The Sketching Complexity of Graph and Hypergraph Counting

John Kallaugher*
jmgk@cs.utexas.edu
UT Austin

Michael Kapralov†
michael.kapralov@epfl.ch
EPFL

Eric Price*
ecprice@cs.utexas.edu
UT Austin

October 12, 2018

Abstract

Subgraph counting is a fundamental primitive in graph processing, with applications in social network analysis (e.g., estimating the clustering coefficient of a graph), database processing and other areas. The space complexity of subgraph counting has been studied extensively in the literature, but many natural settings are still not well understood. In this paper we revisit the subgraph (and hypergraph) counting problem in the sketching model, where the algorithm's state as it processes a stream of updates to the graph is a linear function of the stream. This model has recently received a lot of attention in the literature, and has become a standard model for solving dynamic graph streaming problems.

In this paper we give a tight bound on the sketching complexity of counting the number of occurrences of a small subgraph H in a bounded degree graph G presented as a stream of edge updates. Specifically, we show that the space complexity of the problem is governed by the fractional vertex cover number of the graph H. Our subgraph counting algorithm implements a natural vertex sampling approach, with sampling probabilities governed by the vertex cover of H. Our main technical contribution lies in a new set of Fourier analytic tools that we develop to analyze multiplayer communication protocols in the simultaneous communication model, allowing us to prove a tight lower bound. We believe that our techniques are likely to find applications in other settings. Besides giving tight bounds for all graphs H, both our algorithm and lower bounds extend to the hypergraph setting, albeit with some loss in space complexity.

*This work was done in part while the authors were visiting the Simons Institute for the Theory of Computing.
†Supported in part by ERC Starting Grant 759471-Sublinear.

1 Introduction

Triangle counting is one of the most well-studied problems in streaming graph algorithms. In the standard "turnstile" version of this problem, one maintains a small-space "sketch" of a graph under a stream of insertions and deletions of edges, and at the end of the stream outputs a $(1 \pm \varepsilon)$ multiplicative approximation to the number of triangles T in the graph; unless otherwise specified, we assume ε to be a small constant.

Turnstile streaming algorithms are almost invariably constructed as *linear* sketches, where the sketch maintained is a linear function of the indicator vector of edges; this makes it easy to process insertions and deletions. Linear sketches are also useful in other settings such as distributed computation, since sketches can be merged. There is evidence that any turnstile streaming algorithm can be efficiently implemented using linear sketches [LNW14a, AHLW16], although these results do not quite apply to graph streams.

For worst-case graphs, counting triangles is impossible in sublinear space: $\Omega(m)$ space is required to distinguish between a graph with 0 triangles and one with $T = \Omega(m)$ triangles [BOV13]. However, the hard case is degenerate in that all the triangles share a common edge. If at most Δ_E triangles share any single edge, then this bound becomes $\Omega(m\Delta_E/T)$. In [PT12] an algorithm was given that counts triangles with

$$O\left(m\left(\frac{1}{\sqrt{T}} + \frac{\Delta_E}{T}\right)\right)$$

space, where the $m/\sqrt{T}$ term improves upon a $m/T^{1/3}$ term in [TKMF09, TKM11]. In [KP17] this was shown to be tight for worst-case graphs, but the hard case is again degenerate: all the triangles share a common vertex. If Δ_V bounds the maximum number of triangles to share a vertex, this bound becomes $m\sqrt{\Delta_V}/T$. The algorithm in [KP17] requires

$$\widetilde{O}\left(m\left(\frac{1}{T^{2/3}} + \frac{\sqrt{\Delta_V}}{T} + \frac{\Delta_E}{T}\right)\right)$$

space. A natural question is whether this $\frac{m}{T^{2/3}}$ is necessary.

1.1 Linear Sketching

Suppose we have a problem of the following form: we receive a vector $v \in \mathbb{Z}^n$ as a series of updates $(v_i)_{i=1}^t$, so $v = \sum_{i=1}^t v_i$, and we want to approximate some function $f(v)$. A *linear sketch* for this problem is a linear transformation $A \in \mathbb{Z}^{n \times d}$ and a post-processing function g, so that $g(Av)$ approximates $f(v)$ (with the exact definition of "approximates" depending on the problem). The space complexity of such a sketch is the space needed to store the sketch vector, which is $\Theta(d \log n)$ bits if the maximum size of entries of v is bounded by some $M = poly(n)$ (this holds even if intermediate stages of the stream exceed M, as the sketch vector may be stored mod M).

In [LNW14a], it was shown that any turnstile streaming algorithm (an algorithm that can solve a problem of the above form when the updates v_i are allowed to be negative, but that is allowed to maintain arbitrary state) can be converted into a linear sketch with only logarithmic loss in space complexity. In [AHLW16], this result was extended to *strict* turnstile streaming algorithms, that is algorithms which require that $\sum_{i=1}^s v_i \geq 0$ for each $s \leq t$.

These results come with two important caveats. Firstly, they do not necessarily give an $O(d \log n)$-space streaming algorithm, as neither the linear transformation A nor the post-processing

function g is known to be calculable in $O(d \log n)$ space. However, our sketching lower bounds will be based on communication complexity arguments that bound the size of the sketch vector, circumventing this issue.

Secondly, [LNW14a] assumes that the turnstile algorithm in question works regardless of the value of the partial sums $\sum_{i=1}^{s} v_i$, while [AHLW16] only requires that these sums be non-negative. Therefore, our lower bounds do not rule out the possibility of a turnstile algorithm that requires every partial sum to form a valid graph (i.e. edges can neither be deleted before they arrive, nor can they arrive multiple times before being deleted). However, existing turnstile algorithms do not typically require this property—in particular, any sampling-based algorithm, whether adaptive or non-adaptive, can handle it by the addition of a counter to each stored edge.

1.2 Our Results

Lower bound. We show that any linear sketching algorithm for triangle counting requires $\Omega(\frac{m}{T^{2/3}})$ space, even for constant degree graphs. Such a result is not true for the insertion-only model, where triangles can be counted in graphs with max degree d in $O(md^2/T)$ space by subsampling the edges at rate d/T and storing all subsequent edges that touch the sampled edges [JG05].

Our result generalizes to counting the number of copies of any constant-size connected subgraph H. Such problems appear, for example, in estimating the size of database joins when planning queries [AGM08]. We show that the linear sketching complexity of distinguishing between 0 and T copies of H in constant-degree graphs is at least

$$\Omega\left(\frac{m}{T^{1/\tau}}\right)$$

where τ is the fractional vertex cover number of H, the minimum value such that there exists $f : V(H) \to [0, 1]$ with $\sum_{v \in V(H)} f(v) \leq \tau$ and $f(u) + f(v) \geq 1$ for all $uv \in E(H)$.

Upper bound. We also give a matching upper bound: by subsampling the vertices with probabilities dependent on their weight in the fractional vertex cover, we give an algorithm that estimates T using $O(m/T^{1/\tau})$ words of space, as long as the graph has constant degree. Additionally, the constant-degree restriction can be lifted for many graphs: if an optimal fractional vertex cover of H can place nonzero weight on every vertex (as occurs, for example, if H is a cycle) then the algorithm works for degree up to $T^{1/(2\tau)}$ graphs.

Hypergraph counting. Both our upper and lower bounds extend to counting hypergraphs H, but the exponent on T no longer matches for all hypergraphs. The upper bound remains $O(m/T^{1/\tau})$, while the lower bound becomes $O(m/T^{1/\mu})$ for an exponent μ that equals the fractional vertex cover number τ on many hypergraphs but not all.

All of our results extend to $\varepsilon \ll 1$; the full statements of these results are given in Theorem 25 for the upper bound, Theorem 19 for the general lower bound, and Corollary 24 for the tight lower bound specific to non-hypergraphs.

1.3 Sampling and Sketching

Our upper bounds will take the form of *non-adapative* sampling algorithms. By "sampling algorithm" we mean that the only state maintained between updates is a subset of the input edges,

and by "non-adaptive" we mean that the probability of keeping an edge does not depend on which other edges have been seen so far in the stream[1].

Non-adaptive sampling algorithms may be modifed into linear sketching algorithms by the use of L_0-sampler sketches. An L_0-sampler sketch is a linear sketch which, if v is the frequency vector of the input stream, returns a non-zero co-ordinate of v, chosen uniformly at random (more generally, an L_p-sampler samples v_i with probability proportional to $|v_i|^p$). A linear sketching algorithm for this problem was first presented in [CMR05], while [MW10] defined L_p sampling and gave algorithms for all $p \in [0, 2]$.

In [JST11], it was shown that, if the set to be sampled from has size n and the sample is required to be within δ of uniform for some constant δ, the space required is exactly $\Theta(\log^2 n)$. In [KNP+17], the optimal bound in terms of n and δ was shown to be $\Theta\left(\min\left(n, \log(1/\delta)\log^2\left(\frac{n}{\log(1/\delta)}\right)\right)\right)$.

Therefore, as our sketching lower bound and sampling upper bounds match up to polylog factors, it follows that both are themselves tight up to polylog factors.

1.4 Our Techniques

The core of our lower bound proof is a new set of Fourier analytic techniques for analyzing multiplayer simultaneous communication protocols. Our approach is inspired by the Fourier analytic analysis of the Boolean Hidden Matching problem, but develops several new ideas that we think are likely to find applications beyond subgraph counting lower bounds. We now proceed to describe the Boolean Hidden Matching problem, the main ideas behind its analysis, and then describe our techniques.

The Boolean Hidden Matching problem of Gavinsky et al [GKK+07] is a two player one way communication problem where Alice holds a binary string $x \in \{0,1\}^n$ that she compresses to a message m of s bits and sends to Bob. Bob, besides the message from Alice, gets two pieces of input: a uniformly random matching of size $n/10$ on the set $\{1, 2, \ldots, n\}$, along with a vector of binary labels $w_e \in \{0,1\}, e \in M$. In the YES case of the problem the vector w satisfies $w = Mx$, and in the NO case of the problem the vector w satisfies $w = Mx \oplus 1^{|M|}$, where we abuse notation somewhat by letting M denote the edge incidence matrix of the matching M (where each column corresponds to a vertex, and each edge to a row, with ones in the two co-ordinates corresponding to the vertices the edge is incident to).

The Boolean Hidden Matching problem and the related Boolean Hidden Hypermatching problem of Verbin and Yu [VY11] have been very influential in streaming lower bounds: streaming problems that have recently been shown to admit reductions from Boolean Hidden (Hyper)Matching include approximating maximum matching size [EHL+15, AKLY15, AKL17], approximating MAX-CUT value [KKS15, KK15, KKSV17], subgraph counting [VY11, KP17], and approximating Schatten p-norms [LW16], among others. Most recent streaming lower bounds (with the exception of [KKSV17]) use reductions from Boolean Hidden (Hyper)Matching, without modifying the Fourier analytic techniques involved in the proof.

In this paper we develop several new Fourier analytic ideas that go beyond the Boolean Hidden Matching problem in several directions:

[1]Note that this does not mean that the edges are sampled *independently* of one another—for instance, if we choose a vertex at random and keep all edges incident on that vertex, the event that we keep the edge uv is independent of whether the edge vw is present in the stream, but it is not independent of the event that that we keep the edge vw.

Analyzing simultaneous multiplayer communication. In the Boolean Hidden Matching problem Alice is the only player transmitting a message, but our communication problem features simultaneous communication from multiple players to a referee. We show how to use the convolution theorem from Fourier analysis to analyze the effect of combining information sent by multiple players in the one way simultaneous communication model. While the technique of combining information from two players using the convolution theorem was recently used by [KKSV17] to analyze a three-player game, to the best of our knowledge our work is the first to analyze games with an arbitrary number of players in this manner.

Analyzing a promise version of a communication problem via Fourier analysis. While in the Boolean Hidden Matching problem Alice's string is sampled from the uniform distribution, in our problem multiple players receive correlated binary strings (conditioned on a linear constraint over the binary field). It turns out that this specific form of conditioning lends itself naturally to a Fourier analytic approach due to the linearity of the constraints imposed on the strings, and analysing such correlated settings gives us tight bounds on the subgraph counting version of our problem.

Sharing M among the players. In the Boolean Hidden Matching problem, only Bob has the linear function M, while Alice must send her message based only on x. In our communication problem each (hyper)edge in H corresponds to a player, and every player receives a linear sized set of edges, together with parities of a hidden string x over these edges. Similarly to the Boolean Hidden Matching problem, these parities are either correct (YES case) or flipped simultaneously. A crucial new component, however, is the fact that instead of M being held by the recipient alone, each player holds part of it, and the parts the players hold are correlated.

Analyzing such correlations is in fact necessary even if one only wants to prove a simple lower bound on the space complexity of 'sampling-type' algorithms for triangle counting. We show how to analyze such correlations when H is an arbitrary hypergraph through a purely combinatorial lemma. The weights lemma (Section 4) is primarily concerned with the ability of the players to co-ordinate "weight" functions. This can be used to lower bound the space complexity of sampling-based protocols—we apply it to bound the Fourier coefficients of the referee's posterior distribution on the players' inputs when the players send *arbitrary* messages.

1.5 Related Work

The past decade has seen a large amount of work on the space complexity of graph problems in the streaming model of computation (see, e.g. the recent survey by McGregor [McG17]). The semi-streaming model of computation, which allows $\tilde{O}(n)$ space to process a graph on n vertices, has been extensively studied, with space efficient algorithms known for many fundamental graph problems such as spanning trees [AGM12a], sparsifiers [AG09, KL11, AGM12b, KLM$^+$14], matchings [AG11, AG13, GKK12, Kap13, GO12, HRVZ15, Kon15, AKLY15], spanners [AGM12b, KW14]. Beyond the semi-streaming model, it has recently been shown that it is sometimes possible to approximate the *cost* of the solution to a graph problem in the streaming model even when the amount of space available does not suffice to store the vertex set of the graph (e.g. [KKS14, EHL$^+$15, CCE$^+$15, Cor17, MV18, PS18]). The problem of designing lower bounds for graph sketches has received a lot of attention recently due to the success of graph sketching as an approach to solving dynamic graph streaming problems (e.g., [LNW14b, AKLY15, AKL17]). Similarly to our approach, such

lower bounds normally make use of the simultaneous communication model.

Subgraph counting. The streaming subgraph counting problem was introduced in [BKS02] for the case where H is a triangle. This was followed by alternative algorithms in [BFL$^+$06, JG05]. The lower bounds in [BOV13] and [KP17] were achieved by reductions to one-way communication complexity problems, the indexing problem and the Boolean Hidden Matching problem, respectively. Triangle detection has also been studied as a pure communication problem, for instance [FGO17], as well as in the adjacency-list [KMPT10, BFL$^+$06, MVV16], multi-pass [BOV13, CJ14], and query models [ELRS15].

Work on counting non-triangle subgraphs includes [BFLS07], which presented an algorithm for counting copies of $K_{3,3}$, [BDGL08], which studied subgraphs of size 3 and 4, [MMPS11], which studied cycles of arbitrary size, and [KMSS12], which studied arbitrary subgraphs. The problem has also been studied in the query [JSP15, ANRD15, PSV17] and distributed [ESBD15, ESBD16] models.

Join size estimation. The size of a database join can be viewed as a "labeled" version of hypergraph counting, where each vertex of G can only match a particular vertex ("attribute") of H, and each hyperedge of G can only match a particular hyperedge ("relation") of H. (Both our upper and lower bounds apply in this labeled setting.) In [AGM08] it was shown for a database G with m hyperedges, the size of the join given by a query H can be up to $\Theta(m^\rho)$, where ρ is the fractional *edge* cover number of H.

This result is from a very different regime from ours because it involves very dense graphs and ours involves sparse ones. But one intriguing connection is through the $\Omega(m/T^{2/3})$ lower bound given in [KP17] for the restricted class of "triangle sampling" algorithms. Generalizing that proof for arbitrary H would use [AGM08] to get a lower bound of $\Omega(m(\frac{1}{Tn^{2\rho-|V|}})^{1/\rho})$, as opposed to our $\Omega(m/T^{1/\tau})$ bound. These are the same for some graphs, such as odd cycles, where $\rho = \tau = |V|/2$, and $\Omega(m/T^{1/\tau})$ is stronger for sparse graphs, but the two bounds are incomparable in general. It seems possible that the sample complexity for dense graphs will depend on ρ in some fashion.

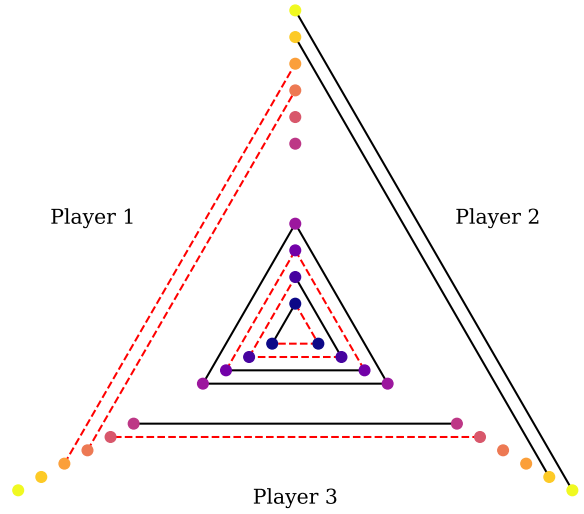
2 Proof Overview

2.1 Lower Bound

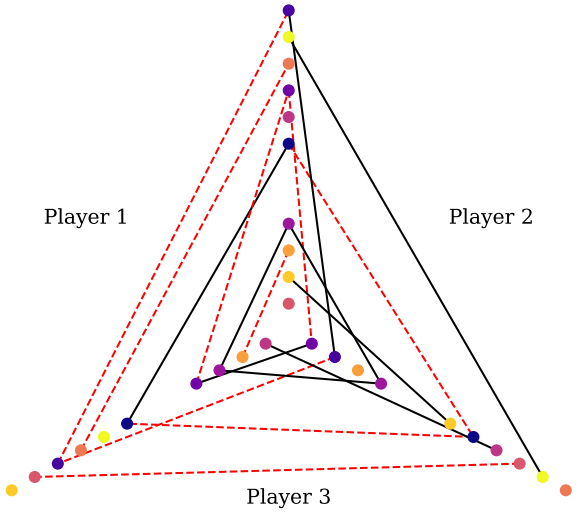
For a fixed graph H with fractional vertex cover τ, we prove an $\Omega(n/T^{1/\tau})$ lower bound for determining whether a constant-degree graph on $\Theta(n)$ vertices has 0 or $\Theta(T)$ copies of H. For illustration, in this section we focus on the case where H is a triangle.

We consider the three-party simultaneous-message communication problem illustrated in Figure 1, where each player is associated with an edge of H. First, we construct a set of $N = \Theta(n)$ vertices V_u for each vertex $u \in H$. The player associated with edge $e = (u, v)$ receives an input consisting of n disjoint edges on $V_u \times V_v$, along with binary labels associated with each edge. We are guaranteed that the three players' inputs collectively contain T triangles, with all the other edges disjoint. Each set of vertices are randomly permuted so that the players do not know which of their edges participate in triangles.

