THEORY OF COMPUTING, Volume 2 (2006), pp. 65–90
http://theoryofcomputing.org

Rank Bounds and Integrality Gaps for Cutting Planes Procedures

Joshua Buresh-Oppenheim* Nicola Galesi† Shlomo Hoory‡
Avner Magen§ Toniann Pitassi¶

Received: May 6, 2005; published: March 25, 2006.

Abstract: We present a new method for proving rank lower bounds for the cutting planes procedures of Gomory and Chvátal (GC) and Lovász and Schrijver (LS), when viewed as proof systems for unsatisfiability. We apply this method to obtain the following new results: First, we prove near-optimal rank bounds for GC and LS proofs for several prominent unsatisfiable CNF examples, including random kCNF formulas and the Tseitin graph formulas. It follows from these lower bounds that a linear number of rounds of GC or LS procedures when applied to the standard MAXSAT linear relaxation does not reduce the integrality gap. Second, we give unsatisfiable examples that have constant rank GC and LS proofs but that require linear rank Resolution proofs. Third, we give examples where the GC rank is $O(\log n)$ but the LS rank is linear. Finally, we address the question of size

*Supported in part by a grant from the PIMS Institute
†Supported by NSERC and by Spanish grant TIC-2001-1577-C03-02
‡Supported by NSERC
§Supported by NSERC
¶Supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) and a Premier's Research Excellence Award

ACM Classification: G.1.6, F.4.1, F.1.3

AMS Classification: 68Q25, 03F20, 90C05, 90C09

Key words and phrases: Cutting Planes, Proof Complexity, Approximation Algorithms

 DOI: 10.4086/toc.2006.v002a004

versus rank; we show that, for both proof systems, rank does not accurately reflect proof size. Specifically, there are examples which have polynomial-size GC/LS proofs but require linear rank.

1 Introduction

Integer linear programming is the problem of optimizing a linear objective function over the integral points of a given (bounded or unbounded) polyhedron. In his seminal paper, Khachian [31] proposed the ellipsoid method for (nonintegral) linear programming, showing that the optimization problem over a polytope is polytime. The additional integrality constraints change the complexity of the problem dramatically: it is well-known that general integer LP is NP-hard. In both the unrestricted and the integral cases, one can also look at feasibility problems instead of at optimization problems. Here, the question is whether a polytope given by a set of linear inequalities is empty. The feasibility problem is closely related to the linear optimization problem, and here too the nonintegral version (checking whether the polytope contains any points at all) is easy while the integral one is NP-complete.

Cutting planes methods for integer linear programming are instrumental in bridging the gap between the true, computationally complex structures (the integral solutions to the problem, or, rather, their convex hull) and their relaxed counterpart, which are generally simple polytopes that contain the convex hull of the integral solutions but also contain other, extraneous nonintegral points. These are methods in which the initial, relaxed polytope P is transformed through a sequence of ever-decreasing (contained) polytopes to the integral hull of P, i.e. the smallest polytope containing the integral points of P. In this sequence, a polytope is produced from its predecessor by using the integrality constraint locally. A simple example of this kind of reasoning is that if one knows that a certain coordinate is at least β, then a stronger conclusion, that this coordinate is at least $\lceil \beta \rceil$, is valid for the integral hull of P. For optimization problems this sequence of polytopes produces a sequence of optimal values that get closer and closer to the desired optimal integral solution, and for feasibility problems, the sequence terminates with the empty polytope if and only if the initial polytope contained no integral points. In either case, then, it seems natural to view this sequence as a proof, either of optimality or of infeasibility.

From the complexity standpoint, there are three important desirable properties of the above sequence: (i) the local operations transforming a polytope to its successor are efficient (ii) the length of the sequence is small, and (iii) there is an efficient algorithm producing the sequence. Properties (i) and (ii) guarantee a small size proof, while (iii) guarantees that we can find the proof efficiently if it is small.

In this paper, we study several prominent cutting planes methods: Gomory-Chvátal cuts [10, 22], and a collection of "matrix-cut" or "lift-and-project" operations defined by Lovász and Schrijver [33]. These methods are currently among the most important techniques for solving or approximating a range of NP-hard 0/1 optimization problems. There are two standard complexity measures of interest for these procedures: rank and size. The size is the total number of cut operations that must be applied and the rank is the total number of rounds of cut operations that must be applied. Rank, therefore, measures the amount of inherent sequentialism in the proof.

Superpolynomial lower bounds on size for a cutting planes method are important since they show that *any* algorithm that produces a cutting planes proof will not be polynomial-time. Superpolynomial size lower bounds are known for the Gomory-Chvátal cutting planes method [34]. There are three distinct types of matrix cuts defined by Lovász and Schrijver, LS_0, LS and LS_+. Exponential lower bounds have been proven for LS_0, the weakest of these [15, 16]. For LS and LS_+, no nontrivial size bounds are known.

Rank is also a natural complexity measure when it comes to proofs. In some proof systems for unsatisfiability, there is a natural rank-based procedure for generating a proof which is practical in certain cases. For example, a rank-based method for Resolution is the familiar Davis-Putnam procedure [17], and a rank-based method for the Polynomial Calculus is a variation on the Gröbner basis algorithm [13]. In both of these cases, it is important that it can be determined if there is a d-round/rank derivation in time at most $n^{O(d)}$. It turns out that matrix cut systems have a somewhat similar property and therefore rank is a particularly interesting measure in this case. In [33] it was shown that for any polytope P, there is an algorithm for optimizing over $P^{(r)}$ in time $n^{O(r)}$, where $P^{(r)}$ is the polytope obtained by applying r rounds of any of the LS methods and n is the total description size of the polytope P. Using similar arguments it can be shown that the same is true when considering the feasibility question rather than optimization for LS and LS_0. It follows that there is a deterministic algorithm that can "search through" all LS proofs of rank d in time $n^{O(d)}$. While this holds for other proof systems such as Resolution, it is less obvious here because the number of faces in the rank-r polytope is not easily bounded, even for small r.

1.1 Our Results and Context

Prior to this work some limitations on the rank-based application of the LS procedure to the problem of approximating vertex cover were shown [2]. In this paper, we study rank-based limitations of all the above-mentioned cutting planes methods both in the case of feasibilty for polytopes defined by unsatisfiable CNF formulas and unsatisfiable sets of mod-2 linear equations and in the case of the optimization problems MAXSAT and MAXLIN.

We present a new method for proving rank lower bounds that applies to both Gomory-Chvátal cutting planes and matrix-cut proof systems. This method can be viewed as a game which produces a tree of (nonintegral) points in the polytope, whose depth is a lower bound on the rank of the polytope in all of the above proof systems. This game allows us to prove asymptotically tight rank bounds for many classes of unsatisfiable boolean formulas, especially those which contain a certain measure of expansion, like random kCNFs and the Tseitin principle on expander graphs. The idea of playing a game on expanding CNFs to achieve proof-complexity lower bounds was largely pioneered by [6].

Result 1: The following holds for GC, LS_0, LS and LS_+:

(1) The Tseitin tautology on a graph H has rank at least $(c-2)n/2$, where c is the edge-expansion of H;

(2) Let $k \geq 5$. There exists a constant c such that, for all $\Delta > c$, a random set of Δn k-mod-2 equations over n variables requires rank $\Omega(n)$ with high probability;

(3) Let $k \geq 5$. There exists a constant c such that, for all $\Delta > c$, a random set of Δn k-clauses over n variables requires rank $\Omega(n)$ with high probability.

Prior to our result, the only high-rank bounds for unsatisfiable boolean examples were for the clique-vs-coloring ([34]) formulas in Gomory-Chvátal cutting planes, and for the PHP in LS ([23]). Concurrently with this work, however, [5] examined GC-rank and, in particular, proved a tight lower bound for the PHP. Subsequent to our result, however, [1] improved the above result for the LS systems to require only $k \geq 3$ in parts (2) and (3).

The *integrality gap* of a linear relaxation is the ratio of the optimal value of the relaxation to the optimum over all integer points. If a linear relaxation of a boolean optimization problem has a small integrality gap, then it is feasible to approximate the optimum of the original problem by solving the linear relaxation. We show that there are MAX-k-SAT and MAX-k-LIN examples where cutting planes procedures are not helpful in the sense that after linearly-many rounds, the integrality gap of the relaxation of the problem is still as big as possible.

Result 2: Let $k \geq 5$ and fix any $\varepsilon > 0$. Let F be a set of $\Theta(n)$ random k-mod-2 equations. The relaxation that results from applying $\Omega(n)$ rounds of GC, LS_0, LS or LS_+ to the standard MAX-k-LIN relaxation of F has integrality gap at least $2 - \varepsilon$ with high probability. Similarly, let C be a set of $\Theta(n)$ random k-clauses. The relaxation that results from applying $\Omega(n)$ rounds of GC, LS_0, LS or LS_+ to the standard MAX-k-SAT relaxation of C has integrality gap at least $\frac{2^k}{2^k-1} - \varepsilon$ with high probability.

To the best of our knowledge there were no results of this form (see also [2, 19]) that give hardness of approximation for more than a logarithmic-number of rounds. Again, subsequently, [1, 38] has proven linear rank lower bounds in the LS systems for various optimization problems, including MAX-3-SAT and MAX-3-LIN. All of these results rule out a particular type of subexponential-time approximation algorithm that works by applying a sublinear number of rounds of an LS system to the obvious LP relaxation to generate an LP with small integrality gap. As noted by [1], many recent successes in approximation algorithms can be viewed as applications of this algorithm. In particular, the SDP relaxations of the Goemans-Williamson maxcut approximation ([21]) and of the Arora-Rao-Vazirani sparsest cut approximation ([4]) are implied by a constant number of rounds of LS_+. While there are optimal PCP results that rule out approximations of MAX-k-SAT and MAX-k-LIN by general algorithms [28], these results rely on unproven complexity assumptions—the stronger the time lower bound desired, the stronger the assumption must be. Our results are unconditional.

Finally, we give examples separating LS-, GC-, and Resolution-rank, and examples with polynomial-size Resolution/GC/LS proofs, that require large rank.

Result 3: There are examples of unsatisfiable CNFs that have

(1) Constant LS and GC-rank, but linear Resolution-rank;

(2) Constant LS_+-rank, $\Theta(\log n)$ GC-rank, and linear LS-rank.

Result 4: There are examples of unsatisfiable CNFs that have

(1) Polynomial-size GC-proofs, but linear GC-rank;

(2) Polynomial-size LS-proofs, but linear LS-rank.

