Exponential lower bounds and Integrality Gaps for Tree-like Lovász-Schrijver Procedures

Toniann Pitassi*
Computer Science Department
University of Toronto
Toronto, Ontario Canada M5S 1A4
toni@cs.toronto.edu

Nathan Segerlind
INTEL Corporation
Hillsboro, Oregon, USA
nathan.l.segerlind@intel.com

June 18, 2008

Abstract

The matrix cuts of Lovász and Schrijver are methods for tightening linear relaxations of zero-one programs by the addition of new linear inequalities. We address the question of how many new inequalities are necessary to approximate certain combinatorial problems, and to solve certain instances of Boolean satisfiability.

Our first result is a size/rank tradeoff for tree-like Lovász-Schrijver refutations, showing that any refutation that has small size also has small rank. This allows us to immediately derive exponential size lower bounds for tree-like refutations of many unsatisfiable systems of inequalities where prior to our work, only strong rank bounds were known.

Unfortunately, we show that this tradeoff does not hold more generally for *derivations* of arbitrary inequalities. We give a very simple example showing that derivations can be very small but nonetheless require maximal rank. This rules out a *generic* argument for obtaining a size-based integrality gap from the corresponding rank-based integrality gap. Our second contribution is to show that a modified argument can often be used to prove size-based integrality gaps from rank-based integrality gaps. We apply this method to prove size-based integrality gaps for several prominant examples where prior to our work, only rank-based integrality gaps were known.

Our third contribution is to prove new separation results. Using our machinery for converting rank-based lower bounds and integrality gaps into size-based lower bounds, we show that tree-like LS_+ cannot polynomially simulate tree-like Cutting Planes, and that tree-like LS_+ cannot polynomially simulate resolution.

We conclude by examining size/rank tradeoffs beyond the LS systems. We show that for Shirali-Adams and Lasserre systems, size/rank tradeoffs continue to hold, even in the general (non-tree) case.

A full version of this paper is available at the Electronic Colloquium on Computational Complexity [23].

1 Introduction

The method of semidefinite relaxations has emerged as a powerful tool for approximating NP-complete problems. Central among these techniques are the lift-and-project methods of Lovász and Schrijver [22] (called LS and LS_+) for tightening a linear relaxation of a zero-one programming problem. For several optimization problems, a small number of applications of the semidefinite LS_+ Lovász-Schrijver operator transforms a simple linear programming relaxation into a tighter linear program that better approximates the zero-one program and yields a state-of-the-art approximation algorithm. For example, one round of LS_+, starting from the natural linear program for the independent set problem gives the Lovász Theta functions [21]; one round starting from the natural linear program for the max cut problem gives the famous Goemans-Williamson relaxation for approximating the maximum cut in a graph [14]; and three rounds gives the breakthrough Arora Rao Vazirani relaxation for the sparsest cut problem [6]. Moreover, linear and semi-definite programming methods are widely viewed as a catch-all approach for solving other approximation problems. To back this up, very recent work [7, 24] shows that for a general family of constraint satisfaction problems, the optimal approximation factor (which is actually unknown!) will be equal to the integrality gap obtained after a small number of rounds of matrix cut operators (under the unique games conjecture.)

Due to the importance and seemingly ubiquitous

* Research supported by NSERC.

nature of this family of algorithms, there has been a growing body of research aimed at ruling out low-rank LS_+ approximation algorithms for prominent approximation problems. These results prove that a very large family of semidefinite programs (those obtained by optimizing over a low-rank LS_+ polytope) will fail to achieve a good approximation by proving an *integrality gap*. (That is, exhibiting a nonintegral point lying in the polytope, whose value is off from the integral optimal by a certain approximation factor.) Such integrality gaps are important as they show that one of the most promising family of algorithms for solving these problems will not succeed in polynomial time. At present there are rank-based integrality gaps for LS and LS_+ for many important problems, including max-k-SAT, max-k-LIN, and vertex cover. (For example, see [5, 13, 25, 26, 27, 8, 20, 2, 10].)