We also guarantee that the XOR of the labels associated with a triangle is the same for every



(a) Input for triangle-counting before permutation. Each player e sees n edges with associated binary labels (pictured as solid/dashed). The edges match up into T triangles (center) and $n-T$ isolated edges (outside). The goal is to determine whether every triangle has an even number of solid edges, or an odd number.



(b) In the hard distribution, we randomly permute the vertices on top, on left, and on right. Each player sees their edges, with associated labels, in a random order; they do not see the pre-permutation vertex identities (represented by color).

Figure 1: Lower bound instance for triangle counting

triangle—either every triangle has an even number of 1s, or an odd number. The goal of the players is to send messages to a referee who knows the edges but not their labels, and for the referee to figure out if every triangle has an odd number of 1 labels.

We will show that a uniformly random instance of this problem requires $\Omega(n/T^{2/3})$ communication for the referee to succeed 2/3 of the time. At the same time, it directly reduces to triangle counting: each player sketches their edges labeled 0 and sends the sketch to the referee. The referee adds the linear sketches up to get a sketch of all 0-labeled edges in the graph. This subgraph either contains zero triangles (if every triangle has an odd number of 1s) or very close to $T/4$ (otherwise), so successfully counting triangles will distinguish the two cases.

Our lower bound for the communication problem consists of two main pieces. First, we give a combinatorial lemma that bounds the players' ability to co-ordinate any assignment of "weights" to edges or subsets of edges, based on the structure of the graph. Then we use Fourier-analytic techniques to extend this to a lower bound of the communication required by any protocol for the problem.

Combinatorial lemma. One approach the players could take to solve the problem would be for each player to look at their n edges and pick a p fraction to send to the referee. If the referee receives a complete triangle, he can solve the problem. What is the expected number of triangles the referee receives, if the players coordinate optimally?

The naive solution where players pick independently at random would yield p^3T triangles. Vertex sampling—picking a $\sqrt{p}$ fraction of vertices, for example those of smallest index, and only sampling edges between picked vertices—increases this to $p^{3/2}T$. In [KP17], a simple counting

argument showed that "oblivious" strategies, which decide whether to sample an edge based only on the edge and not the rest of the player's input, cannot do better than this.

The combinatorial lemma we need for the Fourier-analytic proof is a stronger, generalized version of this sampling lemma. It considers players that receive some private randomness ψ_e and partially-shared randomness ϕ_u for each $u \in e$, and output an arbitrary deterministic function

$$g_e = g_e(\psi_e, (\phi_u)_{u \in e}) \in [-1, 1]$$

of their inputs. If the ϕ_u are fully independent and the ψ_e are $(|E| - 1)$-wise independent, and

$$\max_e \mathop{\mathbb{E}}_{\psi,\phi} \left[g_e^2\right] \leq p$$

for some p, then we show:

$$\mathop{\mathbb{E}}_{\psi,\phi} \left[\prod_e g_e\right] \lesssim p^{3/2}. \tag{1}$$

To relate this to sampling, we note that the communication problem in Figure 1 can be constructed with randomness in the form above: ϕ_u contains the permutation of the vertices V_u associated with u, and ψ_e contains player e's edge labels x_e (which are 2-wise independent) and the random order π_e in which they see their edges. Consider picking a random triangle edge $s \in [T]$ and adding to ψ_e the index of s in player e's list. (One can show that ψ_e remains pairwise independent, despite the shared dependence on s.) If we only allow $g_e \in \{0, 1\}$, then we can think of g_e as the event that player e samples their edge in the sth triangle. The condition on $\mathbb{E}\left[g_e^2\right]$ says that each player can pick at most a p fraction of their edges, on average over their inputs. The conclusion is that the expected fraction of triangles completely sampled is at most $p^{3/2}$.

This combinatorial lemma is different from the simple sampling lemma of [KP17] in several ways. First, it allows players to look at their entire inputs before deciding which edges to keep. Second, while the lemma of [KP17] was based on defining a fixed subset of edges to keep (so the number kept depended only on which edges were seen), in our lemma the players only need to keep a p fraction of their inputs *on average*, but the players do not have shared randomness. If they had shared randomness, there would be a trivial counterexample: with probability p every player samples every edge, giving the referee pT triangles in expectation. Without shared randomness, they can still use the correlation of their input for nontrivial algorithms: for example, for $p = 2^{1-n}$ they can send their entire input if every edge has the same label; because of the promise, if two players sample their inputs then the third is much more likely to. But this coordination is less effective than vertex sampling.

The combinatorial lemma is also more general, in ways that are important for the Fourier-analytic component of the proof. It allows for "fractional" choices of edges $g_e \in [-1, 1]$, with an ℓ_2 constraint. This allows for alternative competitive strategies—for example, placing $\sqrt{p}$ weight on every edge also yields $p^{3/2}$—but no strictly better ones. Additionally, the lemma will extend to cases where instead of placing weight on individual edges, the players place weight on *sets* of k triangle edges for some $k \geq 1$. These will correspond to weight k Fourier coefficients.

Fourier-analytic argument. Our approach for lower bounding the communication problem is inspired by [GKK+07]. Let $x_e \in \{0, 1\}^n$ be the player e's labels before permutation (i.e., from Figure 1a). We consider the referee's posterior distribution p on the triangle parities, $x_1^{1:T} \oplus x_2^{1:T} \oplus x_3^{1:T}$. p is supported on $\{0^T, 1^T\}$, and our goal is to show that it is nearly uniform.

First, we express the referee's posterior distribution on the labels $x = (x_e)_{e \in E}$. Let $f_e(y) = 1$ if player e's message to the referee is consistent with $x_e = y$, and 0 otherwise, and let $f : \{0,1\}^{|E|n} \to \{0,1\}$ be given by $f(x) = \prod_e f_e(x_e)$. The referee has two constraints on x: the message consistency constraint $f(x)$, and a parity constraint $q(x)$. His posterior distribution is uniform on $\mathrm{supp}(fq)$.

The first observation we make relates the referee's total variation distance to the Fourier spectrum of fq. For indicator functions $g : \{0,1\}^m \to \{0,1\}$ it makes sense to consider the renormalized Fourier transform

$$\widetilde{g}(s) := \frac{2^m}{|\mathrm{supp}(g)|}\widehat{g}(s) = \mathop{\mathbb{E}}_{x \in \mathrm{supp}(g)}\left[(-1)^{s \cdot x}\right].$$

With this normalization, we observe that

$$\Delta := \|p - \mathcal{U}(\{0^T, 1^T\})\|_{TV} = \frac{1}{2}\widetilde{fq}(e_1, e_1, e_1)$$

where $e_1 = (1, 0, \ldots, 0) \in \{0,1\}^n$. Using the structure of q's spectrum and the Fourier convolution theorem, we turn this into

$$\Delta = C \sum_{\substack{t \in \{0,1\}^T \\ |t| \equiv 1 \mod 2}} \prod_e \widetilde{f}_e(t0^{n-T})$$

where C is a normalising factor that is constant in expectation over x_1, x_2, x_3. The combinatorial lemma applied to $\widetilde{f}_e$ lets us bound the sum for a fixed $|t| = k$ (in expectation over the input). The bound is, for some constant $D > 0$,

$$\sum_{\substack{t \in \{0,1\}^T \\ |t| = k}} \prod_e \widetilde{f}_e(t0^{n-T}) \le D\binom{T}{k}\left(\max_e \frac{1}{\binom{n}{k}}\mathbb{E}\left[\sum_{\substack{s \in \{0,1\}^n \\ |s| = k}} \widetilde{f}_e(s)^2\right]\right)^{3/2}$$

which can be bounded in terms of the players' c bits of communication by the KKL lemma (for small k) and Parseval's identity (for high k). See Section 3.3 for statements of the bounds used. The dominant term when summing over k is $k = 1$, whence we get that the referee's total variation distance has

$$\mathbb{E}\left[\Delta\right] \lesssim T(c/n)^{3/2}.$$

This implies the players must send at least $n/T^{2/3}$ bits to distinguish the two cases with significant probability.

Changes for non-triangle graphs. For counting general (hyper)graphs, the combinatorial lemma as described gives a bound of $p^{MVC_{1/2}(H)}$, where $MVC_{1/2}(H)$ is a "modified" fractional vertex cover in which weight can be placed directly on edges for half price. For odd cycles such as triangles, $MVC_{1/2}(H)$ equals the non-modified fractional vertex cover τ, giving the desired $\Omega(n/T^{1/\tau})$ bound.

For other graphs, such as the length-3 path, $MVC_{1/2}(H)$ can be less than τ leading to a suboptimal result. For these graphs we use a somewhat different proof, in which the referee is identified with a particular edge e^* in the graph. The other players' inputs are then completely independent of one another, with no promise on the XOR of their labels. We follow a slightly

different Fourier-analytic approach that requires bounding

$$\mathbb{E}\left[\prod_{e\neq e^*} \widetilde{f_e}^2\right].$$

rather than $\mathbb{E}\left[\prod_e \widetilde{f_e}\right]$. We apply the combinatorial lemma to the $\widetilde{f_e}^2$, on which we have an ℓ_1 constraint, giving us the bound

$$\mathbb{E}\left[\prod_{e\neq e^*} \widetilde{f_e}^2\right] \leq p^{MVC_1(H\setminus e^*)}$$

where the exponent is the non-modified fractional vertex cover of $H \setminus e^*$. This gives a lower bound of $\Omega(n/T^{1/MVC_1(H\setminus e^*)})$. For every connected (non-hyper-)graph H that is neither an odd cycle nor a single edge, this equals $\Omega(n/T^{1/\tau})$ for at least one e^*.

For graphs that *are* single edges or odd cycles, $MVC_{1/2}(H) = \tau$, and so the combination of these bounds gives $\Omega(n/T^{1/\tau})$ for every connected graph with more than one edge. For hypergraphs, the individual lower bounds still hold, but their maximum is not necessarily $\Omega(n/T^{1/\tau})$.

Dependence on ε. The above approach is a lower bound for distinguishing T triangles from 0 triangles. Distinguishing T triangles from $(1-\varepsilon)T$ triangles should require more space for small ε. In the non-promise version of the proof used for graphs that are not odd cycles, we use the noise operator, an operator that takes a binary function and "noises" it by randomly flipping input bits, to get an $\varepsilon^{-2/\tau}$ dependence. In the promise version, the bound we get is only $\varepsilon^{-1/\tau}$.

2.2 Upper Bound

For purposes of this overview, we describe a sampling algorithm for the "labeled" version of the problem used in join size estimation, where edges and vertices in G correspond to edges and vertices in H, and we only want to count subgraphs with matching labels. Since H has constant size, we can solve the non-labeled version by trying many random labelings.

Consider a hypergraph H with minimal fractional vertex cover f, so $f(u) \in [0,1]$ for each vertex $u \in V_H$ and $\sum_u f(u) = \tau$. Let $\chi : V_G \to V_H$ be the labels. For a parameter $p \in (0,1)$ to be determined later, we sample each vertex $v \in V_G$ with probability $p^{f(\chi(v))}$, and we keep a hyperedge $e \in E_G$ if and only if we sample all $v \in e$.

The chance we keep any given copy of H is $\prod_{u\in H} p^{f(u)} = p^\tau$. Therefore, if we set $p = 100/T^{1/\tau}$, the expected number of copies of H we see will be $p^\tau T \geq 100$. On the other hand, the chance we keep any single edge e is $\prod_{v\in e} p^{f(\chi(v))} \leq p$, because f covers the edge associated with e and so $\sum_{v\in e} f(\chi(v)) \geq 1$. This gives a algorithm with $O(mp) = O(m/T^{1/\tau})$ space that sees $p^\tau T \geq 100$ copies of H in expectation; from this T can be estimated.

The only tricky bit is to show that the variance of the number of sampled copies of H is small. We bound this in terms of the maximum degree of G and the maximum correlation between sampling two copies of H in G. If this correlation is 1 as can happen in general, the sampling algorithm only works for constant-degree graphs. However, if the vertex cover places at least 0.5 weight on each vertex of H, then the correlation is at most $\sqrt{p} = \Theta(1/T^{1/(2\tau)})$. This lets the

algorithm work for degree $O(T^{1/(2\tau)})$ graphs. In the case of triangles, this $O(T^{1/3})$ degree bound is the correct regime for $O(m/T^{2/3})$ samples to be possible—above this threshold, the maximum number of triangles sharing a single vertex can be larger than $T^{2/3}$ and so the $\Omega(m\sqrt{\Delta_V}/T)$ lower bound of [KP17] precludes it.

3 Preliminaries

3.1 Roadmap

This section will cover notation and certain basic facts about Boolean Fourier analysis. Section 4 introduces a combinatorial lemma that will be needed for our lower bounds. Sections 5 and 6 will show lower bounds for two similar communication games, one where the players are given a promise on their inputs and one where they are not. Section 7 contains reductions from both of these problems to hypergraph counting, giving two non-comparable lower bounds, and a proof that these bounds combine for a tight bound in the case of non-hypergraphs. Finally, Section 8 gives a counting algorithm with a matching upper bound.

3.2 Notation

We will write e_i for the n-bit string w such that $w_j = 1$ when $j = i$ and 0 otherwise. When x is an n bit string and A is a set, $(x)_{a\in A}$ will denote the $|A|$-tuple of strings given by repeating x $|A|$ times. When there is a natural correspondence between elements of A and subsets of $[|A|n]$, we will use tuples $(x_a)_{a\in A}$ interchangeably with $|A|n$ bit strings. If x is a string, $x^{a:b}$ is the $(1+b-a)$-bit substring consisting of the a^{th} to the b^{th} bit of x.

Let H be a multi-hypergraph where empty edges are allowed, i.e. $H = (V, E)$, where E is a multi-set of subsets of V. We define a modification to the standard fractional vertex cover wherein mass can be placed directly on edges, for some price:

Definition 1. *For a weighted hypergraph $H = (V, E)$ with weights $w : E \to [0, \infty]$, we define the* λ-modified fractional vertex cover number *$MVC_\lambda(H, w)$ to be:*

$$MVC_\lambda(H, w) = \min_f \left(\sum_{v\in V} f(v) + \lambda \sum_{e\in E} f(e) \right)$$

over all $f : V \cup E \to [0, \infty]$ satisfying

$$\sum_{v\in e} f(v) + f(e) \geq w(e) \qquad \forall e \in E.$$

When w is omitted, it is assumed that $w(e) = 1$ for all e. We note that $MVC_\lambda(H)$ equals the standard fractional vertex cover number of H whenever $\lambda \geq 1$ and H has no empty hyperedges.

3.3 Basic Facts About Boolean Fourier Analysis

Definition 2. *Let $f : \{0, 1\} \to \mathbb{R}$. The* Fourier transform *$\widehat{f} : \{0, 1\} \to \mathbb{R}$ of f is given by:*

$$\widehat{f}(s) = \frac{1}{2^n} \sum_{z\in\{0,1\}^n} f(z)\chi_s(z)$$

Where $\chi_s(z) = (-1)^{s \cdot z}$.

Lemma 3. *Let* $f, g : \{0,1\}^n \to \mathbb{R}$ *be functions. Then:*

$$\widehat{fg}(s) = \sum_{t \in \{0,1\}^n} \widehat{f}(s)\widehat{g}(s \oplus t)$$

Proof. See section 2.3 of [Wol08]. □

Lemma 4. *Let* $f : \{0,1\}^{kn} \to \mathbb{R}$ *be given by:* $f((z_i)_{i=1}^k) = \prod_{i=1}^k f_i(z_i)$ *for functions* $f_i : \{0,1\}^n \to \mathbb{R}$*. Then, for any* $(s_i)_{i=1}^k \in \{0,1\}^{kn}$*:*

$$\widehat{f}((s_i)_{i=1}^k) = \prod_{i=1}^k \widehat{f}_i(s_i)$$

Proof. See Appendix A. □

One of our two main tools for bounding sums of Fourier coefficients will be Parseval's identity:

Lemma 5 (Parseval)**.** *For every function* $f : \{0,1\}^n \to \mathbb{R}$ *we have:*

$$\sum_{z \in \{0,1\}} f(z)^2 = 2^n \sum_{s \in \{0,1\}^n} \widehat{f}(s)^2$$

The other will be the KKL lemma:

Lemma 6 ([KKL88])**.** *Let* f *be a function* $f : \{0,1\}^n \to \{-1,0,1\}$*. Let* $A = \{x | f(x) \neq 0\}$*, and let* s *denote the Hamming weight of* $s \in \{0,1\}^n$*. Then for every* $\delta \in [0,1]$ *we have*

$$\sum_{s \in \{0,1\}^n} \delta^{|s|} \widehat{f}(s)^2 \leq \left(\frac{|A|}{2^n}\right)^{\frac{2}{1+\delta}}$$

We will make use of the following corollary of this lemma, similar to a corollary from [GKK$^+$07]:

Lemma 7. *For any set* $A \subseteq \{0,1\}^n$ *and* $\lambda \in (1, \infty)$*, let* $f : \{0,1\}^n \to \{0,1\}$ *be the characteristic function of* A*, and suppose that* $|A| \geq 2^{n-c}$ *for some* $c \in \mathbb{N}$*. Then, for each* $k \in [\lfloor \lambda c \rfloor]$ *one has* $\frac{2^{2n}}{|A|^2} \sum_{s \in \{0,1\}^n : |s| = k} \widehat{f}(s)^2 \leq \left(\frac{2\lambda c}{k}\right)^k$.

The proof closely follows Lemma 6 in [GKK$^+$07] and is given in Appendix A for completeness.