The rest of the paper is organized as follows. In Section 2 we define the Resolution/GC/LS proof systems, and give some background. In Section 3 we provide a general scheme for proving rank lower bounds. In Section 4 we prove rank lower bounds when the constraints are expanding. Section 5 deals with integrality gaps that are based on our rank lower bounds. Section 6 gives various separation examples for LS-, GC-, and Resolution-rank. Section 7 gives an example where both the Resolution/GC/LS proof size and rank are polynomial. In Section 8 we describe an algorithm for showing the unsatisfiability of formulas of LS-rank d in time $n^{O(d)}$ based on the results of [33]. This section is largely expository in that we suspect the results were known, but not written down in the generality we supply.

Finally, we would like to note that an extended abstract of this paper appeared in FOCS 2003 [9]. The results here are the same as in that version, but the exposition is more complete.

2 Definitions and Background

Resolution: Resolution proofs work with clauses, viewed as sets of literals. If C and D are sets of literals, then the clause $(C \vee D)$ is derivable from the clauses $(x \vee C)$ and $(\neg x \vee D)$ by the Resolution rule. A resolution refutation of a CNF formula f is a sequence of clauses $C_1, \dots, C_q$ such that each clause is either a clause of f, or follows from two previous clauses by the resolution rule, and the final clause, C_q, is the empty clause. Let S be a resolution refutation of a CNF formula f, represented as a directed acyclic graph (with nodes corresponding to clauses). The *size* of S is the number of clauses in S; the *depth* or *rank* of S is the length of the longest path in the directed acyclic graph. The resolution *size* (or depth) of f is the minimal size (depth) over all resolution refutations of f. S is *tree-like* if the directed acyclic graph is a tree.

Proof systems based on linear programming: We describe several proof systems for systems of linear inequalities where the values of the variables are restricted to be boolean. In these proof systems, we begin with a polytope P defined by linear inequalities associated with the logical formulation of the problem. In the more common case of CNF-formulas we convert clauses to inequalities in the usual way, i.e.

$$\tau(\ell_1 \vee \cdots \vee \ell_k) \equiv [\tau(\ell_1) + \cdots + \tau(\ell_k) \geq 1] \ ,$$

where each ℓ_i is a literal and $\tau(x) \equiv x$ and $\tau(\neg x) \equiv 1 - x$ for each variable x. For example, $\tau(x \vee \neg y \vee z) \equiv x + (1-y) + z \geq 1$. Notice that the $0/1$ solutions to these inequalities are exactly the satisfying boolean assignments to the formula. Relaxing to $0 \leq x_i \leq 1$ makes the set of solutions a polytope whose integral points are the solutions to the original problem. The following fact is immediate, but will be important later:

Proposition 2.1. *Let C be a clause with at least two variables. Then $\tau(C)$ is a linear inequality that is satisfied if at least two of its underlying variables have value $\frac{1}{2}$.*

We begin by describing Gomory-Chvátal (GC) cutting planes. This proof system is sometimes referred to as simply Cutting Planes in the proof complexity literature. In what follows, let $a_i \in \mathbb{R}^n$ and let x be a vector of n variables. By $\langle\ \rangle$, we mean the standard inner-product. Consider the following two rules: (1) (Linear combinations) From linear inequalities $\langle a_1, x\rangle - b_1 \geq 0, \dots, \langle a_k, x\rangle - b_k \geq 0$, derive $\sum_{i=1}^k (\langle \lambda_i a_i, x\rangle - \lambda_i b_i) \geq 0$, where λ_i are positive rational constants; (2) (Rounding) From $\langle a, x\rangle - \lambda \geq 0$ derive $\langle a, x\rangle - \lceil \lambda \rceil \geq 0$, provided that the coordinates of a are integers. Without loss of generality, we can assume that a rounding operation is always applied after every application of rule (1), and thus we can merge (1) and (2) into a single rule, called a GC *cut*.

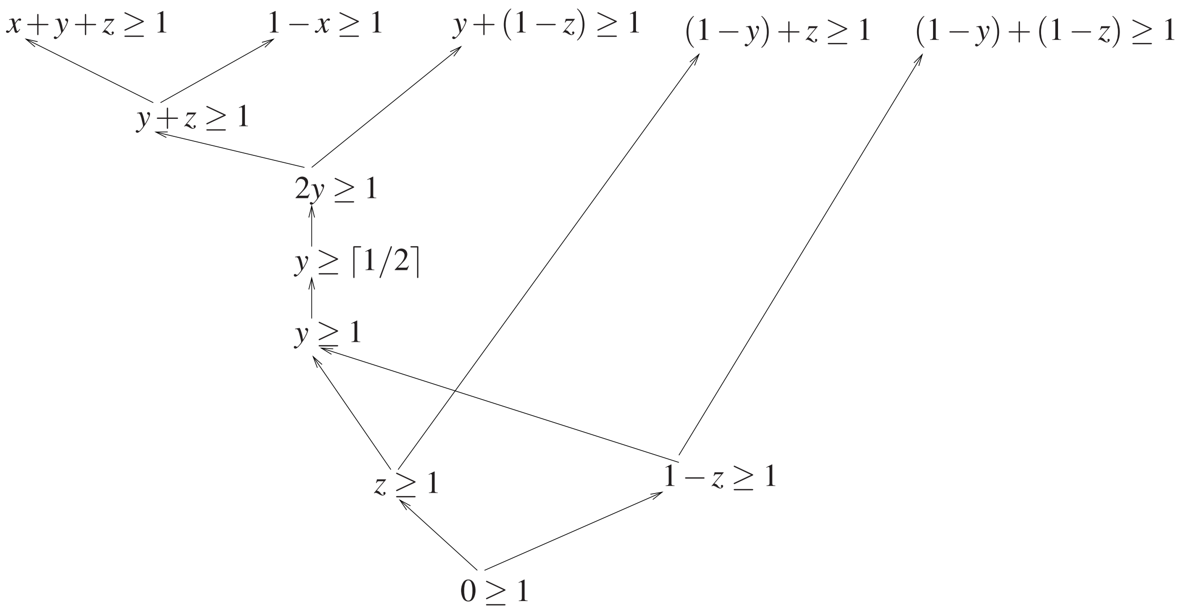

Figure 1: A GC refutation

Definition 2.2. A GC refutation for a set of linear inequalities $f = \{f_1, \ldots, f_m\}$ is a sequence of linear inequalities, $g_1, \ldots, g_q$ such that each g_i is either an inequality from f, or an axiom ($x \geq 0$ or $1-x \geq 0$), or follows from previous inequalities by a GC cut, and the final inequality g_q is $0 \geq 1$.

There are several cutting planes proof systems defined by Lovász and Schrijver, collectively referred to as matrix cuts. These systems allow one to "lift" the linear inequalities to quadratic inequalities, and then "project" back to linear inequalities using the fact that $y^2 = y$ for $y \in \{0,1\}$. Again, let $a_i \in \mathbb{R}^n$ and let x be a vector of n variables. The basic intuition of the following systems is that, if $\langle a_i, x\rangle - b_i \geq 0$ is valid for the integral hull of P, then so are the inequalities $(\langle a_i, x\rangle - b_i)x_j \geq 0$, $(\langle a_i, x\rangle - b_i)x_j \geq 0$, and $(x_j^2 - x_j) = 0$ for each j, since $x_j \in \{0,1\}$. Of course, we can't directly use these quadratic inequalities because it is generally NP-hard to solve the optimization or feasibility problem over such constraints, but it might be helpful, and is certainly valid, to add any linear inequality that is a positive linear combination of them.

Definition 2.3. Given a polytope $P \subseteq [0,1]^n$ defined by $\langle a_i, x\rangle \geq b_i$ for $i = 1, 2, \ldots, m$:

(1) An inequality $\langle c, x\rangle - d \geq 0$ is called an *N-cut* for P if

$$\begin{aligned} \langle c,x\rangle - d &= \sum_{i,j} \alpha_{ij}(\langle a_i,x\rangle - b_i)x_j \\ &+ \sum_{ij} \beta_{ij}(\langle a_i,x\rangle - b_i)(1-x_j) \\ &+ \sum_j \lambda_j(x_j^2 - x_j) \ , \end{aligned}$$

where $\alpha_{ij}, \beta_{ij} \geq 0$ and $\lambda_j \in \mathbb{R}$ 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 linear term $\langle c, x \rangle - d$, we view $x_i x_j$ as distinct from $x_j x_i$.

(3) An inequality $\langle c, x \rangle - d$ is called an N_+-*cut* if

$$\begin{aligned} \langle c,x\rangle - d &= \sum_{i,j} \alpha_{ij}(\langle a_i, x\rangle - b_i)x_j \\ &+ \sum_{ij} \beta_{ij}(\langle a_i, x\rangle - b_i)(1 - x_j) \\ &+ \sum_{j} \lambda_j (x_j^2 - x_j) + \sum_k (\langle h_k, x\rangle + g_k)^2, \end{aligned}$$

where again $\alpha_{ij}, \beta_{ij} \geq 0$, $\lambda_j \in \mathbb{R}$ for $i = 1, \ldots, m$, $j = 1, \ldots, n$ and $g_k + \langle h_k, x \rangle$ is a linear 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 2.4. A *Lovász-Schrijver* (LS) refutation for a set of linear inequalities f is a sequence of inequalities $g_1, \ldots, g_q$ such that each g_i is either an inequality from f or an N-cut for the polytope defined by $g_1, \ldots, g_{i-1}$, 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 2.5. Let $\mathcal{P}$ be one of the proof systems GC, LS, LS_0 or LS_+. Let f be an unsatisfiable set of boolean inequalities and let S be a $\mathcal{P}$-refutation of f, viewed as a directed acyclic graph. The inequalities in S are represented with all coefficients in binary notation. Therefore, the *size* of such an inequality is the sum of all the sizes of its coefficients. The *size* of S is the sum of the sizes of all inequalities in S; the $\mathcal{P}$-size of f is the minimal size over all $\mathcal{P}$ refutations of f.

The complexity measure with which we are primarily concerned is rank. It is defined not only for unsatisfiable sets of boolean inequalities, but for sets of linear inequalities in general.

Definition 2.6. For a set of linear inequalities L that define a polytope in $\mathbb{R}^n$, let $P_L = P_L^{(0)}$ be that polytope. Given $\mathcal{P} \in \{\mathrm{GC}, \mathrm{LS}_0, \mathrm{LS}, \mathrm{LS}_+\}$, let $P_L^{(i)}$ denote the polytope defined by all inequalities that can be derived in depth i from the initial inequalities in $\mathcal{P}$. Clearly $P_L^{(i+1)} \subseteq P_L^{(i)}$. The *rank* of L (or P_L) is the minimal i such that $P_L^{(i)}$ is the convex hull of the integral points in P_L. The *rank* of a point $x \in \mathbb{R}^n$ with respect to P_L is the minimal i such that $x \notin P_L^{(i)}$.

