Lower bounds for Lovász-Schrijver systems and beyond follow from multiparty communication complexity

Paul Beame*
Computer Science & Engineering
University of Washington
Seattle, WA 98195-2350
beame@cs.washington.edu

Toniann Pitassi†
Department of Computer Science
University of Toronto
Toronto, ON M5S 1A4
toni@cs.toronto.edu

Nathan Segerlind‡
Department of Computer Science
Portland State University
Portland, Oregon
nsegerli@cs.pdx.edu

11 December 2006

Abstract

We prove that an $\omega(\log^4 n)$ lower bound for the three-party number-on-the-forehead (NOF) communication complexity of the set-disjointness function implies an $n^{\omega(1)}$ size lower bound for tree-like Lovász-Schrijver systems that refute unsatisfiable CNFs. More generally, we prove that an $n^{\Omega(1)}$ lower bound for the $(k+1)$-party NOF communication complexity of set-disjointness implies a $2^{n^{\Omega(1)}}$ size lower bound for all tree-like proof systems whose formulas are degree k polynomial inequalities.

1 Introduction

Zero-one programming is the problem of optimizing a linear objective function over the zero-one points of a polytope. It is a useful framework for expressing optimization problems. In particular, Boolean CNF satisfiability can be easily recast as a zero-one programming problem, and for this reason zero-one programming was among the first discrete optimization problems proved to be NP-complete. In contrast, linear programming, the problem of optimizing a linear objective function over all points of a polytope, is polynomial-time solvable [19]. Many attempts have been made to transfer efficient techniques from linear programming to zero-one programming, and among them are the Lovász-Schrijver "lift-and-project" methods. In this paper we establish limitations on using such methods to prove unsatisfiability for CNFs, modulo a conjecture in communication complexity.

*Supported by NSF grants CCR-0098066, CCR-0514870, and ITR-0219468

†Supported by an Ontario Premiere's Research Excellence Award, an NSERC grant, and the Institute for Advanced Study. Research done while at the Institute for Advanced Study.

‡Supported by NSF Grant DMS-0303258. Work done while at Institute for Advanced Study and the University of Washington.

Techniques for zero-one programming often come from the slightly more general realm of optimizing over the integral points of a polytope. One approach for reducing these integer programming problems to linear programming problems is to begin with the polytope defined by the original linear program without integrality constraints and systematically pare down the polytope by repeatedly refining the linear program with "cutting planes" that remove only non-integral solutions until we are left with the convex hull of the integral solutions. These are local methods in which an initial polytope Q is transformed by a sequence of local operations to smaller and smaller sub-polytopes until the integral hull of Q is reached. At this point, rational linear programming finds the correct solution. Note that for decision problems, this procedure terminates with the empty polytope if and only if the initial polytope contains no integral points. A well-known method of this kind is the use of Gomory-Chvátal cuts [9] which derive each new cutting plane as a linear combination and shift of existing facet constraints.

For zero-one programming, there are more subtle methods available. Lovász and Schrijver [21] introduced a variety of cutting planes methods that derive new cutting planes by first "lifting" the inequalities to higher degree polynomial inequalities (in particular quadratic inequalities) and then "projecting" them down to linear inequalities using polynomial identities and the fact that $x^2 = x$ for $x \in \{0, 1\}$.

The Lovász-Schrijver methods for solving zero-one programs can be naturally used for propositional proof systems. Consider the problem of proving that a CNF is unsatisfiable (equivalently, proving that a DNF is a tautology). Each clause is mapped to an equivalent linear inequality, for example, $x \vee \neg y \vee z$ is mapped to $x + 1 - y + z \geq 1$. By repeated application of the lift-and-project rules and elementary linear algebra, inequalities of quadratic polynomials are derived from the translated clauses, and we can arrive at the inconsistent inequality $1 \geq 0$ if and only if the CNF is unsatisfiable[1]. In this way, we obtain propositional proof systems for CNF unsatisfiability in which the formulas are quadratic inequalities, and the rules of inference are the algebraic manipulations coming from the lift-and-project steps and elementary linear algebra. Collectively, these propositional proof systems are known as *Lovász-Schrijver systems (LS)*. An important feature of the LS systems is that they can provide exponentially smaller proofs for certain tautologies, such as the pigeonhole principle, than the ones possible with systems such as resolution or constant-depth Frege systems.

There are two complexity measures that are commonly studied for cutting-planes based proof systems such as Lovász-Schrijver and the Gomory-Chvátal cutting planes system: *size* and *rank*. Intuitively, rank is the number of intermediate polytopes that must be passed through before arriving at the integral hull. In [21] it was shown that for any (relaxed) polytope P, if the rank of P is d, then the optimization and decision problems for P can be solved exactly deterministically in time $n^{O(d)}$. This makes Lovász-Schrijver systems especially appealing for solving or approximating NP-hard optimization problems via semidefinite programming. A variety of rank lower bounds for exact solution are known, even for the case of unsatisfiable systems [4, 11, 15, 10, 16]. Moreover, interesting bounds on the ranks required for good approximations to vertex cover [1] and MaxSAT [7] have been obtained. This, in turn, implies inapproximability results for these problems for *any* polynomial-time algorithm based on rank.

While there is a rich and growing body of results concerning rank, very little is known about the size of LS proofs. From the proof theoretic perspective, the size of a proof is defined in the usual manner, but from an informal geometric perspective, the size of a LS procedure with respect to some polytope P is the smallest number of hyperplanes defining all of the polytopes that we need to pass through before arriving at the integral hull. Clearly size lower bounds imply rank lower bounds, and indeed, size lower bounds for tree-like proofs[2] imply rank lower bounds, but whether the converse holds is open.

[1] The proof systems are made more precise in Subsection 2.2.

[2] A proof is tree-like if each formula is used at most once as an antecedent to an inference. That is, each time a formula is

At the time of this writing, it is unknown whether or not every unsatisfiable CNF Φ possesses a tree-like LS proof of unsatisfiability whose size is bounded by a polynomial in the number of symbols in Φ. Of course, if every unsatisfiable CNF has such a small refutation, then $\mathsf{NP} = \mathsf{coNP}$, so one might say that because this is unlikely, the problem is "resolved modulo a plausible complexity theoretic conjecture". However, this is really begging the question as one would expect that establishing limitations for specific proof methods is prerequisite to establishing limitations for *all* propositional proof systems. Moreover, there are several similar results and conditional lower bounds based on weaker assumptions:

1. Results of Pavel Pudlák [25] , extended by Sanjeeb Dash [12, 13], establish that certain formulations of the Lovász-Schrijver refutation systems possess "effective interpolation". Therefore, under the conjecture that there are disjoint NP pairs that are not separable by a polynomial-size circuit, these systems require super-polynomial size to refute some CNFs. The hypothesis that some NP disjoint pairs cannot be separated by polynomial-size circuits is not known to imply $\mathsf{NP} \neq \mathsf{coNP}$, so these results provide further evidence that Lovász-Schrijver proofs require super-polynomial size to refute some CNFs.

2. Grigoriev, Hirsch and Pasechnik showed that there are unconditional superpolynomial size lower bounds known for tree-like LS proofs that certain "non-CNF" polytopes contain no zero-one points [16].

3. Several unconditional lower bounds are known for similar systems that are incomparable with or apparently weaker than tree-like Lovász-Schrijver systems with respect to proof size. An exciting series of papers, culminating in the celebrated result of Pudlák, showed that unconditionally, DAG-like cutting planes proof system (a kind of logic whose formulas are linear inequalities and whose inference rules are based on Gomory-Chvátal cuts) requires super-polynomial size to refute certain CNFs [17, 5, 24]. Sanjeeb Dash has extended this work and proved unconditionally that a restricted form of Lovász-Schrijver system (one that makes only "non-commutative cuts") requires superpolynomial size to refute certain CNFs [12, 13].

In this paper, we develop a new method for approaching size lower bounds for tree-like LS and for systems that generalize tree-like LS. Our main result is that lower bounds on the three-party communication complexity of set disjointness (in the number-on-forehead model) imply lower bounds on the size of tree-like LS proofs for a particular family of unsatisfiable CNF formulas. We also generalize this result to a much more general family of proof systems known as semantic $\mathbf{Th}(\mathbf{k})$, where lines are now degree k polynomial inequalities. All versions of LS are special cases of $\mathbf{Th}(\mathbf{2})$, and Chvátal's Cutting Planes proof system is a special case of $\mathbf{Th}(\mathbf{1})$.

More generally, we show that proving lower bounds on the $(k+1)$-party communication complexity of set disjointness implies lower bounds on the size of tree-like semantic $\mathbf{Th}(\mathbf{k})$ proofs. A lower bound showing that $\textsc{Disj}_k$ is not in $(k+1)$-RP^{cc} would give excellent lower bounds for $\mathbf{Th}(\mathbf{k})$ proofs.

Admittedly, this is another conditional result towards proving size lower bounds for tree-like LS proofs. However, we feel that there is a significant difference between an approach that is based on the conjecture "set disjointness requires $\omega(\log^4 n)$ bits of communication in the three-player number-on-the-forehead model" and assumptions such as "$\mathsf{NP} \neq \mathsf{coNP}$" or "there exists an NP disjoint pair that cannot be separated by a polynomial size circuit". The latter problems both imply that $\mathsf{P} \neq \mathsf{NP}$, one of the most famous and difficult problems in contemporary computer science and mathematics, with few persons making serious claims of substantial progress towards its resolution, and the task of establishing proof-size lower bounds

reused, it must be re-derived. For many proof systems, there are CNFs for which the smallest tree-like proofs of unsatisfiability are exponentially larger than the smallest unrestricted proofs of unsatisfiability, but it is open whether or not this holds for LS proofs.

can be viewed as establishing partial evidence in support of these conjectures. On the other hand, number-on-the-forehead communication complexity is an area in which there has been substantial progress in the decades since its introduction in 1983 [8]. There are specific, concrete functions on n-bit inputs for which the three-player number-on-the-forehead communication complexity is known to require $\Omega(n)$ bits of communication, so the problem is one of establishing the bound for the set-disjointness function in particular. Moreover, in the two-player model, set-disjointness is known to require $\Omega(n)$ bits of communication. Finally, the authors of this paper with Avi Wigderson have established $n^{\Omega(1)}$ lower bounds for the computation of the set-disjointness function by certain restricted protocols in the number-on-the-forehead model.

Our proof can be seen as a generalization of [17] to arbitrary k but the extension requires a number of new ideas and a substantially more complicated argument that includes a detailed analysis of large sets of vertex-disjoint paths in expander graphs.

Our work is incomparable to interpolation-based results of Pudlák and Dash. While our bound is conditional based upon what seems to be a more earthly conjecture than strengthenings of $\mathsf{P} \neq \mathsf{NP}$, our bounds apply only for the tree-like case whereas the results of Pudlák and Dash apply to the DAG-like case as well. On the other hand, their interpolation theorems depend highly on the form of the cuts used whereas our semantic approach allows the result to apply to almost any system for manipulating polynomial inequalities with reasonable inference rules.

While the results of Grigoriev et al [16] are unconditional, they do not apply to systems of inequalities that arise from the translation of CNFs. Rather, an exponential size lower bound was proved for all tree-like LS refutations of the equality $x_1 + \ldots + x_n = \alpha$, where α is a non-integer in the range $(\lceil n/4 \rceil, \lfloor 3n/4 \rfloor)$. As this equality cannot arise as the translation of a CNF into inequalities, their bound says nothing about the LS systems for propositional unsatisfiability. Indeed, proving tree-like size lower bounds for CNF polytopes was given as one of the main problems left open in their paper.

2 Definitions

2.1 Multiparty Communication Complexity and Set Disjointness

The *k-party number-on-the-forehead (NOF) model of communication complexity* computes functions (or relations) of input vectors $(x_1, \ldots, x_k) \in X_1 \times \ldots \times X_k$ distributed among k parties, such that party $i \in [k]$ sees all x_j for all $j \in [k]$, $j \neq i$. It is as if player i has the i'th input on his forehead, hence the name. The players communicate by transmitting bits over a channel shared by all players. The communication complexity of a protocol is the number of bits exchanged. For a function $f : X_1 \times \ldots \times X_k \to \{0,1\}$, we define $R^k_\epsilon(f)$ to be the minimum cost of a randomized protocol that computes f with probability of error at most ϵ. For a more thorough treatment of communication complexity, see the monograph by Kushilevitz and Nisan [20].

The *k-party set disjointness problem* $\textsc{Disj}_{k,m} : (\{0,1\}^m)^k \to \{0,1\}$ is defined by $\textsc{Disj}_{k,m}(\vec{x}) = 1$ if and only if there is some $j \in [m]$ such that $x_{i,j} = 1$ for all $i \in [k]$. Although it might be more appropriate to call this function set intersection rather than disjointness, we follow standard terminology. A *$(0,\epsilon)$-error k-party NOF communication protocol* for set disjointness is a protocol that for every disjoint input produces output 0 and for intersecting inputs outputs 1 with probability at least $1 - \epsilon$.

It is conjectured that for all constants $k \geq 2$ the k-party set disjointness problem of length n requires randomized NOF communication complexity that is $\Omega(n/2^k)$ [3]. The conjecture is proven for $k = 2$ [18], but the best known lower bound for $k \geq 3$ is $\Omega(\log n)$ for general models and $\Omega(n^{1/k})$ for more restricted models [3].

