query complexity of CC in the adjacency-matrix query model. Next, we show that for the seemingly easier problem of approximating the optimal clustering cost, the query complexity increases dramatically under memory constraints, albeit in the random query model.

A query-space tradeoff for approximating the optimal clustering cost

What happens when memory is scarce? To understand the additional query cost of limited memory algorithms, we turn to a model where the algorithm receives a stream of uniformly random vertex pairs and indicators of whether these pairs connect or not in the underlying graph. This is the random-query model. For computing boolean functions, time-space tradeoffs in the random query model have been introduced and studied in [36, 24]. In our second result, the task is to estimate the *value* of the optimal clustering, a seemingly easier goal than finding the partition.

¹ We do not include its formal statement and full proof here due to space constraints; a full version with complete details will be posted online.

Theorem 2 (Informal – restated as Theorem 1)

In the random-query model, an algorithm using only $\gamma\sqrt{n}$ bits of memory to estimate the optimal CC cost, within additive error of εn^2 , needs $\Omega(\min\{q, \frac{n^{3/2}}{\gamma}\})$ queries where

$$q = \begin{cases} \frac{n}{\varepsilon^2\sqrt{\gamma}} & \text{if } \gamma < 1 \\ \frac{n}{\varepsilon^2\gamma^2} & \text{if } \gamma \geq 1 \end{cases}$$

To the best of our knowledge, this is the first query-space tradeoff for any approximation problem as well as a graph problem in the random query model. It demonstrates that memory is not free: with sub-polynomial space (when $\gamma = o(1)$), the query complexity blows past the $\tilde{O}(n/\varepsilon^2)$ baseline, hitting a barrier of $\tilde{\Omega}(n^2)$ for even modest approximation goals. Our lower bound for the case $\gamma \geq 1$ completes the picture and shows that the query complexity remains high even when the memory is large. Memory and queries are provably and strictly intertwined.² The proof of the above theorem faces various technical challenges, which we elaborate on in Section 1.1.

Compared to Theorem 1, Theorem 2 weakens the setting in two ways: random queries instead of arbitrary queries, but turning to the cost variant of the problem. We now explain the reason for doing so. First, if instead of the random-query model we allowed arbitrary queries, then the same lower bound would hold for Turing machines using space $c \log_2 n$, for a constant $c > 1$. Finding a natural problem for which such a bound holds is conceptually difficult. Thus, restricting how we access the input circumvents this difficulty and makes the problem amenable to analysis. Second, we should explain why we use the *cost/value* formulation. There are two reasons. First, clarity. Second, which is crucial here, under $O(\sqrt{n})$ space, the search formulation is vacuous. Merely writing down a near-optimal partition already needs $\Omega(n)$ bits (even for two clusters). Intuitively, any lower bound would then be dominated by output size rather than information. Focusing on value estimation removes this artifact and isolates the genuine memory–query tradeoff. This seems to be the “correct” problem variant to study here.

A lower bound in the general graph model

Finally, we consider a stronger query model, the *general graph model*. In this model, an algorithm can make pair, degree, and neighbor queries. This model was introduced and studied in [33, 2]. This model is significantly stronger, as neighbor queries can reveal global graph structure and create complex statistical dependencies that are challenging to analyze. Specifically, it is known from [33] that testing bipartiteness in the general graph model is strictly stronger than algorithms in the adjacency list or (exclusively) adjacency matrix models alone. However, proving lower bounds in the general model is conceptually difficult.

Theorem 3 (Informal)

Even in the general graph model, any algorithm that finds a clustering with an additive error of εn^2 requires $\Omega(n/\varepsilon)$ queries.

² Here is a word of caution on interpretation. Relative to Theorem 1, in Theorem 2 we change three parameters of the setting. (i) We impose a memory cap. (ii) We switch to the random–query model [36]. (iii) We study the *value* (cost) problem rather than the search problem. Our tradeoff should be read and contrasted with the other results in light of all three changes.

While this bound is not as tight as Theorem 1, it establishes for the first time that CC remains information-theoretically hard even when algorithms are equipped with the strongest local query arsenal. In our proof, we deal with the challenges of this model by analyzing query interactions on carefully constructed regular graphs (graphs where symmetry in some sense turns neighbor queries useless).³

Our results set forth the first query–space tradeoffs for graph problems. Beyond settling the adjacency–matrix complexity and showing hardness in the general model, Theorem 2 in some sense isolates the “price of memory”. Its proof provides a Boolean–Fourier and information-theoretic toolkit for random–query lower bounds that could potentially be of independent interest. We offer Correlation Clustering as the canonical testbed and this paper as the beginning of a broader research direction on query–space tradeoffs across graph primitives.

1.1 Our techniques

Due to space constraints, we only sketch the proof ideas of our main result here—the memory–query tradeoffs for correlation clustering (Theorem 1); the full version will include the proof techniques for the other two results.

To illustrate the proof techniques, we fix $\varepsilon = n^{-1/4}$ and consider (poly log n)-memory algorithms that approximate the clustering cost within an additive error of εn^2 . Given a graph G on n vertices, in the random query model, at each time step t , the algorithm receives a uniformly random vertex pair (u, v) along with the indicator bit specifying whether (u, v) is an edge of G . Using a VC-dimension argument [11], it follows that $O(n/\varepsilon^2) = O(n^{1.5})$ random queries suffice to produce a partition whose cost is within an additive error of $\varepsilon n^2 = n^{7/4}$ from optimal. Although representing such a partition requires $\Omega(n)$ bits of memory, it is natural to expect that just approximating the optimal clustering cost might be possible with significantly less memory⁴. Before our work, it was not known whether even $O(\log n)$ -memory algorithms could achieve an additive $O(n^{7/4})$ -approximation using only $O(n^{1.5})$ random queries⁵. We show that any poly log n -memory algorithm achieving this guarantee must in fact use $\tilde{\Omega}(n^{7/4}) \gg n^{1.5}$ random queries.

The random-query model was introduced in [36] to study memory–query tradeoffs for computing n -bit Boolean functions, when the algorithm receives a random input bit at each time step. This and the subsequent work of [24] established non-trivial tradeoffs for functions with high sensitivity and total influence, respectively. However, for promise or approximation problems, no small set of edges has a strong influence, and therefore, these prior techniques do not apply to our setting. Instead, our result builds on the approach of [31], which established the hardness of approximating MAX-CUT (and the clustering cost) within a multiplicative factor of $(2 - \delta)$ over sparse graphs, in the one-pass streaming model. Specifically, the paper leveraged tight lower bounds for a two-player one-way communication problem – called the Distributional Boolean Hidden Partition (D-BHP) problem – to show that any (random-order) streaming algorithm that distinguishes between a bipartite graph and a random graph must use $\Omega(\sqrt{n})$ bits of memory.

³ The formal statement and full proof are omitted for brevity; a full version including details will be available separately.

⁴ Indeed, low-space streaming algorithms are known for certain approximation parameters [10, 4].

⁵ While [4] established poly log n -space hardness for approximating the clustering cost within an additive error of $o(n^2)$ in the worst-case streaming model, proving hardness in the random query model introduces new technical challenges, even relative to the random-order streaming setting.

There are two main challenges in extending this approach to prove an $\tilde{\Omega}(n^{7/4})$ random-query lower bound for additive approximation of the optimal clustering cost (under memory constraints). First, [31] establishes lower bounds only for sparse graphs with $O(n)$ edges; for such sparse graphs, an additive approximation of $O(n^{1.5})$ to the clustering cost is trivial. Second, in the random-query model, after $\Omega(n)$ queries, edge repetitions occur with high probability, unlike in the random-order one-pass streaming model, where each edge is guaranteed to appear exactly once. The ability to go over the same edge twice significantly increases the technical difficulties in proving hardness results⁶.

To overcome the first challenge, we introduce and study a noisy variant of D-BHP, which we call the Perturbed Distributional Boolean Hidden Partition (PD-BHP) problem. This formulation allows us to work with dense graphs that satisfy the bipartite partition “approximately”. Starting with the Boolean Hidden Matching (BHM) problem introduced by [26], BHM, D-BHP, and their variants have been extensively used to prove streaming and sketching lower bounds (see, e.g., [37, 30, 31, 34, 28, 32, 29, 27, 18, 3]). The tight bounds we establish for PD-BHP may therefore be of independent interest. Similar to [31], we use a hybrid argument to lift the two-player one-way communication lower bound to a multi-player one-way communication bound, where each player receives an independent graph obeying the hidden partition. In the random-query model, however, there is a single underlying graph G dictating the inputs to all players; in particular, repeated queries to the same edge must be consistent with previous observations. This brings in the second challenge. To overcome it, we show that any algorithm capable of distinguishing between the case where all edge queries are consistent and the case where each queried edge is independently resampled must use either $\text{poly log } n$ bits of memory or $\Omega(n^2/\text{poly log } n)$ random queries. We refer to this as the same vector problem, and we establish tight memory–query tradeoffs for it.

Tight communication complexity for PD-BHP problem

Consider the following two-player communication problem. Alice receives a uniformly random n -bit vector $x \in \{0, 1\}^n$. Bob receives a graph $G = (V, E)$ on n vertices with a uniformly random set of r edges, and a vector $w \in \{0, 1\}^r$. In the NO case, each w_i is an independent uniformly random bit. In the YES case, for the i th edge $(u_i, v_i) \in E$, we have

$$w_i = \begin{cases} x_{u_i} \oplus x_{v_i}, & \text{with probability } \frac{1}{2} - 10\varepsilon, \\ 1 - (x_{u_i} \oplus x_{v_i}), & \text{with probability } \frac{1}{2} + 10\varepsilon. \end{cases}$$

In contrast, in the D-BHP problem studied by [31], the YES case is noiseless, that is $w_i = x_{u_i} \oplus x_{v_i}$. The goal is for Bob to distinguish between the two cases with as little communication from Alice as possible. We show that when Bob receives $r = \alpha n/\varepsilon^2$ edges (for some $0 < \alpha < 1$), and Alice sends at most $\gamma\sqrt{n}$ bits (for some $\gamma > 0$), Bob’s distinguishing advantage is at most $O((\gamma + \alpha)\alpha^{1/2})$. This bound is tight: if Alice simply sends the first $O(\sqrt{n}/\alpha)$ coordinates of x to Bob, then with high probability, the graph G will contain at least $\Omega(1/\varepsilon^2)$ edges whose endpoints lie entirely within these coordinates, enabling Bob to distinguish the two cases with constant advantage. To lift the two-player communication bound to the multi-player setting, we parametrize $0 < \alpha = \gamma$ when $\gamma < 1$, and $\alpha = O(1/\gamma^2)$

⁶ Only recently, the breakthrough work of [25] showed that any constant-pass streaming algorithm for distinguishing between a sparse bipartite graph and a random graph requires $\Omega(n^{1/3})$ bits of space. Since multi-pass streaming algorithms read the stream in the same order, this model is incomparable to the random query model.

when $\gamma \geq 1$. Our proof builds on the Fourier-analytic approach of [31] for the D-BHP problem. A key challenge in extending their proof is that, since Bob now receives $\gg n$ edges, the graph G necessarily contains many cycles (of length at least 3). In contrast, the argument of [31] relies crucially on the fact that when Bob receives $\ll n$ edges, the graph is cycle-free. By carefully balancing the number of cycles against the distinguishing advantage they provide, we are able to extend the Fourier-analytic proof to handle the noisy case on denser graphs.