That the rank of any bounded polytope in any of these proof systems is finite is a well-known fact ([22, 10, 33]). Note that, if P contains no integral points, then the rank of the polytope is the maximum rank of its points.

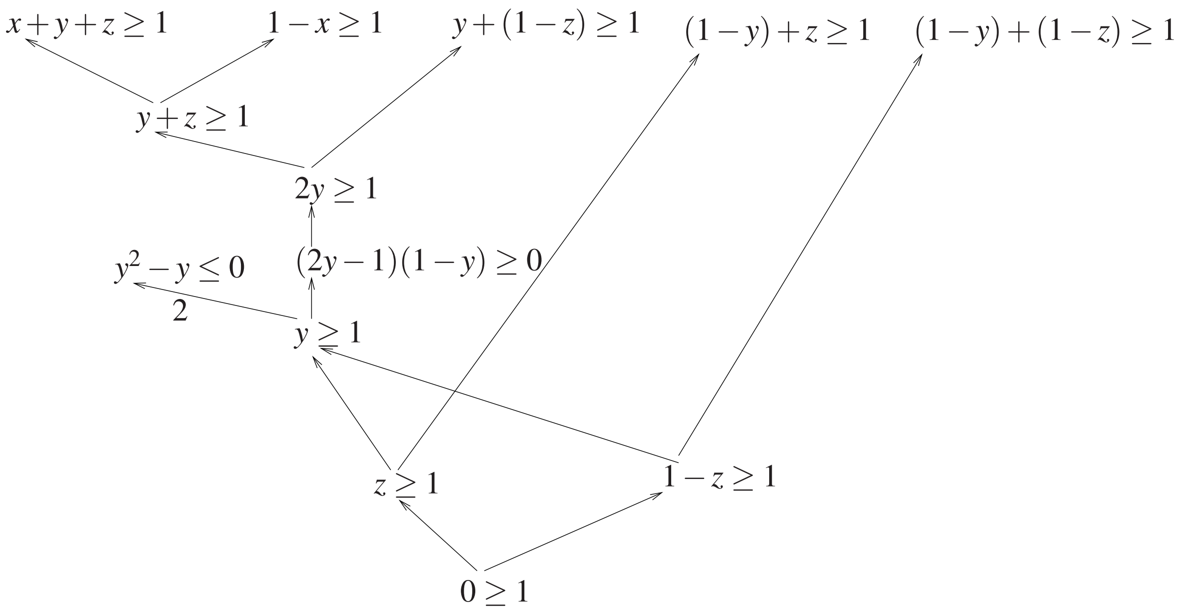

Figure 2: An LS refutation

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 for Resolution, we are restricted to fanin-two. However, Caratheodory's theorem tells us that for any set of vectors V, any vector that is a positive combination of vectors in V can be generated as a positive combination of $dim(V)$ such vectors. Viewing inequalities as vectors, we see that a GC cut is a positive combination of vectors in dimension $n+1$ and an LS (respectively LS_0, LS_+) cut is a positive combination of vectors in dimension n^2+n+1. Therefore, we can assume wlog that the fanin is at most $n+1$ in GC and n^2+n+1 in LS, and so the rank and size would not increase significantly if instead our proof systems were defined to have fanin 2.

Definition 2.7. Let $\mathcal{P}_1$ and $\mathcal{P}_2$ be two refutation systems. We say that $\mathcal{P}_1$ *p-simulates* $\mathcal{P}_2$ if there is a polynomial p such that, for every unsatisfiable formula f, $size_{\mathcal{P}_1}(f) \leq p(size_{\mathcal{P}_2}(f))$, where $size_{\mathcal{P}}(f)$ denotes the size of the minimum refutation of f in system $\mathcal{P}$.

2.1 Alternative definitions

The above definitions of the cutting planes methods lead one to visualize the process syntactically, in a way similar to most proof systems. It is often helpful, however, to look at the dual definitions, which indicate which points remain after applying one round of each of the methods. In fact, this is the way they are usually viewed in optimization.

In general, for a polytope $P \in [0,1]^n$, we let P' be the points that remain after applying one round of the cutting planes method in question.

Definition 2.8. Given a polytope $P \subset [0,1]^n$, the result of applying one round of GC cuts is

$$P' = \{x \in P : \langle a,x\rangle \geq \lceil b \rceil \text{ whenever } a \in \mathbb{Z}^n, b \in \mathbb{R}, \\ \text{and } \langle a,y\rangle \geq b \text{ for all } y \in P\} \ .$$

For the case of GC, it is not hard to see that for every polytope, $P^{(1)} = P'$, and hence $P^{(i+1)} = (P^{(i)})'$ for every $i \geq 0$.

The dual definitions of the matrix-cut systems are most easily stated for the *projective cone*, $\overline{P} \in \mathbb{R}^{n+1}$, of a polytope $P \subset [0,1]^n$. That is,

$$\overline{P} \equiv \{(a, a \cdot w_1, \ldots, a \cdot w_n) : a \geq 0 \text{ and } (w_1, \ldots, w_n) \in P\} \ .$$

If the coordinates of $\mathbb{R}^n$ are $x_1, \ldots, x_n$, we usually refer to this extra coordinate in $\overline{P}$ as x_0. Note that P is exactly the intersection of $\overline{P}$ with the hyperplane $x_0 = 1$. Hence we will often refer to a point $w = (w_0, w_1, \ldots, w_n)$ in $\overline{P} \setminus \{\overline{0}\}$ *projected* onto $[x_0 = 1]$. This simply means the point $(w_1/w_0, \ldots, w_n/w_0) \in P$.

Definition 2.9. (i) A point $w \in \mathbb{R}^{n+1}$ is in $\overline{P}'$ for LS_0 if there is an $(n+1) \times (n+1)$ matrix Y such that $Ye_0 = (e_0^T Y)^T = diag(Y) = w$ and, for all i, $Ye_i, Ye_0 - Ye_i \in \overline{P}$.
(ii) A point w is in $\overline{P}'$ for LS if (i) holds with the extra constraint that Y is symmetric.
(iii) A point w is in $\overline{P}'$ for LS_+ if (ii) holds with the extra constraint that Y is positive semidefinite.

Definition 2.10. For any of LS, LS_+, LS_0, we define P' to be $\overline{P}' \cap [x_0 = 1]$.

To gain some intuition about Definition 2.9, we prove the following useful fact:

Fact 2.11. Given a polytope $P \in [0,1]^n$, after one round of LS_0 we have

$$P' = \bigcap_{i=1}^{n} conv(P \cap [x_i = 0], P \cap [x_i = 1]) \ ,$$

where $conv()$ denotes the convex hull of its arguments.

Proof. Consider $\overline{P}$ and some point $w \in \overline{P}'$. In the matrix Y, the point Ye_i (when projected to $[x_0 = 1]$) is a point in $P \cap [x_i = 1]$ because $Y_{ii} = Y_{0i}$. Also, $Ye_0 - Ye_i$ is a point in $P \cap [x_i = 0]$ because $Y_{i0} = Y_{ii}$. The fact that $Ye_0 = w$ forces w to be a convex combination of Ye_i and $Ye_0 - Ye_i$, for all i.

For the other direction, consider a point $w \in \cap_{i=1}^{n} conv(P \cap [x_i = 0], P \cap [x_i = 1])$. Say $w = (1-\lambda_i)w_i^0 + \lambda_i w_i^1$ for each i, where $w_i^0 \in P \cap [x_i = 0]$, $w_i^1 \in P \cap [x_i = 1]$ and $0 \leq \lambda_i \leq 1$. It is easy to check that the matrix with 0-th column $\binom{1}{w}$ and i-th column $\binom{\lambda_i}{\lambda_i w_i^1}$ fulfills the requirements of Definition 2.9. $\square$

To see that the dual definitions for the matrix cut systems are equivalent to the initial definitions (that is, $P' = P^{(1)}$) consult [15].

2.2 What's known about complexity

By definition LS p-simulates LS_0 and LS_+ p-simulates LS, and these simulations are rank preserving. Moreover for unsatisfiable CNF formulas, GC, LS_0, LS and LS_+ can all p-simulate Resolution and this simulation is rank-preserving [14]. It has also been shown that GC can p-simulate small-weight LS_0 [29]. In terms of negative results for simulations, the propositional pigeonhole principle (PHP) provides a family of unsatisfiable CNF examples requiring exponential-size Resolution proofs [26] but with polynomial-size GC, LS_0, LS and LS_+ proofs [14]. For GC and LS_0, exponential size lower bounds for one specific family of boolean examples are known [34, 16]. For LS and LS_+, no superpolynomial lower bounds are known.

Now let us review what is known with respect to rank. Every system of linear inequalities has a rank n LS proof [33]. For GC, the rank of every polytope in the unit cube is at most $O(n^2 \log n)$ [18], and moreover there are examples requiring GC-rank more than n [18]. However for unsatisfiable examples, the GC-rank is at most n [7]. For GC, linear rank bounds for unsatisfiable CNF examples were first obtained in [11]; however, these examples have exponentially-many faces (inequalities) and thus the rank is still small in the input size. Linear rank bounds for GC (as a function of the input size) for unsatisfiable CNF examples were first proven in [30], and also follow from the size bounds [34]. For LS, a wealth of rank lower bounds are known for the non-empty or non-CNF case (see, for example, [33, 20, 36]). Otherwise, [23] prove linear rank lower bounds for the PHP. In summary, the only known high-rank, unsatisfiable CNF examples were the clique-vs-coloring formulas for GC and the PHP for LS. In this paper, we prove rank bounds for all of these proof systems for many unsatisfiable CNFs satisfying certain combinatorial conditions.

3 Proving Rank Lower Bounds

In what follows, we give methods for proving rank lower bounds for many natural, polysize sets L of contradictory linear inequalities. These lower bounds follow by characterizing some of the points in $P_L^{(i)}$ that survive in $P_L^{(i+1)}$. We call these characterizations "protection lemmas," because they argue that certain points are protected from removal in the next round provided certain other points survived the previous round. These sorts of lemmas have been used in the past to prove rank lower bounds for specific polytopes in specific cutting planes procedures (see [11, 20], for example). We develop a common protection lemma that works for many examples in any of the proof systems we define. Moreover, we define a simple, two-player game that uses this common protection lemma to establish lower bounds.

3.1 Protection Lemmas

We begin with some notation. For $x \in \mathbb{R}^n$, $e \in \{1,\dots,n\}$, and $a \in \mathbb{R}$, we denote by $x^{(e,a)}$ the point that is the same as x except that the e-th coordinate has value a. For $x \in \mathbb{R}^n$, we denote by $E(x)$ the set of coordinates on which x is non-integral.