While these algorithms rule out a large collection of SDP algorithms, they do not rule out all polynomial-time SDP algorithms. For example, it is certainly conceivable that there are inequalities that one might add that are natural for the problem at hand, but that are not derivable by low rank LS_+ from the initial set of inequalities. Such programs would not be ruled out by rank-based integrality gaps. In this paper we study the *size* of the LS_+ derivation needed to yield good approximations to optimization problems. Exponential (or even superpolynomial) *size*-based integrality gaps are the ultimate negative result as they show that *any* polynomial-time procedure based on LS (or LS_+) will fail to efficiently find an approximate solution (via standard rounding schemes.) In contrast, rank-based lower bounds only rule out algorithms that generate low-rank tightenings of the initial polytope.

We point out that lower bounds for LS_+ are incomparable to PCP-based lower bounds since on the one hand they are unconditional, but on the other hand, they rule out a large class of important algorithms (as opposed to all algorithms). As discussed above, there is an abundance of *rank-based* lower bounds and integrality gaps; however, with respect to the stronger size measure, very little is known. In fact the only unconditional size bound known is due to Kojevnikov and Itsykon [19]. Building on results from [15, 17, 16, 18]) they prove exponential size lower bounds for tree-like LS_+ derivations of certain unsatisfiable formulas (the Tseitin formulas). For integrality gaps, there were no size bounds at all. Our paper is largely inspired by the results of [19]. Can we prove size bounds for other unsatisfiable formula? What about size-based integrality gaps? Finally, what is the connection between size and rank?

1.1 Summary of results Our first result is a size/rank tradeoff for tree-like LS_0, LS, LS_+ refutations, showing that tree-like refutations can be converted into somewhat balanced refutations. More precisely, we prove the following. Suppose that I is a system of inequalities with a tree-like LS_+ (or LS, LS_0) refutation of size S. Then there is a refutation of I of rank at most $O(\sqrt{n \ln S})$. In particular, if I has a polynomial-size refutation, then it has a refutation of rank $O(\sqrt{n \log n})$. This tradeoff allows us to immediately derive exponential size lower bounds for tree-like refutations for several unsatisfiable systems of inequalities where prior to our work, only rank bounds were known (random 3CNF formulas and random systems of mod 2 equations). In other words, our lower bounds show that a large class of algorithms (those based on constructing tree-like LS_+ proofs) cannot solve SAT exactly in subexponential time. We note that this result is unconditional and rules out a broader class of algorithms than those ruled out by rank bounds.

The main idea behind our size/rank tradeoff is to define a new measure of complexity for a tree-like proof called the *variable rank*. We view a proof derivation as a tree where we label nodes with inequalities and edges with variables that are lifted on in this step. The rank of a proof is thus the longest path in the proof, whereas the variable rank is the largest number of *distinct* variable labels over all paths. Our key insight is to show that for any refutation, the variable rank equals the rank. This allows us to apply well-known methods for balancing the proof by iteratively applying restrictions to kill off long paths. We show that our tradeoff is optimal by exhibiting a family of formulas where our size/rank tradeoff is tight.

Next we try to attack the more interesting problem of proving superpolynomial size bounds for any LS_+ algorithm for approximating an optimization problem. This class of algorithms, say for max-k-SAT, is defined as follows. Begin with the natural polytope corresponding to an instance of max-k-SAT. Apply *any* sequence of LS_+ cuts to the initial polytope to obtain a new refined polytope. The size of the refined polytope is the number of cuts used to derive it from the initial polytope. The tree-size is the number of cuts used where we require that the underlying derivation is a tree. For a maximization problem, the refined polytope has an integrality gap of k if there is a solution with value at least k times OPT; for a minimization problem, the integrality gap is k if there is a solution with value OPT/k. For example, for vertex cover, we would like to show that any subexponential-size tree LS_+ algorithm has an integrality gap of 2. The most natural way to show this is to prove a stronger size/rank tradeoff for LS_+ that holds for *deriations* of arbitrary inequalities (instead of just for refutations, which are derivations of $0 \geq 1$.)