Same vector problem

Let $N = \binom{n}{2}$. Consider the following distinguishing problem, where the goal is to decide whether the random queries are “consistent”. At each time step t , the algorithm receives a random index $i_t \in [N]$ and a bit b_t . In the NO case, b_t is uniformly random, whereas in the YES case, $b_t = x_{i_t}$, where $x \in \{0, 1\}^N$ is fixed at the start. We show that any algorithm that solves the distinguishing problem, in $\sqrt{N} < T < N$ time-steps, must use at least $\Omega(N/T)$ bits of memory.⁷ This bound is tight (up to log factors): by storing the first s queries, an algorithm will, with high probability, encounter a repeated index within $O(N/s)$ time steps, and thus distinguish with constant advantage. To obtain the tight result, we use an information-theoretic potential function that measures the maximum progress any s -memory algorithm can make towards distinguishing at a time-step. For our main theorem, we generalize the same vector problem to non-uniform bits.

2 Preliminaries

We always denote $[k]$ as the set $\{1, \dots, k\}$. For every binary string S , we use $|S|$ to denote its Hamming weight. For every two binary strings S, S' of the same length, we use $d_H(S, S')$ to denote their Hamming distance.

Basics of information theory

Given a random variable Z , we use $H(Z)$ to denote the Shannon entropy of Z , i.e., $H(Z) = -\sum_z \Pr[Z = z] \log(1/\Pr[Z = z])$. For a scalar $p \in (0, 1)$, $H(p)$ denotes the entropy of the Bernoulli random variable with p probability to be 1. We use $I(X; Y|Z)$ to denote the mutual information between X and Y conditioned on the random variable Z . $I(X; Y|Z) = H(X|Z) - H(X|Y, Z)$, where $H(X|Y) = \mathbb{E}_y H(X|Y = y) \leq H(X)$. Next, we describe some of the properties of mutual information used in the paper.

1. (Chain Rule) $I(X, Y; Z) = I(X; Z) + I(Y; Z|X)$.
2. For discrete random variables, X, Y, Z , $I(X; Y|Z) = 0 \iff X \perp Y|Z$.
3. If $I(Z_1; Z_2|X, Y) = 0$, then $I(X; Z_2|Y) \geq I(X; Z_2|Y, Z_1)$.
4. If $I(Z_1; Z_2|Y) = 0$, then $I(X; Z_2|Y) \leq I(X; Z_2|Y, Z_1)$.

Property 1 follows from the chain rule for Shannon entropy. For property 2, it is easy to see that if $X \perp Y|Z$, then $I(X; Y|Z) = 0$; the other direction uses strict concavity of the log function. Properties 3 and 4 follow from the observation that

$$I(X; Z_2|Y) + I(Z_1; Z_2|X, Y) = I(X, Z_1; Z_2|Y) = I(Z_1; Z_2|Y) + I(X; Z_2|Y, Z_1).$$

⁷ Recall that by the Birthday Paradox, the probability of sampling the same index twice is negligible when $T \ll \sqrt{N}$. Our result covers the entire non-trivial range of T where collisions are possible but not guaranteed.

As mutual information is non-negative, if $I(Z_1; Z_2|X, Y) = 0$, then $I(X; Z_2|Y) \geq I(X; Z_2|Y, Z_1)$ (because $I(Z_1; Z_2|Y) \geq 0$) and if $I(Z_1; Z_2|Y) = 0$, then $I(X; Z_2|Y) \leq I(X; Z_2|Y, Z_1)$.

Graph notations and problem definitions

Throughout this paper, we use $G = (V, E)$ to denote the input graph. We denote $n = |V|$ as the number of vertices. We use $\binom{V}{2}$ to denote the set of unordered pairs of vertices. Given two disjoint subsets V_1, V_2 of a graph G , we use $G[V_1, V_2]$ to denote the induced bipartite subgraph where V_1 and V_2 are the two parts. Given a vertex u and a number i , we use $N(u, i)$ to denote the i -th neighbor of vertex u .

We study the following three problems in this work in both versions where the output is an approximately optimal cost or a partition that yields an approximately optimal cost: the correlation clustering problem, the max cut problem, and the minimum bisection problem. The formal definition and statements to the latter two problems are deferred to the full version of the paper.

Correlation clustering

Given an undirected and unweighted graph $G = (V, E)$. A *clustering* \mathcal{C} of the graph is a partition C_1, C_2, \dots, C_k of its vertex set V where k is unfixed. A clustering also defines an equivalence between vertices: $u \sim_{\mathcal{C}} v$ if u and v belong to the same cluster, and $u \not\sim_{\mathcal{C}} v$ otherwise. The cost of a clustering is defined as the total number of missing edges inside each cluster plus the total number of edges crossing different clusters. Formally, we define it as

$$\text{cost}_G(\mathcal{C}) := |\{(u, v) \notin E : u \sim_{\mathcal{C}} v\}| + |\{(u, v) \in E : u \not\sim_{\mathcal{C}} v\}|$$

Fix a parameter $\varepsilon \in (0, 1/2)$, the *correlation clustering cost* problem asks for an approximate value A such that $|A - \min_{\mathcal{C}} \text{cost}_G(\mathcal{C})| \leq \varepsilon n^2$ given G as the input. The *correlation clustering partition* problem asks for a clustering \mathcal{C}' such that $\text{cost}_G(\mathcal{C}') \leq \min_{\mathcal{C}} \text{cost}_G(\mathcal{C}) + \varepsilon n^2$.

A popular and equivalent definition of correlation clustering is based on a signed graph, where each edge is associated with a positive (+) or a negative (-) sign. The goal is to find a clustering that minimizes the number of disagreements, which corresponds to our cost function $\text{cost}_G(\mathcal{C})$. The problem can also be framed as maximizing the number of agreements. In the additive error setting, these two goals are equivalent, as the sum of agreements and disagreements is the fixed total number of pairs of vertices, $\binom{n}{2}$. In our paper, we model positive edges as existing edges and negative edges as non-edges, and we focus on the case of a complete signed input graph. Our lower bounds for this complete graph model also imply a lower bound for general graph cases.

3 Memory-query tradeoffs in the random query model

We study the memory-query tradeoffs of the correlation clustering cost problem in the *random query model*. This model was introduced in [36] and subsequently studied in [24]. Our main result is the first memory-query tradeoff lower bound for graph problems in the random query model. Different from the previous works, we adapt the proof structure from [31], which is based on Fourier analysis.

The random query model

In the random query model, an algorithm can make random queries to the input graph $G = (V, E)$. Each query returns an independent and identically distributed (i.i.d.) uniformly random pair of vertices (u, v) from $\binom{V}{2}$ and an indicator $\mathbb{1}_{(u,v) \in E}$ of whether or not $(u, v) \in E$. We denote the return of the i -th query as (u_i, v_i, e_i) , where $e_i = \mathbb{1}_{(u_i, v_i) \in E}$.

► **Theorem 1.** *Let $G = (V, E)$ be an undirected simple graph with n vertices. Let Π be any randomized algorithm that, in the random query model, approximates the correlation clustering cost of G to within an additive error of εn^2 with probability at least $99/100$. For this algorithm, if the worst-case query complexity is q and the space used is at most $\gamma\sqrt{n}$ bits, then the following lower bound holds:*

$$q = \begin{cases} \Omega\left(\min\left(\frac{n}{\varepsilon^2\sqrt{\gamma}}, \frac{n\sqrt{n}}{\gamma}\right)\right) & \text{if } \gamma < 1 \\ \Omega\left(\min\left(\frac{n}{\varepsilon^2\gamma^2}, \frac{n\sqrt{n}}{\gamma}\right)\right) & \text{if } \gamma \geq 1 \end{cases}$$

for parameters $\varepsilon \in \left(\omega\left(\frac{1}{\sqrt{n}}\right), 0.05\right)$ and $\gamma \in \left(\omega\left(\frac{\log n}{\sqrt{n}}\right), o(\sqrt{n})\right]$.

We note that our lower bound, which holds for deterministic algorithms, also applies to randomized algorithms by a standard application of Yao's minimax principle. In addition, the same lower bound applies to the max cut and minimum bisection problems. The three problems share the same hard distribution. We defer the lower bound statement and its proof to the latter two problems to the full version of this paper, and focus on the correlation clustering problem in this section.

Our proof extends the streaming lower bound for max cut in [31]. They showed an $\tilde{\Omega}(\sqrt{n})$ space lower bound for approximating max cut with $(2 - \varepsilon)$ -multiplicative error for any constant $\varepsilon > 0$. We adapt their proof to establish lower bounds for a wide range of additive errors.

The core of our proof strategy is a reduction to two auxiliary problems. The first is a noisy variant of the *distributional Boolean Hidden Partition (D-BHP)* problem introduced in [26] and extended in [37, 31], and the second is what we call the *same vector problem*. The desired query-space lower bound follows from establishing lower bounds for these two problems. We will define these auxiliary problems formally in the following subsections.

3.1 Input distribution

We use an input distribution for the correlation clustering that differs from the one in [31]. While they studied the indistinguishability between sparse random graphs, we focus on the indistinguishability between two types of dense graphs: a random dense Erdős-Rényi graph and a random dense Erdős-Rényi graph with $\Theta(\varepsilon n^2)$ edges perturbed in expectation.

Let $\mathcal{G}_{n,p}$ denote the distribution of Erdős-Rényi graphs with parameters n and p . In a random graph from this distribution, each of the $\binom{n}{2}$ possible edges is included with an independent probability p .