Lemma 3.1 (GC Lemma). *The following holds for GC: Let P be a bounded polytope in $\mathbb{R}^n$. Let $x \in \frac{1}{2}\mathbb{Z}^n$, and let $E = E(x)$ be partitioned into sets $E_1, E_2 \dots, E_t$. Suppose that for every $j \in \{1,2,\dots,t\}$*

we can represent x as an average of vectors in P that are 0-1 on E_j and agree with x elsewhere. Then $x \in P'$.

Proof. Assume for contradiction that $x \notin P'$. Then there is a vector $a \in \mathbb{Z}^n$ and a non integral scalar b, such that $\langle a,y \rangle \geq b$ for all $y \in P$ and $\langle a,x \rangle < \lceil b \rceil$. Clearly $x \in P$, being an average of points in that polytope. So $\langle a,x \rangle \geq b$ and it follows that $\langle a,x \rangle$ must be in $\frac{1}{2} + \mathbb{Z}$. Thus $\sum_{e \in E(x)} a_e$ must be odd, and since $\sum_{e \in E(x)} a_e = \sum_i \sum_{e \in E_i} a_e$, there is a j such that $\sum_{e \in E_j} a_e$ is odd. Consider the set of vectors $V \subset P$ that average to x and that differ from x exactly on E_j where they take $0/1$ values. Since $\sum_{e \in E_j} a_e$ is odd we can see that $\langle a,v \rangle$ is integral for all $v \in V$. But then $\langle a,v \rangle \geq \lceil b \rceil$. Since x is an average of the $v \in V$, we also get $\langle a,x \rangle \geq \lceil b \rceil$. Contradiction. ☐

The following lemma is immediate from Fact 2.11:

Lemma 3.2 (LS$_0$ Lemma). *The following holds for LS_0: Let $P \subset [0,1]^n$ be a polytope, and x be a point in P. Then, if for every $i \in E(x)$ there is a set of points $S_i \subset P$ with i-th coordinate in $\{0,1\}$ such that $x \in conv(S_i)$, then $x \in P'$.*

Lemma 3.3 (LS/LS$_+$ Lemma ([20], Theorem 4.1)). *The following holds for LS and LS_+: Let $P \subset [0,1]^n$ be a polytope, and x be a point in P. If, for every $i \in E(x)$, $x^{(i,0)}, x^{(i,1)} \in P$, then $x \in P'$.*

Proof. Let $\bar{x}$ be the vector $(1,x_1,\dots,x_n)^T$ and let $A = \bar{x}\bar{x}^T$. A is certainly symmetric and positive semidefinite, but it has $\bar{x}^2$ instead of $\bar{x}$ on the diagonal ($\bar{x}^2$ is the vector whose ith coordinate is $\bar{x}_i^2$). Let B be the diagonal matrix with $\bar{x} - \bar{x}^2$ on the diagonal. Because $x \in [0,1]^n$, B is positive semidefinite. Finally, let $Y = A + B$; it is clearly symmetric and positive semidefinite. Notice that for every i, Ye_i (projected onto $[x_0 = 1]$) is $x^{(i,1)}$ and that $Ye_0 - Ye_i$ is $x^{(i,0)}$. These are both guaranteed to be in P by the lemma's hypothesis, so x is in P'. ☐

3.2 A game

Lemmas 3.1, 3.2 and 3.3 all conclude the same thing from different hypotheses. We now state a protection lemma that holds for all of the proof systems because it uses a hypothesis that is stronger than any of those in the previous protection lemmas:

Lemma 3.4 (Game Lemma). *The following holds for GC, LS_0, LS and LS_+: Let $P \subset [0,1]^n$ be a polytope, and $x \in \{0,\frac{1}{2},1\}^n \cap P$. If, for every $i \in E(x)$, $x^{(i,0)}, x^{(i,1)} \in P$, then $x \in P'$.*

This lemma gives us the following Prover-Adversary game for showing a lower-bound on the rank of a point $w \in \{0,\frac{1}{2},1\}^n$ with respect to P. We think of the Prover as trying to show that w has high rank, while the Adversary is trying to foil that proof. The game proceeds in rounds. During each round, there is a current point $x \in \{0,\frac{1}{2},1\}^n$, whose initial value is w. At each round, the Prover either moves or allows the Adversary to move:

Prover-move The Prover generates a set of points $Y \subset \{0,\frac{1}{2},1\}^n$ such that x is a convex combination of those points ($x \notin Y$). The Adversary selects one point $y \in Y$ to be the new x.

Adversary-move The Adversary selects a coordinate e such that x_e is $\frac{1}{2}$ and a value $a \in \{0,1\}$. The new x is $x^{(e,a)}$.

The game ends when x is no longer in P or $x \in \mathbb{Z}^n$. The Prover gets one reward-point (as distinguished from geometric points) for each Adversary-move.

Lemma 3.5. *If a Prover has a strategy to earn m reward-points against every adversary, then the (GC, LS_0, LS, or LS_+)-rank of w with respect to P is at least m.*

Proof. Let $r = r(w,P)$ be the maximum, over all adversaries, of the number of rounds in the Prover-Adversary game for the given prover, the point w and the polytope P. We proceed by induction on r. If $r = 0$, then $m = 0$, but the rank of w can never be less than 0. For arbitrary $r > 0$, the Prover can start by making a Prover-move or an Adversary-move. If it is a Prover-move, then the Prover presents Y and, no matter which $y \in Y$ the Adversary chooses, $r(y,P) < r(w,P)$ and the Prover has a strategy to earn m reward-points. Hence, by induction, each $y \in Y$ has rank at least m. By convexity, the rank of w, which is a convex combination of points in Y, is at least m. If it is an Adversary-move, then, no matter which e and a the Adversary chooses, $r(w^{(e,a)},P) < r(w,P)$ and the Prover has a strategy to earn $m-1$ reward-points. Again, by induction, $w^{(e,a)}$ has rank at least $m-1$ for all possible (e,a), so by Lemma 3.4, w has rank at least m. □

4 Expanding Constraints

In what follows, we deal with F, a set of mod-2 equations over n variables. That is, each equation in F is of the form $\sum_{i \in S} x_i \equiv a \pmod 2$, where $S \subset [n]$ and $a \in \{0,1\}$. Notice that each such equation can be represented by the conjunction of $2^{|S|-1}$ clauses, each of which can be represented as a linear inequality. We denote by P_F the polytope bounded by these inequalities and by the inequalities $0 \le x_i \le 1$.

Let G_F be the bipartite graph from the set F to the set of variables where each equation is connected to the variables it contains. We prove a rank lower bound for P_F as a function of the expansion of G_F.

We will need the following notions of expansion:

Definition 4.1. Let $e(V_1,V_2)$ be the number of edges (v_1,v_2) with $v_i \in V_i$. The edge-expansion of a graph $G = (V,E)$ is

$$\min_{S \subset V, |S| \le |V|/2} \frac{e(S, V \setminus S)}{|S|} .$$

Definition 4.2. A bipartite graph from V to U is an (r,ε)-expander if, for all subsets $X \subset V$ where $|X| \le r$, we have $\Gamma(X) \ge \varepsilon|X|$. The *expansion* of a set $X \subset V$, $e(X)$, is the value $|\Gamma(X)|/|X|$.

Definition 4.3. Let G be a bipartite graph from V to U. The boundary of a set $X \subset V$ is $\partial X \stackrel{d}{=} \{u \in U : |\Gamma(u) \cap X| = 1\}$. G is an (r,ε)-boundary expander if for all subsets $X \subset V$ where $|X| \le r$, we have $|\partial X| \ge \varepsilon|X|$. The *boundary expansion* of a set $X \subset V$ is the value $|\partial X|/|X|$.

The following fact relates bipartite expansion with boundary-expansion.

Fact 4.4. If G is a bipartite graph from V to U where V has maximal degree d and if G is an (r,ε)-expander, then G is a $(r,2\varepsilon-d)$-boundary expander.

The reason that we require G_F to be a good expander is that it allows us to satisfy subsets of F:

Lemma 4.5. *Consider a set F of m mod-2 equations over n variables. Assume that for every variable v and every value $a \in \{0,1\}$, there is a solution to F where v assumes the value a. Then all the 0-1 solutions to F average to the all-$\frac{1}{2}$ assignment.*

Proof. Let $S_{v,a}$ be a solution to F in which variable v is set to a. It is easy to see that the mapping $S \mapsto S+S_{v,1}-S_{v,0}$ mod 2 is a one-to-one mapping from solutions with $v=0$ onto solutions with $v=1$. Therefore the average over all solutions to F is $\frac{1}{2}$ on v. $\square$

Lemma 4.6. *Let F be a set of m mod-2 equations over n variables. Assume G_F is an (m,δ)-boundary expander for some $\delta > 0$. Then F has a 0-1 solution.*

Proof. We begin by pairing each equation in F to a variable it includes. For i with $0 \le i < m$ assume we have a pairing $(f_1,v_1),\ldots,(f_i,v_i)$ such that $v_1,\ldots,v_i$ are not in the boundary of $F \setminus \{f_1,\ldots,f_i\}$. Since $i<m$, $F \setminus \{f_1,\ldots,f_i\}$ is not empty, so there must be some $v_{i+1} \notin \{v_1,\ldots,v_i\}$ in $\partial(F \setminus \{f_1,\ldots,f_i\})$. It is connected to some equation f_{i+1} in $F \setminus \{f_1,\ldots,f_i\}$ and to no other equations in that set. Add (f_{i+1},v_{i+1}) to the set of pairs. Eventually we have the set of pairs $(f_1,v_1),\ldots,(f_m,v_m)$. To satisfy F, set all variables not in $\{v_1,\ldots,v_m\}$ arbitrarily. Now, for $i=m$ to 1, set v_i so that it satisfies equation f_i (notice that in this order, v_i is the last unassigned variable of f_i). $\square$

We now use the game to show a rank lower bound for expanding sets of equations. For $x \in \{0,\frac{1}{2},1\}^n$, say an equation $f \in F$ is *fixed* with respect to x if x sets all the variables of f to 0/1 and f is satisfied by x. Let $G_F(x)$ be the subgraph of G_F induced by the set of variables $E(x)$ and the set of non-fixed equations.