Unfortunately, we prove that this tradeoff does not hold more generally for *derivations* of arbitrary inequalities. We present a very simple example showing that derivations can be very small, but nonetheless require maximal rank. This rules out a *generic* argument for obtaining size-based integrality gaps from the corresponding rank-based integrality gaps. Despite our lack of a general tree-size/rank trade-off for derivations of arbitrary linear inequalities, our second main contribution is to show that a modified argument can often be used to prove size-based integrality gaps from rank-based integrality gaps. We illustrate this method by proving size-based integrality gaps for several optimization problems: We show that for max-k-SAT, every polytope that is obtained by applying an LS_+ tightening of sub-exponential tree-size has integrality gap $1+\frac{1}{2^k-1}$. Similarly we prove a size-based integrality gap of $2-\epsilon$ for max-k-LIN, and 7/6 for vertex cover.

Our third main contribution is to prove new separation results in proof complexity. Using our new machinery for converting rank-based lower bounds and integrality gaps into size-based lower bounds (combined with several new ideas), we show that tree-like LS_+ cannot polynomially simulate tree-like Cutting Planes, and that tree-like LS cannot polynomially simulate resolution. This shows in particular that low rank LS_+ cannot polynomially simulate Resolution. We conclude by examining size/rank tradeoffs beyond the LS systems. We show that for Shirali-Adams and Lasserre systems, size/rank tradeoffs continue to hold, even in the general (non-tree) case.

2 Matrix-cut proof systems

There are several cutting planes proof systems defined by Lovász and Schrijver, collectively referred to as matrix cuts [22]. In these proof systems, we begin with a system of linear inequalities over the variables X. We will present dual definitions for these systems: In the "proof-theoretic" one, we start with a system of linear inequalities and describe precise "cut" rules for obtaining new inequalities from previous ones. In the second "model-theoretic" definition, we will begin with a polytope defined as the set of solutions to the initial system of linear inequalities, and at each round, we will describe a new tightened polytope defined as the set of vectors in the original polytope that have a "protection matrix" associated with them.

2.1 Proof-theoretic View

DEFINITION 2.1. *Given a system of inequalities over $[0,1]^n$ defined by $a_i^T X \geq b_i$ for $i = 1,2,\ldots,m$: $j = 1,\ldots,n$. An inequality $c^T X - d$ is called an N_+-cut if*

$$\begin{aligned} c^T X - d \quad = \quad & \sum_{i=1}^{m}\sum_{j=1}^{n} \alpha_{i,j}(a_i^T X - b_i)X_j \\ & + \sum_{i=1}^{m}\sum_{j=1}^{n} \beta_{ij}(a_i^T X - b_i)(1-X_j) \\ & + \sum_{j=1}^{n} \lambda_j (X_j^2 - X_j) + \sum_k (g_k + h_k^T X)^2 \end{aligned}$$

where $\alpha_{i,j}, \beta_{i,j} \geq 0$, $\lambda_j \in \mathbb{R}$ for $i = 1,\ldots,m$, $j = 1,\ldots,n$ and for each k, $g_k \in \mathbb{R}$, $h_k \in \mathbb{R}^n$. An N-cut is a N_+-cut if $k = 0$. (That is, we cannot use squares of arbitrary linear inequalities.) An N_0-cut is an N-cut if the equality holds when we view X_iX_j as distinct from X_jX_i, $1 \leq i < j \leq n$. For each of the above cuts, we say that the inequality $a_i^T \geq b_i$ is a hypothesis of a lifting on the literal X_j (or $1 - X_j$) *if $\alpha_{ij} > 0$ (or $\beta_{ij} > 0$.)*

DEFINITION 2.2. *A Lovász-Schrijver (LS) derivation of $a^T X \geq b$ from a set of linear inequalities I is a sequence of inequalities $g_1, \ldots, g_q$ such that each g_i is either an inequality from I, or follows from previous inequalities by an N-cut as defined above, and such that the final inequality is $a^T X \geq b$. Similarly, a LS_0 derivation uses N_0-cuts and LS_+ uses N_+-cuts. An* elimination *of a point $x \in \mathbb{R}^n$ from I is a derivation from I of an inequality $c^T X \geq d$ such that $c^T x < d$. A* refutation of *I is a derivation of $0 \geq 1$ from I.*