2.2 Threshold Logics

The best known classes of threshold logics are Gomory-Chvátal cutting planes [9], the matrix cuts of Lovász and Schrijver [21], and the lift-and-project relaxations of Sherali and Adams [27]. First we briefly describe Gomory-Chvátal cutting planes, which is referred to in the literature as simply Cutting Planes (CP). A CP proof of unsatisfiability of a set of integer linear inequalities $f = \{\vec{a}_1 \cdot \vec{x} \geq b_1, \ldots \vec{a}_m \cdot \vec{x} \geq b_m\}$ is a sequence of integer linear inequalities $\vec{c}_1 \cdot \vec{x} \geq t_1, \ldots, \vec{c}_q \cdot \vec{x} \geq t_q$ such that each $\vec{c}_i \cdot \vec{x} \geq t_i$ is either an inequality from f, an axiom ($x \geq 0$ or $1 - x \geq 0$), or is obtained by one of the two rules: (i) $\vec{c}_i \cdot \vec{x} \geq t_i$ is a positive integer linear combination of some previously derived inequalities; or (ii) $\vec{c}_i \cdot \vec{x} \geq t_i$ is obtained from a previous inequality $d\vec{c}_i \cdot \vec{x} \geq t_i$ by rounding (to obtain $\vec{c}_i \cdot \vec{x} \geq \lceil t_i/d \rceil$).

There are several cutting planes proof systems defined by Lovász and Schrijver [21], collectively referred to as matrix cuts. These systems allow one to "lift" the linear inequalities to degree-two polynomials and then project back to degree one, using the fact that $x^2 = x$ for $x \in \{0, 1\}$. To see that the definitions below are equivalent to the original definitions of Lovász and Schrijver, see [12].

Definition 2.1. *Given a polytope $P \subseteq \mathbb{Q}^n$ defined by $\vec{a}_i \cdot \vec{x} \geq b_i$ for $i = 1, 2, \ldots, m$:*

(1) An inequality $d - \vec{c} \cdot \vec{x} \geq 0$ is called an N-cut for P if

$$d - \vec{c} \cdot \vec{x} \;=\; \sum_{i,j} \alpha_{ij}(b_i - \vec{a}_i \cdot \vec{x})x_j + \sum_{ij} \beta_{ij}(b_i - \vec{a}_i \cdot \vec{x})(1 - x_j) + \sum_j \lambda_j(x_j^2 - x_j),$$

where $\alpha_{ij}, \beta_{ij} \geq 0$ and $\lambda_j \in R$ for $i = 1, \ldots, m$, $j = 1, \ldots, n$.

(2) A weakening of N-cuts, called N_0-cuts can be obtained if when simplifying to the term $d - \vec{c} \cdot \vec{x}$, we view $x_i x_j$ as distinct from $x_j x_i$.

(3) An inequality $d - \vec{c} \cdot \vec{x}$ is called an N_+-cut if

$$\begin{aligned} d - \vec{c} \cdot \vec{x} \;=\; & \sum_{i,j} \alpha_{ij}(b_i - \vec{a}_i \cdot \vec{x})x_j + \sum_{ij} \beta_{ij}(b_i - \vec{a}_i \cdot \vec{x})(1 - x_j) \\ & + \sum_j \lambda_j(x_j^2 - x_j) + \sum_k (g_k + \vec{h}_k \cdot \vec{x})^2, \end{aligned}$$

where again $\alpha_{ij}, \beta_{ij} \geq 0$, $\lambda_j \in R$ for $i = 1, \ldots, m$, $j = 1, \ldots, n$ and $g_k + \vec{h}_k \cdot \vec{x}$ is an affine function for $k = 1, \ldots, n + 1$.

The operators N, N_0 and N_+ are called the *commutative*, *non-commutative* and *semidefinite* operators, respectively. All three are collectively called *matrix-cut* operators.

Definition 2.2. *A Lovász-Schrijver (LS) refutation for f is a sequence of inequalities $g_1, \ldots, g_q$ such that each g_i is either an inequality from f or follows from previous inequalities by an N-cut as defined above, and such that the final inequality is $0 \geq 1$. Similarly, a LS$_0$ refutation uses N_0-cuts and LS$_+$ uses N_+-cuts.*

Definition 2.3. *Let $\mathcal{P}$ be one of the proof systems CP, LS, LS$_0$ or LS$_+$. Let S be an $\mathcal{P}$-refutation of f, viewed as a directed acyclic graph. If the underlying directed acyclic graph is a tree, then S is a tree-like $\mathcal{P}$-refutation of f. The inequalities in S are represented with all coefficients in binary notation. The size of S is the sum of the sizes of all inequalities in S; the rank of S is the depth of the underlying directed acyclic graph. For a set of boolean inequalities f, the $\mathcal{P}$-size of f is the minimal size over all $\mathcal{P}$ refutations of f. Similarly the $\mathcal{P}$-treesize of f is the minimal size over all tree-like $\mathcal{P}$-refutations of f.*

The inference rules and axioms for the CP, LS, LS_0, and LS_+ systems are easily seen to be sound. Furthermore, it has been shown that, in their tree-like forms, each of CP, LS, LS_0, and LS_+ can p-simulate tree-like resolution (cf. [2]). Therefore, by the completeness of tree-like resolution, the tree-like systems CP, LS, LS_0, and LS_+ can refute every unsatisfiable CNF.

All of above proof systems are special cases of the more general *semantic* threshold logic proof systems which we define now. A *k-threshold formula* over Boolean variables $x_1, \ldots, x_n$ is a formula of the form $\sum_j \gamma_j m_j \geq t$, where γ_j, t are integers, and for all j, m_j is a multilinear monomial of degree at most k. The *size* of a k-threshold formula is the sum of the sizes of γ_j and t, written in binary notation. Let f_1, f_2, g be k-threshold formulas in the variables $\vec{x}$. We say that *g is semantically entailed* by f_1 and f_2 if for every $0/1$ assignment to $\vec{x}$ that satisfies both f_1 and f_2, g is also satisfied.

Let f be an unsatisfiable CNF formula over $x_1, \ldots, x_n$, and let $t_1, \ldots, t_m$ be the underlying set of clauses of f, written as 1-threshold inequalities. A **Th(k)** *refutation of f*, $\mathcal{P}$, is a sequence of k-threshold formulas, $L_1, \ldots, L_q$, where each L_j is one of the inequalities t_i, $i \in [m]$, or is semantically entailed by two formulas L_i and $L_{i'}$ with $i, i' < j$, and the final formula L_q is $0 \geq 1$. The *size* of $\mathcal{P}$ is the sum of the sizes of all k-threshold formulas occurring in $\mathcal{P}$. The proof is *tree-like* if the underlying directed acyclic graph, representing the implication structure of the proof, is a tree. (That is, every formula in the proof, is used at most once as an antecedent of an implication. It is allowed, and quite often necessary, that $L_i = L_j$, and L_i and L_j are used as antecedents for two different inferences. In this way, a formula must be re-derived each time it is used.)

Note that in our definition of these cutting planes systems, we can derive a new inequality from any number of previous inequalities in one step, whereas in the $\mathbf{Th}(\mathbf{k})$ proof system, we are restricted to fan-in two. Because the vector space of degree-at-most-one inequalities has dimension at most $n+1$, in light of Caratheodory's theorem, every inequality derived by purely linear operations in a CP refutation can be derived from at most $n+2$ many previous equations. Therefore, we can assume without loss of generality that the fan-in is at most $n+2$ in CP, and similarly, at most $\binom{n}{2} + n + 2$ in LS, LS_0, and LS_+. Because of this bound on fan-in, refutation size increases by at most an $O(n^2)$ factor when the sums are taken by fan-in two inferences of the form "from $f \geq a$ and $g \geq b$ infer $f + g \geq a + b$". Thus, superpolynomial size lower bounds for tree-like **Th(2)** semantic refutations imply superpolynomial size lower bounds for all tree-like Lovász-Schrijver systems.

Because the inference rule in $\mathbf{Th}(\mathbf{k})$ is semantic entailment, lower bounds for the $\mathbf{Th}(\mathbf{k})$ system apply to almost any tree-like system for deriving polynomial inequalities with reasonable axioms and inference rules, not only the Lovász-Schrijver systems. For example, division operators such as "from $\vec{c} \cdot \vec{x} > 0$ and $(b - \vec{a} \cdot \vec{x})\vec{c} \cdot \vec{x} \geq 0$ infer $\vec{a} \cdot \vec{x} \geq b$" are semantically valid inference rules of fan-in two, and variants of the LS system incorporating such rules fall under our analysis. Furthermore, CP refutations are a special case of **Th(1)** semantic refutations, and thus lower bounds for tree-like **Th(1)** semantic refutations imply similar lower bounds for tree-like CP. This connection was exploited to prove lower bounds for tree-like CP [17].

2.3 Miscellaneous Notation

We use the standard asymptotic notation of Ω, O, ω, and o that is found in theoretical computer science and discrete mathematics. We use the $\pm$ notation in the following nonstandard way: When we write $x = a \pm b$, we mean that $x \in [a - b, a + b]$. We use this in asymptotic sense as well. When we write $x = (1 \pm o(1))M$ we mean that there is a value t with $|t| = o(1)$ so that $x \in [(1 - t)M, (1 + t)M]$.

3 Relating the Size of Threshold Logic Refutations to the Communication Complexity of Search Problems

Let f be an unsatisfiable CNF formula. We will be interested in the following search problem, $Search_f$ associated with f: Given a truth assignment α, find a clause from f which is falsified by α. The model for this computation is a decision tree whose nodes evaluate polynomial threshold functions:

A *k-threshold decision tree* is a rooted, directed tree whose vertices are labeled with k-threshold functions and edges are labeled with either 0 or 1. The leaves of the tree are labeled with clauses of f. A k-threshold decision tree solves $Search_f$ in the obvious way: Start at the root and evaluate the threshold function; follow the edge that is consistent with the value of the threshold function; continue until the computation reaches a leaf and output the associated clause. The size S of a k-threshold decision tree is the sum of the sizes of all threshold formulas in the tree, where the coefficients are written in binary. The depth of a k-threshold decision tree is the depth of the underlying tree.

The following lemma, similar to the degree 1 case in [17], shows that from a small tree-like **Th(k)** refutation of an unsatisfiable formula f, a small-size, small-depth k-threshold decision tree for $Search_f$ can be extracted.

Lemma 3.1. *Let $\mathcal{P}$ be a tree-like* **Th(k)** *refutation of f of size S. Then there is a k-threshold decision tree for $Search_f$ of depth $O(\log S)$ and size $O(S)$. Furthermore, every threshold formula labeling a node of the decision tree is either a formula in the refutation $\mathcal{P}$, or the vacuously true inequality $0 \geq 0$.*

Proof. Assume that $\mathcal{P}$ is a size S tree-like **Th(k)** refutation of f. We will describe a depth $O(\log S)$, size $O(S)$, k-threshold decision tree which computes the search problem associated with f. The proof is by induction on S; clearly if $S = 1$ then the unsatisfiable formula is a single, false threshold formula, so the lemma holds. For the inductive statement, assume that the size of $\mathcal{P}$ is $S > 1$. Because the DAG of the proof is a binary tree, there is an intermediate formula f in $\mathcal{P}$ such that the number of formulas above f (ancestors in the tree) is at least $S/3$ and at most $2S/3$. Let the subtree of $\mathcal{P}$ with root formula f be denoted by $\mathcal{A}$ and write $\mathcal{B}$ for the remainder of $\mathcal{P}$, that is, all formulas of $\mathcal{P}$ that are not in $\mathcal{A}$ and with f replaced by $0 \geq 0$. In the decision tree, the root is labeled with f. Beneath the edge labeled 0, we inductively apply the lemma to the subtree $\mathcal{A}$, and beneath the edge labeled 1, we inductively apply the lemma on the subtree $\mathcal{B}$. Both $\mathcal{A}$ and $\mathcal{B}$ have size at most $2S/3$, so we may apply the induction hypothesis and conclude that the height of the decision tree obtained will be at most $\log_{3/2}(S) + 1$ which is $O(\log S)$. To see that the decision tree computes the search function, notice that if f evaluates to false on a given truth assignment ϕ, then we proceed on the subproof $\mathcal{A}$. By soundness of the proof, at least one of the leaf formulas of $\mathcal{A}$ must be falsified by ϕ. A similar argument holds when f evaluates to true. □

The next lemmas, adapted from arguments in [23], show that any relation computed by a shallow k-threshold decision tree can also be efficiently computed by a $k + 1$ player communication complexity protocol in the number-on-forehead model, over any partition of the variables.