Lemma 8. *Let* $m : \{0,1\}^n \to \{0,1\}^l$ *be a function, and let* $l \leq c - \alpha$ *for some* $\alpha > 0$*, and let* X *be uniformly distributed over* $\{0,1\}^n$*. Define* $F = \{z \in \{0,1\}^n : m(z) = m(X)\}$*. Then, with probability at least* $1 - 2^{-\alpha}$ *over* X*,*

$$|F| \geq 2^{n-c}.$$

Proof. As m takes only 2^l different values, there are at most $\frac{1}{2^\alpha} 2^c$ distinct possible values for F, which partition $\{0,1\}^n$, and so no more than a $\frac{1}{2^\alpha}$ fraction of strings in $\{0,1\}^n$ are in sets of size $\leq 2^{n-c}$. Therefore the probability of a random string being in such a set is $\leq \frac{1}{2^\alpha}$. □

Following [Wol08], we define the noise operator $\mathcal{T}_\varepsilon$ on functions of n-bit strings as follows:

$$\mathcal{T}_\varepsilon(f)(x) = \mathop{\mathbb{E}}_y [f(y)]$$

Where y is the random variable obtained by, for each bit of x, independently flipping it with probability $1/2 - \varepsilon/2$. We will use the fact (also in [Wol08]) that, for all $s \in \{0,1\}^n$:

$$\widehat{\mathcal{T}_\varepsilon(f)}(s) = \varepsilon^{|s|} \widehat{f}(s).$$

4 The Weights Lemma

Definition 9 (Totally disconnected hypergraph)**.** *For a hypergraph $H = (V_H, E_H)$ we say that H is totally disconnected if edges of H are pairwise disjoint, i.e. for every $a, b \in E$ one has $a \cap b = \emptyset$.*

Lemma 10. *Consider any hypergraph $H = (V, E)$ and weight function $w : E \to [0, \infty]$. Suppose that H is not totally disconnected as per Definition 9, i.e., there exist $a, b \in E$ such that $a \cap b \neq \emptyset$.*

Consider any collection of random variables g_e (for $e \in E$) that can be expressed as deterministic functions of some random variables ϕ_u (for $u \in V$) that are independent, and ψ_e (for $e \in E$) that are independent of the ϕ_u and $(|E| - 1)$-wise independent themselves, i.e.,

$$g_e = g_e(\psi_e, (\phi_u)_{u \in e}).$$

Let $q \in \{1, 2\}$ and $0 < p < 1$, and suppose for all $e \in E$ that $|g_e| \leq 1$ always and that

$$\mathbb{E}\left[|g_e|^q\right] \leq p^{w(e)}.$$

Then

$$\mathbb{E}\left[\prod_{e \in E} g_e\right] \leq d^{|V|} p^{MVC_{1/q}(H,w)}$$

where $MVC_{1/q}(H, w)$ is the modified fractional vertex cover number per Definition 1, and d is the maximum of 1 and the greatest degree of a vertex $v \in V$.

Proof. We will induct on the number of vertices that lie in at least two edges of E. By assumption, this is at least 1; let u be one such vertex.

Let $H^u = (V^u, E^u)$ denote the hypergraph obtained from H by removing the vertex u from V and every edge in E. Let $\kappa : E \to E^u$ be the mapping associated with this transformation, and define $\psi_{\kappa(e)} = \psi_e$. For any given ϕ, we define $g^\phi_{\kappa(e)}$ to be g_e conditioned on $\phi_u = \phi$:

$$g^\phi_{\kappa(e)} = (g_e \mid \phi_u = \phi) = g_e(\psi_e, (\phi_v)_{v \in e}|_{\phi_u = \phi}).$$

Let γ denote $\mathbb{E}\left[\prod_{e \in E} g_e\right]$, the expectation we want to bound, and let Φ_V, Ψ_E denote the collection of ϕ_v and ψ_e, respectively. We can then rewrite our desired quantity as

$$\gamma = \mathop{\mathbb{E}}_{\phi_u}\left[\mathop{\mathbb{E}}_{\Phi_{V^u}, \Psi_{E^u}}\left[\prod_{e \in E^u} g_e^{\phi_u}\right]\right]. \tag{2}$$

For any fixed ϕ we can define the modified weight function $w_\phi : E^u \to [0, \infty]$ that results from conditioning on $\phi_u = \phi$:

$$w_\phi(e) := \log_p \mathop{\mathbb{E}}_{\Phi_{V^u}, \psi_e} \left[|g_e^\phi|^q \right].$$

We know for all $e \in E$ that

$$\mathop{\mathbb{E}}_{\phi_u} \left[p^{w_{\phi_u}(\kappa(e))} \right] = \mathop{\mathbb{E}}_{\Phi_V, \psi_e} [|g_e|^q] \leq p^{w(e)}. \tag{3}$$

Additionally, when $u \notin e$, g_e is independent of ϕ_u so $w_\phi(\kappa(e))$ is independent of ϕ, and hence $w_\phi(\kappa(e)) \geq w(e)$.

We now proceed to show for every ϕ that the inner term in (2) satisfies

$$\mathop{\mathbb{E}}_{\Phi_{V^u}} \left[\mathop{\mathbb{E}}_{\Psi_E} \left[\prod_{e \in E^u} g_e^\phi \right] \right] \leq d^{|V|-1} p^{MVC_{1/q}(H,w) - \max_{e \ni u}(w(e) - w_\phi(\kappa(e)))}. \tag{4}$$

For each ϕ, we consider two cases:

Inductive step: H^u is not totally disconnected. In this case u was not the only vertex in H that appears in at least two edges, then H^u satisfies the constraints of our lemma and has fewer vertices that appear in at least two edges. Furthermore, the random variables $g^\phi_{\kappa(e)}$ satisfy the constraints for the lemma with weight function w_ϕ. Therefore by the inductive hypothesis:

$$\mathbb{E} \left[\prod_{e \in E^u} g^\phi_{\kappa(e)} \right] \leq d^{|V|-1} p^{MVC_{1/q}(H^u, w_\phi)},$$

and it suffices to estimate $MVC_{1/q}(H^u, w_\phi)$. As previously noted, every edge e such that $w_\phi(\kappa(e)) < w(e)$ contains u, so we can cover (H, w) by taking any cover of (H^u, w_ϕ) and placing $\max_{e \ni u}(w(e) - w_\phi(\kappa(e)))$ weight on u. Hence

$$MVC_{1/q}(H, w) \leq MVC_{1/q}(H^u, w_\phi) + \max_{e \ni u}(w(e) - w_\phi(\kappa(e)))$$

which, with the previous equation, gives (4).

Base case: H^u is totally disconnected. In this case, u is the only vertex that appears in at least two edges of E. Let $E_1 = \{e \in E \mid u \in e\}$ and $E_2 = E \setminus E_1$. Let $e' = \arg\max_{e \in E_1} w_\phi(e)$ and $e'' = \arg\max_{e \in E_1 \setminus \{e'\}} w_\phi(e)$. We note that

$$MVC_{1/q}(H, w_\phi) = w_\phi(e'') + \frac{1}{q}(w_\phi(e') - w_\phi(e'')) + \frac{1}{q} \sum_{e \in E_2} w_\phi(e)$$

because $q \in \{1, 2\}$.

Let $h_e = g^\phi_{\kappa(e)} = (g_e \mid \phi_u = \phi)$, a function of ψ_e and $(\phi_v)_{v \in e}$. The h_e for all $e \in E_2 \cup \{e'\}$ are independent of each other, because these are $|E_2| + 1 \leq |E| - 1$ variables, so the ψ_e are fully independent, and no ϕ_v variable appears in more than one such h_e. The LHS of (4) which we want to bound is equal to

$$\begin{aligned} \mathop{\mathbb{E}}_{\Phi_{V^u}, \Psi_E} \left[\prod_{e \in E} h_e \right] &\leq \mathop{\mathbb{E}}_{(h_e)_{e \in E}} \left[\prod_{e \in E_2 \cup \{e', e''\}} |h_e| \right] \\ &= \mathop{\mathbb{E}}_{(h_e)_{e \in E_2}} \left[\prod_{e \in E_2} |h_e| \mathop{\mathbb{E}}_{h_{e'}, h_{e''}} [|h_{e'} h_{e''}| \mid (h_e)_{e \in E_2}] \right] \end{aligned}$$

On the other hand, the dependency structure implies

$$\mathbb{E}\left[|h_{e'}|^q \mid (h_e)_{e\in E_2}\right] = \mathbb{E}\left[|h_{e'}|^q\right] = p^{w_\phi(\kappa(e'))}$$

regardless of the values of h_e being conditioned upon. By the same logic,

$$\mathbb{E}\left[|h_{e''}|^q \mid (h_e)_{e\in E_2}\right] = p^{w_\phi(\kappa(e''))}.$$

Splitting into cases for $q \in \{1, 2\}$, we have by Hölder's inequality that

$$\mathbb{E}\left[|h_{e'}h_{e''}| \mid (h_e)_{e\in E_2}\right] \le \begin{cases} \mathbb{E}\left[h_{e'}^2\right]^{1/2}\mathbb{E}\left[h_{e''}^2\right]^{1/2} & = p^{\frac{1}{2}w_\phi(\kappa(e'))+\frac{1}{2}w_\phi(\kappa(e''))} \quad \text{for } q=2 \\ \mathbb{E}\left[|h_{e'}|\right]\cdot 1 & = p^{w_\phi(\kappa(e'))} \quad \text{for } q=1 \end{cases}$$

In either case,

$$\mathbb{E}\left[|h_{e'}h_{e''}| \mid (h_e)_{e\in E_2}\right] \le p^{(1-\frac{1}{q})w_\phi(\kappa(e''))+\frac{1}{q}w_\phi(\kappa(e'))}$$

so the quantity we want to bound is

$$\begin{aligned}
\mathbb{E}\left[\prod_{e\in E} h_e\right] &\le p^{(1-\frac{1}{q})w_\phi(\kappa(e''))+\frac{1}{q}w_\phi(\kappa(e'))}\prod_{e\in E_2}\mathbb{E}\left[|h_e|\right] \\
&\le p^{(1-\frac{1}{q})w_\phi(\kappa(e''))+\frac{1}{q}w_\phi(\kappa(e'))}\prod_{e\in E_2}\mathbb{E}\left[|h_e|^q\right]^{1/q} \\
&\le p^{(1-\frac{1}{q})w_\phi(\kappa(e''))+\frac{1}{q}w_\phi(\kappa(e'))+\frac{1}{q}\sum_{e\in E_2}w(e)} \\
&= p^{MVC_{1/q}(H,w)}\cdot p^{(1-\frac{1}{q})(w_\phi(\kappa(e''))-w(e''))+\frac{1}{q}(w_\phi(\kappa(e'))-w(e'))} \\
&\le p^{MVC_{1/q}(H,w)}\cdot p^{\min_{e\in\{e',e''\}}(w_\phi(\kappa(e))-w(e))} \\
&\le p^{MVC_{1/q}(H,w)}\cdot p^{-\max_{e\ni u}(w(e)-w_\phi(\kappa(e)))}
\end{aligned}$$

giving (4).

Combining the two cases gives (4) for all ϕ unconditionally. Plugging into (2) gives

$$\begin{aligned}
\gamma &\le d^{|V|-1}p^{MVC_{1/q}(H,w)}\mathop{\mathbb{E}}_{\phi_u}\left[p^{-\max_{e\ni u}(w(e)-w_{\phi_u}(\kappa(e)))}\right] \\
&= d^{|V|-1}p^{MVC_{1/q}(H,w)}\mathop{\mathbb{E}}_{\phi_u}\left[\max_{e\ni u} p^{w_{\phi_u}(\kappa(e))-w(e)}\right]
\end{aligned}$$

For any given e, from (3) we have

$$\mathop{\mathbb{E}}_{\phi_u}\left[p^{w_{\phi_u}(\kappa(e))-w(e)}\right] \le p^{w(e)}\times p^{-w(e)} = 1.$$

Since no more than d edges include u, this means

$$\mathop{\mathbb{E}}_{\phi_u}\left[\max_{e\ni u} p^{w_{\phi_u}(\kappa(e))-w(e)}\right] \le \mathop{\mathbb{E}}_{\phi_u}\left[\sum_{e\ni u} p^{w_{\phi_u}(\kappa(e))-w(e)}\right] \le d$$

which gives

$$\gamma \le d^{|V|}p^{MVC_{1/q}(H,w)}$$

as desired. □

5 Hypergraph Counting with a Promise

In both of the games that follow, we will assume that the players are deterministic. This is without loss of generality by Yao's minimax principle since the inputs are sampled from a fixed distribution.

5.1 Game

We will define a $|E|+1$-player game $\texttt{PromiseCounting}(H, n, T, \varepsilon)$ (with H a hypergraph, $n, T \in \mathbb{N}, \varepsilon \in \{1/T, 2/T, \ldots, 1\}$) , as follows: There is one referee, who receives messages from every other player. No other communication takes place. Each player besides the referee corresponds to an edge $e \in E$.

Let $N = T + (n-T)|E|$. For each edge $e \in E$, let $L_e \subset [N]$ be an n-element set containing $[T]$ and $n-T$ elements disjoint from every other L_e, so that if $a \neq b$, $L_a \cap L_b = [T]$, and $\bigcup_{e \in E} L_e = [N]$. For each $e \in E$, let $\rho_e : [n] \to L_e$ be a fixed bijection such that $\rho_e|_{[T]}$ is the identity.

An instance of $\texttt{PromiseCounting}(H, n, T, \varepsilon)$ is as follows:

- For each edge $e \in E$:
  - A string $x_e \in \{0,1\}^n$.
  - A permutation π_e on L_e.
- For each vertex $v \in V$:
  - A permutation π_v on $[N]$.
- A string $\tau \in \{0^{\varepsilon T}, 1^{\varepsilon T}\}$

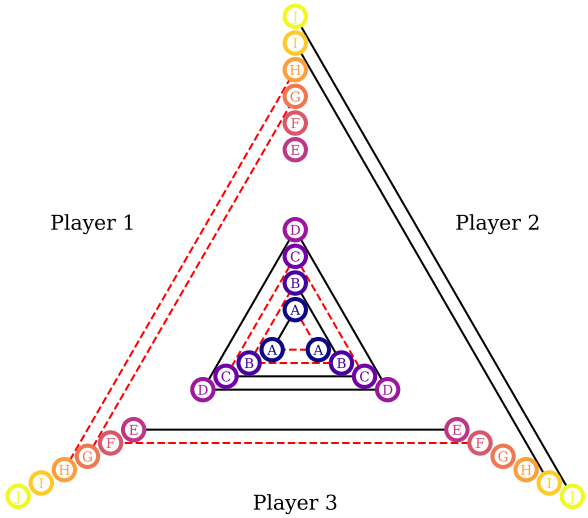
The players have the following promise:

$$\bigoplus_{e \in E} x_e^{1:\varepsilon T} = \tau$$

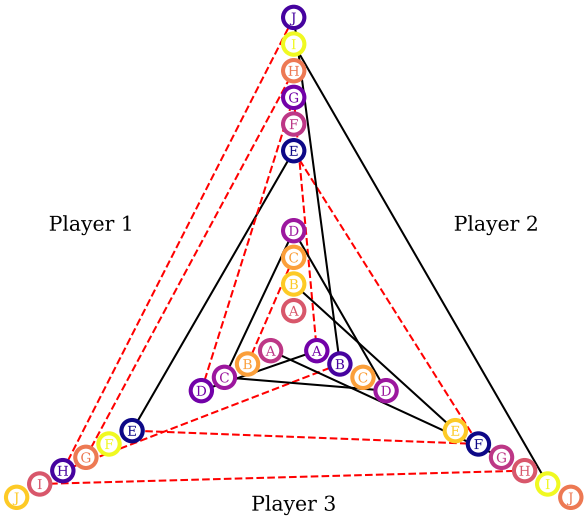
We will write X for the strings $(x_e)_{e \in E}$, Π_E for the permutations $(\pi_e)_{e \in E}$, and Π_V for the permutations $(\pi_v)_{v \in V}$. They have access to the following information:

- For each player $e \in E$:
  - $x_e \rho_e \pi_e$
  - $(\pi_v(\pi_e^{-1}(i)))_{v \in e, i \in L_e}$
- For the referee:
  - $\Pi = (\Pi_E, \Pi_V)$

Given the messages received from the players, the referee's task will be to determine whether $\tau = 0^{\varepsilon T}$ or $\tau = 1^{\varepsilon T}$.



(a) The player's instance ignoring the permutations. The x_e are the indices of red edges, read from inside out: $x_1 = [0, 1, 1, 0, 1, 1]$, $x_2 = [1, 0, 1, 0, 0, 0]$, $x_3 = [1, 1, 0, 0, 0, 1]$



(b) The hard distribution permutes each set of vertices. The players see their edges and associated labels, but not the vertex colors (which represent the pre-permutation identities).

Player 1			Player 2			Player 3		
x	u	v	x	u	v	x	u	v
0	E	E	0	B	E	0	D	C
1	H	J	0	D	D	0	G	A
1	G	H	1	E	F	1	H	I
1	B	C	0	I	I	0	A	D
1	D	G	1	G	A	1	F	E
0	C	D	0	J	B	1	B	H

(c) Each player's input consists of their edges in (b) in a random order. u represents the vertex counterclockwise of the player, and v represents the vertex clockwise.

Figure 2: Encoding of lower bound instance for triangle counting

5.2 Hard Instance

We will lower bound the complexity of this problem under the following hard input distribution: τ is chosen uniformly from $\{0^{\varepsilon T}, 1^{\varepsilon T}\}$, and then the strings $(x_e)_{e \in E}$ are chosen uniformly from:

$$\left\{ (x_e)_{e \in E} \in \{0,1\}^{|E|} : \bigoplus_{e \in E} x_e^{1:\varepsilon T} = \tau \right\}$$

Every permutation π_u, π_e is chosen uniformly at random and independently of each other and the strings.

5.3 Lower Bound

For each player e, write $m_e(x_e\rho_e\pi_e, (\pi_v(\pi_e^{-1}(i)))_{v\in e, i\in L_e})$ for the message the player sends to the referee on seeing $x_e\rho_e\pi_e$ and $(\pi_v(\pi_e^{-1}(i)))_{v\in e, i\in L_e}$.

Theorem 11. *Let H be a connected hypergraph with more than one edge. Let $c \in [n]$. Suppose that, for all inputs (X, Π) to the game, no player sends a message of more than c bits, and suppose that $\varepsilon T \leq n/10$.*

Let $p : \{0^{\varepsilon T}, 1^{\varepsilon T}\} \to [0,1]$ be the referee's posterior distribution on τ. Let ν be the distribution of $\mathcal{U}(\{0^{\varepsilon T}, 1^{\varepsilon T}\})$, the uniform distribution on the two-element set $\{0^{\varepsilon T}, 1^{\varepsilon T}\}$. Let $\mu = MVC_{1/2}(H)$, and let $0 < \delta < 1$.

There exists a constant γ, depending only on H, such that, if $c \leq \gamma \frac{n}{(\delta^2 \varepsilon T)^{1/\mu}}$:

$$\mathop{\mathbb{E}}_{X,\Pi}[||p - \nu||_{TV}] \leq \delta$$

We will prove a weaker form of the theorem in which no player sends a message of more than $c - 2\log(3|E|/\delta) - C'$ bits, for a sufficiently large constant C'. This implies the lemma statement for a slightly larger γ, since the adjustment is $O(\frac{n}{(\delta^2\varepsilon T)^{1/\mu}})$.

We will prove this by relating the distance of p from uniform to the Fourier coefficients of the indicator functions associated with the messages sent by the players.