DEFINITION 2.3. *Let $\mathcal{P}$ be one of the proof systems LS, LS_0 or LS_+. Let Γ be an $\mathcal{P}$-derivation from I, viewed as a directed acyclic graph. The derivation Γ is tree-like if each inequality in the derivation, other than the initial inequalities, is used at most once. The size of Γ is the total bit size of representing all inequalities, with all coefficients in binary notation. The rank of Γ is the depth of the underlying directed acyclic graph. For a set of boolean inequalities I, the $\mathcal{P}$-size ($\mathcal{P}$-tree-size, $\mathcal{P}$-rank) of I is the minimal size (tree-size, rank) over all $\mathcal{P}$ refutations of I. Define $LS_0^r(I)$ ($LS^r(I)$, $LS_+^r(I)$) to be the set of all linear inequalities with LS_0 (LS, LS_+) derivations from I of rank at most r.*

LEMMA 2.1. *(Closure under restrictions) Let Γ be an LS_0 (LS, LS_+) derivation of $c^T X \geq d$ from hypotheses I. Let ρ be a restriction to the variables of X. Then $\Gamma \restriction_\rho$ is an LS_0 (LS, LS_+) derivation of $\left(c^T X \geq d\right) \restriction_\rho$ from the hypotheses $I \restriction_\rho$.*

2.2 Model-theoretic view

DEFINITION 2.4. *Let $I = \{a_i^T X \geq b_i \mid i = 1,\ldots,m\}$ be a system of linear equalities in the variables $X_1, \ldots X_n$.*

Define the polytope of I as $P_I = \{x \in \mathbb{R}^n \mid \forall i \in [m], a_i^T x \geq b_i\}$.

Following the usual conventions, we will change the setting slightly by working with a convex cone rather than a convex set. Our object of interest is the convex set $P_I \subseteq [0,1]^n$. We first convert it into the *homogenized cone* $K_I \subseteq \mathbb{R}^{n+1}$, defined as $K_I = \{x \in \mathbb{R}^{n+1} \mid \forall i \in [m], a_i^T x - b_i x_0 \geq 0\}$. We will now define the various LS operators, N, N_+ and N_0 such that if K is a cone, then $N_+(K)$, $N(K)$ and $N_0(K)$ are also cones.

Definition 2.5. *Let $y \in \mathbb{R}^{n+1}$ be given, and let $K \subseteq \mathbb{R}^{n+1}$ be a cone. An* LS_0 protection matrix for y with respect to K *is a matrix $Y \in \mathbb{R}^{(n+1)\times(n+1)}$ such that: (1) $Ye_0 = diag(Y) = Y^T e_0 = y$. (The top row, leftmost column, and diagonal of Y are y.); (2) For all $i = 0, \ldots n$, $Ye_i \in K$ and $Y(e_0 - e_i) \in K$. (The i^{th} column and (y minus the i^{th} column) are in K.); (3) If $y_i = 0$ then $Ye_i = 0$, and if $y_i = y_0$ then $Ye_i = y$. If Y is also symmetric, then Y is said to be an* LS protection matrix. *If Y is also positive semidefinite, then Y is said to be an* LS_+ protection matrix.

Definition 2.6. *Let $K \subseteq \mathbb{R}^{n+1}$ be a cone. Define $N_0(K)$ to be set of $y \in \mathbb{R}^{n+1}$ such that there exists an LS_0 protection matrix for y with respect to K. We define $N(K)$ and $N_+(K)$ analogously. The sets $N_0(K)$, $N(K)$ and $N_+(K)$ are easily seen to be cones, and therefore the construction can be iterated. Inductively define $N_0^0(K) = K$ and $N_0^{r+1}(K) = N_0(N_0^r(K))$. Define $N^r(K)$ and $N_+^r(K)$ similarly. After applying the N operator iteratively to tighten the cone we will then want to project back to $X_0 = 1$ in order to get to the "tightened" polytope: let $K \restriction_{X_0=1} = \{x \in \mathbb{R}^n \mid (1, x_1, \ldots, x_n) \in K\}$.*

2.3 Equivalence between the two views

The connection between the N_0, N and N_+ operators, which work on cones in $\mathbb{R}^{n+1}$, and the syntactic definition of the LS_0, LS and LS_+ deduction systems is summarized in the following fundamental theorem of Lovász and Schrijver, stating that the polytope obtained after r rounds of the cut rule is equal to the polytope obtained after r iterations of the corresponding N operators, projected onto $X_0 = 1$.