We define two input graph distributions, \mathcal{G}^Y and \mathcal{G}^N .

- **The YES distribution (\mathcal{G}^Y):** A random graph from \mathcal{G}^Y is sampled as follows. We first uniformly generate a partition $P \in \{0, 1\}^n$. For every pair of vertices (v_i, v_j) , we independently include the edge (v_i, v_j) with probability $1/2 + (-1)^{P_i \oplus P_j} \rho$, where $\rho = 10\varepsilon$. For a fixed partition P , we denote this random graph distribution by \mathcal{G}_P^Y .
- **The NO distribution (\mathcal{G}^N):** This is simply the standard Erdős-Rényi graph distribution $\mathcal{G}_{n,1/2}$.

We will show later that the correlation clustering costs of graphs from these two distributions are $2\epsilon n^2$ -far from each other with high probability. Therefore, the lower bound reduces to the indistinguishability of graphs from the two distributions \mathcal{G}^Y and \mathcal{G}^N .

We will also use the graph distribution $\mathcal{G}_{n,r'}$ to define the perturbed distributional boolean hidden partition (PD-BHP) problem: we uniformly sample r' edges over all pairs of vertices, allowing for repetition; the graph is obtained by taking the union of the sampled edges, which is a graph of $r \leq r'$ edges.

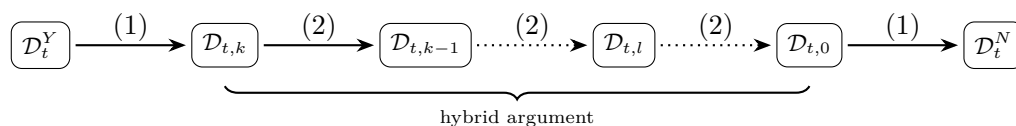
Distribution of Queries

We describe the distributions to the random queries the algorithm makes and their query answers. We define the distributions over query streams. Let a query stream be a sequence of t triples $((u_1, v_1, e_1), \dots, (u_t, v_t, e_t))$, where each pair (u_i, v_i) is sampled uniformly from $\binom{V}{2}$.

- \mathcal{D}_t^Y : The distribution of a stream of t random queries where the graph $G = (V, E)$ is sampled from \mathcal{G}^Y , and each e_i is set to $\mathbb{1}_{(u_i, v_i) \in E}$.
- \mathcal{D}_t^N : The distribution of a stream of t random queries where the graph G is sampled from \mathcal{G}^N .

Our lower bound applies to the mixed distribution $\mathcal{D}_t = \frac{1}{2}\mathcal{D}_t^Y + \frac{1}{2}\mathcal{D}_t^N$.

The indistinguishability of the two distributions is established on the indistinguishability between the two distributions and intermediate distributions. Specifically, fix a number $k \geq 1$ where we assume k divides t . For every $l \in \{0, \dots, k\}$, we let $\mathcal{D}_{t,l}$ to denote the following distribution of query streams: (i) uniformly sample a partition P from the 2^n possible partitions of the graph; (ii) let $G_1, \dots, G_l \leftarrow \mathcal{G}_P^Y$, and let $G_{l+1}, \dots, G_k \leftarrow \mathcal{G}^N$ independently; (iii) for each $i \in [k]$ and every $j = (i-1)t/k + 1, \dots, it/k$, we let $e_j = \mathbb{1}_{(u_j, v_j) \in E(G_i)}$. Intuitively, we divide the input stream into k phases, where the samples of each phase is from an independently sampled graph G from the YES or NO distribution. Our proof idea is as follows



(1): Indistinguishable by lower bounding the same-vector problem

(2): Indistinguishable by lower bounding the PD-BHP problem

■ **Figure 1** Structure of distributions used in the lower bound proof.

Given the distributions, we show that the correlation clustering cost of graphs in \mathcal{D}^Y and \mathcal{D}^N are $2\epsilon n^2$ -far apart from each other with high probability.

► **Lemma 2.** *Let $\epsilon \in \left(\omega\left(\frac{1}{\sqrt{n}}\right), 0.05\right)$. Then with probability $\geq 1 - n^{-\omega(1)}$ a random graph G drawn from \mathcal{G}^Y has correlation clustering cost at most $\frac{n^2}{4} - 5\epsilon n^2 + \tilde{O}(n)$; and a random graph G drawn from \mathcal{G}^N has correlation clustering cost at least $\frac{n^2}{4} - \tilde{O}(n\sqrt{n})$.*

Proof. We first lower bound the optimal clustering cost of a graph from $\mathcal{G}^N = \mathcal{G}(n, 1/2)$. Observe that for every clustering, every pair of vertices (u, v) contribute 1 cost with independent probability $1/2$. Let X denote the cost of an arbitrary clustering. By the standard Chernoff bound,

$$\Pr_{G \leftarrow \mathcal{G}^N} [|X - \mathbb{E}[X]| \geq n\sqrt{n} \log n] \leq 2 \cdot \exp\left(-\frac{n^3 \log^2 n}{3 \cdot \mathbb{E}[X]}\right) \leq 2 \cdot \exp\left(-\frac{4}{3}n \log^2 n\right)$$

67:10 Query Lower Bounds for Correlation Clustering Under Memory Constraints

Since there are at most $n^n < \exp(n \log n)$ clusterings, by union bound, a random graph G drawn from \mathcal{G}^N has correlation clustering cost at least $\frac{n^2}{4} - O(n^{3/2} \log n)$ with $\geq 1 - \exp(-n \log n)$ probability.

For graphs G drawn from \mathcal{G}^Y . By our construction to \mathcal{G}^Y , each pair of vertices (u, v) contribute 1 cost to $\text{cost}_G(P)$ with independent probability $1/2 - \rho$. Let X denote $\text{cost}_G(P)$. Then $\mathbb{E}[X] = \frac{n^2}{4} - 5\varepsilon n^2 - O(n)$. By Chernoff bound,

$$\Pr_{G \leftarrow \mathcal{G}^Y} [|X - \mathbb{E}[X]| \geq n \log n] \leq 2 \cdot \exp\left(-\frac{n^2 \log^2 n}{3 \cdot \mathbb{E}[X]}\right) \leq 2 \cdot \exp\left(-\frac{4}{3} \log^2 n\right) \quad \blacktriangleleft$$

3.2 The boolean hidden partition problem

We analyze the 2-party one-way communication complexity of the perturbed distributional Boolean hidden partition (PD-BHP) problem, a variant of the distributional Boolean hidden partition problem introduced in [31]. In this variant, the input vector is noisy.

Perturbed Distributional Boolean Hidden Partition Problem (PD-BHP)

In this problem, Alice receives a vector $x \in \{0, 1\}^n$, and Bob receives a graph $G = (V, E)$ and a vector $w \in \{0, 1\}^r$, where $r = |E|$. Let $M \in \{0, 1\}^{r \times n}$ be the edge-vertex incidence matrix of G . The problem is to distinguish between two cases based on the relationship between x , w , and G :

1. **YES case:** $w = Mx + \Delta$, where Δ is a random noise vector with each entry independently set to 1 with probability $1/2 + \rho$.
2. **NO case:** The vector w is uniformly random in $\{0, 1\}^r$, independent of x . This is equivalent to $w = Mx + \Delta'$, where Δ' is a uniformly random vector.

Alice sends a message to Bob, who must distinguish between the two cases.

We analyze this problem under a specific distribution where Alice's input x is uniformly random in $\{0, 1\}^n$, and Bob's graph G is sampled from $\mathcal{G}_{n, r'}$, a graph distribution defined by taking a union of r' edges sampled over all pairs of vertices, allowing for repetition. The final answer is YES or NO with probability $1/2$ each. As a result, the number of edges r in the resulting simple graph is exactly the number of distinct edges from the r' samples, and there do not exist two identical rows in the edge-vertex incidence matrix M .

► **Lemma 3.** Fix three parameters $\varepsilon \in \left(\omega\left(\frac{1}{\sqrt{n}}\right), 0.05\right)$, $\gamma > \omega\left(\frac{1}{\sqrt{n}} \log n\right)$ and $\alpha \in \left[\frac{\varepsilon^2}{n}, \min(10^{-7}\gamma^{-2}, 10^{-4})\right]$. Consider an instance of the PD-BHP problem where Alice receives a uniformly random string $x \in \{0, 1\}^n$, and Bob receives a graph $G \in \mathcal{G}_{n, \alpha n / \varepsilon^2}$ together with the corresponding vector w . Any deterministic protocol for the PD-BHP problem that uses at most $\gamma\sqrt{n}$ bits of communication can achieve an advantage of at most $O((\gamma + \alpha)\sqrt{\alpha})$ over random guessing.

This lower bound is tight when both α and γ are constants. In this scenario, Alice can send the first $\Theta(\sqrt{n})$ bits of her input to Bob. In expectation, Bob can then learn $\Theta(1/\varepsilon^2)$ entries of the vector Mx , and thus $\Theta(1/\varepsilon^2)$ entries of the noise vector Δ . Since the noise vector in the YES case and the NO case correspond to Bernoulli distributions that are ρ -far apart, $\Theta(1/\rho^2) = \Theta(1/\varepsilon^2)$ independent samples are sufficient to distinguish them with constant advantage.

This subsection focuses on proving the above lemma. Our proof follows the same framework as [26, 37, 31]. Compared to previous works, our problem involves a key trade-off: while the noise vector Δ makes distinguishing the distributions harder, the denser graphs make it easier. We establish a trade-off between the graph's density, the noise probability, and the communication complexity.

Specifically, Alice's message induces a partition A_1, A_2, \dots, A_{2^c} of $\{0, 1\}^n$, where c is the number of bits sent by Alice. Let the protocol's advantage over random guessing be at least τ , where $\tau := (\gamma + \alpha)\sqrt{\alpha}$. Notice that at least a $1 - \tau$ fraction of all input strings $x \in \{0, 1\}^n$ are contained within sets in the partition that have a size of at least $\tau 2^{n-c}$. We can therefore focus our analysis on one such "typical" set, which we denote by A . Define a parameter $c' = c + \log(1/\tau)$, the size of A is $|A| \geq 2^{n-c'}$.

The central idea is to show that if Alice's input x is drawn uniformly from such a typical set A , the resulting vector Mx is statistically close to uniform. If this holds, Bob is unable to distinguish the YES case (where $\Delta = Mx + w$) from the NO case (where Δ is uniformly random).