For each $e \in E$, define the random function $f_e : \{0,1\}^n \to \{0,1\}$ by:

$$f_e(z) = \begin{cases} 1 & \text{if } m_e(z\rho_e\pi_e, (\pi_v(\pi_e^{-1}(i)))_{v\in e, i\in L_e}) = m_e(x_e\rho_e\pi_e, (\pi_v(\pi_e^{-1}(i)))_{v\in e, i\in L_e}). \\ 0 & \text{otherwise.} \end{cases}$$

And, with $Y = ((y_e)_{e\in E})$, define $f : \{0,1\}^{|E|n} \to \{0,1\}$ by:

$$f(Y) = \prod_{e\in E} f_e(y_e)$$

Let $q : \{0,1\}^{|E|n} \to \{0,1\}$ be given by:

$$q(Y) = \begin{cases} 1 & \text{if } \bigoplus_{e\in E} y_e^{1:\varepsilon T} \in \{0^{\varepsilon T}, 1^{\varepsilon T}\}. \\ 0 & \text{otherwise.} \end{cases}$$

Let $F_e = f_e^{-1}(\{1\}) \subseteq \{0,1\}^n$, and let:

$$F = \prod_{e\in E} F_e \subseteq \{0,1\}^{|E|n}.$$

Let

$$Q = q^{-1}(\{1\}) \subseteq \{0,1\}^{|E|n}$$

and

$$J = F \cap Q.$$

So then, as the game's inputs are uniformly distributed among those such that $\bigoplus_{e\in E} x_e^{1:\varepsilon T} \in \{0^{\varepsilon T}, 1^{\varepsilon T}\}$, the referee's posterior distribution on X is uniform on J. We are now ready to calculate the referee's posterior distribution on τ, writing $Z = (z_e)_{e\in E}$.

$$p(y) = \frac{\left|\left\{Z \in J : \bigoplus_{e\in E} z_e^{1:\varepsilon T} = y\right\}\right|}{|J|}$$

We can now address the total variation distance:

$$\begin{aligned}
||p-\nu||_{TV} &= \frac{1}{2}|p(0^{\varepsilon T}) - p(1^{\varepsilon T})| \\
&= \frac{1}{2|J|}\left|\left(\left|\left\{Z \in J : \bigoplus_{e\in E} z_e^{1:\varepsilon T} = 0^{\varepsilon T}\right\}\right| - \left|\left\{Z \in J : \bigoplus_{e\in E} z_e^{1:\varepsilon T} = 1^{\varepsilon T}\right\}\right|\right)\right| \\
&= \frac{1}{2|J|}\sum_{Z\in\{0,1\}^{|E|n}} f(z)q(z)(-1)^{\sum_{e\in E}(z_e)_1} \\
&= \frac{2^{|E|n-1}}{|J|}\widehat{fq}((e_1)_{e\in E})
\end{aligned}$$

We now introduce a couple of lemmas characterizing the Fourier coefficients of fq.

Lemma 12.

$$\widehat{q}(S) = \begin{cases} 2^{1-\varepsilon T} & \text{if } S = (s0^{n-\varepsilon T})_{e\in E} \text{ with } |s| \text{ even.} \\ 0 & \text{otherwise.} \end{cases}$$

Proof. First, suppose that $S = (s_e)_{e\in E}$ is not of the form $(s)_{e\in E}$ for some $s \in \{0,1\}^n$. Then there exist $a, b \in E, i \in [n]$ such that $(s_a)_i = 1$ and $(s_b)_i = 0$. Partition the strings $z \in \{0,1\}^{|E|n}$ into pairs $z, \widetilde{z}$ by defining $\widetilde{z}$ to be z with $(z_a)_i$ and $(z_b)_i$ flipped. Now, $\bigoplus_{e\in E} z_e^{1:\varepsilon T} = \bigoplus_{e\in E} \widetilde{z}_e^{1:\varepsilon T}$, so $q(z) = q(\widetilde{z})$, while $\chi_S(z) = -\chi_S(z)$, so $q(z)\chi_S(z) + q(\widetilde{z})\chi_S(\widetilde{z}) = 0$. Therefore:

$$\begin{aligned}
\widehat{q}(S) &= \frac{1}{2^{|E|n}}\sum_{z\in\{0,1\}^{|E|n}} q(z)\chi_S(z) \\
&= 0
\end{aligned}$$

Now, suppose that $S = (s)_{e\in E}$ for some $s \in \{0,1\}^n$ with $s^{\varepsilon T+1:n} \neq 0^{n-\varepsilon T}$. Then, let $\varepsilon T < i \leq n$ be such that $s_i = 1$. Choose some edge e' in E arbitrarily. Partition the strings $z \in \{0,1\}^{|E|n}$ into pairs $z, \widetilde{z}$ by defining z to be z with the i^{th} bit of $z_{e'}$ flipped. Now, $\bigoplus_{e\in E} z_e^{1:\varepsilon T} = \bigoplus_{e\in E} \widetilde{z}_e^{1:\varepsilon T}$, so $q(z) = q(\widetilde{z})$, while $\chi_S(z) = -\chi_S(z)$, so $q(z)\chi_S(z) + q(\widetilde{z})\chi_S(\widetilde{z}) = 0$. Therefore:

$$\begin{aligned}
\widehat{q}(S) &= \frac{1}{2^{|E|n}}\sum_{z\in\{0,1\}^{|E|n}} q(z)\chi_S(z) \\
&= 0
\end{aligned}$$

Now suppose $S = (s0^{n-\varepsilon T})_{e\in E}$ for some $s \in \{0,1\}^{\varepsilon T}$ such that $|s|$ is odd. Choose some edge e' in E arbitrarily. Partition the strings in $\{0,1\}^{|E|n}$ into pairs $z, \widetilde{z}$ by defining $\widetilde{z}$ to be z with the first through $\varepsilon T^{\text{th}}$ bits in $z_{e'}$ flipped. Then $\bigoplus_{e\in E} z_e^{1:\varepsilon T} = \bigoplus_{e\in E} \widetilde{z}_e^{1:\varepsilon T} + 1^{\varepsilon T}$, and so $q(\widetilde{z}) = q(z)$. However, $\chi_S(z) = -\chi_S(z)$, so $q(z)\chi_S(z) + q(\widetilde{z})\chi_S(\widetilde{z}) = 0$. Therefore:

$$\begin{aligned}
\widehat{q}(S) &= \frac{1}{2^{|E|n}}\sum_{z\in\{0,1\}^{|E|n}} q(z)\chi_S(z) \\
&= 0
\end{aligned}$$

Finally, suppose $S = (s0^{n-\varepsilon T})_{e\in E}$ for some $s \in \{0,1\}^{\varepsilon T}$ such that $|s|$ is even. Then, for any z such that $q(z) = 1$:

$$\begin{aligned} S \cdot z &= \sum_{e\in E} s0^{n-\varepsilon T} \cdot z_e \\ &\equiv s0^{n-\varepsilon T} \cdot \bigoplus_{e\in E} z_e \mod 2 \\ &= \begin{cases} s \cdot 0^{\varepsilon T} \\ s \cdot 1^{\varepsilon T} \end{cases} \\ &\equiv 0 \mod 2 \end{aligned}$$

So $\chi_S(z) = 1$ for all z such that $q(z) = 1$. Therefore:

$$\begin{aligned} \widehat{q}(z) &= \frac{|q^{-1}(\{1\})|}{2^{|E|n}} \\ &= \frac{1}{2^{|E|n}} \left| \left\{ (z_e)_{e\in E} \in \{0,1\}^{|E|n} \middle| \bigoplus_{e\in E} z_e^{1:\varepsilon T} = 0^{\varepsilon T} \vee \bigoplus_{e\in E} z_e^{1:\varepsilon T} = 1^{\varepsilon T} \right\} \right| \\ &= \frac{1}{2^{|E|n}} 2|\{0,1\}^{|E|n-\varepsilon T}| \\ &= 2^{1-\varepsilon T} \end{aligned}$$

as desired. □

Lemma 13. *For any $f : \{0,1\}^{|E|n} \to \{0,1\}$ and $z \in \{0,1\}^n$:*

$$\widehat{qf}((s)_{e\in E}) = 2^{1-\varepsilon T} \sum_{\substack{t\in\{0,1\}^{\varepsilon T} \\ |t|\equiv 1 \bmod 2}} \widehat{f}((ts^{(\varepsilon T+1):n})_{e\in E})$$

Proof.

$$\begin{aligned} \widehat{qf}((z)_{e\in E}) &= \sum_{y\in\{0,1\}^{|E|n}} \widehat{f}((z)_{e\in E} \oplus y) q(y) \\ &= 2^{1-\varepsilon T} \sum_{\substack{t\in\{0,1\}^{\varepsilon T} \\ |t|\equiv 0 \mod 2}} \widehat{f}((z)_{e\in E} \oplus (t0^{n-\varepsilon T})_{e\in E}) \\ &= 2^{1-\varepsilon T} \sum_{\substack{t\in\{0,1\}^{\varepsilon T} \\ |t|\equiv 1 \bmod 2}} \widehat{f}((ts^{(\varepsilon T+1):n})_{e\in E}). \end{aligned}$$

□

Applying Lemma 13, our total variation bound becomes:

$$||p - \nu||_{TV} = \frac{2^{|E|n-\varepsilon T}}{|J|} \left| \sum_{\substack{t\in\{0,1\}^{\varepsilon T} \\ |t|\equiv 1 \bmod 2}} \widehat{f}(t0^{n-\varepsilon T}) \right|$$

Then, by applying Lemma 4, we can write:

$$||p-\nu||_{TV} = \frac{2^{|E|n-\varepsilon T}}{|J|}\left|\sum_{\substack{t\in\{0,1\}^{\varepsilon T}\\ |t|\equiv 1 \bmod 2}}\prod_{e\in E}\widehat{f}_e(t0^{n-\varepsilon T})\right|$$
$$= \frac{\prod_{e\in E}|F_e|}{2^{\varepsilon T}|J|}\left|\sum_{\substack{t\in\{0,1\}^{\varepsilon T}\\ |t|\equiv 1 \bmod 2}}\prod_{e\in E}\frac{2^n}{|F_e|}\widehat{f}_e(t0^{n-\varepsilon T})\right|$$

We will seek to bound this sum in expectation. To do so, we will use Lemma 10 to show that the players cannot "co-ordinate" the Fourier coefficients of the functions f_e well enough to make the above sum large. In order to apply this lemma, we will need, for each e, a probabilistic bound on

$$\widehat{f}_e(s)^2.$$

We will do this for each value of $k = |s|$, by using Lemma 7 when k is close to 0 or n, and Parseval's identity for other k. To apply Lemma 7, we need to bound the size of F_e from below. By applying Lemma 8 with $\alpha = \log\frac{3|E|}{\delta}$ and then using the union bound, we have that with probability $\geq 1-\delta/3$:

$$\forall e \in E, |F_e| \geq 2^{n-c}.$$

We also need the following bound on the normalizing factor:

Lemma 14. *For any referee input Φ:*

$$\mathop{\mathbb{E}}_{X|\Pi=\Phi}\left[\frac{\prod_{e\in E}|F_e|}{2^{\varepsilon T}|J|}\right] \leq 1.$$

Proof. Condition on $\Pi = \Phi$. Write $(F_e(Y))_{e\in E}, F(Y), J(Y)$ for the values $(F_e)_{e\in E}, F, J$ take when $X = Y$. We use the fact that, for any Π, the sets $\{J(Y)\}_{Y\in Q}$ partition Q, and likewise for each e, the sets $F_e(Y)$ partition $\{0,1\}^n$. Then note that, for any Y, Z, if $J(Y) \neq J(Z)$, $\prod_{e\in E}F_e(X)$ is disjoint from $\prod_{e\in E}F_e(Y)$, as they must be either disjoint or identical, and if they are identical so are $J(Y)$ and $J(Z)$.

So then, writing $\mathcal{J}$ for the set of distinct possible values of $J(X)$, and $R(J)$ for an arbitrary representative element of J:

$$\mathop{\mathbb{E}}_{X}\left[\frac{\prod_{e\in E}|F_e(X)|}{|J(X)|}\right] = \frac{1}{|Q|}\sum_{J\in\mathcal{J}}|J|\frac{\left|\prod_{e\in E}F_e(R(J))\right|}{|J|}$$
$$= 2^{\varepsilon T-|E|n-1}\sum_{J\in\mathcal{J}}\left|\prod_{e\in E}F_e(R(J))\right|$$
$$\leq 2^{\varepsilon T-|E|n-1}|\{0,1\}^{|E|n}|$$
$$= 2^{\varepsilon T}$$

as desired. □

We define $\mathcal{E}$ to be the event that

$$\forall e, |F_e| \geq 2^{n-c} \text{ and } \frac{\prod_{e\in E}|F_e|}{2^{\varepsilon T}|J|} \leq \frac{3}{\delta}$$

which happens with probability at least $1 - 2\delta/3$.

We now define a renormalized and masked version of $\widehat{f_e}$ as follows:

$$\widetilde{f_e}(s) := \begin{cases} \frac{2^n \widehat{f_e}(s)}{|F_e|} & \text{if } |F_e| \geq 2^{n-c}. \\ 0 & \text{otherwise.} \end{cases}$$

Note that $\widetilde{f_e}$ can be expressed as a *deterministic* function of the following form:

$$\widetilde{f_e}(s) = g_e(s\rho_e\pi_e, x_e\rho_e\pi_e, (\pi_v(\pi_e^{-1}(i)))_{v\in e, i\in L_e})$$

To justify this, first recall that F_e is determined by $(x_e\rho_e\pi_e, (\pi_v(\pi_e^{-1}(i)))_{v\in e,i\in L_e})$, and then consider:

$$\begin{aligned}
\widehat{f_e}(s) &= \frac{1}{2^n}\sum_{z\in\{0,1\}^n} f_e(z)(-1)^{z\cdot s} \\
&\propto \sum_{z\in\{0,1\}^n} \mathbb{1}[m_e(z\rho_e\pi_e, (\pi_v(\pi_e^{-1}(i)))_{v\in e,i\in L_e}) = m_e(x_e\rho_e\pi_e, (\pi_v(\pi_e^{-1}(i)))_{v\in e,i\in L_e})(-1)^{z\cdot s}] \\
&= \sum_{z\in\{0,1\}^n} \mathbb{1}[m_e(z\rho_e\pi_e, (\pi_v(\pi_e^{-1}(i)))_{v\in e,i\in L_e}) = m_e(x_e\rho_e\pi_e, (\pi_v(\pi_e^{-1}(i)))_{v\in e,i\in L_e})(-1)^{z\rho_e\pi_e\cdot s\rho_e\pi_e}] \\
&= \sum_{z\in\{0,1\}^{L_e}} \mathbb{1}[m_e(z, (\pi_v(\pi_e^{-1}(i)))_{v\in e,i\in L_e}) = m_e(x_e\rho_e\pi_e, (\pi_v(\pi_e^{-1}(i)))_{v\in e,i\in L_e})](-1)^{z\cdot s\rho_e\pi_e}
\end{aligned}$$

which, as L_e is fixed, is a deterministic function of $(s\rho_e\pi_e, x_e\rho_e\pi_e, (\pi_v(\pi_e^{-1}(i)))_{v\in e,i\in L_e})$.

Now, for any X and Π that satisfy $\mathcal{E}$:

$$\begin{aligned}
||p(X,\Pi) - \nu||_{TV} &= \frac{2^{|E|n-\varepsilon T}}{|J|}\left|\sum_{\substack{t\in\{0,1\}^{\varepsilon T}\\ |t|\equiv 1 \bmod 2}}\prod_{e\in E}\widehat{f_e}(t0^{n-\varepsilon T})\right| \\
&= \frac{\prod_{e\in E}|F_e|}{2^{\varepsilon T}|J|}\left|\sum_{\substack{t\in\{0,1\}^{\varepsilon T}\\ |t|\equiv 1 \bmod 2}}\prod_{e\in E}\widetilde{f_e}(t0^{n-\varepsilon T})\right| \\
&\leq \frac{3}{\delta}\left|\sum_{\substack{t\in\{0,1\}^{\varepsilon T}\\ |t|\equiv 1 \bmod 2}}\prod_{e\in E}\widetilde{f_e}(t0^{n-\varepsilon T})\right| \qquad (5)
\end{aligned}$$

For any $k \in [n]$, note that the distribution of a single

$$\widetilde{f_e}(s) = g_e(s\rho_e\pi_e, x_e\rho_e\pi_e, (\pi_v^{-1}\pi_e)_{v\in e})$$

is identical for every $s \in \{0,1\}^n$ of Hamming weight k: π_e permutes the first argument, and x_e and π_v independently permute the other ones given π_e. Therefore for any fixed $s \in \{0,1\}^n$ of Hamming weight k, we have:

$$\beta_k := \max_e \mathop{\mathbb{E}}_{X,\Pi}\left[\widetilde{f}_e(s)^2\right] = \max_e \frac{1}{\binom{n}{k}} \mathop{\mathbb{E}}_{X,\Pi}\left[\sum_{\substack{s' \in \{0,1\}^n \\ |s'|=k}} \widetilde{f}_e(s')^2\right]$$

is independent of which such s is chosen.

Because any function f and all s have $|\widehat{f}(s)| \le \mathbb{E}_{y \sim \mathcal{U}(\{0,1\}^n)}[|f(y)|]$, we also have:

$$|\widetilde{f}_e(s)| \in [0,1].$$

Therefore Lemma 10 with $q = 2$ says for any s with $|s| = k$ that

$$\mathop{\mathbb{E}}_{X,\Pi}\left[\prod_{e \in E} \widetilde{f}_e(s)\right] \le C\beta_k^\mu$$

for some constant C depending on the hypergraph H. This lets us bound the expectation of

$$\sigma_k = \left|\sum_{\substack{t \in \{0,1\}^{\varepsilon T} \\ |t|=k}} \prod_{e \in E} \widetilde{f}_e(t0^{n-\varepsilon T})\right|$$

by

$$\mathop{\mathbb{E}}_{X,\Pi}[\sigma_k] \le C\binom{\varepsilon T}{k}\beta_k^\mu.$$

Our goal now is to bound the sum of this over all $1 \le k \le \varepsilon T$.

Low-weight terms: For $k \le c$, by Lemma 7 we have

$$\beta_k \le \frac{1}{\binom{n}{k}}\left(\frac{2c}{k}\right)^k.$$

Therefore

$$\begin{aligned}\sum_{k=1}^{c} \mathop{\mathbb{E}}_{X,\Pi}[\sigma_k] &\le C\sum_{k=1}^{c}\binom{\varepsilon T}{k}\binom{n}{k}^{-\mu}\left(\frac{2c}{k}\right)^{k\mu} \\ &\le C\sum_{k=1}^{c}\left(\frac{2^\mu e\varepsilon T c^\mu}{kn^\mu}\right)^k \\ &\le \frac{1}{20}\delta^2\end{aligned}$$

as long as $c \le \gamma n(\frac{\delta^2}{\varepsilon T})^{1/\mu}$ for a sufficiently small constant γ.