Theorem 2.1. *[22] Let I be a set of inequalities in $\{X_1, \ldots X_n\}$ that includes the inequalities $0 \leq X_i \leq 1$ for all $i \in [n]$, and let $K_I \subseteq \mathbb{R}^{n+1}$ be the polyhedral cone given by the homogenization of I. Then $P_{LS_0^r(I)} = N_0^r(K_I) \restriction_{X_0=1}$, $P_{LS^r(I)} = N^r(K_I) \restriction_{X_0=1}$, and $P_{LS_+^r(I)} = N_+^r(K_I) \restriction_{X_0=1}$.*

Corollary 2.1. *Let I be a set of inequalities in $\{X_1, \ldots X_n\}$ that includes the inequalities $0 \leq X_i \leq 1$ for all $i \in [n]$, and let $K_I \subseteq \mathbb{R}^{n+1}$ be the polyhedral cone given by the homogenization of I. The following statements are equivalent: (1) There exists a rank $\leq r$ LS refutation of I; (2) Every point of $N^r(K_I)$ satisfies $0 \geq X_0$; (3) $N^r(K_I) \restriction_{X_0=1}$ is empty. Also, there exists a LS elimination of $x \in \mathbb{R}^n$ from I of rank at most r if and only if $\binom{1}{x} \notin N^r(K_I)$. Analogous statements relate LS_0 with N_0, and LS_+ with N_+.*

Definition 2.7. *Let $x \in [0,1]^n$. $Supp(x)$ are those coordinates i such that x_i is equal to 0 or 1. $E(x)$ are the other coordinates j such that x_j is not integral. Clearly $[n] = Supp(x) \cup E(x)$.*

Definition 2.8. *Let $x \in \mathbb{R}^n$ be given, and let Y be an LS_0 protection matrix for $\binom{1}{x}$. For each $i = 0, \ldots n$, let y^i be the bottom n entries of the $n+1$ dimensional column vector Ye_i, so that $Ye_i = \binom{x_i}{y^i}$. For $i \in E(x)$, let $PV_{i,1}(Y)$ denote the vector y^i/x_i and let $PV_{i,0}(Y)$ denote the vector $(x - y^i)/(1 - x_i)$. For $i \in Supp(x)$, let $PV_{i,0}(Y) = PV_{i,1}(Y) = x$. These $2n$ vectors are collectively known as the* protection vectors for x from Y.

The following lemma shows that if some $x \in K$ fails to make it into the next round of LS_+ tightening, then any candidate protection matrix Y for x will fail in the sense that one of the $2n$ alleged protection vectors will fail to be in K.

Lemma 2.2. *(proof in full version) Let $I = \{a_1^T X \geq b_1, \ldots a_m^T X \geq b_m\}$ be a system of inequalities. Let $c^T X \geq d$ be an inequality obtained by one round of LS_+ lift-and-project from I. Let $x \in \mathbb{R}^n$ be given such that $c^T x < d$. Let Y be a matrix for $\binom{1}{x}$ in the sense that it satisfies the definition of a protection matrix with the possible exception of property (2). Then there exists an $i \in [m]$ and a $j \in [n]$ so that either: (i) $a_i^T X \geq b_i$ is used as the hypothesis for a lifting inference on X_j, $x_j \neq 0$, and $a_i^T PV_{j,1}(Y) < b_i$, or (ii) $a_i^T X \geq b_i$ is used as the hypothesis for a lifting inference on $1 - X_j$, $x_j \neq 1$, and $a_i^T PV_{j,0}(Y) < b_i$.*

We will use the following form of Theorem 2.1, stating that if x is in $N(K)$, then there is a protection matrix Y for x such that all integral bits of x are preserved in all $2n$ protection vectors, and furthermore, the protection vector $PV(Y)_{i,\epsilon}$ that corresponds to lifting on x_i^ϵ also has its i^{th} bit set to ϵ.