Our main technical contribution is an extension of Fourier analysis techniques from prior work [26, 37, 31]. While previous work dealt with sparse, cycle-free graphs [31], our analysis is adapted to handle the presence of cycles.

We use the following normalization of the Fourier transform:

$$\hat{f}(v) = \frac{1}{2^n} \sum_{x \in \{0,1\}^n} f(x)(-1)^{x \cdot v}$$

We use the following bounds on the Fourier mass of f contributed by coefficients of various weight:

► **Lemma 4** (Lemma 6 in [26]). *Let $A \subseteq \{0, 1\}^n$ of size at least $2^{n-c'}$ and f its indicator function. Then for every $l \in [4c']$*

$$\frac{2^{2n}}{|A|^2} \sum_{v:|v|=l} \hat{f}(v)^2 \leq \left(\frac{4\sqrt{2}c'}{l} \right)^l$$

Recall that G is the graph Bob receives as input. The number of edges in G is r where $r \leq \alpha n/\varepsilon^2$. $M \in \{0, 1\}^{r \times n}$ denotes the incidence matrix of G where $M_{eu} = 1$ iff $u \in V$ is an endpoint of $e \in \binom{V}{2}$. We are interested in the distribution of $Mx + \Delta$ where x is uniformly randomly selected from A and Δ is a random noise vector where each entry is 1 with probability $1/2 + \rho$. For $z \in \{0, 1\}^r$, let

$$p_M(z) = \frac{\sum_{x \in A} \Pr_{\Delta}[Mx + \Delta = z]}{|A|}$$

Note that p_M is a function of the message A . We will prove that $p_M(z)$ is close to uniform by bounding the sum of squares of its Fourier coefficients for non-zero weight vectors. Specifically, let U_r denote the uniform distribution over r -bit binary strings, then

$$\|p_M - U_r\|_{\text{tvd}}^2 \leq 2^r \|p_M - U_r\|_2^2 = 2^{2r} \sum_{s \in \{0,1\}^r, s \neq 0} \widehat{p}_M(s)^2$$

67:12 Query Lower Bounds for Correlation Clustering Under Memory Constraints

By expanding the Fourier coefficients of p_M , we have

$$\begin{aligned}
\widehat{p}_M(s) &= \frac{1}{2^r} \sum_{z \in \{0,1\}^r} p_M(z) (-1)^{z \cdot s} \\
&= \frac{1}{|A|2^r} \sum_{\Delta' \in \{0,1\}^r} \Pr[\Delta = \Delta'] \cdot (|\{x \in A : (Mx + \Delta') \cdot s = 0\}| - |\{x \in A : (Mx + \Delta') \cdot s = 1\}|) \\
&= \frac{1}{|A|2^r} \sum_{\Delta' \in \{0,1\}^r} \Pr[\Delta = \Delta'] \cdot (|\{x \in A : x \cdot (M^T s) = (\Delta')^T s\}| - |\{x \in A : x \cdot (M^T s) = (\Delta')^T s + 1\}|) \\
&= \frac{1}{|A|2^r} \sum_{\Delta' \in \{0,1\}^r} \Pr[\Delta = \Delta'] \sum_x f(x) \cdot (-1)^{x \cdot (M^T s)} \cdot (-1)^{(\Delta')^T s} \\
&= \frac{2^n \widehat{f}(M^T s)}{|A|2^r} \sum_{\Delta' \in \{0,1\}^r} \Pr[\Delta = \Delta'] \cdot (-1)^{(\Delta')^T s} \\
&= \frac{2^n \widehat{f}(M^T s)}{|A|2^r} (-2\rho)^{|s|}
\end{aligned}$$

where the last equation is derived from the fact that the summation is exactly the expansion of the polynomial

$$((1/2 + \rho) + (1/2 - \rho))^{r-|s|} ((1/2 - \rho) - (1/2 + \rho))^{|s|} = (-2\rho)^{|s|}$$

Therefore, we can rewrite the TV distance as

$$\begin{aligned}
\|p_M - U_r\|_{\text{TV}}^2 &\leq 2^{2r} \sum_{s \in \{0,1\}^r, s \neq 0} \widehat{p}_M(s)^2 \\
&= \frac{2^{2n}}{|A|^2} \sum_{s \in \{0,1\}^r, s \neq 0} \widehat{f}(M^T s)^2 \cdot (2\rho)^{2|s|} \\
&= \frac{2^{2n}}{|A|^2} \sum_{v \in \{0,1\}^n} \widehat{f}(v)^2 \sum_{s \neq 0} (2\rho)^{2|s|} \mathbb{1}_{v=M^T s} \\
&= \frac{2^{2n}}{|A|^2} \sum_{l \geq 0} \sum_{v \in \{0,1\}^n, |v|=l} \widehat{f}(v)^2 \sum_{s \neq 0} (2\rho)^{2|s|} \mathbb{1}_{v=M^T s}
\end{aligned} \tag{1}$$

where the first inequality is by Cauchy-Schwartz, the second is Parseval's equality. We will show that $\sum_{s \neq 0} (2\rho)^{2|s|} \mathbb{1}_{v=M^T s}$ is well-bounded for every v . Combined with Lemma 4, we get our desired bound.

If we regard s as an indicator of a subgraph of G , where each edge e_i is selected if $s_i = 1$, the summation represents the total weight of different vector s such that the parity of degrees of vertices in the subgraph induced by s is exactly the vector v .

► **Lemma 5.** *Let M be the edge incidence matrix of a graph G drawn from $\mathcal{G}_{n, \alpha n / \varepsilon^2}$. For any vector $v \in \{0,1\}^n$ with Hamming weight $l = |v|$, the following bound on the expected value holds for parameters $\varepsilon \in \left(\omega\left(\frac{1}{\sqrt{n}}\right), 0.05\right)$ and $\alpha \in \left[\frac{\varepsilon^2}{n}, 10^{-4}\right]$:*

$$\mathbb{E}_M \left[\sum_{s \in \{0,1\}^r, s \neq 0} (2\rho)^{2|s|} \mathbb{1}_{v=M^T s} \right] \leq \begin{cases} 10^{10} \alpha^3 & \text{if } l = 0 \\ (10^4 \alpha / n)^{l/2} & \text{if } l > 0 \end{cases}$$

The above expectation bound holds because the weight $\rho = 10\varepsilon$ is small enough, and because r is upper bounded by $\alpha n / \varepsilon^2$.

Proof. We will first bound the expectation for every fixed r . Notice that the graph distribution $\mathcal{G}_{n,\alpha n/\varepsilon^2}$ given the number of edges r fixed is uniform over the set of all r -edge simple graphs.

Now fix an arbitrary $r \leq \alpha n/\varepsilon^2$. We bound the expectation of the summation by counting the number of matrices M such that $v = M^T s$ for every fixed vector $s \in \{0, 1\}^r$. Assume l to be an even number as otherwise the summation is always 0. Denote by $G_{M,s}$ the graph induced by M and s . Then the set of all edges in $G_{M,s}$ can be decomposed into $\geq l/2$ walks, each walk either starts and finishes at the same (arbitrary) vertex, or starts and finishes at vertices where $v_i = 1$. There are exactly $l/2$ walks of the second type, which forms the vector v .

We first bound the probability that k random edges form a walk. Either the walk has determined start/end points, or it is a closed walk. For closed walks, this probability that k selected (and distinct) edges from M of a fixed order form a closed walk is at most

$$n(2.01/n)^{k-1}(2.01/n^2) = 2.01^k/n^k$$

The factor of n denotes the number of start vertices of the walk. At each step of the walk, with probability at most $\frac{n-1}{\binom{n}{2}-r} \leq 2.01/n$ the next sampled edge shares an endpoint with the previous edge. The last step of the walk must go back to the start point, hence with probability $\frac{1}{\binom{n}{2}} \leq 2.01/n^2$ one samples this edge.

Analogously, for walks with fixed start and end vertices, the probability is at most

$$(2.01/n)^{k-1}(2.01/n^2) = 2.01^k/n^{k+1}$$

Therefore, for every fixed r , we have

$$\begin{aligned} & \mathbb{E}_{M: M \in \{0,1\}^{r \times n}} \left[\sum_{s \in \{0,1\}^r, s \neq 0} (2\rho)^{2|s|} \mathbb{1}_{v=M^T s} \right] \\ & \leq \sum_{t=\begin{cases} 3 & \text{if } l=0 \\ l/2 & \text{if } l>0 \end{cases}}^r \binom{r}{t} (2\rho)^{2t} 2^t t! 2.01^t / n^{t+l/2} \\ & = \sum_{t=\begin{cases} 3 & \text{if } l=0 \\ l/2 & \text{if } l>0 \end{cases}}^r \binom{r}{t} t! 1608^t \varepsilon^{2t} / n^{t+l/2} \\ & \leq \sum_{t=\begin{cases} 3 & \text{if } l=0 \\ l/2 & \text{if } l>0 \end{cases}}^r r^t 1608^t \varepsilon^{2t} / n^{t+l/2} \end{aligned}$$

where the last inequality is due to the fact that $\binom{r}{t} t! \leq r^t$. The factor of $2^t t!$ accounts for ordering the t edges arbitrarily ($t!$ ways) and independently partitioning it to multiple walks (2^{t-1} ways).

Notice that for $l=0$, one needs to select at least 3 distinct edges to form $v=0^n$ since $s \neq 0$. The number of edges cannot be 2 because two edges do not form a cycle in a simple graph. For $l>0$, one needs at least $l/2$ edges. Note that the desired expectation is a weighted

67:14 Query Lower Bounds for Correlation Clustering Under Memory Constraints

sum of the expectations for every fixed r . We have

$$\begin{aligned}
 & \mathbb{E}_{M:G_M \leftarrow \mathcal{G}_{n,\alpha n/\varepsilon^2}} \left[\sum_{s \in \{0,1\}^r, s \neq 0} (2\rho)^{2|s|} \mathbb{1}_{v=M^T s} \right] \\
 & \leq \max_{r \leq \alpha n/\varepsilon^2} \mathbb{E}_{M:M \in \{0,1\}^{r \times n}} \left[\sum_{s \in \{0,1\}^r, s \neq 0} (2\rho)^{2|s|} \mathbb{1}_{v=M^T s} \right] \\
 & \leq \max_{r \leq \alpha n/\varepsilon^2} \sum_{t=0}^r r^t 1608^t \varepsilon^{2t} / n^{t+l/2} \\
 & \quad t = \begin{cases} 3 & \text{if } l = 0 \\ l/2 & \text{if } l > 0 \end{cases} \\
 & \leq \sum_{t=0}^r (1608\alpha)^t / n^{l/2} \\
 & \quad t = \begin{cases} 3 & \text{if } l = 0 \\ l/2 & \text{if } l > 0 \end{cases} \\
 & \leq \begin{cases} 10^{10} \alpha^3 & \text{if } l = 0 \\ (10^4 \alpha/n)^{l/2} & \text{if } l > 0 \end{cases}
 \end{aligned}$$

The sum is dominated by small t because it is a decreasing geometric sum when $\alpha \leq 10^{-4}$. ◀

With Lemma 5, we are able to bound the statistical distance between p_M and the uniform distribution.