High-weight terms: By Parseval's identity,

$$\sum_{s\in\{0,1\}^n} \widehat{f_e}(s)^2 = \frac{1}{2^n} \sum_{z\in\{0,1\}^n} f_e(z)^2 = \frac{|F_e|}{2^n}$$

so

$$\sum_{s\in\{0,1\}^n} \widetilde{f_e}(s)^2 \leq 2^c.$$

and hence

$$\sum_{k=0}^{n} \binom{n}{k} \beta_k \leq |E| \cdot 2^c.$$

Therefore, since $\mu \geq 1$,

$$\begin{aligned}
\sum_{k=c+1}^{\varepsilon T} \mathop{\mathbb{E}}_{X,\Pi}[\sigma_k] &\leq C \sum_{k=c+1}^{\varepsilon T} \binom{\varepsilon T}{k} \beta_k^{\mu} \\
&\leq C|E| \cdot 2^c \max_{k:c\leq k\leq \varepsilon T} \frac{\binom{\varepsilon T}{k}}{\binom{n}{k}} \\
&\leq C|E| \left(\frac{2e\varepsilon T}{n}\right)^c.
\end{aligned}$$

Since $\varepsilon T \leq n/10$ and $c \geq 2\log(1/\delta) + C'$ for a chosen constant C', we may choose C' to be large enough that this gives

$$\sum_{k=c+1}^{\varepsilon T} \mathop{\mathbb{E}}_{X,\Pi}[\sigma_k] \leq \delta^2/20.$$

Combining the two cases, we have

$$\mathop{\mathbb{E}}_{X,\Pi}\left[\sum_{k=1}^{\varepsilon T} \sigma_k\right] \leq \delta^2/9$$

and recall from (5) and the definition of σ_k that

$$\mathop{\mathbb{E}}_{X,\Pi|\mathcal{E}}[||p(X,\Pi) - \nu||_{TV}] \leq \frac{3}{\delta} \mathop{\mathbb{E}}_{X,\Pi|\mathcal{E}}\left[\sum_{k=1}^{\varepsilon T} \sigma_k\right].$$

Since $||p(X,\Pi) - \nu||_{TV} \leq 1$ always, this gives:

$$\begin{aligned}
\mathop{\mathbb{E}}_{X,\Pi}[||p(X,\Pi) - \nu||_{TV}] &\leq \frac{3}{\delta} \mathop{\mathbb{E}}_{X,\Pi}\left[\sum_{k=1}^{\varepsilon T} \sigma_k\right] + \mathbb{P}\left[\overline{\mathcal{E}}\right] \\
&\leq \delta/3 + 2\delta/3 \\
&= \delta.
\end{aligned}$$

finishing the proof of Theorem 11.

Corollary 15. *Let H be a connected hypergraph with more than one edge. Let $c \in [n]$. Suppose that, for all inputs (X, Π) to* `PromiseCounting`(H, n, T, ε)*, no player sends a message of more than c bits.*

Let $\mu = MVC_{1/2}(H)$, and let $0 < \delta < 1$.

There exists a constant γ, depending only on H, such that, if $c \leq \gamma \frac{n}{(\delta^2 \varepsilon T)^{1/\mu}}$, the players succeed at the game with probability at most $1/2 + \delta$.

Proof. By Yao's principle [**?**], as we have a fixed distribution on inputs to our game, it is sufficient to consider deterministic protocols. Suppose we have such a protocol with maximum message size no more than c.

By Theorem 11, the referee's posterior distribution on τ is at most δ from uniform after receiving the messages associated with the protocol, and therefore whatever function of the messages is used to guess τ, it will be correct with probability at most $1/2 + \delta$. □

6 Hypergraph Counting with No Promise

6.1 Game

We will define a $|E| + 1$-player game `Counting`(H, n, T, ε) (with H a hypergraph, $n, T \in \mathbb{N}, \varepsilon \in (0, 1)$) , as follows: There is one referee, who receives messages from every other player. No other communication takes place. Each player besides the referee corresponds to an edge $e \in E$.

Let $N = T + (n - T)|E|$. For each edge $e \in E$, let $L_e \subset [N]$ be an n-element set containing $[T]$ and $n - T$ elements disjoint from every other L_e, so that if $a \neq b$, $L_a \cap L_b = [T]$, and $\bigcup_{e \in E} L_e = [N]$. For each $e \in E$, let $\rho_e : [n] \to L_e$ be a fixed bijection such that $\rho_e|_{[T]}$ is the identity.

An instance of `Counting`(H, n, T, ε) is as follows:

- For each edge $e \in E$:
  - A string $x_e \in \{0, 1\}^n$.
  - A string $\widetilde{x}_e$ generated by, for each bit of x_e, flipping that bit with probability $1/2 - \varepsilon^{1/|E|}/2$.
  - A permutation π_e on L_e.
- For each vertex $v \in V$:
  - A permutation π_v on $[N]$.
- A string $\tau \in \{0^T, 1^T\}$

We will write X for the strings $(x_e)_{e \in E}$, $\widetilde{X}$ for the strings $(\widetilde{x}_e)_{e \in E}$, χ for $(X, \widetilde{X})$, Π_E for the permutations $(\pi_e)_{e \in E}$, and Π_V for the permutations $(\pi_v)_{v \in V}$. They have access to the following information:

- For each player $e \in E$:
  - $\widetilde{x}_e \rho_e \pi_e$

- $(\pi_v(\pi_e^{-1}(i)))_{v\in e, i\in L_e}$

- For the referee:
  - $\Pi = (\Pi_E, \Pi_V)$
  - $\tau \oplus \bigoplus_{e\in E} x_e^{1:T}$

Given the messages received from the players, the referee's task will be to determine whether $\tau = 0^T$ or $\tau = 1^T$.

6.2 Hard Instance

We will lower bound the complexity of this problem under the following hard instance: τ is chosen uniformly from $\{0^{\varepsilon T}, 1^{\varepsilon T}\}$, the strings $(x_e)_{e\in E}$ are each chosen uniformly and independently from $\{0,1\}^n$, and every permutation is chosen uniformly at random and independently of each other and the strings.

6.3 Lower Bound

For each player e, write $m_e(\widetilde{x}_e, (\pi_v(\pi_e^{-1}(i)))_{v\in e, i\in L_e})$ for the message the player sends to the referee on seeing $\widetilde{x}_e$ and $(\pi_v^{-1}\pi_e)_{v\in e}$.

Theorem 16. *Let H be a connected hypergraph with more than one edge. Let $c \in [n]$. Suppose that, for all inputs (χ, Π) to the game, no player sends a message of more than c bits, and suppose $T < n/10$.*

Let $p : \{0,1\}^n \to [0,1]$ be the referee's posterior distribution on $\bigoplus_{e\in E} x_e^{1:T}$ before considering $\tau \oplus \bigoplus_{e\in E} x_e^{1:T}$. Let υ be the distribution of $\mathcal{U}(\{0,1\}^T)$, the uniform distribution on the set $\{0,1\}^T$. Let $\mu = MVC_1(H)$.

There exists a constant γ that depends on H such that, if $c \le \gamma \frac{n}{(\delta^2\varepsilon^2 T)^{1/\mu}}$:

$$\mathop{\mathbb{E}}_{\chi,\Pi}[||p - \upsilon||_{TV}] \le \delta$$

We will prove a weaker form of the theorem in which no player sends a message of more than $c - 2\log(3|E|/\delta) - C'$ bits, for a sufficiently large constant C'. This implies the lemma statement for a slightly larger γ, since the adjustment is $O(\frac{n}{(\delta^2\varepsilon T)^{1/\mu}})$.

We will prove this by examining the Fourier coefficients of p. We define the functions $(f_e)_{e\in E}, f$ and the sets $(F_e)_{e\in E}$ in a similar manner to the previous game.

For each e, define the random function $f_e : \{0,1\}^n \to \{0,1\}$ by:

$$f_e(z) = \begin{cases} 1 & \text{if } m_e(z\rho_e\pi_e, (\pi_v(\pi_e^{-1}(i)))_{v\in e, i\in L_e}) = m_e(\widetilde{x}_e\rho_e\pi_e, (\pi_v(\pi_e^{-1}(i)))_{v\in e, i\in L_e}). \\ 0 & \text{otherwise.} \end{cases}$$

And, with $Y = ((y_e)_{e\in E})$, define $f : \{0,1\}^{|E|n} \to \{0,1\}$ by:

$$f(Y) = \prod_{e\in E} f_e(y_e)$$

Let $F_e = f_e^{-1}(\{1\})$, and let:

$$F = \prod_{e \in E} F_e$$

So then, as the game's inputs are uniformly distributed, the referee's posterior distribution on $\widetilde{X}$ is uniform on F, and so the posterior probability that $\widetilde{X} = \widetilde{Y}$ is:

$$\frac{f(\widetilde{Y})}{|F|}$$

Therefore, the posterior probability that $X = Y$ is given by the sum over all $\widetilde{Y}$ of the probability that $\widetilde{X} = \widetilde{Y}$ times the probability of obtaining Y from $\widetilde{Y}$ by flipping bits with probability $1/2 - \varepsilon^{1/|E|}/2$; that is:

$$\frac{\mathcal{T}_{\varepsilon^{1/|E|}}(f)(Y)}{|F|}$$

Therefore, writing $Z = (z_e)_{e \in E}$, the referee's posterior distribution on $\bigoplus_{e \in E} x_e^{1:T}$ is given by:

$$p(y) = \frac{1}{|F|} \sum_{\substack{z \in \{0,1\}^{|E|n} \\ \bigoplus_{e \in E} z_e^{1:T} = y}} \mathcal{T}_{\varepsilon^{1/|E|}}(f)(z)$$

So the Fourier coefficients of p are given by:

$$\begin{aligned}
\widehat{p}(s) &= \frac{1}{2^T} \sum_{y \in \{0,1\}^T} p(y)(-1)^{y \cdot s} \\
&= \frac{1}{2^T |F|} \sum_{y \in \{0,1\}^T} \sum_{\substack{z \in \{0,1\}^{|E|n} \\ \bigoplus_{e \in E} z_e^{1:T} = y}} \mathcal{T}_{\varepsilon^{1/|E|}}(f)(z)(-1)^{y \cdot s} \\
&= \frac{1}{2^T |F|} \left(\sum_{\substack{z \in \{0,1\}^{|E|n} \\ \bigoplus_{e \in E} z_e^{1:T} \cdot s = 0}} \mathcal{T}_{\varepsilon^{1/|E|}}(f)(z) - \sum_{\substack{z \in \{0,1\}^{|E|n} \\ \bigoplus_{e \in E} z_e^{1:T} \cdot s = 1}} \mathcal{T}_{\varepsilon^{1/|E|}}(f)(z) \right) \\
&= \frac{1}{2^T |F|} \left(\sum_{\substack{z \in \{0,1\}^{|E|n} \\ z \cdot (s 0^{n-T})_{e \in E} = 0}} \mathcal{T}_{\varepsilon^{1/|E|}}(f)(z) - \sum_{\substack{z \in \{0,1\}^{|E|n} \\ z \cdot (s 0^{n-T})_{e \in E} = 1}} \mathcal{T}_{\varepsilon^{1/|E|}}(f)(z) \right) \\
&= \frac{2^{|E|n-T}}{|F|} \widehat{\mathcal{T}_{\varepsilon^{1/|E|}}(f)}((s 0^{n-T})_{e \in E}) \\
&= \frac{2^{|E|n-T}}{|F|} \varepsilon^{|s|} \widehat{f}((s 0^{n-T})_{e \in E})
\end{aligned}$$

And so by applying Lemma 4:

$$\widehat{p}(s) = \frac{2^{|E|n-T}}{|F|} \varepsilon^{|s|} \prod_{e \in E} \widehat{f}(s 0^{n-T}).$$

By Parseval's identity, as for any probability distribution q on $\{0,1\}^T$, $\widehat{q}(0^T) = \frac{1}{2^T}$ and $\widehat{v}(s) = 0$ for all $s \neq 0^T$:

$$\sum_{z\in\{0,1\}^T} (p(z) - v(z))^2 = \frac{2^{2|E|n-T}}{|F|^2} \sum_{s\in\{0,1\}^T\setminus\{0^T\}} \varepsilon^{2|s|} \prod_{e\in E} \widehat{f}(s0^{n-T})^2$$
$$= 2^{-T} \sum_{s\in\{0,1\}^T\setminus\{0^T\}} \varepsilon^{2|s|} \prod_{e\in E} \frac{2^{2n}}{|F_e|^2} \widehat{f}(s0^{n-T})^2$$

By applying Lemma 8 with $\alpha = \log 2|E|/\delta$ and applying the union bound:

$$\forall e \in E, |F_e| \leq 2^{n-c}$$

conditioned on an event $\mathcal{E}$ with probability at least $1 - \delta/2$.

We will now define a renormalized and masked version of $\widehat{f_e}$ as follows:

$$\widetilde{f_e}(s) := \begin{cases} \frac{2^n \widehat{f_e}(s)}{|F_e|} & \text{if } |F_e| \geq 2^{n-c} \\ 0 & \text{otherwise.} \end{cases}$$

Note that $\widetilde{f_e}$ can be expressed as a *deterministic* function of the randomness in the following form:

$$\widetilde{f_e}(s) = g_e(s\rho_e\pi_e, \widetilde{x}_e\rho_e\pi_e, (\pi_v(\pi_e^{-1}(i)))_{v\in e, i\in L_e})$$

To justify this, first recall that F_e is determined by $(\widetilde{x}_e\rho_e\pi_e, \widetilde{x}_e\rho_e\pi_e, (\pi_v(\pi_e^{-1}(i)))_{v\in e,i\in L_e}$, and then consider:

$$\begin{aligned}
\widehat{f_e}(s) &= \frac{1}{2^n} \sum_{z\in\{0,1\}^n} f_e(z)(-1)^{z\cdot s} \\
&\propto \sum_{z\in\{0,1\}^n} \mathbb{1}[m_e(z\rho_e\pi_e, (\pi_v(\pi_e^{-1}(i)))_{v\in e,i\in L_e}) = m_e(\widetilde{x}_e\rho_e\pi_e, (\pi_v(\pi_e^{-1}(i)))_{v\in e,i\in L_e})(-1)^{z\cdot s}] \\
&= \sum_{z\in\{0,1\}^n} \mathbb{1}[m_e(z\rho_e\pi_e, (\pi_v(\pi_e^{-1}(i)))_{v\in e,i\in L_e}) = m_e(\widetilde{x}_e\rho_e\pi_e, (\pi_v(\pi_e^{-1}(i)))_{v\in e,i\in L_e})(-1)^{z\rho_e\pi_e\cdot s\rho_e\pi_e}] \\
&= \sum_{z\in L_e} \mathbb{1}[m_e(z, (\pi_v(\pi_e^{-1}(i)))_{v\in e,i\in L_e}) = m_e(\widetilde{x}_e\rho_e\pi_e, (\pi_v(\pi_e^{-1}(i)))_{v\in e,i\in L_e})](-1)^{z\cdot s\rho_e\pi_e}
\end{aligned}$$

As L_e is fixed, this is a deterministic function of $(s\rho_e\pi_e, (\widetilde{x}_e\rho_e\pi_e, (\pi_v(\pi_e^{-1}(i)))_{v\in e,i\in L_e}))$.

Now, conditioned on $\mathcal{E}$:

$$\sum_{z\in\{0,1\}^T} (p(z) - v(z))^2 = 2^{-T} \sum_{s\in\{0,1\}^T\setminus\{0^T\}} \varepsilon^{2|s|} \prod_{e\in E} \widetilde{f}(s0^{n-T})^2. \tag{6}$$

For any $k \in [n]$, note that the distribution of a single

$$\widetilde{f_e}(s) = g_e(s\rho_e\pi_e, \widetilde{x}_e\rho_e\pi_e, (\pi_v^{-1}\pi_e)_{v\in e})$$

is identical for every $s \in \{0,1\}^n$ of Hamming weight k: π_e permutes the first argument, and x_e and π_v independently permute the other ones given π_e. Therefore for any fixed $s \in \{0,1\}^n$ of Hamming

weight k, we have:

$$\beta_k := \max_e \mathop{\mathbb{E}}_{X,\Pi}\left[\widetilde{f}_e(s)^2\right] = \max_e \frac{1}{\binom{n}{k}} \mathop{\mathbb{E}}_{X,\Pi}\left[\sum_{\substack{s'\in\{0,1\}^n \\ |s'|=k}} \widetilde{f}_e(s')^2\right]$$

independent of which such s is chosen.

Because any function f and all s have $|\widehat{f}(s)| \le \mathbb{E}_{y\sim\mathcal{U}(\{0,1\}^n)}[|f(y)|]$, we also have:

$$\widetilde{f}_e(s)^2 \in [0,1].$$

Therefore Lemma 10 with $q=1$ says for any s with $|s|=k$ that

$$\mathop{\mathbb{E}}_{X,\Pi}\left[\prod_{e\in E}\widetilde{f}_e(s)^2\right] \le C\beta_k^\mu$$

for some constant C depending on the hypergraph H. This lets us bound the expectation of

$$\sigma_k = \sum_{\substack{t\in\{0,1\}^T \\ |t|=k}} \varepsilon^{2k}\prod_{e\in E}\widetilde{f}_e(t0^{n-T})^2$$

by

$$\mathop{\mathbb{E}}_{X,\Pi}[\sigma_k] \le C\varepsilon^{2k}\binom{T}{k}\beta_k^\mu.$$

Our goal now is to bound the sum of this over all $1\le k\le T$.

Low-weight terms: For $k\le c$, by Lemma 7 we have

$$\beta_k \le \frac{1}{\binom{n}{k}}\left(\frac{2c}{k}\right)^k.$$

Therefore

$$\begin{aligned}\sum_{k=1}^{c}\mathop{\mathbb{E}}_{X,\Pi}[\sigma_k] &\le C\sum_{k=1}^{c}\varepsilon^{2k}\binom{T}{k}\binom{n}{k}^{-\mu}\left(\frac{2c}{k}\right)^{k\mu}\\ &\le C\sum_{k=1}^{c}\left(\frac{2^\mu e\varepsilon^2 Tc^\mu}{kn^\mu}\right)^k\\ &\le \frac{1}{20}\delta^2\end{aligned}$$

as long as $c\le\gamma n(\frac{\delta^2}{\varepsilon^2 T})^{1/\mu}$ for a sufficiently small constant γ.