Lemma 2.3. *(Proof in the full version) Let $x \in \mathbb{R}^n$. and let $K \subseteq \mathbb{R}^{n+1}$ be a cone that satisfies $0 \leq X_i \leq X_0$*

for all $i \in [n]$. Let $\binom{1}{x} \in N_0(K)$ $(N(K), N_+(K))$. Then there exists a LS_0 (LS, LS_+) protection matrix Y for $\binom{1}{x}$ with respect to K_I such that for each $i \in [n]$, $\epsilon \in \{0,1\}$, $Supp(x) \cup \{i\} \subseteq Supp(PV_{i,\epsilon}(Y))$.

Finally, we will use Farkas lemma which is kind of "completeness theorem" for linear programming:

LEMMA 2.4. *Let $I = \{a_i^T X \geq b_i \mid i = 1, \ldots, m\}$ be a system of inequalities so that for all x satisfying each inequality in I, $c^T x \geq d$. Then there exists $\alpha_1, \ldots, \alpha_m$, each $\alpha_i \geq 0$ such that $c^T X - d = \sum_{i=1}^m \alpha_i (a_i^T X - b_i)$.*

3 Tree-size versus rank

The high-level strategy for the our size/rank tradeoff is very similar to that used by Clegg, Edmonds and Impagliazzo, showing a relationship between degree and size for the polynomial calculus [11]. We first outline this general approach, and then explain the obstacles in using this approach and how we overcome them. As an example, we will outline how to transform a polynomial-size tree refutation into a low rank refutation. Consider the skeleton of the proof tree where nodes are labelled with inequalities and edges are labelled with the literal that is being lifted upon (multiplied by). If we can hit the proof with a restriction such that each long path contains at least one literal set to false, then this will result in a low rank proof, under the restriction. However the low-rank refutation will only be a refutation under the restriction and thus we must continue recursively and argue that there is also a low rank restriction under all other settings to the restricted variables. This will be possible since the size of the restriction will be small. Finally, we will combine all of the low rank refutations (one for each assignment to the restricted variables) in order to obtain a low rank refutation of the entire formula.

In our actual argument, we will select the restriction and recursive somewhat differently than described above, but the intuition is similar. Rather than selecting the whole restriction at once to kill all long paths simultaneously, we will select one variable setting at a time. We will always choose the next variable setting greedily, by picking the variable assignment that kills off the largest number of long paths. We argue that when the variable is set this way (the first case), the number of long paths drops by a large fraction, and when the variable is set the other way (the second case), the total number of variables is reduced by 1. In the first case, we will argue inductively that we can obtain a low rank $r-1$ refutation, and in the second case, a rank r refutation, and finally argue that they can be combined to obtain a rank r refutation.

When applying this argument we run into trouble because a path can be long without mentioning a lot of distinct literals on the edges of the path. A proof is called *regular* if for every path in the proof, a variable occurs in at most one edge labelling along the path. If the proof is regular, then we can apply the above argument. Unfortunately, the proof might be highly *irregular*, potentially making it impossible to apply the restriction argument. An extreme example would be a refutation tree containing two very long paths, one that mentions a literal x_i repeatedly, and another that mentions $\neg x_i$ repeatedly, thereby making it impossible to kill off both long paths simultaneously.

We get around this problem by arguing that in any refutation, if there is a long path, then there must exist another long *regular* path. More precisely, the rank of a tree refutation is the length of the longest path, and we define the *variable rank* of the tree refutation to be the maximum number of variables that are mentioned on a single path. (If the proof is regular, then these two notions of rank are equal.) Theorem 3.1 shows that rank and variable rank are equal. Note that we do not show that for any refutation tree, we can convert it into a regular refutation tree of the same rank. Nonetheless by controlling the irregularities in the proof, we can make the argument outlined above go through. We show that rank and variable rank are equal in Subsection 3.1, and we use this to prove the tree-size/rank trade-off in Subsection 3.2.