► **Lemma 6.** Fix three parameters $\varepsilon \in \left(\omega\left(\frac{1}{\sqrt{n}}\right), 0.05\right)$, $\gamma > \omega\left(\frac{1}{\sqrt{n}} \log n\right)$ and $\alpha \in \left[\frac{\varepsilon^2}{n}, \min(10^{-7}\gamma^{-2}, 10^{-4})\right]$. Let $\tau = (\gamma + \alpha)\sqrt{\alpha}$ and $c' = \gamma\sqrt{n} + \log(1/\tau)$. Let $A \subseteq \{0,1\}^n$ of size at least $2^{n-c'}$, and let $f: \{0,1\}^n \rightarrow \{0,1\}$ be the indicator of A . Let G be a random graph sampled from $\mathcal{G}_{n,\alpha n/\varepsilon^2}$. Then

$$\mathbb{E}_M[\|p_M - U_r\|_{tvd}^2] = O((\alpha^2 + \gamma^2)\alpha)$$

Proof. Recall that by inequality (1),

$$\|p_M - U_r\|_{tvd}^2 \leq \frac{2^{2n}}{|A|^2} \sum_{l \geq 0} \sum_{v \in \{0,1\}^n, |v|=l} \hat{f}(v)^2 \sum_{s \neq 0} (2\rho)^{2|s|} \mathbb{1}_{v=M^T s}$$

We break this summation into two parts: the part $l \in [0, 4c']$ and the part $l \in [4c', n]$. For the first part, by Lemma 4 and Lemma 5

$$\begin{aligned}
 & \mathbb{E}_M \left[\frac{2^{2n}}{|A|^2} \sum_{l=0}^{4c'-1} \sum_{l \text{ is even}} \sum_{v \in \{0,1\}^n, |v|=l} \hat{f}(v)^2 \sum_{s \neq 0} (2\rho)^{2|s|} \mathbb{1}_{v=M^T s} \right] \\
 & \leq 10^{10} \alpha^3 + \sum_{l=2, l \text{ is even}}^{4c'-1} \left(\frac{4\sqrt{2}c'}{l}\right)^l \cdot (10^4 \alpha/n)^{l/2} \\
 & \leq O(\alpha^3) + \sum_{l=2, l \text{ is even}} (10^6 \gamma^2 \alpha / l^2)^{l/2} \\
 & = O((\alpha^2 + \gamma^2)\alpha)
 \end{aligned}$$

where $\frac{2^{2n}}{|A|^2} \hat{f}(0^n) = 1$ and $10^6 \gamma^2 \alpha < 0.1$.

For the part $l \in [4c', n]$, we have

$$\begin{aligned} & \mathbb{E}_M \left[\frac{2^{2n}}{|A|^2} \sum_{l \geq 4c'} \sum_{v \in \{0,1\}^n, |v|=l} \hat{f}(v)^2 \sum_{s \neq 0} (2\rho)^{2|s|} \mathbb{1}_{v=M^T s} \right] \\ & \leq 2^{c'} \cdot (10^4 \alpha/n)^{2c'} \\ & = (\sqrt{2} \cdot 10^4 \alpha/n)^{2\gamma\sqrt{n} + O(\log n)} \\ & = n^{-\omega(\log n)} \end{aligned}$$

where we used the fact that

$$\frac{2^{2n}}{|A|^2} \sum_{v \in \{0,1\}^n, |v|=l} \hat{f}(v)^2 \leq \frac{2^{2n}}{|A|^2} \sum_{v \in \{0,1\}^n} \hat{f}(v)^2 = \frac{2^n}{|A|} \leq 2^{c'}$$

Together with the part $l \in [0, 4c')$, we have

$$\mathbb{E}_M [\|p_M - U_r\|_{tvd}^2] = O((\alpha^2 + \gamma^2)\alpha) \quad \blacktriangleleft$$

We will need the following lemma, which is the Lemma 5.6 from [31].

► **Lemma 7** (Lemma 5.6 of [31]). *Let $(X, Y^1), (X, Y^2)$ be random variables taking values on finite sample space $\Omega = \Omega_1 \times \Omega_2$. For any $x \in \Omega_1$ let $Y_x^i, i = 1, 2$ denote the conditional distribution of Y^i given the event $\{X = x\}$. Then*

$$\|(X, Y^1) - (X, Y^2)\|_{tvd} = \mathbb{E}_X [\|Y_X^1 - Y_X^2\|_{tvd}]$$

We now prove Lemma 3.

Proof of Lemma 3. Let $P(x)$ be Alice's message function and let $Q(M, i, w)$ be Bob's function, where M is the edge incidence matrix of the input graph, w is input vector of Bob, and i is Alice's message. Since we are analyzing the protocol under a fixed input distribution, we can assume P and Q are deterministic.

Let D^1 denote the distribution of $(M, P(x), w)$ under YES instances, and D^2 the same under NO instances. We aim to show that

$$\|D^1 - D^2\|_{tvd} = O\left((\gamma + \alpha)\alpha^{1/2}\right),$$

which implies that the protocol's distinguishing advantage is at most $O((\gamma + \alpha)\sqrt{\alpha})$ on PD-BHP.

The function $P(x)$ partitions $\{0, 1\}^n$ into sets A_1, \dots, A_{2^c} , where $c = \gamma\sqrt{n}$ is the number of bits communicated. Since there are 2^c such sets, at least a $1 - \tau$ fraction of $\{0, 1\}^n$ is contained in those A_i of size at least $\tau 2^{n-c}$, where we define $\tau := (\alpha + \gamma)\sqrt{\alpha}$. We call any message i with $|A_i| \geq \tau 2^{n-c}$ a *typical* message. Then the probability that $i = P(x)$ is not typical on a uniformly random input x is at most τ .

Let us write $D^1 = (M, i, p_{M,i})$, where $p_{M,i}$ is the conditional distribution of w given M and i under the YES distribution, and write $D^2 = (M, i, U_r)$, where U_r is the uniform distribution over $\{0, 1\}^r$. For each M and i , let $D_{(M,i)}^1 := p_{M,i}$ and $D_{(M,i)}^2 := U_r$ denote the respective conditional distributions of w .

Applying Lemma 7, we get:

$$\begin{aligned} \|D^1 - D^2\|_{tvd} &= \mathbb{E}_i \left[\mathbb{E}_M \left[\|D_{(M,i)}^1 - D_{(M,i)}^2\|_{tvd} \right] \right] \\ &\leq \Pr[i \text{ not typical}] + \mathbb{E}_M [\|p_{M,i'} - U_r\|_{tvd}] \quad (i' \text{ any typical message}) \\ &\leq \tau + \mathbb{E}_M [\|p_{M,i'} - U_r\|_{tvd}] \end{aligned}$$

67:16 Query Lower Bounds for Correlation Clustering Under Memory Constraints

Now fix any typical message i' . Then, by definition, $|A_{i'}| \geq 2^{n-c'}$ where $c' = \gamma\sqrt{n} + \log(1/\tau)$. Lemma 6 then implies

$$\mathbb{E}_M [\|p_{M,i'} - U_r\|_{\text{tvd}}^2] = O((\gamma^2 + \alpha^2)\alpha).$$

Applying Jensen's inequality yields

$$\mathbb{E}_M [\|p_{M,i'} - U_r\|_{\text{tvd}}] \leq \sqrt{\mathbb{E}_M [\|p_{M,i'} - U_r\|_{\text{tvd}}^2]} = O((\gamma + \alpha)\sqrt{\alpha}).$$

Combining this with the earlier bound, we conclude that

$$\|D^1 - D^2\|_{\text{tvd}} = O((\gamma + \alpha)\sqrt{\alpha}),$$

as claimed. ◀

3.3 Same vector problem

We prove a query-space lower bound for the following problem in the random query model.

Same vector problem

Let $N = \binom{n}{2}$, $p \in [0, 1]$ and $k = o(N)$. Let $q < N$ divides N . Let X, Y_1, \dots, Y_k be binary vectors drawn from $\text{Bern}(p)^N$ independently, i.e., they are length- N vectors with each bit being 1 with probability p . An algorithm for this problem has k phases. At each phase $k' \in [k]$, the algorithm makes q random queries. The return of each random query is a pair (I_t, W_{I_t}) where $I_t \in_R [N]$ independently. For W_{I_t} , (i) in the YES case, $W_{I_t} = X_{I_t}$; (ii) in the NO case, $W_{I_t} = Y_{k', I_t}$ at phase k' . The algorithm must distinguish between the two cases above.

When $k > 1$, a trivial upper bound is to memorize $\tilde{O}(N/kq)$ queries from the first $kq/2$ queries, and check if there is any collision with the latter $kq/2$ samples. We show that the upper bound is tight using information theory. Formally, we obtain the following lower bound.

► **Lemma 8.** *Fix $k \geq 1$ and $q \geq 2n$. Fix $p \in [0, 1]$ to be a function of n . For every algorithm that solves the same vector problem correctly with probability $\geq 2/3$, the algorithm must use $\Omega(N/kq)$ bits of memory. Specifically, this is true when the algorithm outputs correctly w.p. $\geq 2/3$ on an input distribution where the input is in the YES case and the NO case uniformly.*

We will need to use Fano's inequality and Shearer's lemma in the proof.

► **Proposition 9** (Fano's inequality). *Let $X \rightarrow Y \rightarrow \tilde{X}$ be a Markov chain, and let $p_e = \Pr[\tilde{X} \neq X]$. Let $H(x) = -x \log x - (1-x) \log(1-x)$ denote the binary entropy function. Let \mathcal{X} be the support of X . Then*

$$H(p_e) + p_e \log(|\mathcal{X}| - 1) \geq H(X|Y)$$

► **Proposition 10** (Shearer's lemma [19]). *Let X_1, \dots, X_n be random variables, and let $S_1, \dots, S_m \subseteq [n]$ be subsets such that each $i \in [n]$ belongs to at least k sets. Then*

$$k \cdot H(X_1, \dots, X_n) \leq \sum_{j=1}^m H(X_i : i \in S_j)$$

Proof to Lemma 8. We will prove the lower bound to any algorithm over the distribution where the input is in the YES case and the NO case uniformly.