Theorem 4.7. *Let $\varepsilon > 0$ and let $w \in \{0,\frac{1}{2},1\}^n$. If $G_F(w)$ is an $(r,2+\varepsilon)$-boundary expander, then w has (GC, LS_0, LS, LS_+)-rank at least $r\varepsilon$ with respect to P_F.*

Proof. We start the game with $x=w$. Clearly $x \in P_F$ since each equation $f \in F$ is either fixed or has two underlying variables set to $\frac{1}{2}$ by the expansion requirement. In the latter case, each linear inequality representing f is satisfied by Proposition 2.1. Let $\Gamma_x(R)$ be the neighbor set of $R \subset F$ in $G_F(x)$. Let ℓ initially be set to r. The Prover's strategy is as follows:

1. Let the Adversary move as long as all subsets $R \subset F$ in $G_F(x)$ of size at most ℓ have boundary expansion >2 in $G_F(x)$. Note that after such a move we have $x \in P_F$ since all equations in $G_F(x)$ have degree at least 2.

2. Let B be a maximal subset of equations in $G_F(x)$ with boundary expansion ≤ 2 such that $|B| \le \ell$. Only one variable has been set (by an Adversary move) since B had boundary expansion >2, so now B must have boundary expansion >1. Now the Prover moves. Let Y be the set of all assignments satisfying B that are $0-1$ on $\Gamma_x(B)$ and that agree with x elsewhere. To see that Y is nonempty and that it does indeed average to x, consider an arbitrary variable v in $\Gamma_x(B)$ and an

arbitrary value $a \in \{0,1\}$. By Lemma 4.5 it is enough to show that there is a point in Y in which v is set to a. Notice that B still has boundary-expansion greater than 0 on the graph $G_F(x)$ minus v, and so Lemma 4.6 implies that, regardless of the setting of v, there exists a $0-1$ assignment on $\Gamma_x(B) \setminus \{v\}$ satisfying B. The Adversary selects one $y \in Y$ to be the new x.

Set ℓ to $\ell - |B|$. If $\ell = 0$, stop the game. Otherwise, we argue that $x \in P_F$. Indeed, in that case $|B|$ is strictly smaller than ℓ, and it is always the case that every equation f not in B has at least two neighbors in $G_F(x)$ since otherwise $B \bigcup \{f\}$ would also have boundary-expansion at most 2 contradicting the maximality of B.

3. Repeat until the game is over.

Now we will show that the Prover always earns at least $r\varepsilon$ reward-points. Assume the game ends after k rounds of the strategy. For any round $i \le k$, let B_i be the set of vertices designated in step 2 and let $S = \bigcup_{j=1}^{k} B_j$. The size of S is r, so S had a boundary of size at least $(2+\varepsilon)r$ in G_F. At the end of the game, S has no boundary (in fact it has no neighbors) in $G_F(x)$. At most $2r$ of these boundary nodes were removed by the Prover: at the beginning of step 2 of round i, B_i has at most $2|B_i|$ boundary nodes and every boundary node of S is a boundary node for exactly one B_i. Hence at least εr of S's original boundary nodes were removed by the Adversary. By Lemma 3.5, w has the required rank. ☐

It turns out that many common formulas are examples of boundary-expanding mod-2 equations.

Definition 4.8 ([39]). The Tseitin tautology for an odd-size graph $G = (V,E)$, denoted $TS(G)$, is the following: given a boolean variable X_{uv} for each edge $(u,v) \in E$, there exists no assignment to all the variables satisfying

$$\sum_{v \in \Gamma(u)} X_{uv} \equiv 1 \pmod 2 \ ,$$

for every $u \in V$.

Definition 4.9. There are $2\binom{n}{k}$ linear, mod-2 equations over n variables that contain exactly k different variables. Let $\mathcal{M}_m^{k,n}$ be the probability distribution induced by choosing m of these equations uniformly and independently. There are $2^k\binom{n}{k}$ clauses over n variables that contain exactly k different variables. Let $\mathcal{N}_m^{k,n}$ be the probability distribution induced by choosing m of these clauses uniformly and independently.

Theorem 4.7 enables us to prove our main result:

Corollary 4.10. *The following holds for GC, LS_0, LS and LS_+:*

(1) The Tseitin tautology on a graph H has rank at least $(c-2)n/2$, where c is the edge-expansion of H;

(2) Let $k \ge 5$. There exists a constant c such that, for all $\Delta > c$, $F \sim \mathcal{M}_{\Delta n}^{k,n}$ requires rank $\Omega(n)$ with high probability;

(3) Let $k \ge 5$. There exists a constant c such that, for all $\Delta > c$, $C \sim \mathcal{N}_{\Delta n}^{k,n}$ requires rank $\Omega(n)$ with high probability.

Proof. Throughout, let w be the all $\frac{1}{2}$ point.

(1) The edge-expansion of a graph $H = (V, E)$ is the density of the sparsest cut:

$$\min_{S \subset V, |S| \leq |V|/2} \frac{e(S, V \setminus S)}{|S|} .$$

It is easy to see that $G_{TS(H)}(w)$ is an $(n/2, c)$-boundary-expander: consider any subset of nodes in H, $S \subset V$ and look at the corresponding set R of equations in $TS(H)$. Every variable corresponding to an edge across the cut $(S, V \setminus S)$ is in the boundary of R in the bipartite graph $G_{TS(H)}$. Now simply apply Theorem 4.7.

(2) It is well-known that $G_F(w)$ is an excellent expander ([12]): for any constant Δ, ε,k, there exists a constant $\alpha > 0$ such that $G_F(w)$ is almost always an $(\alpha n, k-1-\varepsilon)$-expander. By Fact 4.4, every (r, δ) bipartite expander graph on (V, U) where V has maximal degree d is an $(r, 2\delta - d)$-boundary-expander. Hence $G_F(w)$ is an $(\alpha n, k-2-2\varepsilon)$-boundary-expander. For $k \geq 5$ and small ε, the boundary-expansion is more than 2, so w has rank $\Omega(n)$ by Theorem 4.7. Lastly, we need to fix c such that, whenever $\Delta > c$, F is unsatisfiable with high probability (otherwise, F might not have high rank, despite the fact that w does). The corollary follows.

(3) $G_C(w)$, the bipartite graph associated with the clauses of C, is the same as $G_F(w)$ for random F. Generate C' by adding, for each $e \in C$, the following clauses: if e has an even (odd) number of positive literals, all clauses on the same variables as e that have an even (odd) number of positive literals. Clearly w's rank with respect to P_C is at least its rank with respect to $P_{C'}$, but C' is equivalent to a set of $|C|$ mod-2 equations such that $G_{C'}(w)$ is an $(\alpha n, k-2-2\varepsilon)$-boundary expander (with high probability, given $\Delta, \varepsilon, k, \alpha$ as in (2)). Again, fix c so that, whenever $\Delta > c$, C is unsatisfiable with high probability.

□

5 Integrality Gaps from Rank Lower Bounds

The problem MAX-k-SAT (MAX-k-LIN) is the following: given a set of k-clauses (mod-2 equations), determine the maximum number of clauses (equations) that can be satisfied simultaneously. This problem is well-studied in the theory of approximation algorithms and optimal inapproximation results are known under the assumption that $P \neq NP$ [28]. Here we show optimal inapproximation results (that are unconditional) for a restricted class of approximation algorithms that involve applying GC or LS procedures to a relaxation of the standard integer program. These algorithms are not necessarily polytime. Similar results have been shown for LS-relaxations of vertex cover ([2], see also the improvements [3, 37]) and maximum independent set ([19]). Both show that a large integrality gap remains after $\Omega(\log n)$ rounds of *LS*.

Given a set of k-mod-2 equations $F = \{f_1, \ldots, f_m\}$ over variables $x_1, \ldots, x_n$, add a new set of variables $y_1, \ldots, y_m$. For each f_i: $\sum_{j \in I_i} x_j \equiv a_i \pmod 2$, let f_i' be the equation $y_i + \sum_{j \in I_i} x_j \equiv a_i + 1 \pmod 2$. Let F' be the set of f_i''s. If y_i is 1, then f_i' is satisfied if and only if f_i is satisfied. Hence we want to maximize the linear function $\sum_{i=1}^m y_i$ over the constraints F' within the boolean cube. Convert these mod-2 equations into linear constraints, and call the resulting linear program L_F. The integrality gap of this LP is at most 2 since, given any set of mod-2 equations, there must be a boolean assignment satisfying at least half of them. An r-round GC- (respectively, LS_0-, LS-, LS_+-) relaxation of (the integer version of) L_F (or any linear program) is a linear program with the same optimization function but with all additional constraints that can be generated in depth r from the original constraints using GC (respectively, LS_0, LS, LS_+).

Theorem 5.1. *Let $k \geq 5$. For every constant $\varepsilon > 0$, there are constants $\Delta, \beta > 0$ such that if $F \sim \mathcal{M}^{k,n}_{\Delta n}$ then the integrality gap of every βn-round GC- (resp., LS_0-, LS-, LS_+-) relaxation of L_F is at least $2-\varepsilon$ with high probability.*

Proof. Given ε, fix $\Delta \gg 8\ln 2/\varepsilon'^2$, where $(\frac{1}{2}+\varepsilon')(2-\varepsilon)=1$. An arbitrary assignment satisfies each of F's equations with probability $\frac{1}{2}$, so the expected number of satisfied equations is $\frac{1}{2}\Delta n$. The probability that it satisfies more than $(\frac{1}{2}+\varepsilon')\Delta n$ equations is at most $exp(-\frac{\varepsilon'^2\Delta n}{8})$ by Chernoff. Given the choice of Δ, this expression is much less than 2^{-n}, so with high probability no assignment satisfies more than a $\frac{1}{2}+\varepsilon'$-fraction of F's equations.

On the other hand, consider an assignment w that sets the variables $y_1,\dots,y_{\Delta n}$ to 1 and sets $x_1,\dots,x_n$ to $\frac{1}{2}$. Clearly, w satisfies all of the equations of F'. Furthermore, it is well-known that $G_{F'}(w)$ is almost surely an $(\alpha n, 2+\delta)$-boundary expander for some $\alpha,\delta > 0$ that depend on Δ. Let $\beta = \alpha\delta$. Hence, by Theorem 4.7, w remains a feasible solution for every βn-round GC- (resp., LS_0-, LS-, LS_+-) relaxation of L_F. □

We can form a linear program L_C for a set of k-clauses C in an analogous manner. Similarly,

Theorem 5.2. *Let $k \geq 5$. For every $\varepsilon > 0$, there exists $\Delta, \beta > 0$ such that if $C \sim \mathcal{N}^{k,n}_{\Delta n}$, then the integrality gap of every βn-round relaxation of L_C is at least $\frac{2^k}{2^k-1}-\varepsilon$ with high probability.*

Again, this inapproximation result is optimal since, given any set of k-clauses, there is a boolean assignment that satisfies at least a $\frac{2^k-1}{2^k}$ fraction of them.