Lemma 3.2. *Suppose that a relation $R(x_1, \ldots, x_{kn})$ is computed by a depth d k-threshold decision tree in which all coefficients are bounded by $N \geq n$. For any partition of the inputs into k sets, there is a $k+1$-party deterministic NOF communication complexity protocol for R in which $O(d \log N)$ bits are communicated in total.*

Proof. Fix a partition of $x_1, \ldots x_{kn}$. Observe that for each monomial in each k-threshold formula there is at least one party that can evaluate the monomial. Let $\alpha_1 m_1 + \ldots + \alpha_q m_q \geq t$ be the k-threshold formula

queried at the root of the k-threshold decision tree for f. The set of monomials m_j can be partitioned into $k+1$ groups, where group i contains monomials that can be "seen" by the i^{th} player. Each player (in turn) communicates the weighted linear combination of their monomials to the other players. After all players have spoken, each player can simply add up the total sum and see if it is greater than the target t, in order to evaluate the k-threshold formula. The $k+1$ players then continue on the half of the decision tree which agrees with the value of this formula. The protocol terminates after d rounds, and each round requires $O(\log N)$ bits of communication. □

In order to prove the randomized version of the above lemma we use a standard randomized protocol for testing linear inequalities. The protocol works in the *number-in-hand* model which is more restricted than the number-on-the-forehead model. In the number-in-the-hand model, each player $i = 1, \ldots k$ has private access to the input x_i (whereas in the NOF model, player i sees inputs $x_1, \ldots x_{i-1}, x_{i+1}, \ldots x_k$).

Lemma 3.3. *Let $y_1, \ldots, y_{k+1}$ be (signed) integers with n-bit binary representations and let $c > 0$. Then there is an $O(k \log^2 n)$-bit $(k+1)$-player number-in-hand probabilistic protocol with error less than $1/n^c$ for determining whether $y_1 + \ldots + y_{k+1} \geq 0$.*

Proof. The players follow a binary search strategy on the bits of the y_i.

Suppose $n \geq 2$. (If $n \leq 2$ the parties simply send their inputs.) Let y_i^H be the high order $\lceil n/2 \rceil$ bits that under-approximate $y_i/2^{\lfloor n/2 \rfloor}$ and y_i^L be the corresponding low order bits for $1 \leq i \leq k+1$. (Some of the y_i may be negative but then the y_i^L will all be positive.) If $\sum_i y_i^H > 0$ then $\sum_i y_i > 0$; similarly, if $\sum_i y_i^H < -k$ then $\sum_i y_i < 0$. Thus, unless $\sum_i y_i^H \in \{-k, \ldots, 0\}$, the answer can be found by determining whether $y_1^H + \ldots + y_{k+1}^H \geq 0$. If $\sum_i y_i^H = -j \in \{-k, \ldots, 0\}$ then the answer can be found by comparing $y_1^L + \ldots + y_k^L + y_{k+1}^L - j \cdot 2^{\lfloor n/2 \rfloor}$ to 0.

Player 1 randomly selects a prime number $p \in [n^{c+2} \log n, 2n^{c+2} \log n]$ and sends $(p, y_1^H \bmod p)$. For $i = 2$ to k, player i, sends $y_i^H \bmod p$. Then, using these values and his own private input, player $k+1$ computes $z = \sum_{i=1}^{k+1} y_i^H \bmod p$.

If $z \not\equiv -j \pmod p$ for $j \in \{0, \ldots, k\}$ then player $k+1$ sends the bit 1 and the protocol continues recursively, using $y_1^H, \ldots, y_{k+1}^H$ instead of $y_1, \ldots, y_{k+1}$.

If $z \equiv -j \pmod p$ for $j \in \{0, \ldots, k\}$ then player $k+1$ sends the bit 0 and j and the protocol continues recursively with players 1 to j using $(y_1^L - 2^{\lfloor n/2 \rfloor}), \ldots, (y_j^L - 2^{\lfloor n/2 \rfloor})$ instead of $y_1, \ldots, y_j$ and players $j+1$ to $k+1$ using $y_{j+1}^L, \ldots, y_{k+1}^L$ instead of $y_{j+1}, \ldots, y_{k+1}$.

In both of the recursive calls the integers each have at most $\lfloor n/2 \rfloor + 1$ bits. At each stage, we send $O(\log n)$ bits, and the total number of stages is $O(\log n)$ for a total of $O(\log n)^2$ bits sent. The probability of error at each stage is $O(1/n^{c+1})$ and therefore the total error is less than $1/n^c$ (for sufficiently large n). □

Lemma 3.4. *Suppose that a relation $R(x_1, \ldots, x_{kn})$ is computed by a depth d k-threshold decision tree in which all coefficients are bounded by $N \geq n$. For any partition of the inputs into k sets, there is a $(k+1)$-party randomized NOF communication complexity protocol for R in which $O(d(\log \log N)^2)$ bits are communicated in total which is correct with probability at least $1 - 1/n$.*

Proof. As in the proof of Lemma 3.2, the players proceed in d rounds, at each step evaluating the threshold formula and proceeding on the consistent subtree. Let $p(\vec{x}) \geq t$ be the threshold formula at the root of the decision tree. As before, partition the monomials of $p(\vec{x})$ into $k+1$ groups where the i^{th} player can "see" the monomials in group i. Each of the $k+1$ players computes the weighted sum of their respective monomials. Call these sums $y_1, \ldots, y_{k+1}$, respectively. Player $k+1$ uses $y'_{k+1} = y_{k+1} - t$ and by applying Lemma 3.3

with $n = \log_2 N$ and c such that $1/n^c < 1/(dn)$, there is a probabilistic protocol allowing the players to determine with error at most $1/(dn)$ whether the sum of the y_i's is at least t, where $O((\log\log N)^2)$ bits are exchanged. After evaluating $p(\vec{x}) \geq t$, the players then continue on the branch of the decision tree which agrees with the value of $p(\vec{x}) \geq t$. The protocol terminates after d rounds, for a total of $O(d(\log\log N)^2)$ bits of communication. By the union bound, the probability of encountering an error at some level of the recursion is at most $d \cdot 1/(dn) = 1/n$ □

The following theorem is an easy corollary of the above lemmas.

Theorem 3.5. *Suppose that f has a tree-like* **Th(k)** *refutation of size S. Then there exists a $(k+1)$-party randomized NOF communication complexity protocol for $Search_f$ (over any partition of the variables into k groups) that communicates $O(\log^3 S)$ bits and has error probability at most $1/n$.*

Further, if all k-threshold formulas in the **Th(k)** *refutation have coefficients bounded by a polynomial in n, then there is a randomized protocol using $O(\log S(\log\log n)^2)$ many bits and error probability at most $1/n$, and a deterministic protocol using $O(\log S \log n)$ bits.*

Proof. We apply Lemma 3.1 to produce a k-threshold decision tree for $Search_f$ of depth $O(\log S)$ and size $O(S)$. Because every label of a node of the decision tree is a formula of the refutation, or the triviality $0 \geq 0$, N is no larger than the maximum absolute value of a coefficient in the refutation.

For the first claim, set N to be the maximum absolute value of any coefficient appearing in the decision tree; by the definition of the size of a proof, $N = 2^{O(S)}$. For the second claim, we by hypothesis that $N = n^{O(1)}$. We apply Lemmas 3.2 and 3.4 to this decision tree to yield the claimed size and error bounds. □

4 The Difficult CNFs, their Search Problems, and an Outline of the Lower Bound Proof

Our hard examples are based on the well-known Tseitin graph formulas. Let $G = (V, E)$ be any connected, undirected graph and let $\vec{c} \in \{0,1\}^V$. The *Tseitin formula for G with respect to charge vector $\vec{c}$*, $TS(G, \vec{c})$, has variables $\text{Vars}(G) = \{y_e \mid e \in E\}$. The formula states that for every vertex $v \in V$, the parity of the edges incident with v is equal to the charge, c_v, at node v. It is expressed propositionally as the conjunction of the clauses obtained by expanding $\oplus_{e \ni v} y_e = c_v$ for each $v \in V$. Note that for a graph with maximum degree d, each clause is of width at most d and the number of clauses is at most $|V|2^d$.

Notice that $TS(G, \vec{c})$ is satisfiable if and only if $\sum_{v \in V} c_v$ is even. For odd $\vec{c}$, the search problem $Search_{TS(G,\vec{c})}$ takes a 0/1 assignment α to $\text{Vars}(G)$ and outputs a clause of $TS(G, \vec{c})$ that is unsatisfied. In other words, a solution to $Search_{TS(G,\vec{c})}$ on input α is a vertex v such that a parity equation at the vertex v is violated by α.

To make the search problem hard for k-party NOF communication protocols, and by Theorem 3.5, hard for $(k-1)$-threshold decision trees, we modify $TS(G, \vec{c})$ by replacing each variable y_e by the conjunction of k variables, $\bigwedge_{i=1}^k y_e^i$, and expanding the result into clauses by use of deMorgan's law. We call the resulting *k-fold Tseitin formula*, $TS^k(G, \vec{c})$, and its variable set, $\text{Vars}^k(G) = \{y_e^i \mid e \in E, i \in [k]\}$.

For a fixed graph G and different odd-charge vectors $\vec{c} \in \{0,1\}^{V(G)}$, the problems $Search_{TS^k(G,\vec{c})}$ are very closely related. Define $\text{ODDCHARGE}^k(G)$ to be the k-party NOF communication search problem which takes as input an odd charge vector $\vec{c} \in \{0,1\}^{V(G)}$, seen by all players, and an assignment α to $\text{Vars}^k(G)$, in which player i sees all values but the assignment α_e^i to y_e^i for $e \in E(G)$, and requires that the players output a vertex v that is a solution to $Search_{TS^k(G,\vec{c})}$.

The communication complexity of $\text{ODDCHARGE}^k(G)$ depends on the graph G, and we use a carefully modified expander to obtain our lower bounds. We use a family of graphs H_n such that each H_n is the union of two edge-disjoint graphs on the same set of n vertices $[n]$, G_n and T_n. G_n is a Δ-regular expander graph of the form defined by Lubotzky, Phillips, and Sarnak [22] for $\Delta = \Theta(\log n)$. Since $\overline{G_n}$ has degree $> n/2$, there is a spanning tree T_n of maximum degree 2 (one can take the Hamiltonian path guaranteed by Dirac's Theorem, cf. [14]) in $\overline{G_n}$. Clearly H_n also has maximum degree $\Theta(\log n)$ and thus $TS^k(H_n, \vec{c})$ has size $n^{O(k)}$. (Notice that the graph H_n has degree $O(\log n)$, so the CNF $TS^k(H_n, \vec{c})$ has size $n^{O(1)}$ and width $O(\log n)$.)

Now we are ready to describe the sequence of reductions to show that an efficient k-party NOF communication complexity protocol for $\text{ODDCHARGE}^k(H_n)$ will imply an efficient 1-sided error randomized k-party NOF protocol for the set disjointness relation. The reduction passes through two intermediate problems: A search problem called $\text{EVENCHARGE}^k(H_n)$, and set-disjointness with the promise that for every input under consideration, the size of the intersection is either zero or one. Reducing the general set disjointness problem to this *zero/one set disjointness* problem is a standard application of Valiant-Vazirani isolation (Lemma 5.4, after [28]). Our reduction from zero/one set disjointness to $\text{ODDCHARGE}^k(H_n)$ goes via an intermediate problem, $\text{EVENCHARGE}^k(H_n)$, which is the exact analog of $\text{ODDCHARGE}^k(H_n)$ except that the input charge vector $\vec{c}$ is even rather than odd and the task is to either find a charge violation or to determine that no charge violation exists. For an assignment α and a charge vector $\vec{c}$, we define $\text{Err}(\alpha, \vec{c})$ to be the set of vertices at which the parity constraints are violated by α.

Theorem 4.1. *Let $k \geq 2$ and $m = n^{1/3}/\log n$. For each n there is an odd charge vector $\vec{c} \in \{0,1\}^n$ such that for any $\epsilon < 1/2$ the size of any tree-like* **Th(k-1)** *refutation of $TS^k(H_n, \vec{c})$ is at least $2^{\Omega((R^k_\epsilon(\text{DISJ}_{k,m})/\log n)^{1/3})}$. Further if the coefficients in the* **Th(k-1)** *refutations are bounded by a polynomial in n then the refutation size must be at least $2^{\Omega(R^k_\epsilon(\text{DISJ}_{k,m})/(\log n(\log\log n)^2))}$.*

The proof of Theorem 4.1 is presented at the end of Section 5. Here we provide a high-level outline of the proof and its component lemmas. In the sketch, quantities are left out, and definitions are not precise.
Proof Sketch: Suppose the sake of contradiction that there is a small $\mathbf{Th(k-1)}$ refutation of $TS^k(H_n, \vec{c})$.

1. We apply the refutation-to-search conversion of Theorem 3.5 to obtain a low-communication k-player NOF protocol for the $\text{ODDCHARGE}^k(H_n)$ search problem.

2. Using Lemma 5.1, we convert the search protocol for $\text{ODDCHARGE}^k(G)$ to a search protocol for $\text{EVENCHARGE}^k(H_n)$ that correctly solves "most" $\text{EVENCHARGE}^k(H_n)$ instances. "Most" is measured by a distribution $\mathcal{D}_t$ on the $\text{EVENCHARGE}^k(H_n)$ instances in which there are exactly $2t$ nodes at which the parity constraints are violated. The distribution is $\mathcal{D}_t$ defined in Definition 5.2 of Section 5.

3. In Lemma 5.2 we show that the $0/1$ set-disjointness problem randomly reduces to $\text{EVENCHARGE}^k(H_n)$ in the following sense: For each set disjointness instance $\vec{x}$, there is a distribution $R(\vec{x})$ on $\text{EVENCHARGE}^k(H_n)$ instances so that if $|\cap \vec{x}| = 0$, the instance generated satisfies all parity constraints, and if $|\cap \vec{x}| = 1$, the instance generated has exactly two nodes at which the parity constraints are violated.