Let c denote the length of the memory. For every $j \in [kq]$, let $M_j \in \{0, 1\}^c$ denote the memory after receiving the j -th random query. Specifically, $M_0 = 0^N$ denotes the initial memory.

Without loss of generality assume that M_{kq} is the output of the algorithm. Let D be an indicator such that $D = 1$ at the YES case, and $D = 0$ at the NO case. Let $D \rightarrow M_{kq} \rightarrow M_{kq}$ be the Markov chain. By Fano's inequality and $p_e \leq 1/3$, we get

$$0.92 \geq H(1/3) \geq H(p_e) \geq H(D|M_{kq}) = 1 - I(D; M_{kq}) \quad (2)$$

We will decompose the mutual information $I(D; M_{kq})$ into the sum of increases of mutual information in each phase. Formally, we denote by $I_{(t)}$ the list of random variables $I_{t(q-1)+1}, \dots, I_{tq}$ in phase t , and we use the same notation for other variables.

$$\begin{aligned} & I(D; M_{kq}) \\ &= \sum_{t=1}^k I(D; M_{tq}) - I(D; M_{(t-1)q}) \\ &\leq \sum_{t=1}^k I(D; M_{(t-1)q}, I_{(t)}, W_{I_{(t)}}) - I(D; M_{(t-1)q}) \\ &= \sum_{t=1}^k I(D; I_{(t)}, W_{I_{(t)}} | M_{(t-1)q}) \\ &= \sum_{t=1}^k I(D; W_{I_{(t)}} | I_{(t)}, M_{(t-1)q}) \end{aligned} \quad (3)$$

by the data processing inequality.

We will upper bound each $I(D; W_{I_{(t)}} | I_{(t)}, M_{(t-1)q})$ by $I(M_{(t-1)q}; X_{I_{(t)}} | D, I_{(t)})$. For every t, m and $i_{(t)}$, let $\gamma_{t,m,i_{(t)}} = \Pr[D = 0 | M_{t-1} = m, I_t = i_{(t)}]$. We can rewrite

$$\begin{aligned} & \sum_{t=1}^k I(D; W_{I_{(t)}} | I_{(t)}, M_{(t-1)q}) \\ &= \sum_{t=1}^k H(W_{I_{(t)}} | I_{(t)}, M_{(t-1)q}) - H(W_{I_{(t)}} | D, I_{(t)}, M_{(t-1)q}) \\ &\leq \sum_{t=1}^k H(W_{I_{(t)}} | I_{(t)}) - \mathbb{E}_{m, i_{(t)}} [H(W_{I_{(t)}} | D, I_{(t)} = i_{(t)}, M_{(t-1)q} = m)] \\ &= \sum_{t=1}^k H(W_{I_{(t)}} | I_{(t)}) - \mathbb{E}_{m, i_{(t)}} [\gamma_{t,m,i} H(W_{I_{(t)}} | D = 0, I_{(t)} = i_{(t)}, M_{(t-1)q} = m) + \\ & \quad (1 - \gamma_{t,m,i}) H(W_{I_{(t)}} | D = 1, I_{(t)} = i_{(t)}, M_{(t-1)q} = m)] \end{aligned}$$

where the inequality follows from the fact that conditioning reduces entropy.

And we can rewrite the above as the summation of $I(M_{(t-1)q}; X_{I_{(t)}} | D, I_{(t)})$:

$$\begin{aligned}
 & \sum_{t=1}^k I(M_{(t-1)q}; X_{I_{(t)}} | D, I_{(t)}) \\
 &= \sum_{t=1}^k H(X_{I_{(t)}} | D, I_{(t)}) - H(X_{I_{(t)}} | D, M_{(t-1)q}, I_{(t)}) \\
 &= \sum_{t=1}^k H(X_{I_{(t)}} | I_{(t)}) - \mathbb{E}_{m, i_{(t)}} [H(X_{I_{(t)}} | D, M_{(t-1)q} = m, I_{(t)} = i_{(t)})] \\
 &= \sum_{t=1}^k H(W_{I_{(t)}} | I_{(t)}) - \mathbb{E}_{m, i_{(t)}} [\gamma_{t, m, i} H(X_{I_{(t)}} | D = 0, I_{(t)} = i_{(t)}, M_{(t-1)q} = m) + \\
 & \quad (1 - \gamma_{t, m, i}) H(X_{I_{(t)}} | D = 1, I_{(t)} = i_{(t)}, M_{(t-1)q} = m)] \\
 &= \sum_{t=1}^k H(W_{I_{(t)}} | I_{(t)}) - \mathbb{E}_{m, i_{(t)}} [\gamma_{t, m, i} H(W_{I_{(t)}} | D = 0, I_{(t)} = i_{(t)}, M_{(t-1)q} = m) + \\
 & \quad (1 - \gamma_{t, m, i}) H(W_{I_{(t)}} | D = 1, I_{(t)} = i_{(t)}, M_{(t-1)q} = m)] \\
 &\geq \sum_{t=1}^k I(D; W_{I_{(t)}} | I_{(t)}, M_{(t-1)q})
 \end{aligned} \tag{4}$$

We are able to replace the conditional entropy of $X_{I_{(t)}}$ by $W_{I_{(t)}}$ regardless of whether the answer is YES or NO. When $D = 1$ (the YES case), $W_{I_{(t)}} = X_{I_{(t)}}$ by our definition. When $D = 0$ (the NO case), $W_{I_{(t)}} = Y_{t, I_{(t)}}$ is independent from previous samples, and follows the same (conditional) distribution as $X_{I_{(t)}}$.

Let $S \leftarrow [N]^q$ denote a random subset $S \subseteq [N]$ to be the range of a random vector from $[N]^q$. We have

$$\begin{aligned}
 I(M_{(t-1)q}; X_{I_{(t)}} | D, I_{(t)}) &= \frac{1}{2} I(M_{(t-1)q}; X_{I_{(t)}} | D = 1, I_{(t)}) \\
 &= \frac{1}{2} \mathbb{E}_{S \leftarrow [N]^q} [I(M_{(t-1)q}; X_S | D = 1)] \\
 &= \frac{1}{2} \mathbb{E}_{S \leftarrow [N]^q} [H(X_S | D = 1) - H(X_S | D = 1, M_{(t-1)q})] \\
 &= \frac{1}{2} \mathbb{E}_{S \leftarrow [N]^q} [H(X_S | D = 1)] - \frac{1}{2} \mathbb{E}_{S \leftarrow [N]^q} [H(X_S | D = 1, M_{(t-1)q})] \\
 &= \frac{1}{2} \mathbb{E}_{S \leftarrow [N]^q} \left[\sum_{i \in S} H(X_i | D = 1) \right] - \frac{1}{2} \mathbb{E}_{S \leftarrow [N]^q} [H(X_S | D = 1, M_{(t-1)q})] \tag{5}
 \end{aligned}$$

Let $\mu = \Pr_{S \leftarrow [N]^q} [i \in S], \forall i \in [N]$. We have $\mu = 1 - (1 - 1/N)^q \leq q/N$. Then, by Shearer's Lemma (Proposition 10),

$$\mathbb{E}_{S \sim I_{(t)}} [H(X_S | D = 1, M_{(t-1)q})] \geq \mu \cdot H(X | D = 1, M_{(t-1)q}).$$

And,

$$\begin{aligned}
 \mathbb{E}_{S \sim I_{(t)}} \left[\sum_{i \in S} H(X_i | D = 1) \right] &= \sum_{i \in [N]} \Pr_{S \leftarrow [N]^q} [i \in S] \cdot H(X_i | D = 1) \\
 &= \mu \left(\sum_{i \in [N]} H(X_i | D = 1) \right) \\
 &= \mu \cdot H(X | D = 1).
 \end{aligned}$$

Substituting in Equation (5), we get

$$\begin{aligned} I(M_{(t-1)q}; X_{I(t)} | D = 1, I(t)) &\leq \frac{\mu}{2} H(X|D = 1) - \frac{\mu}{2} H(X|D = 1, M_{(t-1)q}) \\ &= \frac{\mu}{2} I(M_{(t-1)q}, X|D = 1) \\ &\leq \frac{cq}{2N}. \end{aligned}$$

Combined with inequalities (2), (3), (4), we obtain

$$0.92 \geq 1 - I(D; M_{kq}) \geq 1 - \sum_{t=1}^k I(D; W_{I(t)} | I(t), M_{(t-1)q}) \geq 1 - \sum_{t=1}^k I(M_{(t-1)q}; X_{I(t)} | D, I(t)) \geq 1 - \frac{ckq}{2N}$$

and we get a space lower bound $c \geq 0.16 \frac{N}{kq}$. \blacktriangleleft

3.4 Proof to the main theorem

In this section, we prove the main theorem, which is a query-space lower bound to approximating the max cut in the random query model. We repeat the theorem here.

► **Theorem 1.** *Let $G = (V, E)$ be an undirected simple graph with n vertices. Let Π be any randomized algorithm that, in the random query model, approximates the correlation clustering cost of G to within an additive error of εn^2 with probability at least $99/100$. For this algorithm, if the worst-case query complexity is q and the space used is at most $\gamma\sqrt{n}$ bits, then the following lower bound holds:*

$$q = \begin{cases} \Omega\left(\min\left(\frac{n}{\varepsilon^2\sqrt{\gamma}}, \frac{n\sqrt{n}}{\gamma}\right)\right) & \text{if } \gamma < 1 \\ \Omega\left(\min\left(\frac{n}{\varepsilon^2\gamma^2}, \frac{n\sqrt{n}}{\gamma}\right)\right) & \text{if } \gamma \geq 1 \end{cases}$$

for parameters $\varepsilon \in \left(\omega\left(\frac{1}{\sqrt{n}}\right), 0.05\right)$ and $\gamma \in \left(\omega\left(\frac{\log n}{\sqrt{n}}\right), o(\sqrt{n})\right]$.