6 Separating GC, LS and Resolution Ranks

We consider the following generalization of PHP_n, the Pigeonhole Principle on $n+1$ pigeons and n holes, first suggested in [6]. Let $G=(U,V,E)$ be a bipartite graph, where $|U|=n+1$ and $|V|=n$. The tautology PHP(G) is the statement that G doesn't have a perfect matching. The formal statement of this is (1) For each $i \in U$, $\sum_{j\in\Gamma(i)} x_{i,j} \geq 1$; (2) For all $j \in V$, $i,i' \in \Gamma(j)$, such that $i \neq i'$, $x_{i,j}+x_{i',j} \leq 1$. The standard PHP_n is just PHP$(K_{n+1,n})$, where $K_{n+1,n}$ is the complete $n+1,n$ bipartite graph.

In this section we show the following separations: (1) PHP_n has LS-rank n but GC-rank $O(\log n)$; (2) For an expander graph G with constant degree d, the Resolution-rank of $PHP(G)$ is $\Omega(n)$, while its LS-rank and GC-rank are $O(d)$.

The Resolution-rank lower bound is proven in [6]; it is implied by their size lower bound. The LS- and GC-rank upper bounds are very similar to each other: for every $j \in V$, it is possible to derive $\sum_{i\in\Gamma(j)} x_{i,j} \leq 1$ in rank $O(d)$ in both systems (Corollary 3.1.1 of [27] and Corollary 2.8 of [33]). The point is that the polytope defined by adding these new inequalities is the empty polytope, and therefore we can get the desired contradiction in one LS or GC step.

For the separation result of the GC and LS ranks, we start with the upper bound on the GC-rank of PHP_n. This result was proved independently by [5]. Actually, the theorem follows almost immediately from Theorem 3.1.1 of [27], which preceeds both works. We include its proof for illustrative purposes.

Theorem 6.1. *The GC-rank of PHP_n is $O(\log n)$.*

Proof. For a subset $S \subset \{1,2,\ldots,n+1\}$ and $1 \le j \le n$ let $f_{S,j}$ be the inequality $\sum_{i\in S} x_{ij} \le 1$. We claim that it is possible to deduce, from $f_{S,j}$ for every S of size k, any $f_{T,j}$ with T of size $< 2k$ in one GC-cut. In other words, if $f_{S,j}$ are valid for $PHP^{(r)}$ (recall the notation of Definition 2.6) for every S of size k and every j, then $f_{T,j}$ is valid for $PHP^{(r+1)}$ for every T of size $< 2k$. This means that for all j, $\sum_{i=1}^{n+1} x_{ij} \le 1$ is valid for $PHP^{(O(\log n))}$. On the other hand, no solution that satisfies these inequalities can satisfy all the axioms $\sum_{j=1}^{n} x_{ij} \ge 1$ for every i. Therefore $PHP^{(O(\log n))} = \emptyset$, and the Chvátal -rank of PHP_n is $O(\log n)$. To see the claim, take any j and T of size $l < 2k$, and sum up with coefficients $1/\binom{l-1}{k-1}$ the inequalities $f_{S,j}$ over all subsets $S \subset T$ of size k. After rounding the deduced inequality is

$$\sum_{i\in T} x_{i,j} \le \left\lfloor \frac{\binom{l}{k}}{\binom{l-1}{k-1}} \right\rfloor = \lfloor l/k \rfloor \le 1 \ , \tag{6.1}$$

namely, $f_{T,j}$. A good way to think of (6.1) is that when using the symmetric sum, we care only about the average threshold for a single variable. In $f_{S,j}$ it is $1/|S|$, and so basically all we do is take the threshold $x_i \le 1/|S|$ and turn it into $\sum_{i\in T} x_i \le |T|/|S|$, and if $|T| < 2|S|$ we get $\sum_{i\in T} \le \lfloor |T|/|S| \rfloor \le 1$. □

In fact, this bound is tight by [5]. Again, Theorem 3.1.1 of [27] proves something very similar.

[24] prove that the PHP has constant-rank proofs in LS_+. This fact also follows immediately from Corollary 2.15 of [33]. In light of this, LS_+ is separated from GC with respect to rank.

A linear lower bound for the LS-rank of PHP_n was given by [23]. See Corollary 2.8 of [33] for a very similar proof. We will give a proof for the LS_0-rank using a protection lemma, which we think is simple and illuminating.

Theorem 6.2. *([23]) The LS_0-rank of PHP_n is $n-1$.*

Proof. The proof proceeds by induction on n. PHP_2 consists of a single point, and its LS_0-rank is therefore 1. For PHP_n, we argue that the all $1/n$ point has rank $n-1$. Given $1 \le i \le n+1$ and $1 \le \ell \le n$, let $x^{i,\ell}$ be the following point: $x^{i,\ell}_{i,\ell} = 1$; $x^{i,\ell}_{i,\ell'} = 0$ for all $\ell' \ne \ell$; $x^{i,\ell}_{i',\ell} = 0$ for all $i' \ne i$; $x^{i,\ell}$ is $1/(n-1)$ everywhere else. For every coordinate (i,j), let S_{ij} be the set of $x^{i,\ell}$ for $1 \le \ell \le n$. Note that for every point in S_{ij}, the coordinate (i,j) has value in $\{0,1\}$. Furthermore, the average of all points in S_{ij} is the all $1/n$ point. By Lemma 3.2, the all $1/n$ point has rank one more than the minimum rank of the points in S_{ij}. But each such point is the all $1/(n-1)$ point for PHP_{n-1}, so it must have rank $n-2$ by induction. □

The PHP has polynomial-size (tree-like) LS_0 proofs. The fact that LS requires rank $\Omega(n)$ for the PHP shows that for both LS and LS_0 proofs, large rank is not a good indicator of large size (even in the tree-like systems). Since GC and LS_+ prove the PHP in small rank, and since Resolution requires large proofs, the PHP does not resolve this question for these proof systems. In the next section, we give a different formula which shows that GC and Resolution can have large rank and small size. In fact, it is not difficult to see that tree-like Resolution can have large rank and small size. The open questions, then, are whether large rank implies large size in tree-like GC or in LS_+ (tree-like or not).

7 GC Proofs with Large Rank and Small Size

In Theorem 6.1 of [11] and Theorem 4 of [5], it is shown that the size s of a GC proof of a tautology is $O(n^r)$ where n is the number of variables and r is the GC-rank of the polytope associated with the tautology. Here we show an example where this bound is very far from being tight. Specifically, we show an example of a tautology which has a quadratic-size GC proof (in fact even a Resolution proof with that size) and linear GC-rank. It turns out that such a separation between size and rank can be witnessed by any formula that has polysize GC refutations, but requires exponential tree-like GC refutations ([5]).

The unsatisfiable formula we take is GT_n which is the negation of the property that every total ordering on n elements has a maximal element (alternatively, that a directed graph closed under transitivity and with no cycles of size two has a source node). More formally, given the set of boolean variables $\{X_{ij} \,:\, i,j \in [n], i \neq j\}$, assert

(1) $X_{ij} \equiv \neg X_{ji}$ for each $i \neq j$ (totality);

(2) $X_{ij} \wedge X_{jk} \rightarrow X_{ik}$ for each $i \neq j \neq k$ (transitivity);

(3) $\bigvee_{j \neq i} X_{ij}$ for each i (no maximal element).

The formula was introduced by [32] and is formulated using $n(n-1)$ variables. The natural way to state it uses width-n clauses, but it can be encoded with constant-width. Stalmark [35] shows that GT_n has polynomial-size Resolution refutations, but Bonet and Galesi [8] show that it requires width $\Omega(n)$ (even when stated with narrow clauses). Since Resolution-width is at most Resolution-rank (the length of the path from a width-w clause to the empty clause in a Resolution refutation is at least w), the Resolution-rank is also $\Omega(n)$. Since GC polynomially simulates resolution ([14]), there is a also a polynomial size GC proof of the formula. It remains to show:

Theorem 7.1. *The GC-rank and the LS_0-rank of the polytope associated with GT_n is $\Omega(n)$.*

We associate a partial ordering $\prec$ on $[n]$ with a vector $x_\prec \in \{0, \frac{1}{2}, 1\}^{n(n-1)}$ by the assignment $x_{ij} = 0, 1, \frac{1}{2}$ when i is smaller than, bigger than, or incomparable to j, respectively.

Definition 7.2. A (partial) order $\prec$ is called *s-scaled* if there is a partition of $[n]$ into sets $A_1, A_2, \ldots, A_s$, such that $\prec$ is a total ordering within each of the A_i's and is not defined between elements in different A_i's.

Claim 7.3. *If $\prec$ is s-scaled with $s > 2$, then $x_\prec$ remains after $s-3$ rounds of GC or LS_0 cuts.*

The claim immediately provides a lower bound of $n-2$ for the rank of $P = GT_n$ since the vector associated with the empty order (which is n-scaled) has that rank.

Proof of Claim 7.3. By induction on s. Suppose $\prec$ is 3-scaled. We need to show that $x_\prec \in P = P^{(0)}$. Transitivity inequalities clearly hold for three elements in the same A_i. A transitivity inequality that involves more than one A_i must contain at least two variables with value $\frac{1}{2}$ and therefore must be satisfied by Proposition 2.1. The "no maximal element" inequalities also hold, because for every element there are at least two others to which it is not comparable, and the two associated $\frac{1}{2}$ values alone satisfy the inequality. For a general s we let $x = x_\prec$. Notice that $E(x)$ is a set of all edges connecting different components of the graph when we associate $\prec$ with a graph which is a union of s complete graphs. We partition the edges in $E(x)$ to $\binom{s}{2}$ sets by the components they connect and argue that x and this partition satisfy the conditions of Lemma 3.1. Indeed, for a choice of components A and B we denote by $\prec_A$ the order which is the same as $\prec$ except all the elemens of A are bigger than those of B. Similarly we define $\prec_B$. It is easy to see that $x = (x_{\prec_A} + x_{\prec_B})/2$. Since $\prec_A, \prec_B$ are $(s-1)$-scaled we inductively have that $rank(x_{\prec_A}), rank(x_{\prec_B}) \geq s-3$, and by Lemma 3.1 $rank(x) \geq s-2$. Notice that since Lemma 3.1 is strictly weaker than Lemma 3.2, the proof is valid for LS_0 in addition to GC. □

8 Automatizability of the LS-systems for Small-Rank CNF Formulas

In this section, we show that if $P \in [0,1]^n$ is a polytope that can be described by a polynomial number of halfspaces each with polynomial-length coefficients, then there is a procedure to test $P^{(r)}$ (the r-th round of LS or LS_0) for emptyness in time $n^{O(r)}$. This is important since it means that if P is a polytope that comes from an unsatisfiable CNF or set of mod-2 equations and P has small rank, then its unsatisfiability can be efficiently witnessed. The general idea of this theorem is evident in [33], but it is not explicitly stated in this form. Also, we rely heavily on the techniques explained in [25].