4. The distributions $R(\vec{x})$ and $\mathcal{D}_t$ do not coincide, but they are close enough. In Lemma 5.3, it is shown that when $|\cap \vec{x}| = 0$, $R(\vec{x})$ and $\mathcal{D}_0$ are ϵ-close in l_1 distance, and similarly, when $|\cap \vec{x}| = 1$, $R(\vec{x})$ and $\mathcal{D}_1$ are ϵ-close in l_1 distance. Therefore, using the protocol of Lemma 5.1 on inputs generated by $R(\vec{x})$ correctly solves $0/1$ set-disjointness with added probability of error at most ϵ. Lemma 5.3 is the most delicate part of the argument, and it is where most of the work is invested.

A simple argument shows that the lower bound of Theorem 4.1 holds for all odd charge vectors.

Theorem 4.2. *The same lower bounds as Theorem 4.1 hold for* every *odd charge vector* $\vec{c} \in \{0,1\}^n$.

Proof. Observe that distributions $\mathcal{D}_t$ and $R(\vec{x})$ on the assignments to $\text{Vars}^k(H_n)$ both have the property that for each edge e of T_n, $\alpha_e^1 = \cdots = \alpha_e^k$. Therefore in the proof of Theorem 4.1 observe that we can replace $TS^k(H_n, \vec{c})$ by $\widetilde{TS}^k(H_n, \vec{c}) = TS^k(H_n, \vec{c}) \wedge EQ(T_n)$ where $EQ(T_n)$ is the conjunction of $(\neg y_e^i \vee y_e^j)$ for every $i \neq j \in [k]$ and every $e \in T_n$. The size of any **Th(k-1)** refutation of $TS^k(H_n, \vec{c})$ is at least that of $\widetilde{TS}^k(H_n, \vec{c})$. Moreover, it is not hard to see that for any odd weight vectors $\vec{c}, \vec{d} \in \{0,1\}^n$, $\widetilde{TS}^k(H_n, \vec{c})$ and $\widetilde{TS}^k(H_n, \vec{d})$ have proof sizes that differ by at most a polynomial additive term: Given a small proof of $\widetilde{TS}^k(H_n, \vec{d})$, let $S \subset [n]$ be the set of vertices v for which $c_v \neq d_v$. Since both c and d are odd weight vectors, $|S|$ is even. Let $M \subset E(T_n)$ be the set of edges of corresponding to $|S|/2$ disjoint sub-paths in T_n that match the elements in S.

Applying the substitution of $y_e^i = \neg y_e^i$ for each $e \in M$ and $i \in [k]$ *almost* converts a refutation of $\widetilde{TS}^k(H_n, \vec{d})$ into a refutation of $\widetilde{TS}^k(H_n, \vec{c})$. A positive literal y_e^i in a clause from $TS^k(H_n, \vec{c})$, with e on a toggled path, becomes $\neg y_e^i$, rather than $\neg y_e^1 \vee \ldots \vee \neg y_e^k$ which is the proper form for negative literals in clauses of $TS^k(H_n, \vec{d})$. This is corrected by application of the subsumption rule. A disjunction of negative literals $\neg y_e^1 \vee \ldots \vee \neg y_e^k$ in a clause from $TS^k(H_n, \vec{c})$, with e on a toggled path, becomes $y_e^1 \vee \ldots \vee y_e^k$, rather than one of $y_e^1, \ldots, y_e^k$, as is the proper form for positive literals in clauses of $TS^k(H_n, \vec{d})$. Application of the axioms $\neg y_e^j \vee y_e^i$, with resolution steps, correct this. These corrections increase the size of the proof by at most an $O(k)$ factor. □

5 Reduction from Set Disjointness to ODDCHARGE

The reduction from $\text{EVENCHARGE}^k(H_n)$ to $\text{ODDCHARGE}^k(H_n)$ works by planting a single randomly chosen additional charge violation. This yields a protocol for $\text{EVENCHARGE}^k(H_n)$ that works well on average for each class of inputs with a given number of charge violations. This is similar in spirit to a reduction of Raz and Wigderson [26], and the reader might profit by first becoming familiar with that argument.

The difficult part of our argument is the reduction from zero/one set disjointness to $\text{EVENCHARGE}^k(H_n)$. The key idea is that for even $\vec{c}$, charge violations of $TS^k(H_n, \vec{c})$ come in pairs: Given an instance $\vec{x} \in (\{0,1\}^m)^k$ of zero/one set disjointness, using the public coins, the players randomly choose an even charge vector $\vec{c}$ and m vertex-disjoint paths in H_n, $p_1, \ldots, p_m$, for each $j \in [m]$, the players plant the $x_{1,j}, \ldots, x_{k,j}$ as the assignment along each edge of path p_j, in a random solution that otherwise meets the chosen charge constraint. By construction, a charge violation can occur only at the endpoints of a path and only if there is an intersection in the set disjointness problem.

The challenges arise when we would like to apply the average case properties of the $\text{EVENCHARGE}^k(H_n)$ protocol to the instances created by the above distribution. Unfortunately, this distribution is not quite uniform and we need that the distribution is close to uniform. The bulk of the work is in using the properties of H_n, rapid mixing, modest degree, and high girth, to show that the distribution generated by the reduction is sufficiently close to uniform.

Distributions on labeled graphs Let n be given, let H_n be the graph described in Section 4, and let $\vec{c}$ be an even charge vector.

Definition 5.1. *We define $Sol(H_n, \vec{c})$ to be the set of all 0/1 assignments to the edges of H_n so that for each vertex $v \in [n]$, the parity of edges incident with v is equal to c_v. A uniform random distribution over $Sol(H_n, \vec{c})$ can be obtained by first selecting 0/1 values uniformly at random for all edges in G_n and then choosing the unique assignment to the edges of T_n that fulfill the charge constraints given by $\vec{c}$.*

Given a bit value b associated with an edge $e \in G_n$, we can define a uniform distribution $\mathcal{L}_k(b)$ over the corresponding variables y_e^i, $i \in [k]$. Such an assignment is chosen randomly from $\mathcal{L}_k$ on input b by the following experiment. If $b = 1$ then set all variables associated with edge e, y_e^i, $i \in [k]$ to 1. Otherwise if $b = 0$, set the vector $(\vec{y}_e)_{i \in [k]}$ by choosing uniformly at random from the set of $2^k - 1$ not-all-1 vectors (ie. $\{0,1\}^k \setminus \{0^k\}$).

Definition 5.2. *For any $t \geq 0$ let $\mathcal{D}_t$ be a distribution given by the following experiment on input $H_n = G_n \cup T_n$.*

1. *Choose an even charge vector $\vec{c} \in \{0,1\}^n$ uniformly at random.*
2. *Choose $\beta \in Sol(H_n, \vec{c})$ uniformly at random.*
3. *For each $e \in G_n$, select the values for the vector $(y_e)_{i \in [k]}$ from $\mathcal{L}_k(\beta_e)$ and for each $e \in T_n$, set $y_e^i = \beta_e$ for all $i \in [k]$.*
4. *Select a random subset $U \subseteq [n]$ of $2t$ vertices and produce charge vector $\vec{c}^U$ from $\vec{c}$ by toggling all bits c_v for $v \in U$.*
5. *Return the pair $(\alpha, \vec{c}^U)$ where α is the boolean assignment to the variables y_e^i, $i \in [k]$, $e \in H_n$.*

Reduction from EVENCHARGE **to** ODDCHARGE

Lemma 5.1. *Let n be given, and let Δ be the maximum degree of a vertex in H_n. Suppose that Π_{odd} is a randomized k-party NOF protocol for $\text{ODDCHARGE}^k(H_n)$ that produces a vertex with probability at least $1 - \epsilon$, is correct whenever it produces a vertex, and uses at most s bits of communication. Then there is a randomized k-party NOF protocol Π_{even} for $\text{EVENCHARGE}^k(H_n)$ that uses $s + \Delta$ bits of communication and has the following performance:*

$$\begin{aligned} \Pr_{(\alpha,\vec{c}) \in \mathcal{D}_0}[\Pi_{even}(\alpha, \vec{c}) = \text{true}] &= 1 \\ \Pr_{(\alpha,\vec{c}) \in \mathcal{D}_t}[\Pi_{even}(\alpha, \vec{c}) \in \mathit{Err}(\alpha, \vec{c})] &\geq 2/3 - \epsilon, \text{ for } t \geq 1. \end{aligned}$$

Proof. Let Π_{odd} be a protocol for $\text{ODDCHARGE}^k(H_n)$. We give a protocol Π_{even} for $\text{EVENCHARGE}^k(H_n)$. On input $(\alpha, \vec{c})$ and random public string r: Using r, choose a random vertex $v \in [n]$. Check whether the parity equation associated with vertex v is satisfied by α using at most $\Delta(G)$ bits of communication. (This can be done by having Player 1 broadcast y_e^2 for each $e \ni v$, and then having Player 2 compute whether the constraint at v is obeyed or violated.) If it is not, return v. Otherwise, create an odd charge vector, $\vec{c}^{\{v\}}$, which is just like $\vec{c}$ except that the value of c_v is replaced by $1 - c_v$. Now run Π_{odd} on input $(\vec{c}^{\{v\}}, \alpha)$. If Π_{odd} returns the planted error v or if Π_{odd} does not return a value then return "true"; if Π_{odd} returns $u \neq v$, output u.

Suppose that $(\alpha, \vec{c}) \in \mathcal{D}_0$. Then α satisfies all charges specified by $\vec{c}$, so when Π_{odd} returns a vertex the above protocol must output "true" because Π_{odd} has one-sided error–that is, Π_{odd} will only return a vertex u when there is an error on the parity equation associated with u. Now suppose that $(\alpha, \vec{c}) \in \mathcal{D}_t$ so exactly $2t$

parity equations are violated. If the parity constraint about the vertex v that is not satisfied, then the protocol detects this and correctly reports the location of the error. The remaining case is when the parity constraint at v is satisfied, and in this case we call Π_{odd} on a pair $(\alpha, \vec{c}^{\{v\}})$ where exactly $2t+1$ parity equations are violated.

We show the probability bound by conditioning separately on the events $\text{Err}(\alpha, \vec{c}^{\{v\}}) = T$ for each $T \in \binom{[n]}{2t+1}$. Because the events $\text{Err}(\alpha, \vec{c'}) = T$ partition the probability space, this proves the claim. By symmetry, for $T \in \binom{[n]}{2t+1}$ and *any* function g whose range is a subset of T, we have that $\Pr_{\alpha,\vec{c},v}[g(\alpha, \vec{c}^{\{v\}}) = v \mid \text{Err}(\alpha, \vec{c}^{\{v\}}) = T] = 1/(2t+1)$ since it is equally likely for $\vec{c'} = \vec{c}^{\{v\}}$ to be generated as $\vec{c}^{\{u\}}$ for any $u \in T$. Thus we obtain:

$$\begin{aligned}
&\Pr_{\alpha,\vec{c},v}[\Pi_{even}(\alpha, \vec{c}^{\{v\}}) \text{ errs} \mid \text{Err}(\alpha, \vec{c}^{\{v\}}) = T] \\
&= \Pr_{\alpha,\vec{c},v}[\Pi_{odd}(\alpha, \vec{c}^{\{v\}}) = v \text{ or } \Pi_{odd}(\alpha, \vec{c}^{\{v\}}) \text{ is not defined } \mid \text{Err}(\alpha, \vec{c}^{\{v\}}) = T] \\
&\leq 1/(2t+1) + \epsilon \;\leq\; 1/3 + \epsilon
\end{aligned}$$

for $t \geq 1$. □

Reduction from Zero/One Set Disjointness to EVENCHARGE**:** We now show how to use a k-party NOF communication complexity protocol Π_{even} for $\text{EVENCHARGE}^k(H_n)$ as guaranteed by Lemma 5.1 to produce a k-party NOF protocol for the zero/one set disjointness problem which uses the following definition. In this reduction, we place the set-disjointness variables on the variables labeling some randomly chosen paths in the graph G_n. For the purposes of analyzing the distribution, the paths are chosen to be of length $l = \lceil \frac{\log n}{\log \log n} \rceil$ where $c_1 > 0$ is a constant. The constant c_1 is determined by Proposition 6.6. This is necessary for the proof of Lemma 5.3. For a more thorough discussion of this choice, see Section 6.2.

Definition 5.3. *Let $P_l^{(m)}$ be the set of all sequences of m vertex-disjoint length l paths in G_n.*

Lemma 5.2. *Let $m = n^{1/3}/\log n$. For sufficiently large n and for any even charge vector $\vec{c}$, if there is a probabilistic k-party NOF communication complexity protocol, Π_{even} for $\text{EVENCHARGE}^k(H_n)$ using s bits, satisfying the conditions in Lemma 5.1 for $\mathcal{D}_0$ and $\mathcal{D}_1$, then there is a randomized $(0, 1/3+\epsilon+o(1))$ error k-party NOF communication complexity protocol Π_{01disj} for zero/one set disjointness on input $\vec{x} \in (\{0,1\}^m)^k$ that uses s bits of communication.*

Proof. Let $\vec{x}$ be an instance of zero/one set disjointness. Protocol Π_{01disj} will call Π_{even} on the graph H_n, on a pair $(\alpha, \vec{c})$ chosen according to the following distribution/experiment:

1. On input $\vec{x}$ with public coins r:
   (a) Using public coins r, choose a random even charge vector $\vec{c} \in \{0,1\}^n$.
   (b) Using public coins r, choose a sequence of m vertex-disjoint length l paths, $p_1, \ldots p_m$ uniformly at random from $P_l^{(m)}$.
   (c) Using the public coins r, choose $\beta \in Sol(H_n - \bigcup_{j=1}^m p_j, \vec{c})$
2. For all edges $e \in H_n$, all players other than player i compute α_e^i as follows:
   (a) If $e \in p_j$ for $j \in [m]$, set $\alpha_e^i = x_{i,j}$

(b) If $e \in G_n$ and $e \notin \bigcup_{j=1}^{m} p_j$, choose the vector $\alpha_e^1 \ldots \alpha_e^k$ according to the distribution $\mathcal{L}_k(\beta_e)$.