The lower bound is obtained through a reduction from both the same-vector problem and the PD-BHP problem to the max cut problem. Our reduction follows a similar structure to [31] despite adding a reduction to the same vector problem. This reduction is necessary in our case. We start from introducing the reduction used in [31]. Then we outline how we adapt this reduction to obtain our result. After presenting a full map, we will show the formal reduction.

In [31], the main result is a $\tilde{\Omega}(\sqrt{n})$ space lower bound for $(2 - \varepsilon)$ -approximating the max cut given a stream of edges of the input graph. To that end, they show that every algorithm with limited space cannot distinguish a bipartite graph from an Erdős-Rényi graph with the same expected number of edges. The lower bound is also achieved by a reduction to the D-BHP problem, which is similar to the PD-BHP problem we study in this work but (i) the noise vector is always 0^n ; (ii) the number of edges given to Bob is $\Theta(n/k)$ where $k = \Theta(\log n)$ is the number of phases. Besides, their total number of edges over all k phases in the stream is only $\Theta(n)$.

Their reduction has two steps. First, they show that a stream of k independently sampled random graphs of $\Theta(n/k)$ edges is close in total variation to a stream of the edges of a random graph of $\Theta(n)$ edge. This holds both when the graph is sampled from an Erdős-Rényi distribution or when the graph is a random bipartite graph given a fixed random partition. It is true because the total number of edges is $\Theta(n)$ and it is unlikely an edge will be sampled

twice in two different phases. Given the first step, we only need to show that algorithms cannot distinguish between a stream of k phases of random sparse bipartite graphs given and a stream of k phases of random sparse Erdos-Renyi graphs.

The next step is the standard hybrid argument. For every $i \in \{0, \dots, k\}$, let $\mathcal{D}_{(i)}^Y$ denote the input distribution where the first i phases are random sparse bipartite graphs given a fixed random partition, and the last $k - i$ phases are sparse Erdős-Rényi graphs. By the hybrid argument, if there is an algorithm that can distinguish the two cases with constant advantage, there also exists an index $i \in [k]$ such that the algorithm can distinguish the two distributions $\mathcal{D}_{(i-1)}^Y$ and $\mathcal{D}_{(i)}^Y$ with $\Omega(1/k)$ advantage. Such an algorithm Π can be used to device an algorithm for the D-BHP problem: Alice, given a partition as input, samples $i - 1$ sparse graphs of $\Theta(n/k)$ edges and runs Π on it locally. Then, Alice sends the content of the space of Π to Bob. Bob sets the graph of the i -th phase to be its input graph, and the last $n - i$ phases to be Erdős-Rényi graphs. Bob runs Π on the given space and the last $n - i + 1$ phases, and outputs the output of Π . Finally, by the lower bound to D-BHP, the streaming algorithm also requires $\tilde{\Omega}(\sqrt{n})$ space.

In our case, however, it is impossible to show that a stream of random samples of pairs of vertices to the graph is statistically indistinguishable from a stream of k phases of independent random graphs of $1/k$ size. This is because each phase has $\omega(n)$ pairs of vertices in our case. Therefore, almost certainly many pairs of vertices will be sampled twice and cause a conflict (i.e., a pair of vertices is connected in one phase, but disconnected in another). To that end, we use our lower bound established above for the same-vector problem to show that every algorithm in the random query model with bounded time and space cannot distinguish a stream of consistent random queries of a fixed random graph from a stream of random queries that consists of k phases of independent random graphs. Given that, we still use the hybrid argument to reduce the streaming lower bound to the communication lower bound for the PD-BHP problem studied in Section 3.2. We refer readers to Figure 1 for an illustration to our reductions.

Proof to Theorem 1. Suppose for the sake of contradiction that there is a randomized algorithm in the random query model that makes t queries to the input graph, uses $\gamma\sqrt{n}$ bits of memory, and approximates the correlation clustering cost to within additive error εn^2 with probability at least $99/100$. Here we let

$$t := C \cdot \begin{cases} \Omega\left(\min\left(\frac{n}{\varepsilon^2\sqrt{\gamma}}, \frac{n\sqrt{n}}{\gamma}\right)\right) & \text{if } \gamma < 1 \\ \Omega\left(\min\left(\frac{n}{\varepsilon^2\gamma^2}, \frac{n\sqrt{n}}{\gamma}\right)\right) & \text{if } \gamma \geq 1 \end{cases}$$

where $C > 0$ is a small enough constant. By Yao's minimax principle, there also exists a deterministic algorithm Π with the same query complexity and space complexity such that Π approximates the correlation clustering cost given a random input from $\mathcal{D}_t = \frac{1}{2}\mathcal{D}_t^Y + \frac{1}{2}\mathcal{D}_t^N$ with $\geq 99/100$ probability. Let Π' be the algorithm that outputs 1 if and only if the output of Π is less than $\frac{n^2}{4} - 2.5\varepsilon n^2$. By Lemma 2, Π' is a distinguisher for input distributions \mathcal{D}_t^Y and \mathcal{D}_t^N with high advantage. Formally, we have

$$\Pr_{S \leftarrow \mathcal{D}_t^Y}[\Pi'(S) = 1] \geq 99/100 - n^{-\omega(1)} \quad (6)$$

and

$$\Pr_{S \leftarrow \mathcal{D}_t^N}[\Pi'(S) = 1] \leq 1/100 + n^{-\omega(1)} \quad (7)$$

where S denotes the input stream.

Let $k = \max\left(\frac{1}{\gamma\sqrt{\gamma}}, 1\right)$. Without loss of generality, we assume k divides t . We obtain that

$$\Pr_{S \leftarrow \mathcal{D}_{t,k}} [\Pi'(S) = 1] \geq 0.6$$

Otherwise, by (6), Π' solves the same-vector problem correctly with probability $\geq 0.695 - n^{-\omega(1)}$ on the distribution $\frac{1}{2}\mathcal{D}_t^Y + \frac{1}{2}\mathcal{D}_{t,k}$, which contradicts Lemma 8 since $\frac{N}{t} \leq \gamma\sqrt{n}/C$. Analogously, by (7) and Lemma 8, we have

$$\Pr_{S \leftarrow \mathcal{D}_{t,0}} [\Pi'(S) = 1] \leq 0.4$$

By the standard hybrid argument, there exists a number $i \in [k]$ such that Π' distinguishes between $\mathcal{D}_{t,i-1}$ and $\mathcal{D}_{t,i}$. Formally, if we let $p_i := \Pr_{S \leftarrow \mathcal{D}_{t,i}} [\Pi'(S) = 1]$ for every $i = 0, \dots, k$, we have

$$\max_{i \in [k]} |p_i - p_{i-1}| \geq \frac{1}{k} \sum_{i \in [k]} |p_i - p_{i-1}| \geq \frac{1}{k} |p_k - p_0| \geq (0.6 - 0.4)/k = \frac{1}{5k}$$

Given such an algorithm Π' and index i , we can obtain an algorithm for the PD-BHP problem with $\geq \frac{1}{10k}$ advantage in the communication model.

Formally, the algorithm works as follows: when Alice receives a partition $P \in \{0, 1\}^n$ as input, she independently samples a sequence of $i - 1$ graphs G_1, \dots, G_{i-1} drawn from \mathcal{G}_P^Y locally. Alice then runs the algorithm Π' on $\frac{i-1}{k}t$ random queries to these graphs. After that, she sends the content of the space of Π' to Bob. Bob receives the space of Π' , the input graph G and the input vector w . He samples a sequence of t/k triples $((u_1, v_1, e_1), \dots, (u_{t/k}, v_{t/k}, e_{t/k}))$ on the marginal distribution that a random graph from $\mathcal{G}_{n,t/k}$ is exactly G , where each e_i is the corresponding bit in w . After this, Bob samples a sequence of $k - i$ graphs G_{i+1}, \dots, G_k from \mathcal{G}^N and correspondingly $\frac{k-i}{k}t$ random queries. Bob then simulates Π' on these random queries with its initial content of the space set to the message of Alice. Finally, Bob outputs the output of Π' .

To see its correctness, notice that the input distribution of Bob is exactly the distribution of subgraphs induced by the i -th phase of the random queries. On the YES case, Alice and Bob together simulates Π' on the input distribution $\mathcal{D}_{t,k}$, whereas on the NO case, they simulates Π' on $\mathcal{D}_{t,k-1}$. Hence the above algorithm distinguishes YES cases from NO cases with advantage $\geq \frac{1}{10k}$. However, when $\gamma < 1$, by applying Lemma 3 and setting $\alpha = \frac{t}{kn/\varepsilon^2} \leq C \cdot \gamma$, the advantage should be $\leq (\gamma + \alpha) \cdot \sqrt{\alpha} \leq C' \cdot \gamma\sqrt{\gamma}$, which is smaller than $\frac{1}{10k} = 0.1\gamma\sqrt{\gamma}$ when C' is small enough. When $\gamma \geq 1$, by setting $\alpha = \frac{t}{kn/\varepsilon^2} \leq \frac{C}{\gamma^2}$, the advantage should be $\leq (\gamma + \alpha) \cdot \sqrt{\alpha} \leq C' < \frac{1}{10k} = 0.1$. Here C' is a constant that is decreasing when C tends to 0. Therefore, we get a contradiction. \blacktriangleleft

References

- 1 Nir Ailon, Moses Charikar, and Alantha Newman. Aggregating inconsistent information: ranking and clustering. *Journal of the ACM (JACM)*, 55(5):1–27, 2008. doi:10.1145/1411509.1411513.
- 2 Noga Alon, Tali Kaufman, Michael Krivelevich, and Dana Ron. Testing triangle-freeness in general graphs. *SIAM Journal on Discrete Mathematics*, 22(2):786–819, 2008. doi:10.1137/07067917X.
- 3 Sepehr Assadi, Vaggos Chatziafratis, Jakub Lacki, Vahab Mirrokni, and Chen Wang. Hierarchical clustering in graph streams: Single-pass algorithms and space lower bounds. In Po-Ling Loh and Maxim Raginsky, editors, *Conference on Learning Theory, 2-5 July 2022, London, UK*, volume 178 of *Proceedings of Machine Learning Research*, pages 4643–4702. PMLR, PMLR, 2022. URL: <https://proceedings.mlr.press/v178/assadi22a.html>.