It is important to note that the ability to optimize over a polytope does not imply the ability to test it for emptyness. Indeed, optimization procedures generally assume non-emptyness. In particular, when [33] shows that it is possible to efficiently optimize over $P^{(r)}$ in LS_+ for small r, this does not seem to imply anything about testing for emptyness. See the end of this section for further explanation of this point.

A *strong separation oracle* for a convex body $P \subseteq \mathbb{R}^n$ is a procedure, that given $x \in \mathbb{R}^n$, either states that $x \in P$ or supplies a hyperplane separating x from P.

We say that a convex body $P \subseteq \mathbb{R}^n$ has *facet-complexity* φ if it can be represented as a set of linear inequalities (with rational coefficients) such that each of the inequalities can be encoded in length φ.

Theorem 8.1. *Assume we are given a strong separation oracle for a polytope $P \subseteq [0,1]^n$ of facet-complexity φ. Then, there is an algorithm for the LS and LS_0 proof systems that checks if $P^{(r)}$ is empty with running time $poly(n,\varphi)^r$.*

Note that for a polytope arising from CNF formulas, $\varphi = O(n)$, and consequently the running time is $n^{O(r)}$. The claim follows for LS from the following lemmas. For LS_0, the argument is very similar.

Lemma 8.2. *Let $P \subseteq [0,1]^n$ be a polytope with facet-complexity φ. Given a strong separation oracle $\mathcal{A}$ for P, there is a strong separation oracle for $P^{(1)}$ that makes $poly(n,\varphi)$ calls to $\mathcal{A}$.*

Lemma 8.3. *If a polytope $P \subseteq [0,1]^n$ has facet-complexity φ, then $P^{(1)}$ has facet-complexity bounded by $O(n^6 \cdot \varphi)$.*

Lemma 8.2 implies a strong separation oracle for $P^{(r)}$ with running time $\mathrm{poly}(n,\varphi)^r$. By Lemma 8.3 the facet-complexity of $P^{(r)}$ is bounded by $\varphi \cdot n^{O(r)}$. Theorem 6.4.9 from [25] states that we can check whether a polytope is empty by querying a strong separation oracle for that polytope. The number of queries required is polynomial in the facet-complexity and the dimension.

Proof of Lemma 8.2. As usual, we move to the cone $\overline{P}$ in $\mathbb{R}^{n+1}$ (see Section 2.1). It is easy to see that a strong separation oracle for P implies one for $\overline{P}$, and that the facet-complexity of P and $\overline{P}$ are the same. We define a cone $M(\overline{P})$ in $\mathbb{R}^{(n+1)^2}$ as the collection of $(n+1)\times(n+1)$ matrices Y satisfying

(i) Y is symmetric,

(ii) $Y_0 = \mathrm{diag}(Y)$,

(iii) $Y_i \in \overline{P}$,

(iv) $Y_0 - Y_i \in \overline{P}$,

where we denote by $Y_0, \ldots Y_n$ the columns of Y, and by $\mathrm{diag}(Y)$ its diagonal. From Definition 2.9, we have

$$P^{(1)} = \left\{ x \in \mathbb{R}^n : Y \in M(\overline{P}) \text{ and } Y_0 = \begin{pmatrix} 1 \\ x \end{pmatrix} \right\} \quad .$$

Let $x \in \mathbb{R}^n$. Consider the following polytope $Q_{x,\overline{P}}$ in $\mathbb{R}^{(n+1)^2}$.

$$Q_{x,\overline{P}} = \left\{ Y \in M(\overline{P}) | Y_0 = \begin{pmatrix} 1 \\ x \end{pmatrix} \right\} \quad .$$

By definition $x \in P^{(1)}$ if and only if $Q_{x,\overline{P}}$ is not empty. We first argue that $Q_{x,\overline{P}}$ has a separation oracle. To see that, observe that conditions (i) and (ii) above, as well as the condition that $Y_0 = \binom{1}{x}$, are all simple halfspaces and hyperplanes. Conditions (iii) and (iv) can be checked using the separation oracle for P. Since the facet-complexity of $Q_{x,\overline{P}}$ is bounded by φ, we can apply [25] Theorem 6.4.9 to obtain an algorithm that checks whether $Q_{x,\overline{P}}$ is empty, and consequently whether $x \in P^{(1)}$. Assume now that $x \notin P^{(1)}$. Along the above run of the algorithm (ending with the conclusion $Q_{x,\overline{P}} = \emptyset$), the separation oracle for P has been invoked a polynomial number of times, resulting in a polynomial number of halfspaces containing P. Let R be the intersection of those halfspaces. The crucial point to note here is that $Q_{x,R} = \emptyset$. This is since $Q_{x,R}$ and $Q_{x,\overline{P}}$ are indistinguishable to this run of the algorithm.

Let $(a_j, \cdot) \geq b_j$ be the halfspaces defining R. By the duality theorem, there is a positive combination $\vec{\alpha}$ of the inequalities $(a_j, Y_i) \geq b_j$ and $(a_j, Y_0 - Y_i) \geq b_j$ plus a combination of the inequalities of $M(\overline{P})$, such that (i) the coefficient vector of the Y variables is 0 and (ii) the constant term is of the form $\sum \alpha_i x_i \geq b > 0$. On the other hand, if $x \in P^{(1)}$ then $Q_{x,R}$ is not empty and so the same combination cannot lead to a contradiction and so $\sum \alpha_i x_i \leq 0$. This provides the desired separation. The only thing left to is to find the combination (the vector of coefficients α) that leads to the above contradiction. Here we use the fact that R has a polynomial number of faces, and so to find the combination satisfying both (i) and (ii) above is nothing but solving a polynomial linear program. ☐

We say that a cone has *vertex-complexity* ν if it is the span of a collection of rational vectors, each of which can be encoded in length ν.

Proof of Lemma 8.3. The facet-complexity of $M(\overline{P})$ is at most φ. Lemma 6.2.4 of [25] states that, for any polytope in $\mathbb{R}^d$ of facet-complexity φ and vertex-complexity ν, we have $\nu \leq 4d^2\varphi$ and $\varphi \leq 3d^2\nu$. Therefore, the vertex-complexity of $M(\overline{P})$ is at most $O(n^4\varphi)$. This bound also applies to the vertex-complexity of $\overline{P}^{(1)}$ since it is just a projection of $M(\overline{P})$. By the same lemma, the facet-complexity of $\overline{P}^{(1)}$ is $O(n^2 \cdot n^4\varphi)$, and our claim follows. $\square$

It is not clear how to test $P^{(r)}$ for emptyness efficiently in LS_+ because Lemma 8.2 does not seem to hold. In particular, $M(\overline{P})$ is now required to be positive semidefinite, so $Q_{x,\overline{P}}$ is defined by infinitely many linear inequalities.

9 Open Questions

Two of the major challenges in this area are to prove size lower bounds for GC or LS refutations of, say, random k-CNFs or the Tseitin Tautologies, and to prove rank lower bounds on LS as a means of approximating optimization problems such as Vertex Cover—that is, improving the bound of [2] (see [3, 37] for some improvements on this result). More immediate open questions are the following: Do our techniques for MAXSAT rank lower bounds apply to any other optimization problems? For example, [1] used similar techniques to prove rank bounds for hypergraph Vertex Cover and Set Cover. Does large GC-rank of a CNF imply large tree-like GC-size? Does large LS_+-rank imply large LS_+-size? If a CNF has a rank-r LS_+-refutation, can we find such a refutation in time $n^{O(r)}$? This would lead to a natural automated theorem prover that proves the PHP efficiently.

References

[1] * M. ALEKHNOVICH, S. ARORA, AND I. TOURLAKIS: Towards strong approximability results in the Lovász-Schrijver hierarchy. In *Proceedings of the Thirty-Seventh Annual ACM Symposium on Theory of Computing*, pp. 294–303, 2005. [STOC:1060590.1060634]. 1.1, 9

[2] * S. ARORA, B. BOLLOBÁS, AND L. LOVÁSZ: Proving integrality gaps without knowing the linear program. In *Proceedings of the Forty-Third Annual Symposium on Foundations of Computer Science*, pp. 313–322, 2002. [FOCS:10.1109/SFCS.2002.1181954]. 1.1, 5, 9

[3] * S. ARORA, B. BOLLOBÁS, L. LOVÁSZ, AND I. TOURLAKIS: Proving integrality gaps without knowing the linear program. *Theory of Computing*, 2(2):19–51, 2006. [ToC:v002/a002]. 5, 9

[4] * S. ARORA, S. RAO, AND U. VAZIRANI: Expander flows, geometric embeddings and graph partitioning. In *Proceedings of the Thirty-Sixth Annual ACM Symposium on Theory of Computing*, pp. 222–231, Chicago, IL, June 2004. [STOC:1007352.1007355]. 1.1

[5] * A. ATSERIAS, M. L. BONET, AND J. LEVY: On Chvátal rank and cutting planes proofs. Technical Report TR03-041, ECCC, 2003. [ECCC:TR03-041]. 1.1, 6, 6, 7

[6] * E. BEN-SASSON AND A. WIGDERSON: Short proofs are narrow – Resolution made simple. In *Proceedings of the Thirty-First Annual ACM Symposium on Theory of Computing*, pp. 517–526, Atlanta, GA, May 1999. [STOC:301250.301392]. 1.1, 6

[7] * A. BOCKMAYR, F. EISENBRAND, M. E. HARTMANN, AND A. S. SCHULZ: On the Chvátal rank of polytopes in the 0/1 cube. *Discrete Applied Mathematics*, 98(1-2):21–27, October 1999. [10.1016S0166-218X(99)00156-0]. 2.2

[8] * M. L. BONET AND N. GALESI: A study of proof search algorithms for Resolution and Polynomial Calculus. In *Proceedings of the Fortieth Annual Symposium on Foundations of Computer Science*, pp. 422–431, 1999. [FOCS:10.1109/SFFCS.1999.814614]. 7

[9] * J. BURESH-OPPENHEIM, N. GALESI, S. HOORY, A. MAGEN, AND T. PITASSI: Rank bounds and integrality gaps for cutting planes procedures. In *Proceedings of the Forty-Fourth Annual Symposium on Foundations of Computer Science*, pp. 318–327, 2003. [FOCS:10.1109/SFCS.2003.1238206]. 1.1