High-weight terms: By Parseval's identity,

$$\sum_{s\in\{0,1\}^n}\widehat{f}_e(s)^2 = \frac{1}{2^n}\sum_{z\in\{0,1\}^n}f_e(z)^2 = \frac{|F_e|}{2^n}$$

so

$$\sum_{s\in\{0,1\}^n} \widetilde{f}_e(s)^2 \le 2^c$$

and hence

$$\sum_{k=0}^{n} \binom{n}{k} \beta_k \le |E| \cdot 2^c.$$

Therefore, since $\mu \ge 1$ and $\varepsilon^2 T \le n/10$,

$$\begin{aligned}
\sum_{k=c+1}^{T} \mathop{\mathbb{E}}_{X,\Pi}[\sigma_k] &\le C \sum_{k=c+1}^{T} \varepsilon^{2k} \binom{T}{k} \beta_k^{\mu} \\
&\le C|E| \cdot 2^c \max_{k:c\le k\le T} \frac{\varepsilon^{2k}\binom{T}{k}}{\binom{n}{k}} \\
&\le C|E| \left(\frac{2e\varepsilon^2 T}{n}\right)^c \\
&\le C|E| 2^{-c}.
\end{aligned}$$

Since $c \ge 2\log(1/\delta) + C'$ for a sufficiently large constant C', this gives

$$\sum_{k=c+1}^{T} \mathop{\mathbb{E}}_{X,\Pi}[\sigma_k] \le \delta^2/20.$$

Combining the two cases, we have

$$\mathop{\mathbb{E}}_{X,\Pi}\left[\sum_{k=1}^{T} \sigma_k\right] \le \delta^2/9.$$

Then:

$$\mathop{\mathbb{E}}_{X,\Pi|\mathcal{E}}\left[\sum_{z\in\{0,1\}^T} (p(z) - v(z))^2\right] \le 2^{-T} \mathop{\mathbb{E}}_{X,\Pi|\mathcal{E}}\left[\sum_{k=1}^{T} \sigma_k\right]$$

So:

$$\mathop{\mathbb{E}}_{X,\Pi|\mathcal{E}}\left[||p - v||_{TV}^2\right] \le \mathop{\mathbb{E}}_{X,\Pi|\mathcal{E}}\left[\sum_{k=1}^{T} \sigma_k\right]$$

And so, as $||p - v||_{TV}^2 \le 1$ always:

$$\begin{aligned}
\mathop{\mathbb{E}}_{X,\Pi}[||p - v||_{TV}] &\le \mathop{\mathbb{E}}_{X,\Pi}[||p - v||_{TV} \mathbb{1}_{\mathcal{E}}] + \mathop{\mathbb{E}}_{X,\Pi}\left[\mathbb{1}_{\overline{\mathcal{E}}}\right] \\
&\le \mathop{\mathbb{E}}_{X,\Pi}\left[\sum_{k=1}^{T} \sigma_k\right]^{1/2} + \mathbb{P}\left[\overline{\mathcal{E}}\right] \\
&\le \delta/3 + \delta/2 \\
&= \delta
\end{aligned}$$

as desired.

Corollary 17. *Let H be a connected hypergraph with more than one edge. Let $c \in [n]$. Suppose that, for all inputs (X, Π) to* `Counting`(H, n, T, ε)*, no player sends a message of more than c bits.*

Let $\mu = MVC_1(H)$, and let $0 < \delta < 1$.

There exists a constant γ that depends on H such that, if $c \leq \gamma \frac{n}{(\delta^2\varepsilon^2 T)^{1/\mu}}$, the players succeed at the game with probability at most $1/2 + \delta$.

Proof. By Yao's principle [**?**], as we have a fixed distribution on inputs to our game, it is sufficient to consider deterministic protocols. Suppose we have such a protocol with maximum message size no more than c.

By applying Theorem 16 with a smaller choice of constant γ, the referee's posterior distribution on $\bigoplus_{e \in E} x_e^{1:T}$ is at most $\delta/2$ from uniform after receiving the messages associated with the protocol but before looking at $\tau \oplus \bigoplus_{e \in E} x_e^{1:T}$. So then, after looking at $\tau \oplus \bigoplus_{e \in E} x_e^{1:T}$, the referee must determine whether it is more likely that they are looking at $\bigoplus_{e \in E} x_e^{1:T}$ or $1^T \oplus \bigoplus_{e \in E} x_e^{1:T}$. However, the distributions of $\bigoplus_{e \in E} x_e^{1:T}$ and $1^T \oplus \bigoplus_{e \in E} x_e^{1:T}$ conditioned on the messages received are both $\delta/2$-close to uniform, and so by the triangle inequality are at most δ from each other, and so the referee guesses correctly with probability at most $1/2 + \delta$. □

7 Linear Sketching Lower Bound

Definition 18. *Let $\mathcal{A}$ be a randomized graph streaming algorithm, and let $\mathbb{S}$ be the set of possible states $\mathcal{S}$ of $\mathcal{A}$. We will say $\mathcal{A}$ has* composable state *if, for any fixed random seed for $\mathcal{A}$, there is a function $c : \mathbb{S} \times \mathbb{S} \to \mathbb{S}$ such that, if $\mathcal{S}_1$ is the state of $\mathcal{A}$ after receiving the stream of edges E_1 as input, and $\mathcal{S}_2$ is the state of $\mathcal{A}$ after receiving the stream of edges E_2 as input, $c(\mathcal{S}_1, \mathcal{S}_2)$ is the state of $\mathcal{A}a$ after receiving the concatenation of E_1 and E_2 as input.*

Theorem 19. *Let $H = (V, E)$ be a (fixed) connected hypergraph with $|E| > 1$. Let $T \in \mathbb{N}, \varepsilon \in (1/\sqrt{T}, 1]$. Let $\mathcal{A}$ be a graph streaming algorithm that can distinguish between graphs G presented as a stream of edges with at least T copies of H and graphs with at most $(1-\varepsilon)T$ copies of H with probability $99/100$, provided G has no more than m edges. Let $S(m)$ be the maximum space usage of $\mathcal{A}$ across all m-edge inputs.*

Furthermore, let $\mathcal{A}$ have composable state. Then, for all $m \geq O(T)$:

$$S(m) = \Omega\left(\max\left(\frac{m}{(\varepsilon T)^{1/\mu_2}}, \frac{m}{(\varepsilon^2 T)^{1/\mu_1}}\right)\right)$$

where $\mu_2 = MVC_{\frac{1}{2}}(H)$ and $\mu_1 = \max_{e \in E} MVC_1(H \setminus e)$, with constants that may depend on H but nothing else.

We will prove this by reductions to `PromiseCounting` and `Counting`. In both reductions, we will use the following lemma on binomial distributions, from [KB80]:

Lemma 20. *Let m be any median of $Bi(n, p)$. Then:*

$$\lfloor np \rfloor \leq m \leq \lceil np \rceil$$

Our first reduction will be to `PromiseCounting`.

Lemma 21.

$$\forall m \geq O(T), S(m) = \Omega\left(\frac{m}{(\varepsilon T)^{1/\mu_2}}\right)$$

Proof. First, we note that we may assume that $T \geq 10000$ WLOG. An algorithm for $T < 10000$ can distinguish between streams with 0 copies of H and streams with at least 20000 copies of H, and applying the lemma for $T = 10000$ and $\varepsilon = 1$ will get the desired $\Omega(m)$ bound. We may assume $10/\sqrt{T} \leq \varepsilon \leq 1/10$ for similar reasons. We will also assume ε is an integer multiple of $1/T$, as this will cost us at most a factor of 2 in the bound, as if $\mathcal{A}$ can distinguish between graphs with $(1-\varepsilon)T$ and T copies of H, it can distinguish between graphs with $1 - \lfloor \varepsilon T \rfloor$ and T copies.

We will use $\mathcal{A}$ to devise a $S(m)$-bit protocol for $\texttt{PromiseCounting}(H, n, T', \varepsilon)$, where $n = \Theta(m)$, $T' = 2^{|E|}T$, and the instance is distributed as in our "hard instance" from Theorem 11. This will be based on constructing a graph $G = (B, R)$ in pieces to input to $\mathcal{A}$. The protocol is as follows (recall that $\bigcup_{e \in E} L_e = [N]$):

- Let $B = [N] \times V(H)$
- Each player e, on seeing the input:
  - $x_e \rho_e \pi_e \in \{0,1\}^{L_e}$
  - $(\pi_v(\pi_e^{-1}(i)))_{v \in e, i \in L_e}$

  constructs a set of edges R_e as follows: For each $i \in L_e$ such that $(x_e \rho_e \pi_e)_i = 0$, add the hyperedge $\{(\pi_v(\pi_e^{-1}(i))), v) : v \in e\}$.
- Each player e runs $\mathcal{A}$ with R_e as input, and then sends the state of $\mathcal{A}$ to the referee.
- The referee composes all received states, and reads the output of $\mathcal{A}$ on $G = (B, R)$, where $R = \bigcup_{e \in E} R_e$.
- If the algorithm reports that G has no more than $(1 - \varepsilon T)$ triangles, the referee decides that $\tau = 1^{\varepsilon T}$, and otherwise decides that $\tau = 0^{\varepsilon T}$.

Now we would like to know how many copies of H are in $G = (B, R)$. Consider any vertex $(i, v) \in [N] \times V(H)$. If $\pi_v^{-1}(i) \notin [T']$, then at most one edge in R includes (i, v), as there is at exactly one $e \in E$ such that L_e includes i (as the sets L_e were defined to have pairwise intersection $[T']$).

For any i such that $\pi_v^{-1} \in [T']$, and for each $e \ni v$, (i, v) will be contained in the hyperedge $\{(\pi_v(\pi_e^{-1}(j))), v) : v \in e\}$ (if it exists, that is if $(x_e \rho_e \pi_e)_j = 0$), where $j = \pi_e(\pi_v^{-1}(i))$, and no other edges. As this applies for each $v \in V(H)$, the connected component containing (i, v) will be contained in $R^{(i)} = \{\{(\pi_v(\pi_e^{-1}(j))), v) : v \in e\} : e \in E, j = \pi_e(\pi_v^{-1}(i))\}$. Therefore, the other vertices in edges that contain (i, v) will be contained within the set $\{(\pi_u(\pi_v^{-1}(i)), u) : \exists e \ni v, v \in e\}$.

By repeating this argument, this means that the connected component containing (i, v) will be contained in $\{(\pi_u(\pi_v^{-1}(i)), u) : v \in V(H)$. So this component contains exactly one copy of H if the edge $\{(\pi_v(\pi_u^{-1}(i)), v) : v \in e\}$ is present for every $e \in H$, and no copies otherwise. The first will happen iff, for every $e \in E$, the $\pi_e(\pi_v^{-1}(i))$ bit of $x_e \rho_e \pi_e$ is 0, so if the $\pi_v^{-1}(i)$ bit of $x_e \rho_e$ is 0. As $\pi_v^{-1} \in [T']$ and ρ_e is the identity on $[T']$, this happens iff the $\pi_v^{-1}(i)$ bit of x_e is 0.

So, as any copy of H must contain at least one vertex of degree at least 2, the number of copies of H in G is equal to the number of indices $i \in [T']$ such that $\forall e, (x_e)_i = 0$. Recall that in our hard instance of `PromiseCounting` the strings $(x_e)_{e \in E}$ are uniform on strings such that:

$$\bigoplus_{e \in E} x_e^{1:\varepsilon T} = \tau$$

where τ is uniformly distributed on $\{0^{\varepsilon T}, 1^{\varepsilon T}\}$.

Now, for any i such that $\bigoplus_{e \in E}(x_e)_i = 1$, there is at least one $e \in E$ such that $(x_e)_i = 1$, while if $\bigoplus_{e \in E}(x_e)_i = 0$, it is the case that $\forall e \in E, (x_e)_i = 0$ with probability $2^{1-|E|}$. For $\varepsilon T < i \leq T$, $\bigoplus_{e \in E}(x_e)_i$ is equally likely to be either, while for $i \leq \varepsilon T$, $\bigoplus_{e \in E}(x_e)_i = \tau_i$.

Therefore, the number of copies of H in G is distributed as:

$$\begin{aligned} &\mathrm{Bi}((1-\varepsilon)T', 2^{-|E|}) && \text{Conditioned on } \tau = 1^{\varepsilon T'}. \\ &\mathrm{Bi}((1-\varepsilon)T', 2^{-|E|}) + \mathrm{Bi}(\varepsilon T', 2^{1-|E|}) && \text{Conditioned on } \tau = 0^{\varepsilon T'}. \end{aligned}$$

As ε is an integer multiple of T, the unique median of $\mathrm{Bi}((1-\varepsilon)T', 2^{-|E|})$ is:

$$(1-\varepsilon)T$$

Therefore, the probability that $\mathrm{Bi}((1-\varepsilon)T', 2^{-|E|}) \leq (1-\varepsilon)T$ is at least $1/2$. Then, as $\mathrm{Var}(\mathrm{Bi}((1-\varepsilon)T', 2^{-|E|}) + \mathrm{Bi}(\varepsilon T', 2^{1-|E|})) < (1+\varepsilon)T'2^{-|E|} = (1+\varepsilon)T$, by Chebyshev:

$$\mathbb{P}\left[\mathrm{Bi}((1-\varepsilon)T', 2^{-|E|}) + \mathrm{Bi}(\varepsilon T', 2^{1-|E|}) \leq T\right] \leq \frac{(1+\varepsilon)^2}{4\varepsilon^2 T} \leq \frac{1}{100}$$

Therefore, when $\tau = 0^{\varepsilon T}$, the protocol correctly guesses it with probability at least 99/200,and when $\tau = 1^{\varepsilon T}$, the protocol correctly guesses it with probability at least $(99/100)^2 \geq 98/100$. So the success probability of this protocol is at least:

$$\frac{1}{2} \cdot \frac{99}{200} + \frac{1}{2} \cdot \frac{98}{100} = 0.7375$$

Then, by Corollary 15, this implies that:

$$S(m) = \Omega\left(\frac{m}{(\varepsilon T)^{1/\mu_2}}\right)$$

□

Lemma 22.

$$\forall m \geq O(T), \forall e^* \in E, S(m) = \Omega\left(\frac{m}{(\varepsilon^2 T)^{1/\mu_1}}\right)$$

where $\mu_1 = MVC_1(H \setminus e^*)$.

Proof. As in the previous lemma, we will assume that $T \geq 100$, $10000/\sqrt{T} \leq \varepsilon \leq 1/10$, and ε is an integer multiple of $1/T$, at the cost of at most a constant factor in our bound.

Let $\mathcal{A}$ be a composable distinguishing algorithm. We will use it to devise a $S(m)$-bit protocol for $\texttt{Counting}(\texttt{H} \setminus \texttt{e}^*, \texttt{n}, \texttt{T}', \varepsilon)$, where $T' = 2^{|E|}T$ and the state of the game is distributed according to our "hard instance" from Theorem 16. Let $E' = E \setminus e^*$.

- Let $B = [N] \times V(H)$
- Each player e, on seeing the input:
  - $\widetilde{x_e}\rho_e\pi_e \in \{0,1\}^{L_e}$
  - $(\pi_v(\pi_e^{-1}(i)))_{v \in e, i \in L_e}$

  constructs a set of edges R_e as follows: For each $i \in L_e$ such that $(\widetilde{x_e}\rho_e\pi_e)_i = 0$, add the hyperedge $\{(\pi_v(\pi_e^{-1}(i))), v) : v \in e\}$.
- Each player runs $\mathcal{A}$ with R_e as input, and then sends the state of $\mathcal{A}$ to the referee.
- The referee, on seeing the input:
  - $\Pi = (\Pi_E, \Pi_V)$
  - $\tau \oplus \bigoplus_{e \in E} x_e^{1:T}$

  sets $\widetilde{x}_{e^*} = \tau \oplus \bigoplus_{e \in E} x_e^{1:T}$, chooses ρ_{e^*}, π_{e^*} arbitrarily, and constructs R_{e^*} by adding $\{(\pi_v(\pi_{e^*}^{-1}(i))), v) : v \in e^*\}$ for each $i \in [T]$ such that $(\widetilde{x_e}\rho_e\pi_e)_i = 1$.
- The referee runs $\mathcal{A}$ with R_{e^*} as input, and then composes the state of $\mathcal{A}$ with the received states, and reads off the output of $\mathcal{A}$.
- If the algorithm reports that G has no more than $(1 - \varepsilon T)$ triangles, the referee decides that $\tau = 1^T$, and otherwise decides that $\tau = 0^T$.

As in the previous lemma, let G be $(B, R = \bigcup_{e \in E} R_e)$. By the same argument, the number of copies of H in G will be precisely the number of indices $i \in [T]$ such that $\forall e \in E, (\widetilde{x_e})_i = 0$.

To analyze this, first recall that $\widetilde{x_e}$ was generated from x_e by flipping every bit of x_e independently with probability $1/2 - \varepsilon^{1/|E|}/2$, and so for each $e \in E'$, we can write $\widetilde{x}_e = x_e \oplus y_e$, where the y_e are independent and are generated by setting each co-ordinate of y_e independently to 1 with probability $1/2 - \varepsilon^{1/|E|}/2$ and 0 otherwise. Recall also that the strings $(x_e)_{e \in E}$ are uniformly distributed, and so conditioned on $\widetilde{x}_{e^*}$, they are are distributed uniformly among strings that sum to $\widetilde{x}_{e^*}$. Therefore, if we condition on $(y_e)_{e \in E'}$ and τ the $(\widetilde{x}_e)_{e \in E'}$ are distributed uniformly among strings such that:

$$\bigoplus_{e \in E'} \widetilde{x}_e = \tau \oplus \widetilde{x}_{e^*} \oplus \bigoplus_{e \in E'} y_e$$

Using the fact that the probability of $\mathrm{Bi}(n,p)$ being even is $1/2 + (1-2p)^n/2$ (see, e.g. the proof in [**?**]), for each $i \in [T]$ we have that $\bigoplus_{e \in E'}(\widetilde{x}_e)_i = (\tau \oplus \widetilde{x}_{e^*})_i$ with probability $1/2 + \varepsilon/2$. Therefore, when $(\widetilde{x}_{e^*})_i = 0$, the probability that $\forall e \in E, (\widetilde{x}_e)_i = 0$ is:

$$\begin{aligned} &(1+\varepsilon)2^{1-|E|} && \text{If } \tau_i = 0 \\ &(1-\varepsilon)2^{1-|E|} && \text{If } \tau_i = 1 \end{aligned}$$

While when $(\widetilde{x}_{e^*})_i = 1$, the probability is 0 by definition. So, as $\widetilde{x}_{e^*}$ is uniformly distributed when only conditioned on τ, we can write down the distribution on the number of copies of H in G:

$$\begin{aligned} &\mathrm{Bi}(T', (1+\varepsilon)2^{-|E|}) && \text{If } \tau = 0^{T'} \\ &\mathrm{Bi}(T', (1-\varepsilon)2^{-|E|}) && \text{If } \tau = 1^{T'} \end{aligned}$$

So by considering the median, the probability that $\text{Bi}(T', (1-\varepsilon)2^{-|E|}) \leq (1-\varepsilon)T$ is at least $1/2$, while by Chebyshev the probability that $\text{Bi}(T', (1+\varepsilon)2^{-|E|}) \leq (1-\varepsilon)T$ is at most $1/100$, and so as in the previous lemma, the protocol succeeds with probability at least 0.7375.

Therefore, by Corollary 17:

$$S(m) = \Omega\left(\frac{m}{(\varepsilon^2 T)^{1/\mu_1}}\right)$$

□

Theorem 19 then follows directly from the previous two lemmas.

To prove this gives tight bounds (for ε constant) for all 2-uniform hypergraphs (that is, all graphs), we will need the following lemma on graph covers:

Lemma 23. *Let $G = (V, E)$ be a connected graph with $|E| > 1$. Then:*

$$\max(MVC_2(G), \max_{e \in E} MVC_1(G \setminus e)) = MVC_1(G)$$

is the standard fractional vertex cover of G.