- 4 Sepehr Assadi, Vihan Shah, and Chen Wang. Streaming algorithms and lower bounds for estimating correlation clustering cost. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, volume 36, pages 75201–75213, 2023. URL: http://papers.nips.cc/paper_files/paper/2023/hash/ee1a1ecc92f35702b5c29dad3dc909ea-Abstract-Conference.html.
- 5 Sepehr Assadi and Chen Wang. Sublinear time and space algorithms for correlation clustering via sparse-dense decompositions. In Mark Braverman, editor, *13th Innovations in Theoretical Computer Science Conference, ITCS 2022, Berkeley, CA, USA, January 31 - February 3, 2022*, volume 215 of *LIPICs*, pages 10:1–10:20. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022. doi:10.4230/LIPICs.ITCS.2022.10.
- 6 Nikhil Bansal, Avrim Blum, and Shuchi Chawla. Correlation clustering. *Machine Learning*, 56:89–113, 2004. doi:10.1023/B:MACH.0000033116.57574.95.
- 7 Soheil Behnezhad, Moses Charikar, Weiyun Ma, and Li-Yang Tan. Almost 3-approximate correlation clustering in constant rounds. In *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 720–731. IEEE, 2022. doi:10.1109/FOCS54457.2022.00074.
- 8 Soheil Behnezhad, Moses Charikar, Weiyun Ma, and Li-Yang Tan. Single-pass streaming algorithms for correlation clustering. In *Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 819–849. SIAM, 2023. doi:10.1137/1.9781611977554.ch33.
- 9 Soheil Behnezhad, Mahsa Derakhshan, MohammadTaghi Hajiaghayi, Cliff Stein, and Madhu Sudan. Fully dynamic maximal independent set with polylogarithmic update time. In *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 382–405. IEEE, 2019. doi:10.1109/FOCS.2019.00032.
- 10 Aditya Bhaskara, Samira Daruki, and Suresh Venkatasubramanian. Sublinear algorithms for MAXCUT and correlation clustering. In Ioannis Chatzigiannakis, Christos Kaklamanis, Dániel Marx, and Donald Sannella, editors, *45th International Colloquium on Automata, Languages, and Programming, ICALP 2018, Prague, Czech Republic, July 9-13, 2018*, volume 107 of *LIPICs*, pages 16:1–16:14. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2018. doi:10.4230/LIPICs.ICALP.2018.16.
- 11 Marco Bressan, Nicolò Cesa-Bianchi, Andrea Paudice, and Fabio Vitale. Correlation clustering with adaptive similarity queries. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d’Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, volume 32, pages 12510–12519, 2019. URL: <https://proceedings.neurips.cc/paper/2019/hash/b0ba5c44aaf65f6ca34cf116e6d82ebf-Abstract.html>.
- 12 Mélanie Cambus, Fabian Kuhn, Etna Lindy, Shreyas Pai, and Jara Uitto. A $(3+\varepsilon)$ -approximate correlation clustering algorithm in dynamic streams. In *Proceedings of the 2024 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2861–2880. SIAM, 2024. doi:10.1137/1.9781611977912.101.
- 13 Nairen Cao, Vincent Cohen-Addad, Euiwoong Lee, Shi Li, David Rasmussen Lolck, Alantha Newman, Mikkel Thorup, Lukas Vogl, Shuyi Yan, and Hanwen Zhang. Solving the correlation cluster lp in sublinear time. In *Proceedings of the 57th Annual ACM Symposium on Theory of Computing (STOC)*, pages 1154–1165, 2025. doi:10.1145/3717823.3718181.
- 14 Nairen Cao, Vincent Cohen-Addad, Euiwoong Lee, Shi Li, Alantha Newman, and Lukas Vogl. Understanding the cluster linear program for correlation clustering. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing (STOC)*, pages 1605–1616, 2024. doi:10.1145/3618260.3649749.
- 15 Nairen Cao, Shang-En Huang, and Hsin-Hao Su. Breaking 3-factor approximation for correlation clustering in polylogarithmic rounds. In *Proceedings of the 2024 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 4124–4154. SIAM, 2024. doi:10.1137/1.9781611977912.143.

- 16 Moses Charikar, Venkatesan Guruswami, and Anthony Wirth. Clustering with qualitative information. *Journal of Computer and System Sciences*, 71(3):360–383, 2005. doi:10.1016/j.jcss.2004.10.012.
- 17 Shuchi Chawla, Konstantin Makarychev, Tselil Schramm, and Grigory Yaroslavtsev. Near optimal lp rounding algorithm for correlation clustering on complete and complete k-partite graphs. In *Proceedings of the 47th annual ACM Symposium on Theory of Computing (STOC)*, pages 219–228, 2015. doi:10.1145/2746539.2746604.
- 18 Chi-Ning Chou, Alexander Golovnev, and Santhoshini Velusamy. Optimal streaming approximations for all boolean max-2csp and max-ksat. In *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 330–341. IEEE, 2020. doi:10.1109/FOCS46700.2020.00039.
- 19 Fan RK Chung, Ronald L Graham, Peter Frankl, and James B Shearer. Some intersection theorems for ordered sets and graphs. *Journal of Combinatorial Theory, Series A*, 43(1):23–37, 1986. doi:10.1016/0097-3165(86)90019-1.
- 20 Vincent Cohen-Addad, Silvio Lattanzi, Slobodan Mitrović, Ashkan Norouzi-Fard, Nikos Parotsidis, and Jakub Tarnawski. Correlation clustering in constant many parallel rounds. In *International Conference on Machine Learning (ICML)*, pages 2069–2078. PMLR, 2021. URL: <http://proceedings.mlr.press/v139/cohen-addad21b.html>.
- 21 Vincent Cohen-Addad, Euiwoong Lee, Shi Li, and Alantha Newman. Handling correlated rounding error via preclustering: A 1.73-approximation for correlation clustering. In *IEEE 64th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1082–1104. IEEE, 2023. doi:10.1109/FOCS57990.2023.00065.
- 22 Vincent Cohen-Addad, Euiwoong Lee, and Alantha Newman. Correlation clustering with sherali-adams. In *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 651–661. IEEE, 2022. doi:10.1109/FOCS54457.2022.00068.
- 23 Vincent Cohen-Addad, David Rasmussen Lolck, Marcin Pilipczuk, Mikkel Thorup, Shuyi Yan, and Hanwen Zhang. Combinatorial correlation clustering. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing (STOC)*, pages 1617–1628, 2024. doi:10.1145/3618260.3649712.
- 24 Itai Dinur. Time-space lower bounds for bounded-error computation in the random-query model. In *Proceedings of the 2024 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2900–2915. SIAM, 2024. doi:10.1137/1.9781611977912.103.
- 25 Yumou Fei, Dor Minzer, and Shuo Wang. Multi-pass streaming lower bounds for approximating max-cut. *66th IEEE Symposium on Foundations of Computer Science (FOCS) 2025*, 2025.
- 26 Dmitry Gavinsky, Julia Kempe, Iordanis Kerenidis, Ran Raz, and Ronald De Wolf. Exponential separations for one-way quantum communication complexity, with applications to cryptography. In *Proceedings of the 39th Annual ACM Symposium on Theory of Computing (STOC)*, pages 516–525, 2007. doi:10.1145/1250790.1250866.
- 27 Venkatesan Guruswami and Runzhou Tao. Streaming hardness of unique games. In Dimitris Achlioptas and László A. Végh, editors, *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2019, Massachusetts Institute of Technology, Cambridge, MA, USA, September 20-22, 2019*, volume 145 of *LIPICs*, pages 5:1–5:12. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2019. doi:10.4230/LIPICs.APPROX-RANDOM.2019.5.
- 28 Venkatesan Guruswami, Ameya Velingker, and Santhoshini Velusamy. Streaming complexity of approximating max 2csp and max acyclic subgraph. In Klaus Jansen, José D. P. Rolim, David Williamson, and Santosh S. Vempala, editors, *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2017, Berkeley, CA, USA, August 16-18, 2017*, volume 81 of *LIPICs*, pages 8:1–8:19. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2017. doi:10.4230/LIPICs.APPROX-RANDOM.2017.8.

- 29 John Kallaugher, Michael Kapralov, and Eric Price. The sketching complexity of graph and hypergraph counting. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 556–567. IEEE, 2018. doi:10.1109/FOCS.2018.00059.
- 30 Michael Kapralov, Sanjeev Khanna, and Madhu Sudan. Approximating matching size from random streams. In *Proceedings of the 25th annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 734–751. SIAM, 2014. doi:10.1137/1.9781611973402.55.
- 31 Michael Kapralov, Sanjeev Khanna, and Madhu Sudan. Streaming lower bounds for approximating MAX-CUT. In Piotr Indyk, editor, *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2015, San Diego, CA, USA, January 4-6, 2015*, pages 1263–1282. SIAM, SIAM, 2015. doi:10.1137/1.9781611973730.84.
- 32 Michael Kapralov, Sanjeev Khanna, Madhu Sudan, and Ameya Velingker. $(1 + \omega(1))$ -approximation to max-cut requires linear space. In Philip N. Klein, editor, *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2017, Barcelona, Spain, Hotel Porta Fira, January 16-19*, pages 1703–1722. SIAM, SIAM, 2017. doi:10.1137/1.9781611974782.112.
- 33 Tali Kaufman, Michael Krivelevich, and Dana Ron. Tight bounds for testing bipartiteness in general graphs. *SIAM Journal on Computing*, 33(6):1441–1483, 2004. doi:10.1137/S0097539703436424.
- 34 Dmitry Kogan and Robert Krauthgamer. Sketching cuts in graphs and hypergraphs. In *6th Conference on Innovations in Theoretical Computer Science (ITCS)*, pages 367–376, 2015. doi:10.1145/2688073.2688093.
- 35 Konstantin Makarychev and Sayak Chakrabarty. Single-pass pivot algorithm for correlation clustering. keep it simple! In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, volume 36, pages 6412–6421, 2023. URL: http://papers.nips.cc/paper_files/paper/2023/hash/149ad6e32c08b73a3ecc3d11977fcc47-Abstract-Conference.html.
- 36 Ran Raz and Wei Zhan. The random-query model and the memory-bounded coupon collector. In Thomas Vidick, editor, *11th Innovations in Theoretical Computer Science Conference, ITCS 2020, Seattle, Washington, USA, January 12-14, 2020*, volume 151 of *LIPICs*, pages 20:1–20:11. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2020. doi:10.4230/LIPICs.ITCS.2020.20.
- 37 Elad Verbin and Wei Yu. The streaming complexity of cycle counting, sorting by reversals, and other problems. In *Proceedings of the 22nd annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 11–25. SIAM, 2011. doi:10.1137/1.9781611973082.2.