[10] * V. CHVÁTAL: Edmonds polytopes and a hierarchy of combinatorial problems. *Discrete Mathematics*, 4(4):305–337, 1973. [DiscMath:10.1016/0012-365X(73)90167-2]. 1, 2

[11] * V. CHVÁTAL, W. COOK, AND M. HARTMANN: On cutting-plane proofs in combinatorial optimization. *Linear Algebra and its Applications*, 114/115:455–499, 1989. [10.10160024-3795(89)90476-X]. 2.2, 3, 7

[12] * V. CHVÁTAL AND E. SZEMERÉDI: Many hard examples for resolution. *Journal of the ACM*, 35(4):759–768, 1988. [JACM:48014.48016]. 4

[13] * M. CLEGG, J. EDMONDS, AND R. IMPAGLIAZZO: Using the Gröbner basis algorithm to find proofs of unsatisfiability. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*, pp. 174–183, Philadelphia, PA, May 1996. [STOC:237814.237860]. 1

[14] * W. COOK, C. R. COULLARD, AND G. TURÁN: On the complexity of cutting plane proofs. *Discrete Applied Mathematics*, 18:25–38, 1987. [10.10160166-218X(87)90039-4]. 2.2, 7

[15] * 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. 1, 2.1

[16] * S. DASH: An exponential lower bound on the length of some classes of branch-and-cut proofs. *Mathematics of Operations Research*, 30(3):678–700, 2005. 1, 2.2

[17] * M. DAVIS AND H. PUTNAM: A computing procedure for quantification theory. *Journal of the ACM*, 7(3):201–215, 1960. [JACM:321033.321034]. 1

[18] * FRIEDRICH EISENBRAND AND ANDREAS S. SCHULZ: Bounds on the Chvátal rank of polytopes in the $0/1$-cube. In *Proceedings of the Seventh Annual Conference on Integer Programming and Combinatorial Optimization*, number 1610 in Lecture Notes in Computer Science, pp. 137–150, 1999. [IPCO:y0ruretrf6paebe1]. 2.2

[19] * U. FEIGE AND R. KRAUTHGAMER: The probable value of Lovász-Schrijver relaxations for maximum independent set. *SIAM Journal on Computing*, 32(2):345–370, 2003. [SICOMP:10.1137/S009753970240118X]. 1.1, 5

[20] * M. GOEMANS AND L. TUNÇEL: When does the postive semidefiniteness constraint help in lifting procedures. *Mathematics of Operations Research*, 26:796–815, 2001. 2.2, 3, 3.3

[21] * M. GOEMANS AND D. WILLIAMSON: Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM*, 42(6):1115–1145, 1995. [JACM:227683.227684]. 1.1

[22] * R. E. GOMORY: Solving linear programming problems in integers. In R. Bellman and M. Hall, Jr., editors, *Combinatorial Analysis*, pp. 211–215, Providence, RI, 1960. Symposia in Applied Mathematics X, American Mathematical Society. 1, 2

[23] * D. GRIGORIEV, E. A. HIRSCH, AND D. V. PASECHNIK: Complexity of semi-algebraic proofs. In *Proceedings of the Eighteenth Annual Symposium on Theoretical Aspects of Computer Science*, number 2285 in Lecture Notes in Computer Science, pp. 419–430, 2002. [STACS:t7lpe5yvaphpp0j6]. 1.1, 2.2, 6, 6.2

[24] * D. GRIGORIEV, E. A. HIRSCH, AND D. V. PASECHNIK: Exponential lower bound for static semi-algebraic proofs. In *Proceedings of the Twenty-Ninth International Colloquium on Automata, Languages and Programming*, number 2380 in Lecture notes in computer science, pp. 257–268, 2002. [ICALP:j2k07b37hmw2pfgw]. 6

[25] * MARTIN GRÖTSCHEL, LÁSZLÓ LOVÁSZ, AND ALEXANDER SCHRIJVER: *Geometric algorithms and combinatorial optimization*, volume 2 of *Algorithms and Combinatorics*. Springer-Verlag, Berlin, second edition, 1993. 8, 8

[26] * A. HAKEN: The intractability of resolution. *Theoretical Computer Science*, 39:297–305, 1985. [TCS:10.1016/0304-3975(85)90144-6]. 2.2

[27] * MARK HARTMANN: *Cutting Planes and the Complexity of the Integer Hull*. PhD thesis, Cornell University, 1988. Department of Operations Research and Industrial Engineering. 6, 6

[28] * J. HÅSTAD: Some optimal inapproximability results. *Journal of the ACM*, 48(4):798–859, 2001. [JACM:502090.502098]. 1.1, 5

[29] * E. A. HIRSCH AND A. KOJEVNIKOV: Several notes on the power of Gomory-Chvátal cuts. Technical Report TR03-012, ECCC, 2003. [ECCC:TR03-012]. 2.2

[30] * R. IMPAGLIAZZO, T. PITASSI, AND A. URQUHART: Upper and lower bounds on tree-like cutting planes proofs. In *Proceedings from Logic in Computer Science*, 1994. [10.1109LICS.1994.316069]. 2.2

[31] * 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. 1

[32] * B. KRISHNAMURTHY: Short proofs for tricky formulas. *Acta Informatica*, 22:253–275, 1985. [ActaInf:mp65776636126242]. 7

[33] * L. LOVÁSZ AND A. SCHRIJVER: Cones of matrices and set-functions and 0-1 optimization. *SIAM J. Optimization*, 1(2):166–190, 1991. [SIOPT:01/0801013]. 1, 1.1, 2, 2.2, 6, 6, 8

[34] * PAVEL PUDLÁK: Lower bounds for resolution and cutting plane proofs and monotone computations. *Journal of Symbolic Logic*, 62(3):981–998, September 1997. 1, 1.1, 2.2

[35] * G. STALMARK: Short resolution proofs for a sequence of tricky formulas. *Acta Informatica*, 33(3):277–280, 1996. [ActaInf:lhrhe2glkmgflbu1]. 7

[36] * T. STEPHEN AND L. TUNÇEL: On a representation of the matching polytope via semidefinite liftings. *Mathematics of Operations Research*, 24(1):1–7, 1999. 2.2

[37] * I. TOURLAKIS: New lower bounds for vertex cover in the Lovász-Schrijver hierarchy. Manuscript, 2005. 5, 9

[38] * I. TOURLAKIS: Towards optimal integrality gaps for hypergraph vertex cover in the Lovász-Schrijver hierarchy. In *Proceedings of Eighth International Workshop on Approximation Algorithms for Combinatorial Optimization Problems and Ninth International Workshop on Randomization and Computation*, pp. 233–244, 2005. 1.1

[39] * G. S. TSEITIN: On the complexity of derivation in the propositional calculus. In A. O. Slisenko, editor, *Studies in Constructive Mathematics and Mathematical Logic, Part II*. 1968. 4.8

AUTHORS

Joshua Buresh-Oppenheim
Postdoctoral Fellow
Simon Fraser University
jburesho@cs.sfu.ca
http://www.cse.ucsd.edu/users/jboppenheim

Nicola Galesi
Associate Professor
Università degli Studi di Roma La Sapienza
galesi@di.uniroma1.it
http://www.dsi.uniroma1.it/~galesi/

Shlomo Hoory
Post-Doctoral Fellow
University of British Columbia
shlomoh@cs.ubc.ca
http://www.cs.ubc.ca/~shlomoh

Avner Magen
Assistant Professor
University of Toronto
avner@cs.toronto.edu
http://www.cs.toronto.edu/~avner

Toniann Pitassi
Professor
University of Toronto
toni@cs.toronto.edu
http://www.cs.toronto.edu/~toni

ABOUT THE AUTHORS

JOSH BURESH-OPPENHEIM received his Ph. D. from the University of Toronto under the supervision of Toni Pitassi in 2005. Since then, he has been a post-doc at the University of California, San Diego, hosted by Russell Impagliazzo, and at Simon Fraser University, hosted by David Mitchell. His research interests include propositional proof complexity, classifying algorithmic techniques, and computational complexity in general. In his spare time, Josh is heir-apparent to the principality of Liechtenstein. When not on missions of diplomacy, he enjoys reënacting episodes of Welcome Back, Kotter, critiquing trade shows for the New Orleans Times Picayune, and fabricating biographies.

NICOLA GALESI received his Ph. D. in Computer Science from the Universitat Politecnica de Catalunya of Barcelona (Spain) in 2000 under the direction of Maria Luisa Bonet. At present, he is an associate professor in the Department of Computer Science at Università degli Studi di Roma "La Sapienza" (Italy). Formerly (01/05) he held a similar position in the Department of Computer Science of the Universitat Politecnica de Catalunya. He was a postdoc at the Institute for Advanced Study as a member of the School of Mathematics during the Special Year in Computational Complexity (00/01) and in 2002 he was a postdoc in the Department of Computer Science of the University of Toronto (Toronto, Canada) in the Theory Group. His main research interest is in complexity theory, especially in lower bounds in proof complexity.

In addition to spending time with his family, Nicola practices Taoist Tai Chi, is an avid reader and moviegoer, likes to play the guitar, and enjoys Japanese cuisine.

SHLOMO HOORY received his Ph. D. in Computer Science from the Hebrew University Jerusalem in 2002 under the supervision of Nati Linial. Since then he has been a postodoctoral fellow at the theory group at the University of Toronto and the University of British Columbia. His research interests include algebraic graph theory, expanders, graphs of high girth, cryptography, error correcting codes, and computation. He is especially interested in extremal properties of irregular graphs. Apart from academic life, he has practical experience from his long episodes in the high-tech industry. This experience includes programming, real time systems, OS internals, VLSI design, and digital signal processing.

AVNER MAGEN did his BSc, MSc and PhD at the Hebrew University Jerusalem under the supervision of Nati Linial. He is now an assistant professor at the Department of Computer Science, University of Toronto. His main interest areas are metric embeddings, lower bounds to combinatorial optimization problems and approximation algorithms. He enjoys travelling and hiking.

TONI PITASSI received bachelors and masters degrees from Pennsylvania State University and then received a PhD from the University of Toronto in 1992 under the supervision of Stephen Cook. After that, she spent 2 years as a postdoc at UCSD, and then 2 years as an assistant professor (in mathematics with a joint appointment in computer science) at the University of Pittsburgh. For the next four years, she was a faculty member of the Computer Science Department at the University of Arizona. In the fall of 2001, she moved back to Toronto, where she is currently a professor in the Computer Science Department. Her research interests include proof complexity, circuit complexity, classifying algorithmic techniques, analysis of SAT-solvers, and theory of machine learning.