Proof. Consider the dual of the fractional vertex cover problem, the fractional maximum matching problem, where the aim is to find a function $f : E \to [0, 1]$ such that

$$\forall u \in V, \sum_{v \in N(u)} f(uv) \leq 1$$

and $\sum_{e \in E} f(e)$ is maximized. This is known (see, e.g., [**?**]) to have a half-integral optimal solution, and therefore a solution:

$$f(e) = \begin{cases} 1 & \text{if } e \in D \\ 1/2 & \text{if } e \in C \end{cases}$$

where D is a (possibly empty) set of disjoint edges, and C is either empty or an odd cycle disjoint from D. If $G \neq C$ then, as G is connected, there is at least one edge e such that $f(e) = 0$. Therefore, that edge can be deleted from G without changing its maximum matching number and therefore without changing $MVC_1(G)$, and so $MVC_1(G) = \max_{e \in E} MVC_1(G \setminus e)$.

Otherwise, suppose G is an odd cycle. Let $g : V \cup E \to [0, \infty)$ be any function such that:

$$\sum_{v \in e} (g(v) + g(e)) \geq 1, \forall e \in E$$

Then:

$$\begin{aligned} \sum_{v \in V} g(v) + \frac{1}{2} \sum_{e \in E} g(e) &\geq \sum_{v \in V} g(v) + \frac{1}{2} \sum_{uv \in E} (1 - g(u) - g(v)) \\ &= |E|/2 \end{aligned}$$

So $MVC_{\frac{1}{2}}(G) = |E|/2$, which is also $MVC_1(G)$ (by considering a cover that puts weight $1/2$ on each vertex). □

Corollary 24. *Let $H = (V, E)$ be a connected graph with $|E| > 1$. Let $\varepsilon \in (0, 1], T \in \mathbb{N}$. Let $\mathcal{A}$ be a graph streaming algorithm that can distinguish between graphs G presented as a stream of edges with T copies of H, graphs with $(1 - \varepsilon)T$ copies of H with probability $99/100$, provided G has no more than m edges. Let $S(m)$ be the maximum space usage of $\mathcal{A}$ across all m-edge inputs.*

Furthermore, let $\mathcal{A}$ have composable state. Then:

$$\forall m \geq O(T), S(m) = \Omega\left(\frac{m}{(\varepsilon T)^{1/\tau}}\right)$$

where τ is the fractional vertex cover of H, and the constant factor may depend on H but nothing else.

8 Upper Bound

Our main result is

Theorem 25. *For every hypergraph $H = (V_H, E_H)$, $\varepsilon \in (0, 1)$ there exists a sketching algorithm that, for any hypergraph $G = (V_G, E_G)$ on n vertices with degrees bounded by d, approximates the number of copies of H in G to within a $1 + \varepsilon$ multiplicative factor with probability at least $99/100$ using space $s \leq C \cdot \varepsilon^{-2/\tau} \cdot mT^{-1/\tau}$, where T is the number of copies of H in G and τ is the fractional vertex cover of H and C is a constant that depends on H.*

For graphs we get a more powerful result, which allows the graph G to have higher degrees:

Theorem 26. *For every graph $H = (V_H, E_H)$ that admits a minimum vertex cover that assigns nonzero weight to every vertex, for every $\varepsilon \in (0, 1)$ there exists a sketching algorithm that, for any graph $G = (V_G, E_G)$ on n vertices with degrees bounded by $d \leq C' \varepsilon^{1/\tau} T^{1/(2\tau)}$, approximates the number of copies of H in G to within a $1 + \varepsilon$ multiplicative factor with probability $99/100$ using space $C \cdot \varepsilon^{-2/\tau} \cdot mT^{-1/\tau}$, where T is the number of copies of H in G and τ is the fractional vertex cover of H, and $C, C' > 0$ are constants that depend only on H.*

This result requires the minimum vertex cover to assign nonzero weight to every vertex; this happens for cycles but not stars.

Consider the fractional vertex cover of H

$$\begin{aligned} &\min \sum_{a \in V_H} x_a \\ &\text{s.t.} \sum_{a \in e} x_a \geq 1 \text{ for all } e \in E_H, \end{aligned} \tag{7}$$

let $x^* \in \mathbb{R}^{V_H}$ denote an optimal solution and let τ denote its value.

Fix a mapping $\chi : V_G \to V_H$ (see Algorithm 1, line 3). For a subset $S \subseteq V_G$ we write $\chi(S) \sim H$ if the subgraph induced by S equipped with labels $\chi(S)$ contains a copy of H, i.e. for every $a \subseteq S$ one has that if $\chi(a) \in E_H$, then $a \in E_G$. Note that, if $A(H)$ is the number of automorphisms of H, the probability that a randomly chosen χ will give $\chi(S) \sim H$ is $A(H)/k^k$.

Lemma 27. *For every $G = (V_G, E_G)$, every $H = (V_H, E_H)$ with $|V_H| = k$, if E' is the set of edges sampled by Algorithm 1 (line 8), then $\mathbb{E}[|E'|] \leq p|E_G|$.*

Algorithm 1 Subgraph counting by vertex sampling

procedure SAMPLE(H, p) ▷ Input: hypergraph H, sampling probability p
  Compute minimum vertex cover x^* in H
  $\chi \sim UNIF([k]^{V_G})$ ▷ Random mapping of V_G to $V_H = [k]$
  for $u \in V_G$ **do**
    $a \leftarrow \chi(u)$
    $X_u \leftarrow$ independent Bernoulli r.v. with mean $p^{x^*_a}$
  end for
  $E' \leftarrow \{e \in E_G : \chi(e) = |e| \text{ and } \prod_{u \in e} X_u = 1\}$ ▷ Keep colorful edges only
  $Z \leftarrow k^k \cdot p^{-\tau} \cdot \sum_{S \subseteq V_G : \chi(S) \sim H} \prod_{u \in S} X_u$ ▷ Knowing E' and χ suffices to compute Z
  return $Z/A(H)$
end procedure

Proof. For every choice of $\chi : V_G \to [k]$, only edges $e \in E_G$ with $\chi(e) = |e|$ are kept, and each such edge is kept with probability

$$\mathbb{E}\left[\prod_{u \in e} X_u\right] = \prod_{u \in e} \mathbb{E}\left[X_u\right] = \prod_{u \in e} p^{x^*_{\chi(u)}} = p^{\sum_{a \in e} x^*_a} \leq p,$$

where we used the fact that for every $e \in E_H$ one has $\sum_{a \in e} x^*_a \geq 1$ since x^* is a feasible vertex cover. Thus, the number of edges that the algorithm keeps is at most $p|E_G|$ in expectation. □

Lemma 28. *For every $G = (V_G, E_G)$, every $H = (V_H, E_H)$ with $|V_H| = k$ the estimator Z computed by Algorithm 1 satisfies $\mathbb{E}[Z] = A(H)T$.*

Proof. We have $Z = k^k p^{-\tau} \sum_{S \subseteq V_G} \mathbf{I}[\chi(S) \sim H] \cdot \prod_{u \in S} X_u$, so

$$\begin{aligned}
\mathbb{E}[Z] &= k^k p^{-\tau} \sum_{S \subseteq V_G} \mathbb{P}[\chi(S) \sim H] \cdot p^{\tau} \\
&= k^k \sum_{S \subseteq V_G} \mathbb{P}[\chi^{-1} \text{is an isomorphism form } H \text{ to } S] \\
&= A(H)T,
\end{aligned}$$

as required. □

The following simple claim will be useful for upper bounding the variance:

Claim 29. *For every hypergraph H, every hypergraph G with vertex degrees bounded by d the following conditions hold. For every $S \subseteq V_G$ the number of sets $U \subseteq V_G$ such that $\chi(S) \sim H$, $\chi(U) \sim H$ for some $\chi : V_G \to [k]$ and $|S \cap U| = r, r > 0$ is upper bounded by $d^r f(H)$ for some function f of the hypergraph H.*

Proof. We bound the number of $U \subseteq V_G$ such that $\chi(U) \sim H$ for some $\chi : V_G \to [k]$ and $S \cup U \neq \emptyset$. Fix one such U, and define an auxiliary graph $J = (V_J, E_J)$ on vertex set $(U \setminus S) \cup \{s\}$, where s is a supernode corresponding to S, by connecting two vertices $a, b \in J$ by an edge if there exists $\chi|_U : U \to V_H$ and an edge e in $\chi(U) \cap E_H$ that includes both, and give the edge $(a, b) \in E_J$ label e. Here we say that an edge includes supernode s if it has nonempty intersection with S. Since

U is connected, there exists a spanning tree F in the graph J whose edges are labeled by edges of H. We call such a spanning tree F a *template*, and we say that the pair (S, U) is consistent with template F. Given S, the number of possible U's consistent with template F is upper bounded by $(dk^2)^{|F|}$. Indeed, starting with S, one can traverse the edges of the forest F to discover all vertices in $U \setminus S$, with at most d edges incident on every vertex in G by assumption of the lemma, at most k choices of the next vertex within a given edge, and at most k choices of a vertex in S to start from when starting to traverse a subtree subtended at s in F. The number of templates F is a function of the graph H only, and since $|F| \leq r$ (recall that $r = |U \setminus S|$ by definition), we get the bound of $d^r f(H)$ for some function H. □

We will need the following, for a proof see e.g. [**?**]

Claim 30. *For every graph $H = (V_H, E_H)$ the optimal vertex cover x^* can be assumed to be half-integral, i.e. $x^*_a \in \{0, 1/2, 1\}$ for all $a \in V_H$.*

Lemma 31. *If every vertex $v \in V_G$ belongs to at most d hyperedges and H is connected, then one has $\mathbf{Var}[Z] \leq d^k f(H) k^k p^{-\tau} \mathbb{E}[Z]$.*

Furthermore, if $H = (V_H, E_H)$ is a connected graph (i.e. every hyperedge has size 2) that has an optimal vertex cover with full support (i.e. one that does not assign zero weight to any vertex), then for every graph $G = (V_G, E_G)$ with degrees bounded by $d \leq \frac{1}{2} p^{1/2}$ one has $\mathbf{Var}[Z] \leq 2 f(H)(dp^{1/2}) p^{-\tau} \mathbb{E}[Z]$.

Proof. Let k denote the number of vertices in H.

We have

$$\begin{aligned} Z^2 &= \left(k^k p^{-\tau} \sum_{S \subseteq V_G} \mathbf{I}[\chi(S) \sim H] \cdot \prod_{u \in S} X_u \right)^2 \\ &= k^{2k} p^{-2\tau} \sum_{S,U \subseteq V_G} \mathbf{I}[\chi(S) \sim H \text{ and } \chi(U) \sim H] \cdot \prod_{u \in S \cup U} X_u. \end{aligned}$$

Taking expectations over χ and X, we get

$$\begin{aligned} \mathbb{E}[Z^2] &= k^{2k} p^{-2\tau} \sum_{S,U \subseteq V_G} \mathbb{P}[\chi(S) \sim H \text{ and } \chi(U) \sim H] \cdot \mathbb{P}[X_u = 1 \text{ for all } u \in S \cup U] \\ &= k^{2k} p^{-2\tau} \sum_{\substack{S,U \subseteq V_G \\ S \cup U = \emptyset}} \mathbb{P}[\chi(S) \sim H \text{ and } \chi(U) \sim H] \cdot \mathbb{P}[X_u = 1 \text{ for all } u \in S \cup U] \\ &+ k^{2k} p^{-2\tau} \sum_{\substack{S,U \subseteq V_G \\ S \cup U \neq \emptyset}} \mathbb{P}[\chi(S) \sim H \text{ and } \chi(U) \sim H] \cdot \mathbb{P}[X_u = 1 \text{ for all } u \in S \cup U] \\ &= \mathbb{E}[Z]^2 + Q, \end{aligned} \tag{8}$$

where

$$\begin{aligned}
Q &= k^{2k}p^{-2\tau} \sum_{\substack{S,U\subseteq V_G \\ S\cup U\neq\emptyset}} \mathbb{P}[\chi(S)\sim H \text{ and } \chi(U)\sim H]\cdot\mathbb{P}[X_u=1 \text{ for all } u\in S\cup U] \\
&\leq k^{2k}p^{-2\tau} \sum_{\substack{S,U\subseteq V_G \\ S\cup U\neq\emptyset}} \mathbb{P}[\chi(S)\sim H]\cdot\mathbb{P}[X_u=1 \text{ for all } u\in S\cup U] \\
&\leq k^{2k}p^{-2\tau} \sum_{S\subseteq V_G} |\{U\subseteq V_G : U\cap S\neq\emptyset \text{ and } \exists\chi \text{ s.t.}\chi(U)\sim H\}|\cdot \\
&\qquad\qquad \cdot\mathbb{P}[\chi(S)\sim H]\cdot\mathbb{P}[X_u=1 \text{ for all } u\in S\cup U]
\end{aligned} \tag{9}$$

By Claim 29 we get that

$$|\{U\subseteq V_G : U\cap S\neq\emptyset \text{ and } \exists\chi \text{ s.t.}\chi(U)\sim H\}| \leq d^{|U\setminus S|}f(H) \tag{10}$$

for some function H, substituting this bound into (9) and using the upper bound $\mathbb{P}[X_u=1 \text{ for all } u\in S\cup U]\leq\mathbb{P}[X_u=1 \text{ for all } u\in S]$ as well as $|U\setminus S|\leq k$, we get

$$\begin{aligned}
Q &\leq d^k f(H)k^{2k}p^{-2\tau}\sum_{S\subseteq V_G}\mathbb{P}[\chi(S)\sim H]\cdot\mathbb{P}[X_u=1 \text{ for all } u\in S] \\
&= d^k f(H)k^k p^{-\tau}\mathbb{E}[Z].
\end{aligned}$$

Putting this together with (8) and using $\mathbf{Var}(Z)=\mathbb{E}[Z^2]-\mathbb{E}[Z]^2$, we get

$$\mathbf{Var}[Z] = d^k f(H)k^k p^{-\tau}\mathbb{E}[Z],$$

proving the first claim of the lemma.

For the second claim of the lemma first note that by the half-integrality of vertex cover for graphs (Claim 30) as well as the assumption that the vertex cover of H has full support we have

$$\mathbb{P}[X_u=1 \text{ for all } u\in S\cup U]\leq\mathbb{P}[X_u=1 \text{ for all } u\in S]\cdot p^{|U\setminus S|/2}. \tag{11}$$

We now get, using (9), that

$$\begin{aligned}
Q &\leq k^{2k}p^{-2\tau}\sum_{S\subseteq V_G}|\{U\subseteq V_G : U\cap S\neq\emptyset \text{ and } \exists\chi \text{ s.t.}\chi(U)\sim H\}|\cdot \\
&\qquad\qquad\cdot\mathbb{P}[\chi(S)\sim H]\cdot\mathbb{P}[X_u=1 \text{ for all } u\in S\cup U] \\
&= k^{2k}p^{-2\tau}\sum_{S\subseteq V_G}\sum_{r\geq 1}|\{U\subseteq V_G : |U\cap S|=r \text{ and } \exists\chi \text{ s.t.}\chi(U)\sim H\}|\cdot \\
&\qquad\qquad\cdot\mathbb{P}[\chi(S)\sim H]\cdot\mathbb{P}[X_u=1 \text{ for all } u\in S\cup U] \\
&\leq k^{2k}p^{-2\tau}\sum_{S\subseteq V_G}\sum_{r\geq 1} d^r f(H)\cdot\mathbb{P}[\chi(S)\sim H]\cdot\mathbb{P}[X_u=1 \text{ for all } u\in S\cup U] \\
&\leq f(H)k^{2k}p^{-2\tau}\sum_{S\subseteq V_G}\sum_{r\geq 1} d^r p^{r/2}\cdot\mathbb{P}[\chi(S)\sim H]\cdot\mathbb{P}[X_u=1 \text{ for all } u\in S] \\
&= f(H)k^{2k}p^{-2\tau}\left(\sum_{r\geq 1} d^r p^{r/2}\right)\sum_{S\subseteq V_G}\mathbb{P}[\chi(S)\sim H]\cdot\mathbb{P}[X_u=1 \text{ for all } u\in S] \\
&\leq 2f(H)(dp^{1/2})k^{2k}p^{-2\tau}\sum_{S\subseteq V_G}\mathbb{P}[\chi(S)\sim H]\cdot\mathbb{P}[X_u=1 \text{ for all } u\in S]. \\
&= 2f(H)(dp^{1/2})k^k p^{-\tau}\mathbb{E}[Z].
\end{aligned}$$

In the equation above the second inequality is by (10), the third is by (11), and the last is by summing the geometric series, which is justified due to the assumption $d \leq \frac{1}{2}p^{1/2}$ of the lemma.

Putting the bounds above together, we get

$$\mathbf{Var}[Z] \leq 2f(H)(dp^{1/2})p^{-\tau}\mathbb{E}[Z],$$

as required.

□

We now give

Proof of Theorem 25: Let $s = (100d^k f(H))^{1/\tau} \cdot \varepsilon^{-2/\tau} \cdot m \cdot (T/A(H))^{-1/\tau}$, where $m = |E_G|$ is the number of edges in G.

We consider two cases. If $s > m$, then we simply sample all the edges of G and compute the number of copies of H offline. If $s < m$, we use Algorithm 1 with the sampling parameter p set to $p = s/m$. Note that by Lemma 27 the space complexity is at most s. We get by Lemma 28 that our estimator is unbiased, and by Lemma 31 (first part) that its variance is $\mathbf{Var}[Z] \leq d^k f(H)(m/s)^{\tau}\mathbb{E}[Z] = d^k f(H)(m/s)^{\tau}\mathbb{E}[Z]$.

Since $s = (100d^k f(H))^{1/\tau} \cdot \varepsilon^{-2/\tau} \cdot mT^{-1/\tau}$ by our setting above, we get that

$$\mathbf{Var}[Z] \leq d^k f(H)(m/s)^{\tau}\mathbb{E}[Z] \leq \varepsilon^2 T\mathbb{E}[Z] = \frac{1}{100}\varepsilon^2 T^2,$$

and the theorem follows by Chebyshev's inequality. □

Proof of Theorem 26: Let $s = (100d^k f(H))^{1/\tau} \cdot \varepsilon^{-2/\tau} \cdot m \cdot (T/A(H))^{-1/\tau}$, where $m = |E_G|$ is the number of edges in G.

We consider two cases. If $s > m$, then we simply sample all the edges of G and compute the number of copies of H offline. If $s < m$, we use Algorithm 1 with the sampling parameter p set to $p = s/m$. Note that by Lemma 27 the space complexity is at most s. We get by Lemma 28 that our estimator is unbiased, and by Lemma 31 (second part) that its variance is

$$\begin{aligned}\mathbf{Var}[Z] &\leq 2f(H)(dp^{1/2}) \cdot (m/s)^{\tau}\mathbb{E}[Z] \\ &\leq 2f(H)(d(s/m)^{1/2}) \cdot (m/s)^{\tau}\mathbb{E}[Z] \\ &\leq 2f(H) \cdot (m/s)^{\tau}\mathbb{E}[Z]\end{aligned}$$

under the assumption that $d \leq (m/s)^{1/2}$ (we verify this assumption shortly).