(c) For the remaining edges $e \in T_n$, set all variables α_e^i for $i \in [k]$ equal to β_e.

3. Return $(\alpha, \vec{c})$

We write $\mathcal{R}(\vec{x})$ to denote the distribution on assignment/charge pairs produced by reduction Π_{01disj} when given an input $\vec{x}$. The following lemma shows that when $|\cap \vec{x}| = 1$, although $\mathcal{R}(\vec{x})$ is not the same as $\mathcal{D}_1$, $\mathcal{R}(\vec{x})$ is close to the distribution $\mathcal{D}_1$ in the ℓ_1 norm. This is the main technical lemma in the proof. The proof of this lemma can be found in the next section.

Lemma 5.3. *Let $\vec{x} \in (\{0,1\}^m)^k$ and $|\cap \vec{x}| = 1$. Then $||\mathcal{R}(\vec{x}) - \mathcal{D}_1||_1$ is $o(1)$.*

The protocol Π_{01disj} will output 0 if Π_{even} returns "true" and 1 otherwise. If $\cap\vec{x} = \emptyset$, by the above construction, the support of $\mathcal{R}(\vec{x})$ is contained in that of $\mathcal{D}_0$ and thus on $\mathcal{R}(\vec{x})$, Π_{even} must answer "true" and the vector $\vec{x}$ is correctly identified as being disjoint. In the case that $\cap\vec{x}$ contains exactly one element, $\Pr[\Pi_{01disj}(\vec{x})) = 0] \geq 2/3 - \epsilon - o(1)$. This completes the proof of Lemma 5.2. □

Reduction from Set disjointness to Zero/One Set disjointness

Lemma 5.4. *If there is an $(0, \epsilon)$ randomized NOF protocol for the k-party zero-one set-disjointness problem that uses s bits of communication where ϵ is a constant < 1, then there is a $(0, \frac{1}{3})$ randomized NOF protocol for the k-party set-disjointness problem that uses $O(s \log n)$ bits of communication.*

Naturally, our starting point is the well-known result of Valiant and Vazirani [28].

Lemma 5.5 (Valiant-Vazirani)**.** *Let a be a positive integer. Fix a nonempty $S \subseteq \{0,1\}^a$, and choose $w_1, \ldots w_a \in \{0,1\}^a$ independently and uniformly. With probability at least $1/4$, there exists $j \in \{0, \ldots, a\}$ so that $|\{x \in S \mid \forall i \leq j,\ x \cdot w_i = 0\}| = 1$.*

Proof of Lemma 5.4. Let Π be the protocol for the promise problem. Set $a = \lceil \log n \rceil$. Using public coins, independently and uniformly choose $w_1, \ldots w_l \in \{0,1\}^a$. For $j \in \{0, \ldots a\}$, the players run the protocol Π, using the following rule for evaluating the input $x_{i,r}$ for $i \in [k], r \in [m]$: interpret r as a vector in $\{0,1\}^a$, and replace the value of $x_{i,r}$ by zero if for some $j' \leq j$, $w_{j'} \cdot r \neq 0$, and use the value $x_{i,r}$ if for all $j' \leq j$, $w_{j'} \cdot r = 0$. If the protocol Π returns 1, the players halt and output 1, otherwise, the players proceed to round $j + 1$. If no intersection is found after all $a + 1$ rounds, the players announce that the inputs are disjoint.

Clearly, this protocol uses $O(s \log n)$ bits of communication, and by the 0-error property of Π on disjoint inputs, it never outputs 1 when the inputs are disjoint. When the inputs are non-disjoint, the Valiant-Vazirani construction ensures that with probability at least $1/4$, at some round j the protocol Π is used on an input with a unique intersection, and therefore, conditioned on this event, the correct answer is returned with probability at least $1 - \epsilon$. Therefore, the correct answer is returned with probability at least $\frac{1}{4} - \frac{\epsilon}{4}$. Because ϵ is bounded away from 1 and the error is one-sided, a constant number of repetitions decreases the probability of error to $1/3$. □

Combining the reductions to prove Theorem 4.1

Proof. (of Theorem 4.1) By Theorem 3.5 and the definition of $\text{ODDCHARGE}^k(H_n)$, if for every $\vec{c} \in \{0,1\}^n$ there is tree-like **Th(k-1)** refutation of $TS^k(H_n, \vec{c})$ of size at most S, then there is a $1/n$-error randomized k-party NOF communication complexity protocol for $\text{ODDCHARGE}^k(H_n)$ in which at most $O(\log^3 S)$ bits are communicated. By communicating the values of the edges incident to the vertex to be output by this $\text{ODDCHARGE}^k(H_n)$ protocol, the players can check that this vertex is indeed in error and not produce a vertex otherwise. This gives a 0-error protocol that outputs the correct answer with probability at least $1 - 1/n$. By Lemma 5.1 this yields a randomized 0-error k-party NOF protocol Π_{even} for $\text{EVENCHARGE}^k(H_n)$ that uses $O(\log^3 S + \log n)$ bits, produces the correct answer for all inputs in the support of $\mathcal{D}_0$ and for inputs randomly chosen according to $\mathcal{D}_1$ produces a correct answer with probability at least $2/3 - 1/n$. Applying Lemma 5.2 this yields a $(0, 1/3 + 1/n + o(1))$-error k-party protocol for zero/one set disjointness on $(\{0,1\}^m)^k$ also of complexity $O(\log^3 S + \log n)$. Finally applying Lemma 5.4 yields an error $1/3$ randomized k-party NOF protocol for $\text{DISJ}_{k,m}$ of complexity $O(\log^3 S \log n + \log^2 n)$ bits in total. The case for polynomially-bounded coefficients is obtained by applying a similar reduction using the other part of Theorem 3.5. □

6 Proximity of distributions $\mathcal{D}_1$ and $\mathcal{R}(\vec{x})$ when $|\cap \vec{x}| = 1$

In this section we prove Lemma 5.3: that for $|\cap \vec{x}| = 1$ the distributions $\mathcal{R}(\vec{x})$ and $\mathcal{D}_1$ are close in the ℓ_1 norm. Let $\mu_{\mathcal{D}_1}$ and $\mu_{\mathcal{R}(\vec{x})}$ be their associated probability measures. We will show that for all but a set of $(\alpha, \vec{c})$ with $\mu_{\mathcal{D}_1}$ measure $o(1)$, $\mu_{\mathcal{D}_1}(\alpha, \vec{c}) = (1 \pm o(1))\mu_{\mathcal{R}(\vec{x})}(\alpha, \vec{c})$.

Given an instance of the set disjointness variables, $\vec{x} = (\{0,1\}^m)^k$, for $j \in [m]$ we say that the *color* of j is the tuple $(x_{1,j}, \ldots, x_{k,j}) \in \{0,1\}^k$. By construction, the assignment $\mathcal{R}(\vec{x})$ has color $(x_{1,j}, \ldots, x_{k,j})$ on each edge of the path p_j.

Definition 6.1. *Given an ordered sequence of paths $\vec{p} \in P_l^{(m)}$, an $\vec{x} \in (\{0,1\}^m)^k$, and an assignment α, write $\chi(\alpha_{\vec{p}}) = \vec{x}$ if and only if every edge on path p_j has color $(x_{1,j}, \ldots, x_{k,j})$ for every $j \in [m]$.*

We first observe that for any $(\alpha, \vec{c})$ with $|\text{Err}(\alpha, \vec{c})| = 2$ the probability $\mu_{\mathcal{D}_1}(\alpha, \vec{c})$ depends only on the number of edges $e \in G_n$ having color 1^k in α.

Definition 6.2. *Let $\phi(a, b) = 2^{-a}(2^k - 1)^{-(a-b)}$.*

Lemma 6.1. *For any $(\alpha, \vec{c})$ with $|Err(\alpha, \vec{c})| = 2$ and $m_1 = |\{e \in E(G_n) \mid \alpha_e = 1^k\}|$,*

$$\mu_{\mathcal{D}_1}(\alpha, \vec{c}) = \frac{\phi(|E(G_n)|, m_1)}{2^{n-1}\binom{n}{2}}.$$

Proof. Let $U = \text{Err}(\alpha, \vec{c})$. The probability under $\mathcal{D}_1$ that U is chosen to be flipped is $1/\binom{n}{2}$ and, given U, the probability that the charge vector $\vec{c}$ is produced by the experiment is simply the probability that $\vec{c}^U$ is generated by the uniform distribution over all 2^{n-1} many even charge vectors, that is, $2^{-(n-1)}$. Conditioned on the event that U is chosen to be flipped, and that the charge vector is $\vec{c}$, the chance that α labels the edges for the randomly selected element of $Sol(H_n, \vec{c})$ is $2^{-|E(G_n)|}(2^k - 1)^{-(|E(G_n)| - m_1)} = \phi(|E(G_n)|, m_1)$. □

Definition 6.3. *For $U \subset V$ with $|U| = 2$ let $P_l^{(m)}(U)$ be the set of all elements of $P_l^{(m)}$ that have a path whose endpoints are U.*

Now consider the measure $\mu_{\mathcal{R}(\vec{x})}(\alpha, \vec{c})$. Let $\{i\} = \cap\vec{x} \subseteq [n]$, $U = \text{Err}(\alpha, \vec{c})$ with $|U| = 2$, and $m_1 = |\{e \in E(G_n) \mid \alpha_e = 1^k\}|$. By the definition of $\mathcal{R}(\vec{x})$,

$$\begin{aligned}\mu_{\mathcal{R}(\vec{x})}(\alpha, \vec{c}) &= \Pr_{\vec{p}\in P_l^{(m)}}[\text{Ends}(p_i) = \text{Err}(\alpha, \vec{c}) \wedge \chi(\alpha_{\vec{p}}) = \vec{x}] \\ &\quad \cdot \Pr_{\substack{\vec{c'}\in\{0,1\}^n \\ \alpha'\in\mathcal{L}_k(Sol(H_n-\vec{p},\vec{c'}))}}[\alpha' = \alpha_{G_n-\vec{p}} \text{ and } \vec{c'} = \vec{c}] \\ &= \Pr_{\vec{p}\in P_l^{(m)}}[\text{Ends}(p_i) = \text{Err}(U)] \cdot \Pr_{\vec{p}\in P_l^{(m)}(U)}[\chi(\alpha_{\vec{p}}) = \vec{x}] \\ &\quad \cdot \phi(|E(G_n)| - ml, m_1 - l)/2^{n-1}.\end{aligned}$$

Observe that p_i is a uniformly chosen element of P_l and we can analyze the first term using the following property of random paths on LPS expanders proved as part of Lemma 6.9 in Section 6.2.2.

Lemma 6.2. *For $u \neq v \in V(G_n)$ and $l \geq c_1 \log n/\log\log n$, $\Pr_{p\in P_l}[$Ends$(p) = \{u, v\}] = (1 \pm o(1))/\binom{n}{2}$.*

Thus

$$\begin{aligned}\mu_{\mathcal{R}(\vec{x})}(\alpha, \vec{c}) &= (1 \pm o(1))\frac{\phi(|E(G_n)| - ml, m_1 - l)}{\binom{n}{2}2^{n-1}} \cdot \Pr_{\vec{p}\in P_l^{(m)}(U)}[\chi(\alpha_{\vec{p}}) = \vec{x}] \\ &= (1 \pm o(1))\frac{\mu_{\mathcal{D}_1}(\alpha, \vec{c})}{\phi(ml, l)} \cdot \Pr_{\vec{p}\in P_l^{(m)}(U)}[\chi(\alpha_{\vec{p}}) = \vec{x}].\end{aligned}$$

It follows that we will obtain the desired result if we can show that for all but a $o(1)$ measure of $(\alpha, \vec{c})$ under $\mu_{\mathcal{D}_1}$,

$$\Pr_{\vec{p}\in P_l^{(m)}(U)}[\chi(\alpha_{\vec{p}}) = \vec{x}] = (1 \pm o(1))\phi(ml, l) = (1 \pm o(1))2^{-ml}(2^k - 1)^{-(m-1)l}$$

where $U = \text{Err}(\alpha, \vec{c})$. In the case that this happens, we say that $(\alpha, \vec{c})$ is *well-distributed for* $\vec{x}$.

Using the second moment method we prove the following lemma which shows that for all but a $o(1)$ measure of $(\alpha, \vec{c})$ under $\mu_{\mathcal{D}_1}$, $(\alpha, \vec{c})$ is indeed well-distributed for $\vec{x}$. The detailed proof is given in Section 6.1.