Since $s = (100f(H))^{1/\tau} \cdot \varepsilon^{-2/\tau} \cdot mT^{-1/\tau}$ by assumption of the theorem, we get that

$$\mathbf{Var}[Z] \leq f(H)(m/s)^{\tau}\mathbb{E}[Z] \leq \varepsilon^2 T\mathbb{E}[Z] = \frac{1}{100}\varepsilon^2 T^2,$$

and the theorem follows by Chebyshev's inequality. It remains to verify that $d \leq (m/s)^{1/2}$ under our setting of s, which is indeed true since

$$(m/s)^{1/2} = \left((100f(H))^{-1/\tau} \cdot \varepsilon^{2/\tau} \cdot T^{1/\tau}\right)^{1/2} \geq \frac{1}{10}f(H)^{-1/2}\varepsilon^{1/\tau}T^{1/(2\tau)} \geq d$$

by assumption of the theorem, as required. □

References

[AG09] K. Ahn and S. Guha. Graph sparsification in the semi-streaming model. *ICALP*, pages 328–338, 2009.

[AG11] K. Ahn and S. Guha. Linear programming in the semi-streaming model with application to the maximum matching problem. *ICALP*, pages 526–538, 2011.

[AG13] K. Ahn and S. Guha. Access to data and number of iterations: Dual primal algorithms for maximum matching under resource constraints. *CoRR*, abs/1307.4359, 2013.

[AGM08] Albert Atserias, Martin Grohe, and Dániel Marx. Size bounds and query plans for relational joins. In *Foundations of Computer Science, 2008. FOCS'08. IEEE 49th Annual IEEE Symposium on*, pages 739–748. IEEE, 2008.

[AGM12a] Kook Jin Ahn, Sudipto Guha, and Andrew McGregor. Analyzing graph structure via linear measurements. *SODA*, pages 459–467, 2012.

[AGM12b] Kook Jin Ahn, Sudipto Guha, and Andrew McGregor. Graph sketching: Sparsification, spanners, and subgraphs. *PODS*, 2012.

[AHLW16] Yuqing Ai, Wei Hu, Yi Li, and David P Woodruff. New characterizations in turnstile streams with applications. In *LIPIcs-Leibniz International Proceedings in Informatics*, volume 50. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2016.

[AKL17] Sepehr Assadi, Sanjeev Khanna, and Yang Li. On estimating maximum matching size in graph streams. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2017, Barcelona, Spain, Hotel Porta Fira, January 16-19*, pages 1723–1742, 2017.

[AKLY15] Sepehr Assadi, Sanjeev Khanna, Yang Li, and Grigory Yaroslavtsev. Tight bounds for linear sketches of approximate matchings. *CoRR*, 2015.

[ANRD15] Nesreen K. Ahmed, Jennifer Neville, Ryan A. Rossi, and Nick Duffield. Efficient graphlet counting for large networks. In *Proceedings of the 2015 IEEE International Conference on Data Mining (ICDM)*, ICDM '15, pages 1–10, Washington, DC, USA, 2015. IEEE Computer Society.

[BDGL08] I. Bordino, D. Donato, A. Gionis, and S. Leonardi. Mining large networks with subgraph counting. In *2008 Eighth IEEE International Conference on Data Mining*, pages 737–742, Dec 2008.

[BFL+06] Luciana S Buriol, Gereon Frahling, Stefano Leonardi, Alberto Marchetti-Spaccamela, and Christian Sohler. Counting triangles in data streams. In *Proceedings of the twenty-fifth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 253–262. ACM, 2006.

[BFLS07] Luciana S. Buriol, Gereon Frahling, Stefano Leonardi, and Christian Sohler. Estimating clustering indexes in data streams. In Lars Arge, Michael Hoffmann, and Emo Welzl, editors, *Algorithms – ESA 2007*, pages 618–632, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg.

[BKS02] Ziv Bar-Yossef, Ravi Kumar, and D. Sivakumar. Reductions in streaming algorithms, with an application to counting triangles in graphs. In *Proceedings of the Thirteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '02, pages 623–632, Philadelphia, PA, USA, 2002. Society for Industrial and Applied Mathematics.

[BOV13] Vladimir Braverman, Rafail Ostrovsky, and Dan Vilenchik. How hard is counting triangles in the streaming model? In *Automata, Languages, and Programming*, pages 244–254. Springer, 2013.

[CCE+15] Rajesh Hemant Chitnis, Graham Cormode, Hossein Esfandiari, MohammadTaghi Hajiaghayi, Andrew McGregor, Morteza Monemizadeh, and Sofya Vorotnikova. Kernelization via sampling with applications to dynamic graph streams. *CoRR*, abs/1505.01731, 2015.

[CJ14] Graham Cormode and Hossein Jowhari. A second look at counting triangles in graph streams. *Theoretical Computer Science*, 552:44–51, 2014.

[CMR05] Graham Cormode, S. Muthukrishnan, and Irina Rozenbaum. Summarizing and mining inverse distributions on data streams via dynamic inverse sampling. In *Proceedings of the 31st International Conference on Very Large Data Bases*, VLDB '05, pages 25–36. VLDB Endowment, 2005.

[Cor17] The sparse awakens: Streaming algorithms for matching size estimation in sparse graphs. In *25th Annual European Symposium on Algorithms, ESA 2017, September 4-6, 2017, Vienna, Austria*, pages 29:1–29:15, 2017.

[EHL+15] Hossein Esfandiari, Mohammad Taghi Hajiaghayi, Vahid Liaghat, Morteza Monemizadeh, and Krzysztof Onak. Streaming algorithms for estimating the matching size in planar graphs and beyond. In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2015, San Diego, CA, USA, January 4-6, 2015*, pages 1217–1233, 2015.

[ELRS15] Talya Eden, Amit Levi, Dana Ron, and C. Seshadhri. Approximately counting triangles in sublinear time. In *Proceedings of the 56th FOCS*, pages 614–633. IEEE, 2015.

[ESBD15] Ethan R. Elenberg, Karthikeyan Shanmugam, Michael Borokhovich, and Alexandros G. Dimakis. Beyond triangles: A distributed framework for estimating 3-profiles of large graphs. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '15, pages 229–238, New York, NY, USA, 2015. ACM.

[ESBD16] Ethan R. Elenberg, Karthikeyan Shanmugam, Michael Borokhovich, and Alexandros G. Dimakis. Distributed estimation of graph 4-profiles. In *Proceedings of the 25th International Conference on World Wide Web*, WWW '16, pages 483–493, Republic and Canton of Geneva, Switzerland, 2016. International World Wide Web Conferences Steering Committee.

[FGO17] Orr Fischer, Shay Gershtein, and Rotem Oshman. On the multiparty communication complexity of testing triangle-freeness. *CoRR*, abs/1705.08438, 2017.

[GKK+07] 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 Thirty-ninth Annual ACM Symposium on Theory of Computing*, STOC '07, pages 516–525, New York, NY, USA, 2007. ACM.

[GKK12] A. Goel, M. Kapralov, and S. Khanna. On the communication and streaming complexity of maximum bipartite matching. *SODA*, 2012.

[GO12] Venkatesan Guruswami and Krzysztof Onak. Superlinear lower bounds for multipass graph processing. *CCC*, 2012.

[HRVZ15] Zengfeng Huang, Božidar Radunović, Milan Vojnović, and Qin Zhang. Communication complexity of approximate maximum matching in distributed graph data. *STACS*, 2015.

[JG05] Hossein Jowhari and Mohammad Ghodsi. New streaming algorithms for counting triangles in graphs. In *Computing and Combinatorics*, pages 710–716. Springer, 2005.

[JSP15] Madhav Jha, C. Seshadhri, and Ali Pinar. Path sampling: A fast and provable method for estimating 4-vertex subgraph counts. In *Proceedings of the 24th International Conference on World Wide Web*, WWW '15, pages 495–505, Republic and Canton of Geneva, Switzerland, 2015. International World Wide Web Conferences Steering Committee.

[JST11] Hossein Jowhari, Mert Sağlam, and Gábor Tardos. Tight bounds for lp samplers, finding duplicates in streams, and related problems. In *Proceedings of the Thirtieth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, PODS '11, pages 49–58, New York, NY, USA, 2011. ACM.

[Kap13] Michael Kapralov. Better bounds for matchings in the streaming model. *SODA*, 2013.

[KB80] Rob Kaas and Jan M Buhrman. Mean, median and mode in binomial distributions. *Statistica Neerlandica*, 34(1):13–18, 1980.

[KK15] Dmitry Kogan and Robert Krauthgamer. Sketching cuts in graphs and hypergraphs. *ITCS*, 2015.

[KKL88] J. Kahn, G. Kalai, and N. Linial. The influence of variables on boolean functions. In *Proceedings of the 29th Annual Symposium on Foundations of Computer Science*, SFCS '88, pages 68–80, Washington, DC, USA, 1988. IEEE Computer Society.

[KKP] full version placeholder.

[KKS14] Michael Kapralov, Sanjeev Khanna, and Madhu Sudan. Approximating matching size from random streams. In *25th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2014.

[KKS15] Michael Kapralov, Sanjeev Khanna, and Madhu Sudan. Streaming lower bounds for approximating MAX-CUT. In *26th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2015.

[KKSV17] Michael Kapralov, Sanjeev Khanna, Madhu Sudan, and Ameya Velingker. $(1 + \Omega(1))$-Approximation to MAX-CUT requires linear space. In *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, 2017.

[KL11] Jonathan A. Kelner and Alex Levin. Spectral sparsification in the semi-streaming setting. *STACS*, pages 440–451, 2011.

[KLM+14] Michael Kapralov, Yin Tat Lee, Cameron Musco, Christopher Musco, and Aaron Sidford. Single pass spectral sparsification in dynamic streams. *FOCS*, 2014.

[KMPT10] Mihail N. Kolountzakis, Gary L. Miller, Richard Peng, and Charalampos E. Tsourakakis. Efficient triangle counting in large graphs via degree-based vertex partitioning. In Ravi Kumar and Dandapani Sivakumar, editors, *Algorithms and Models for the Web-Graph*, pages 15–24, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.

[KMSS12] Daniel M. Kane, Kurt Mehlhorn, Thomas Sauerwald, and He Sun. Counting arbitrary subgraphs in data streams. In *Proceedings of the 39th International Colloquium Conference on Automata, Languages, and Programming - Volume Part II*, ICALP'12, pages 598–609, Berlin, Heidelberg, 2012. Springer-Verlag.

[KNP+17] Michael Kapralov, Jelani Nelson, Jakub Pachocki, Zhengyu Wang, David P Woodruff, and Mobin Yahyazadeh. Optimal lower bounds for universal relation, and for samplers and finding duplicates in streams. In *Foundations of Computer Science (FOCS), 2017 IEEE 58th Annual Symposium on*, pages 475–486. Ieee, 2017.

[Kon15] Christian Konrad. Maximum matching in turnstile streams. *CoRR*, abs/1505.01460, 2015.

[KP17] John Kallaugher and Eric Price. A hybrid sampling scheme for triangle counting. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1778–1797. SIAM, 2017.

[KW14] Michael Kapralov and David Woodruff. Spanners and sparsifiers in dynamic streams. *PODC*, 2014.

[LNW14a] Yi Li, Huy L Nguyen, and David P Woodruff. Turnstile streaming algorithms might as well be linear sketches. In *Proceedings of the forty-sixth annual ACM symposium on Theory of computing*, pages 174–183. ACM, 2014.

[LNW14b] Yi Li, Huy L. Nguyen, and David P. Woodruff. Turnstile streaming algorithms might as well be linear sketches. In *Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 - June 03, 2014*, pages 174–183, 2014.

[LW16] Yi Li and David P. Woodruff. On approximating functions of the singular values in a stream. In *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016*, pages 726–739, 2016.

[McG17] Andrew McGregor. Graph sketching and streaming: New approaches for analyzing massive graphs. In *Computer Science - Theory and Applications - 12th International Computer Science Symposium in Russia, CSR 2017, Kazan, Russia, June 8-12, 2017, Proceedings*, pages 20–24, 2017.

[MMPS11] Madhusudan Manjunath, Kurt Mehlhorn, Konstantinos Panagiotou, and He Sun. Approximate counting of cycles in streams. In *Proceedings of the 19th European Conference on Algorithms*, ESA'11, pages 677–688, Berlin, Heidelberg, 2011. Springer-Verlag.

[MV18] Andrew McGregor and Sofya Vorotnikova. A simple, space-efficient, streaming algorithm for matchings in low arboricity graphs. In *1st Symposium on Simplicity in Algorithms, SOSA 2018, January 7-10, 2018, New Orleans, LA, USA*, pages 14:1–14:4, 2018.

[MVV16] Andrew McGregor, Sofya Vorotnikova, and Hoa T. Vu. Better algorithms for counting triangles in data streams. In *Proceedings of the 35th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems*, PODS '16, pages 401–411, New York, NY, USA, 2016. ACM.

[MW10] Morteza Monemizadeh and David P. Woodruff. 1passs relative-error lp-sampling with applications. In *Proceedings of the Twenty-first Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '10, pages 1143–1160, Philadelphia, PA, USA, 2010. Society for Industrial and Applied Mathematics.

[PS18] Pan Peng and Christian Sohler. Estimating graph parameters from random order streams. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2018, New Orleans, LA, USA, January 7-10, 2018*, pages 2449–2466, 2018.

[PSV17] Ali Pinar, C. Seshadhri, and Vaidyanathan Vishal. Escape: Efficiently counting all 5-vertex subgraphs. In *Proceedings of the 26th International Conference on World Wide Web*, WWW '17, pages 1431–1440, Republic and Canton of Geneva, Switzerland, 2017. International World Wide Web Conferences Steering Committee.

[PT12] Rasmus Pagh and Charalampos E Tsourakakis. Colorful triangle counting and a mapreduce implementation. *Information Processing Letters*, 112(7):277–281, 2012.

[TKM11] Charalampos E Tsourakakis, Mihail N Kolountzakis, and Gary L Miller. Triangle sparsifiers. *J. Graph Algorithms Appl.*, 15(6):703–726, 2011.

[TKMF09] Charalampos E Tsourakakis, U Kang, Gary L Miller, and Christos Faloutsos. Doulion: counting triangles in massive graphs with a coin. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 837–846. ACM, 2009.

[VY11] Elad Verbin and Wei Yu. The streaming complexity of cycle counting, sorting by reversals, and other problems. *SODA*, pages 11–25, 2011.

[Wol08] Ronald de Wolf. *A Brief Introduction to Fourier Analysis on the Boolean Cube*. Number 1 in Graduate Surveys. Theory of Computing Library, 2008.

A Proofs Omitted from Section 3

Proof of Lemma 4: We will write $f_i' : \{0,1\}^{kn} \to \mathbb{R}$ for the function given by:

$$f_i'((z_i)_{i=1}^k) = f_i(z_i)$$

We first note that, if $s_j \neq 0$ for any $j \neq i$, $\widehat{f_i'}((s_j)_{j=1}^k) = 0$. To show this, let j, l be such that $(s_j)_l = 1$. Then partition the elements of $\{0,1\}^{kn}$ into pairs z, z' where z' is obtained by flipping $(z_j)_l$. Then $f_i'(z) = f_i'(z')$ while $\chi_{(s_j)_{j=1}^k}(z) = -\chi_{(s_j)_{j=1}^k}(z')$, and so $f_i'(z)\chi_{(s_j)_{j=1}^k}(z) + f_i'(z')\chi_{(s_j)_{j=1}^k}(z') = 0$. Therefore:

$$\begin{aligned}\widehat{f_i}((s_j)_{j=1}^k) &= \sum_{z \in \{0,1\}^{kn}} f_i(z)\chi_{(s_j)_{j=1}^k}(z) \\ &= 0\end{aligned}$$

Now, if $(s_j)_{j=1}^k$ has $s_j = 0$ for all $j \neq i$:

$$\begin{aligned}\widehat{f_i'}((s_j)_{j=1}^k) &= \frac{1}{2^{kn}} \sum_{(z_j)_{j=1}^k \in \{0,1\}^{kn}} f_i'((z_j)_{j=1}^k)(-1)^{(z_j)_{j=1}^k \cdot (s_j)_{j=1}^k} \\ &= \frac{1}{2^{kn}} \sum_{(z_j)_{j=1}^k \in \{0,1\}^{kn}} f_i(z_i)(-1)^{z_i \cdot s_i} \\ &= \frac{2^{(k-1)n}}{2^{kn}} \sum_{z \in \{0,1\}^n} f_i(z)(-1)^{z \cdot s_i} \\ &= \widehat{f_i}(s_i)\end{aligned}$$

So then, as

$$f = \prod_{i=1}^k f_i'$$

we can apply the convolution theorem for Fourier transforms:

$$\widehat{f}((s_i)_{i=1}^k) = \sum_{t^{(2)} \in \{0,1\}^{kn}, \ldots, t^{(k)} \in \{0,1\}^{kn}} \widehat{f_1'}\left((s_i)_{i=1}^k \oplus \bigoplus_{i=2}^k t^{(i)}\right) \prod_{i=2}^k \widehat{f_i'}\left(t^{(i)}\right).$$

Now, in the above sum, for each $i = 2, \ldots, |E|$, $\widehat{f_i}(t^{(i)})$ will be zero if $t^{(i)}$ has any ones outside of $(t^{(i)})_i$. So the only non-zero term of this sum is the one where $t^{(i)} = (0, \ldots, s_i, \ldots, 0)$ for $i = 2, \ldots, k$. Therefore:

$$\begin{aligned}\widehat{f}((s_i)_{i=1}^k) &= \prod_{i=1}^k \widehat{f_i'}((0, \ldots, s_i, \ldots, 0)) \\ &= \prod_{e \in E} \widehat{f_i}(s_i).\end{aligned}$$

□

Proof of Lemma 7: We apply the KKL lemma with $\delta = \frac{1}{\lambda c} k \in [0,1]$, getting:

$$\begin{aligned}
\frac{2^{2n}}{|A|^2} \sum_{s \in \{0,1\}^n ; |s|=k} \widehat{f}(s)^2 &\leq \frac{2^{2n}}{|A|^2} \frac{1}{\delta^k} \left(\frac{|A|}{2^n} \right)^{\frac{2}{1+\delta}} \\
&= \frac{1}{\delta^k} \left(\frac{2^n}{|A|} \right)^{\frac{2\delta}{1+\delta}} \\
&\leq \frac{1}{\delta^k} \left(\frac{2^n}{|A|} \right)^{2\delta} \\
&\leq \frac{2^{2\delta c}}{\delta^k} \\
&= \left(\frac{2^{1/\lambda} \lambda c}{k} \right)^k \\
&\leq \left(\frac{2\lambda c}{k} \right)^k
\end{aligned}$$

□