Lemma 6.3. *Let $m \leq n^{1/3}/\log n$ and $l = 2\lceil c_1 \log n/\log\log n\rceil$ and $\vec{x} \in (\{0,1\}^m)^k$ with $|\cap\vec{x}| = 1$. For almost all $U \subset [n]$ with $|U| = 2$,*

$$\Pr_{(\alpha,\vec{c})\in\mathcal{D}_1}[(\alpha, \vec{c}) \text{ is well-distributed for } \vec{x} \mid \text{Err}(\alpha, \vec{c}) = U] = 1 - o(1)$$

Lemma 5.3 follows from this almost immediately.

Proof of Lemma 5.3. Let $\vec{x} \in (\{0,1\}^m)^k$ and $|\cap\vec{x}| = 1$. By Lemma 6.3 and the preceding argument, for all $U \in \binom{[n]}{2}$ except for a set B that forms an $o(1)$ fraction of $\binom{[n]}{2}$,

$$\Pr_{(\alpha,\vec{c})\in\mathcal{D}_1}[\mu_{\mathcal{R}(\vec{x})}(\alpha, \vec{c}) = (1 \pm o(1))\mu_{\mathcal{D}_1}(\alpha, \vec{c}) \mid \text{Err}(\alpha, \vec{c}) = U] = 1 - o(1).$$

By Lemma 6.2, $\Pr_{(\alpha,\vec{c})\in\mathcal{D}_1}[\text{Err}(\alpha, \vec{c}) \in B] = o(1)$. Therefore by summing over distinct choices of U, we obtain that with probability $1-o(1)$ over $(\alpha, \vec{c}) \in \mathcal{D}_1$, $\mu_{\mathcal{R}(\vec{x})}(\alpha, \vec{c}) = (1\pm o(1))\mu_{\mathcal{D}_1}(\alpha, \vec{c})$. This is equivalent to the desired conclusion that $||\mathcal{D}_1 - \mathcal{R}(\vec{x})||_1$ is $o(1)$. □

6.1 Most $(\alpha, \vec{c})$ are well-distributed

In this section we use the second moment method to prove Lemma 6.3. For this purpose we will need the following property of the LPS expander graphs G_n, proved in Section 6.2 which will allow us to show that the correlations considered in the second moment method are low.

Definition 6.4. *For $\vec{p}, \vec{q} \in P_l^{(m)}$ we write $\vec{p} \sim_s \vec{q}$ when $\vec{p}$ and $\vec{q}$ share exactly s edges. Let $\gamma > 0$ be a positive real number. We say that $U \subset V(G_n)$ is* γ-nice *if for all $s \geq 0$, $\Pr_{\vec{p},\vec{q} \in P_l^{(m)}(U)}[\vec{p} \sim_s \vec{q}] \leq \gamma^s$.*

Theorem 6.4. *(proved in § 6.2) Suppose that $m \leq n^{1/3}/\log n$ and $l = 2\lceil c_1 \log n / \log\log n \rceil$. There are constants $c > 0$ and c' such that for all but a $o(1)$ fraction of sets $U = \{u, v\} \subset V(G_n)$, for all $\vec{q} \in P_l^{(m)}(U)$ and every integer $s \geq 0$,*
$\Pr_{\vec{p} \in P_l^{(m)}(U)}[\vec{p} \sim_s \vec{q}] \leq (c'/(\log\log n)^{1/4} + (\log n)^{-c})^s$,
i.e. almost every $U \in V^{(2)}$ is $(c'/(\log\log n)^{1/4} + 1/\log^c n)$-nice.

We now use this in our application of the second moment method to prove that most $(\alpha, \vec{c})$ pairs are well-distributed:

Lemma 6.5. *Let $m \leq n^{1/3}/\log n$ and $l = 2\lceil c_1 \log n / \log\log n \rceil$, $\vec{x} \in (\{0,1\}^m)^k$ with $|\cap \vec{x}| = 1$, and $|U| = 2$. If U is γ-nice with $\gamma = o(2^{-k})$, then*

$$\Pr_{(\alpha,\vec{c}) \in \mathcal{D}_1}[(\alpha, \vec{c}) \text{ is well-distributed for } \vec{x} \mid \mathit{Err}(\alpha, \vec{c}) = U] = 1 - o(1)$$

Proof. For each $\vec{p} \in P_l^{(m)}(U)$, let $X_{\vec{p}}$ denote the indicator variable for the event that $\chi(\alpha_{\vec{p}}) = \vec{x}$.

We now calculate $E_{(\alpha,\vec{c}) \in \mathcal{D}_1}[X_{\vec{p}}]$. For $(\alpha, \vec{c})$ chosen according to $\mathcal{D}_1$, the assignment $\alpha_{\vec{p}}$ is distributed according to $(\mathcal{L}_k)^{ml}$; therefore, since for $\chi(\alpha_{\vec{p}})$ to equal $\vec{x}$, $\alpha_{\vec{p}}$ must have precisely l edges whose color is 1^k and $l(m-1)$ edges whose color is a lift of label 0,

$$E_{(\alpha,\vec{c}) \in \mathcal{D}_1}[X_{\vec{p}}] = \Pr_{(\alpha,\vec{c}) \in \mathcal{D}_1}[X_{\vec{p}} = 1] = \phi(ml, l) = 2^{-ml}(2^k - 1)^{-(m-1)l}.$$

Let $X = \sum_{\vec{p} \in P_l^{(m)}(U)} X_{\vec{p}}$. X is the random variable denoting the number of sequences $\vec{p} \in P_l^{(m)}(U)$ for which $\chi(\alpha_{\vec{p}}) = \vec{x}$. By the linearity of expectation, $E_{(\alpha,\vec{c})}[X] = \phi(ml, l) \cdot |P_l^{(m)}(U)|$.

We use the second moment method to show that X is concentrated near its expectation. For $\vec{p}, \vec{q} \in P_l^{(m)}(U)$, the random variables $X_{\vec{p}}$ and $X_{\vec{q}}$ are correlated if and only if $\vec{p}$ and $\vec{q}$ share an edge. Because U is γ-nice $\Pr_{\vec{p},\vec{q} \in P_l^{(m)}(U)}[\vec{p} \sim_s \vec{q}] \leq \gamma^s$.

When $X_{\vec{p}} = 1$, the colors of all edges of $\vec{p}$ are determined. Therefore given $X_{\vec{p}} = 1$, if $\vec{p} \sim \vec{q}$, either some edge that $\vec{p}$ and $\vec{q}$ share ensures that $X_{\vec{q}} = 0$, or the probability that $X_{\vec{p}} = X_{\vec{q}} = 1$ is non-zero. In the latter case consider $G' = \bigcup_{i=1}^m (p_i \cup q_i)$ which contains $2ml - s$ edges. Because the marginal distribution of α to the edges of G' independently assigns each e of G' a label using the distribution $\mathcal{L}_k$ (per Definition 5.1), we have that the probability that $\chi(\alpha_{\vec{p}}) = \chi(\alpha_{\vec{q}}) = \vec{x}$ is larger than $[\phi(ml, l)]^2$ by a factor of either 2 or $2(2^k - 1)$ per shared edge depending on whether that edge has label 1 or 0.

Let $D = \sum_{\vec{p} \sim_s \vec{q}} \Pr_{(\alpha,\vec{c})}[X_{\vec{p}} = X_{\vec{q}} = 1]$.

$$\begin{aligned}
D &= \sum_{s=1}^{ml} \sum_{\vec{p}\sim_s\vec{q}} \Pr_{(\alpha,\vec{c})}[X_{\vec{p}} = X_{\vec{q}} = 1] \\
&\leq \sum_{s=1}^{ml} \sum_{\vec{p}\sim_s\vec{q}} (2(2^k-1))^s \Pr_{(\alpha,\vec{c})\in\mathcal{D}_1}[X_{\vec{p}} = 1] \Pr_{(\alpha,\vec{c})\in\mathcal{D}_1}[X_{\vec{q}} = 1] \\
&= \sum_{s=1}^{ml} \sum_{\vec{p}\sim_s\vec{q}} (2(2^k-1))^s (\phi(ml,l))^2 \\
&= \sum_{s=1}^{ml} |P_l^{(m)}(U)|^2 \Pr_{\vec{p},\vec{q}\in P_l^{(m)}(U)}[\vec{p} \sim_s \vec{q}] (2(2^k-1))^s (\phi(ml,l))^2 \\
&= (|P_l^{(m)}(U)| \cdot \phi(ml,l))^2 \sum_{s=1}^{lm} \Pr_{\vec{p},\vec{q}\in P_l^{(m)}(U)}[\vec{p} \sim_s \vec{q}] (2(2^k-1))^s \\
&= (E_{(\alpha,\vec{c})\in\mathcal{D}_1}[X])^2 \sum_{s=1}^{ml} \Pr_{\vec{p},\vec{q}\in P_l^{(m)}(U)}[\vec{p} \sim_s \vec{q}] (2(2^k-1))^s \\
&\leq (E_{(\alpha,\vec{c})\in\mathcal{D}_1}[X])^2 \sum_{s=1}^{ml} \gamma^s (2(2^k-1))^s.
\end{aligned}$$

Since $\gamma = o(2^{-k})$ by hypothesis, $\sum_{s=1}^{\infty} \gamma^s (2(2^k-1))^s$ is $o(1)$ and thus D is $o((E_{(\alpha,\vec{c})\in\mathcal{D}_1}[X])^2)$. Therefore, $E_{(\alpha,\vec{c})}(X^2) = D + E_{(\alpha,\vec{c})}(X) = o((E_{(\alpha,\vec{c})}[X])^2) + E_{(\alpha,\vec{c})}(X)$ and by the second moment method,

$$\Pr_{(\alpha,\vec{c})\in\mathcal{D}_1}[|X - E_{(\alpha,\vec{c})\in\mathcal{D}_1}(X)| \geq \epsilon E_{(\alpha,\vec{c})\in\mathcal{D}_1}(X)] \leq \frac{D + E_{(\alpha,\vec{c})\in\mathcal{D}_1}[X]}{\epsilon^2 E_{(\alpha,\vec{c})}(X)^2} = o(1).$$

By choosing ϵ as an appropriate function that is $o(1)$, we obtain that with probability $1 - o(1)$ in the choice of $(\alpha, \vec{c}) \in \mathcal{D}_1$, $X = (1 \pm o(1))\phi(ml,l) \cdot |P_l^{(m)}(U)|$ and therefore with probability $1 - o(1)$ in $(\alpha, \vec{c})$, $\Pr_{\vec{p}\in P_l^{(m)}(U)}[\chi(\alpha_{\vec{p}}) = \vec{x}] = (1 \pm o(1))\phi(ml,l)$ and thus $(\alpha, \vec{c})$ is well-distributed for $\vec{x}$. □

Proof of Lemma 6.3. Let $\vec{x} \in (\{0,1\}^m)^k$ and $|\cap \vec{x}| = 1$. By Theorem 6.4 there is a $\delta > 0$ so that for all but a $o(1)$ fraction of sets $U \subset V(G_n)$ with $|U| = 2$, U is γ-nice for $\gamma = c''/(\log\log n)^{1/4}$ for some constant c'' and γ is $o(2^{-k})$. Therefore, $\Pr_{(\alpha,\vec{c})\in\mathcal{D}_1}[\text{Err}(\alpha,\vec{c}) \text{ is } \gamma\text{-nice}] = 1 - o(1)$ and by Lemma 6.5, $\Pr_{(\alpha,\vec{c})\in\mathcal{D}_1}[(\alpha,\vec{c}) \text{ is well-distributed for } \vec{x} \mid \text{Err}(\alpha,\vec{c}) = U] = 1 - o(1)$. □

6.2 Graph Theoretic Properties of LPS Expanders

6.2.1 The Lubotzky-Phillips-Sarnak Expanders

In the analysis of $R(\vec{x})$, we want the endpoints of the random paths in G_n to be almost uniformly distributed. We base our proof of this upon the fact that the endpoints of random walks in expander graphs are almost uniformly distributed (Proposition 6.6). Since a walk is allowed to repeat vertices but a path does not repeat vertices, the length of the walk is too large with respect to the degree, it is very likely that a random walk will not be a path. To transfer Proposition 6.6 from walks to paths, we use a graph in which random walks of

length l will have their endpoints almost uniformly distributed but the walks are short enough with respect to the degree so that a random walk very likely to be a path. We use the Lubotzky-Phillips-Sarnak expanders. The crucial properties of the expander graphs G_n constructed in [22] that we need are:

1. G_n is regular of degree $\Delta = \Theta(\log n)$.
2. G_n is connected and non-bipartite.
3. The second eigenvalue of G_n is $O(\sqrt{\log n})$.
4. The girth of G_n is $\Omega(\log n / \log\log n)$.

A walk in G_n is chosen by selecting a start node and repeatedly following one of the Δ edges adjacent to the current node.

Proposition 6.6. *There exists $c_1 > 0$ so that for every $u, v \in V(G_n)$, a random walk in G_n of length $l \geq c_1 \log n / \log\log n$ starting at u ends at vertex v with probability at least $1/n - 1/n^2$ and at most $1/n + 1/n^2$.*

We consider random walks and random paths in the G_n graphs of a fixed length $l = l(n) = 2\lceil c_1 \log n / \log\log n \rceil$ that is twice the minimum length specified in Proposition 6.6 so that their midpoints are nearly uniformly distributed.

6.2.2 Approximating Paths by Walks

Remark 1. *In principle one might replace disjoint paths in the definition of Π_{01disj} by disjoint walks of the same length, conditioned on each having distinct endpoints. However, in that case it would be overwhelmingly likely that many walks will repeat edges and therefore, as graphs, they would contain different numbers of edges. This would significantly complicate the second moment argument of Lemma 6.5.*

We show that because G_n is expanding and has high girth, random walks in G_n not only mix well but they are paths almost surely as well. We state some folklore properties of random walks and observe how they translate into properties of random paths.

For $v \in V(G_n)$, let $W_l(v)$ be the set of all Δ^l walks of length l in G_n starting at v and $P_l(v)$ be the set of all paths of length l in G_n with one endpoint v. Let $\mu_{W_l(v)}$ be the measure given by a uniform distribution over $W_l(v)$ and $\mu_{P_l(v)}$ be the measure given by a uniform distribution over $P_l(v)$.

Lemma 6.7. *There exists a universal constant c_3 so that for every $v \in V(G_n)$ and for each path $p \in P_l(v)$, $(1 - c_3/\log\log n)\mu_{P_l(v)}(p) \leq \mu_{W_l(v)}(p) \leq \mu_{P_l(v)}(p)$. Moreover, for w uniformly chosen from $W_l(v)$ the probability that w is not a path is at most $c_3/\log\log n$.*

Proof. Observe that every $p \in P_l(v)$ has equal measure under $\mu_{W_l(v)}$ so $\mu_{W_l(v)}(p) \leq \mu_{P_l(v)}(p)$ and, moreover, $\mu_{W_l(v)}(p) = \mu_{P_l(v)}(p)\mu_{W_l(v)}(P_l(v))$.

Set $g = girth(G_n)$. By the properties of G_n, $g \geq c_0 \log n / \log\log n$ for some constant $c_0 > 0$ and its degree $\Delta \geq c_2 \log n$ for some constant $c_2 \geq 0$. Notice that for any walk w of length l each vertex in w can have at most $l/(g-3)$ many neighbors also in w. (If u is a vertex in w that has two neighbors u' and u'' in G_n within distance $g-3$ on w then there is a cycle of length $g-1$ in $w \cup \{(u,u'),(u,u'')\}$ which is a

subgraph of G_n.) Therefore

$$\begin{aligned}\mu_{W_l(v)}(P_l(v)) &\geq \left(\frac{\Delta - l/(g-3)}{\Delta}\right)^l \geq 1 - \frac{l^2}{\Delta(g-3)} \\ &\geq 1 - \frac{2\lceil c_1 \log n/\log\log n\rceil^2}{c_2 \log n \cdot (c_0 \log n/\log\log n - 3)} \geq 1 - c_3/\log\log n\end{aligned}$$

for some constant c_3. □

The following are folklore properties of random walks in G_n.

Proposition 6.8. *Let W_l be the set of all walks of length l in G_n.*

1. *For each $v \in V(G_n)$, $\Pr_{w\in W_l}[v \in V(w)] \leq (l+1)/n$.*
2. *For each $u \neq v \in V(G_n)$, $\Pr_{w\in W_l}[Ends(w) = \{u,v\}] = (1 \pm 2/n)/\binom{n}{2}$.*

Proof. There is a sequence of $l+1$ vertices (not necessarily distinct) on each walk w in W_l and precisely Δ^l walks in which v is the i-th vertex in w. Therefore, in total there are at most $(l+1)\Delta^l$ walks with $v \in V(w)$. (This is an overcount since v may appear more than once in w.) Since there are precisely $n\Delta^l$ random walks in G_n of length l, $\Pr_{w\in W_l}[v \in V(w)] \leq (l+1)/n$.

By Proposition 6.6 the chance that a particular pair of distinct vertices $\{u,v\}$ appear as endpoints of w is $\frac{2}{n}(1/n \pm 1/n^2)$ which is $(1 \pm 2/n)/\binom{n}{2}$. □

We obtain the following easy corollary which includes a proof of Lemma 6.2.

Lemma 6.9. *Let P_l be the set of all paths in G_n of length l.*

1. *Let $V' \subseteq V(G_n)$. There exists a constant c so that*

$$\Pr_{p\in P_l}[V(p) \cap V' \neq \emptyset] \leq (1 + c/\log\log n)\frac{|V'|(l+1)}{n}.$$

2. *Let $u \neq v \in V(G_n)$. Then $\Pr_{p\in P_l}[Ends(p) = \{u,v\}] = (1 \pm o(1))/\binom{n}{2}$*

Proof. By Proposition 6.8, for w a randomly chosen walk of length l in G_n,

$$\Pr_{w\in W_l}[V(w) \cap V' \neq \emptyset] \leq \frac{|V'|(l+1)}{n},$$

and by Lemma 6.7, $\Pr_{w\in W_l}[w \text{ is a path}] \geq 1 - c_3/\log\log n$. The random distribution of paths p of length l in G_n is the same as the random distribution of walks w of length l in G_n conditioned on w being a path. Therefore

$$\begin{aligned}\Pr_{p\in P_l}[V \cap V(p) \neq \emptyset] &= \Pr_{w\in W_l}[V \cap V(w) \neq \emptyset \mid w \text{ is a path}] \\ &\leq \frac{|V|(l+1)}{(1 - c_3/\log\log n)n} \\ &\leq (1 + c/\log\log n)\frac{|V|(l+1)}{n},\end{aligned}$$

for some constant c.

For $u \neq v \in V(G_n)$, by Lemma 6.7 $\Pr_{p\in P_l}[\text{Ends}(p) = \{u,v\}]$ is within a $1 \pm o(1)$ factor of $\Pr_{w\in W_l}[\text{Ends}(w) = \{u,v\}]$ and by Proposition 6.8 the latter is $(1 \pm o(1))/\binom{n}{2}$ which yields the desired property.

□

6.2.3 The Proof of Theorem 6.4

In this subsection we prove Theorem 6.4. We will actually prove a slightly stronger result in which $\vec{q} \in P_l^{(m)}(U)$ is replaced by any subgraph of G_n with at most $m(l+1)$ vertices and maximum degree at most 2.

It will be convenient to consider sequences of length l paths P_l^m that are not necessarily vertex-disjoint. Let $\mu_{P_l^{(m)}}$ be the uniform measure on $P_l^{(m)}$ and $\mu_{P_l^m}$ be the uniform distribution on P_l^m.

Lemma 6.10. *Suppose that $m \leq n^{1/3}/\log n$ and $l = 2\lceil c_1 \log n/\log\log n\rceil$. For any $\vec{p} \in P_l^{(m)}$, $(1 - o(1))\mu_{P_l^{(m)}}(\vec{p}) \leq \mu_{P_l^m}(\vec{p}) \leq \mu_{P_l^{(m)}}(\vec{p})$.*

Proof. Conditioned on the paths in $\vec{p} \in P_l^m$ being vertex-disjoint $\mu_{P_l^m}$ is uniform over $P_l^{(m)}$. By Lemma 6.9, the probability that the i-th path shares a vertex with paths $p_1, \ldots, p_{i-1}$ is at most $(1 + c/\log\log n)(l+1)^2(m-1)/n \leq 2l^2m/n$ and the probability that the paths in P_l^m are not vertex-disjoint is at most $2l^2m^2/n \leq 1/n^{1/3}$. □

We first observe that if we only we required that $\vec{p} \in P_l^{(m)}$ rather $\vec{p} \in P_l^{(m)}(U)$ – i.e., we had no requirement that one path in $\vec{p}$ have its endpoints in U – then the exponentially-decaying bound on intersection size of Theorem 6.4 would be relatively easy.

Lemma 6.11. *Suppose that $m \leq n^{1/3}/\log n$ and $l = 2\lceil c_1 \log n/\log\log n\rceil$. There is some constant $c \geq 0$ such that for all subgraphs G' of G_n with at most $m(l+1)$ vertices and every integer $s \geq 0$,*

$$\Pr_{\vec{p} \in P_l^{(m)}}[|E(\cup\vec{p}) \cap E(G')| \geq s] \leq (\log n)^{-cs}.$$

Proof. For $\vec{p} \in P_l^{(m)}$, because each component of $\vec{p}$ is a path of length l, if $|E(\cup\vec{p}) \cap E(G')| \geq s$ then there are at least $\lceil s/l\rceil$ paths p_i in $\vec{p}$ that that share an edge (and therefore a vertex) with G'. By Lemma 6.9, the probability that a random p_i from P_l shares a vertex with G' is at most $(1 + c/\log\log n)(l+1)^2m/n < 2l^2m/n$. Therefore for elements of P_l^m, the probability that there are least $r = \lceil s/l\rceil$ such paths is at most $\binom{m}{r}(2l^2m/n)^r < (2l^2m^2/n)^r/2$. By Lemma 6.10, the probability that this happens for elements of $P_l^{(m)}$ is at most $(2l^2m^2/n)^r \leq n^{s/(3l)} = (\log n)^{-cs}$ for some constant $c > 0$. □

The major complication of the proof of Theorem 6.4 is the assumption that $\vec{p}$ contains a path with endpoints u and v for $U = \{u, v\}$, $u \neq v$. We base the analysis of paths with endpoints U on the analysis of walks with endpoints U. For some sets U, for example if u and v are adjacent in G_n, the distributions of random walks and random paths with endpoints U may not be close to each other.[3] We will see that for most choices of U, the probabilities under the two distributions are close to each other and this will be enough to obtain the bound required by Theorem 6.4.

Definition 6.5. *For $U = \{u, v\} \in V(G_n)$ let $W_l(U)$ be the set of all walks in G_n of length n that have endpoints U.*

Lemma 6.12. *There is a constant c_4 such that for all but at most a $c_4/\log\log n$ fraction of pairs $u \neq v \in V(G_n)$*

$$\Pr_{w \in W_l(\{u,v\})}[w \text{ is a path}] \geq 2/3.$$

[3] Even in these cases the distributions may be sufficiently close but we do not need to analyze them.

Proof. By Lemma 6.7,

$$\Pr_{w \in W_l}[w \text{ is not a path}] \leq c_3 / \log\log n.$$

Therefore by definition,

$$\sum_{u \neq v \in V(G_n)} \Pr_{w \in W_l}[\text{Ends}(w) = \{u, v\}] \Pr_{w \in W_l(\{u,v\})}[w \text{ is not a path}] \leq c_3 / \log\log n.$$

By Proposition 6.8, $\Pr_{w \in W_l}[\text{Ends}(w) = \{u, v\}] \geq (1 - 2/n)\binom{n}{2}^{-1}$ and thus

$$(1 - 2/n)\binom{n}{2}^{-1} \sum_{u \neq v \in V(G_n)} \Pr_{w \in W_l(\{u,v\})}[w \text{ is not a path}] \leq c_3 / \log\log n,$$

which says that the expected value

$$E_{u \neq v \in V(G_n)}(\Pr_{w \in W_l(\{u,v\})}[w \text{ is not a path}]) \leq \frac{c_3 / \log\log n}{(1 - 2/n)}.$$

We now apply Markov's inequality to obtain that the fraction of pairs $u \neq v \in V(G_n)$ for which $\Pr_{w \in W_l(\{u,v\})}[w \text{ is not a path}] \geq 1/3$, is at most $\frac{c_3 / \log\log n}{(1-2/n)/3} \leq c_4 / \log\log n$ for some constant c_4.

□

Bounding Intersection Size of Random Walks

Lemma 6.12 will allow us to use the following analysis involving a random walk with endpoints in U rather than a random path.

Lemma 6.13. *Let G' be a subgraph of G_n with the property that every vertex has degree at most d in G'. For fixed $v \in V(G_n)$,*

$$\Pr_{w \in W_l(v)}[|E(w) \cap E(G')| \geq s] \leq \binom{l}{s}\left(\frac{d}{\Delta}\right)^s.$$

Proof. There are at most $\binom{l}{s}$ many choices of steps in the random walk in which the first s shared edges can occur. Fix some such set of steps $S \subseteq [l]$. For each $i \in S$ a necessary condition for the i-th edge in the walk to lie in $E(G')$ is that the endpoint u after step $i - 1$ must lie in $V(G')$. Since $\deg_{G'}(u) \leq d$, given that $u \in V(G')$, the probability that the i-th edge lies in $E(G')$ is then at most d/Δ. That is, conditioned on a shared edge in each of the first j elements in S, the chance of a shared edge in the $j + 1$-st element in S is at most d/Δ because every vertex has degree at most d in G'. This yields a total probability at most $\binom{l}{s}(d/\Delta)^s$ as required. □

In order to analyze the random walks in $W_l(U)$ we need more than the result of Lemma 6.13 since it constrains only one endpoint of the random walk rather than both endpoints. We can view each half of a random walk in which both endpoints are constrained as two random walks of half the length with only one endpoint constrained. (Obviously, these two half-length walks are highly correlated.)

Lemma 6.14. *Let $l = 2\lceil c_1 \log n / \log\log n \rceil$. Let G' be a subgraph of G_n in which every vertex has degree at most d. For $u \neq v \in V(G_n)$,*

$$\Pr_{w \in W_l(\{u,v\})}[|E(w) \cap E(G')| \geq s] < \left(\frac{2dl}{\Delta}\right)^{s/2}.$$

Proof. Without loss of generality, walk $w \in W_l(\{u,v\})$ starts at u and ends at v. Let $l' = l/2$. Let $w = (w_u, w_v)$ where w_u and w_v each have length l'. We first observe that w_u is nearly uniformly distributed in $W_{l'}(u)$:

Let $w^* \in W_{l'}(u)$ and let v^* be the end of w^*.

$$\begin{aligned}
&\Pr_{w \in W_l(\{u,v\})}[w_u = w^* \mid w \text{ starts at } u] \\
&= \frac{\Pr_{w \in W_l(u)}[w_u = w^* \text{ and } w_v, \text{ starting at } v^*, \text{ ends at } v]}{\Pr_{w \in W_l(u)}[w \text{ ends at } v]} \\
&= \frac{\Pr_{w_u \in W_{l'}(u)}[w_u = w^*] \cdot \Pr_{w_v \in W_{l'}(v^*)}[w_v \text{ ends at } v]}{\Pr_{w \in W_l(u)}[w \text{ ends at } v]}
\end{aligned}$$

Clearly $\Pr_{w_u \in W_{l'}(u)}[w_u = w^*] = \Delta^{-l'} = \Delta^{-l/2}$ and since $l > l' \geq c_1 \log n / \log\log n$ by Proposition 6.6, both $\Pr_{w_v \in W_{l'}(v^*)}[w_v \text{ ends at } v]$ and $\Pr_{w \in W_l(u)}[w \text{ ends at } v]$ are $1/n \pm 1/n^2$ and thus

$$\Pr_{w \in W_l(\{u,v\})}[w_u = w^* \mid w \text{ starts at } u] = (1 \pm O(1/n))\Delta^{-l/2}.$$

Since G_n is a *regular* undirected graph, a length l random walk from u to v has the same distribution as a length l random walk from v to u. Thus by symmetry with the above argument, within a $1 \pm O(1/n)$ factor, w_v is distributed as a (nearly) uniform random walk of length l' starting at v.

Now if there are a total of s edges in common between w and G' then at least $\lceil s/2 \rceil$ must be shared between G' and one of the two halves of w, w_u and w_v. By Lemma 6.13 and the above argument each of these probabilities is at most $(1+O(1/n))(\frac{dl'}{\Delta})^{\lceil s/2 \rceil}$ and the total probability is at most $2(1+O(1/n))(\frac{dl}{2\Delta})^{\lceil s/2 \rceil} \leq (2\frac{dl}{\Delta})^{\lceil s/2 \rceil}$. □

Deriving the bound

Lemma 6.15. *Let $l = 2\lceil c_1 \log n / \log\log n \rceil$ and $m \leq n^{1/3}/\log n$. For any fixed subgraph G' of G_n with at most $m(l+1)$ vertices and maximum degree at most 2, and any set $U = \{u, v\} \subset V(G_n)$,*

$$\Pr_{(w,\vec{p})) \in W_l(U) \times P_l^{m-1}}[|(E(w) \cup E(\vec{p})) \cap E(G')| \geq s] \leq (c''/\log\log n)^{s/4} + (\log n)^{-cs/2}.$$

Proof. If there are s edge intersections between $E(w) \cup E(\vec{p})$ and G', then at least $s/2$ of them occur in either w or $\vec{p}$. Lemma 6.14 implies that $\Pr_{w \in W_l(U)}[|E(w) \cap E(G')| \geq s/2] \leq \left(\frac{4l}{\Delta}\right)^{s/4} \leq (c''/\log\log n)^{s/4}$.

By Lemma 6.11, $\Pr_{\vec{p} \in P_l^{m-1}}[|E(\vec{p}) \cap E(G')| \geq s/2] \leq \Pr_{\vec{p} \in P_l^m}[|E(\vec{p}) \cap E(G')| \geq s/2] \leq (\log n)^{-cs/2}$. □

We now obtain Theorem 6.4:

Lemma 6.16. *Suppose that $m \leq n^{1/3}/\log n$ and $l = 2\lceil c_1 \log n / \log\log n \rceil$. For all but a $c_4/\log\log n$ fraction of all $U = \{u, v\}$, $u \neq v \in V(G_n)$, there are constants $c, c' > 0$ such that for all subgraphs G' of G_n with at most $m(l+1)$ vertices and maximum degree 2 and for every integer $s \geq 0$,*

$$\Pr_{\vec{p} \in P_l^{(m)}(U)}[|E(\cup\vec{p}) \cap E(G')| \geq s] \leq ((c'/\log\log n)^{1/4} + (\log n)^{-c})^s.$$

Proof. By Lemma 6.12, for all but a $c_4/\log\log n$ fraction of U, $\Pr_{w\in W_l(U)}[w \text{ is a path}] \geq 2/3$. For any such U, since the distribution of $w \in W_l(U)$ conditional on w being a path is uniform over $P_l(U)$, the measure of any event on $P_l(U) \times P_l^{m-1}$ is at most $3/2$ times that on $W_l(U) \times P_l^{m-1}$. Further, by the same argument as Lemma 6.10, the probability that the paths in $\vec{p}$ chosen from $P_l(U) \times P_l^{m-1}$ are vertex disjoint is at least $1 - o(1)$ conditioned on being vertex disjoint the distribution of $\vec{p}$ is uniform over $P_l^{(m)}(U)$. Therefore the measure of any event on $P_l^{(m)}(U)$ is at most $(1+o(1))3/2 \leq 2$ times that on $W_l(U) \times P_l^{m-1}$. Applying Lemma 6.15 and adjusting constants c and c' yields the bound. □

7 Discussion

There are a couple of interesting open problems related to our work beyond the natural problem of the communication complexity of DISJ_k.

The first regards *automatizability* and the existence of *separation oracles*. In [21] it was shown that if a system of $0/1$ inequalities has a rank $\leq d$ LS refutation, then the system of inequalities possesses a separation oracle that runs in time $n^{O(d)}$. (A separation oracle is a procedure that takes a polytope P and a point $\vec{x}$ and returns either "true" if $\vec{x} \in P$, or it returns a hyperplane separating $\vec{x}$ and P.) Does semantic $\mathbf{Th}(\mathbf{k})$ have an efficiently computable separation oracle as LS does? A refutation system $\mathcal{R}$ is said to be *automatizable* ([6], cf. [2]) if there is an algorithm that, given unsatisifable CNF ψ, the algorithm finds a refutation of ψ in time $S^{O(1)}$ where S is the minimum size of an $\mathcal{R}$ refutation of ψ. The question of the existence of a separation oracle for $\mathbf{Th}(\mathbf{k})$ is closely related to whether or not $\mathbf{Th}(\mathbf{k})$ is automatizable and we conjecture that the answer to both questions is negative.

The second question is whether or not it is possible to extend our lower bounds to other tautologies that would imply inapproximability results for polynomial-time $\mathbf{Th}(\mathbf{k})$-based algorithms. For example, if we could prove superpolynomial lower bounds for tree-like $\mathbf{Th}(\mathbf{k})$ proofs of random 3CNF formulas, this would imply inapproximability results for $\mathbf{Th}(\mathbf{k})$-based linear programming algorithms for MaxSAT [7]. Of course, lower bounds for random 3CNF formulas are open for the $\mathbf{Th}(\mathbf{1})$ systems and even for tree-like cutting planes with unary coefficients. A first-step towards analyzing random 3-CNFs in the $\mathbf{Th}(\mathbf{k})$ systems would be to improve the analysis of this paper to apply to a graph of degree 3 rather than one of degree $\Theta(\log n)$.

Acknowledgments

We are indebted to Avi Wigderson for helpful discussions and insights.

References

[1] S. Arora, B. Bollobás, and L. Lovász. Proving integrality gaps without knowing the linear program. In *Proceedings 43nd Annual Symposium on Foundations of Computer Science*, pages 313–322, Vancouver, BC, November 2002. IEEE.

[2] P. Beame and T. Pitassi. Propositional Proof Complexity: Past, Present, and Future. In G. Paun, G. Rozenberg, and A. Salomaa, editors, *Current Trends in Theoretical Computer Science: Entering the 21st Century*, pages 42–70. World Scientific Publishing, 2001.

[3] P. Beame, T. Pitassi, N. Segerlind, and A. Wigderson. A direct sum theorem for corruption and the multiparty NOF communication complexity of set disjointness. In *Proceedings Twentieth Annual IEEE Conference on Computational Complexity*, pages 52–66, San Jose, CA, June 2005.

[4] A. Bockmayr, F. Eisenbrand, M.E. Hartmann, and A.S. Schulz. On the Chvatal rank of polytopes in the 0/1 cube. *Discrete Applied Mathematics*, 98(1-2):21–27, 1999.

[5] M. L. Bonet, T. Pitassi, and R. Raz. Lower bounds for cutting planes proofs with small coefficients. *Journal of Symbolic Logic*, 62(3):708–728, September 1997.

[6] M. L. Bonet, T. Pitassi, and R. Raz. No feasible interpolation for TC^0 frege proofs. In *Proceedings 38th Annual Symposium on Foundations of Computer Science*. IEEE, October 1997.

[7] J. Buresh-Oppenheim, N. Galesi, S. Hoory, A. Magen, and T. Pitassi. Rank bounds and integrality gaps for cutting planes procedures. In *Proceedings 44th Annual Symposium on Foundations of Computer Science*, pages 318–327, Boston, MA, October 2003. IEEE.

[8] A. K. Chandra, M. L. Furst, and R. J. Lipton. Multi-party protocols. In *Proceedings of the Fifteenth Annual ACM Symposium on Theory of Computing*, pages 94–99, Boston, MA, April 1983.

[9] V. Chvátal. Edmonds polytopes and a hierarchy of combinatorial problems. *Discrete Mathematics*, 4:305–337, 1973.

[10] V. Chvátal, W. Cook, and M. Hartmann. On cutting-plane proofs in combinatorial optimization. *Linear Algebra and its Applications*, 114/115:455–499, 1989.

[11] W. Cook, C. R. Coullard, and G. Turan. On the complexity of cutting plane proofs. *Discrete Applied Mathematics*, 18:25–38, 1987.

[12] S. Dash. *On the matrix cuts of Lovász and Schrijver and their use in Integer Programming*. PhD thesis, Department of Computer Science, Rice University, March 2001.

[13] S. Dash. An exponential lower bound on the length of some classes of branch-and-cut proofs. In W. Cook and A. S. Schulz, editors, *IPCO*, volume 2337 of *Lecture Notes in Computer Science*, pages 145–160. Springer-Verlag, 2002.

[14] R. Diestel. *Graph Theory*. Springer-Verlag, 1997.

[15] F. Eisenbrand and A. S. Schulz. Bounds on the Chvatal rank of polytopes in the $0/1$-cube. *Combinatorica*, 23(2):245–261, 2003.

[16] D. Grigoriev, E. A. Hirsch, and D. V. Pasechnik. Complexity of semi-algebraic proofs. In *(STACS) 2002: 19th Annual Symposium on Theoretical Aspects of Computer Science*, volume 2285 of *Lecture Notes in Computer Science*, pages 419–430, Antibes, France, February 2002. Springer-Verlag.

[17] R. Impagliazzo, T. Pitassi, and A. Urquhart. Upper and lower bounds on tree-like cutting planes proofs. In *9th Annual IEEE Symposium on Logic in Computer Science*, pages 220–228, Paris, France, 1994.

[18] B. Kalyanasundaram and Georg Schnitger. The probabilistic communication complexity of set intersection. In *Proceedings, Structure in Complexity Theory, Second Annual Conference*, pages 41–49, Cornell University, Ithaca, NY, June 1987. IEEE.

[19] L. G. Khachian. A polynomial time algorithm for linear programming. *Doklady Akademii Nauk SSSR, n.s.*, 244(5):1093–1096, 1979. English translation in *Soviet Math. Dokl. 20*, 191–194.

[20] E. Kushilevitz and N. Nisan. *Communication Complexity*. Cambridge University Press, Cambridge, England ; New York, 1997.

[21] L. Lovasz and A. Schrijver. Cones of matrices and set-functions and 0-1 optimization. *SIAM J. Optimization*, 1(2):166–190, 1991.

[22] A. Lubotzky, R. Phillips, and P. Sarnak. Ramanujan graphs. *Combinatorica*, 8(3):261–277, 1988.

[23] N. Nisan. The communication complexity of threshold gates. In V.S.D. Mikl'os and T. Szonyi, editors, *Combinatorics: Paul Erdös is Eighty, Volume I*, pages 301–315. Bolyai Society, 1993.

[24] P. Pudlák. Lower bounds for resolution and cutting plane proofs and monotone computations. *Journal of Symbolic Logic*, 62(3):981–998, September 1997.

[25] P. Pudl'ak. On the complexity of propositional calculus. In *Sets and Proofs, Invited Papers from Logic Colloquium '97*, pages 197–218. Cambridge, 1999.

[26] R. Raz and A. Wigderson. Monotone circuits for matching require linear depth. *Journal of the ACM*, 39(3):736–744, July 1992.

[27] A. Sheralit and W. Adams. A hierarchy of relaxations between the continuous and convex hull representations for zero-one programming problems. *SIAM Journal on Discrete Mathematics*, 3:411–430, 1990.

[28] L. Valiant and V. Vazirani. NP is as easy as detecting unique solutions. *Theoretical Computer Science*, pages 85–93, 1986.