3.1 Variable rank measures how many distinct variables must be lifted upon along some path in a derivation. More precisely: Let I be a set of linear inequalities over the variables $X_1, \ldots, X_n$, and let Γ be a tree-like LS_+ derivation from I. Label the edges of the tree by the literal that is being lifted on in that inference. Let π be a path from an axiom to the final inequality. The *variable rank of* π is the number of distinct variables that appear as lift-variables in the edges of π. The *variable rank* of Γ is the maximum variable rank of any path from an axiom to the final inequality in Γ. For any inequality $c^T X \geq d$, the *variable rank of $c^T X \geq d$ with respect to I, $vrank^I(c^T X \geq d)$*, is defined to be the minimal variable rank of any derivation of $c^T X \geq d$. If there is no such derivation, then the variable rank is defined to be ∞. The variable rank of I, $\mathrm{vrank}(I)$, is defined to be $\mathrm{vrank}(0 \geq 1)$. The variable rank of a vector $x \in [0,1]^n$ with respect to I, $\mathrm{vrank}^I(x)$, is the minimum variable rank with respect to I of an inequality $c^T X \geq d$ such that $c^T x < d$.

THEOREM 3.1. *Let I be a set of inequalities, then for LS_0, LS and LS_+, for any x, $vrank^I(x) = rank^I(x)$.*

Proof. Let $x \in [0,1]^n$. Clearly $\text{vrank}^I(x) \le \text{rank}^I(x)$. We will prove the other direction by induction on $\text{rank}^I(x)$. We will show that for any x, if x has rank r, then any elimination of x must have a path that lifts on at least r distinct variables from $E(x)$. (Recall that $E(x)$ are those indices/coordinates of x that take on nonintegral values.) For $r = 0$ the proof is trivial.

For the inductive step, let x be a vector such that $\text{rank}^I(x) \ge r+1$. Let Γ be a minimum variable rank elimination of x that is frugal in the sense that x satisfies every inequality of Γ except for the final inequality. Let the final inference of Γ derive the inequality $c^T X - d$. By Lemma 2.3, there is a protection matrix Y for $\binom{1}{x}$ with respect to $N_+^r(P_I)$ satisfying the properties of the lemma. By Lemma 2.2, there exists $i \in [m]$ and $j \in [n]$ so that either $a_i^T X \ge b_i$ is the hypothesis of an X_j lifting and $a_i^T PV_{1,j}(Y) < b_i$, or $a_i^T X \ge b_i$ is the hypothesis of an $1 - X_j$ lifting and $a_i^T PV_{0,j}(Y) < b_i$.

Suppose that the lifting is on X_j (the case of $1-X_j$ is exactly the same). We now want to argue that j is not in $\text{Supp}(x)$. Suppose $j \in \text{Supp}(x)$. Then $PV_{0,j}(Y) = PV_{1,j}(Y) = x$. But this implies that $a_i^T x < b_i$ so Γ is not frugal, as we could have removed this last inference. Thus, we can assume that j is not in $\text{Supp}(x)$. Now let $y = PV_{j,1}(Y)$. Because Y is a protection matrix for $\binom{1}{x}$ with respect to $N_+^r(K_I)$, $y = PV_{j,1}(Y) \in N_+^r(K_I)$. Therefore y has rank r and by the induction hypothesis, this implies that this derivation of $a_i^T X \ge b_i$ must have some long path that lifts on at least r variables from $E(y)$. Consider this long path plus the edge labelled X_j from $a_i^T X \ge b_i$ to $c^T X \ge d$. We want to show that this path lifts on $r+1$ distinct variables from $E(x)$. First, let S be the set of r distinct variables from $E(y)$ that label the long path in the derivation of $a_i^T X \ge b_i$. By Lemma 2.3, these r variables are also in $E(x)$. Now consider the extra variable X_j labelling the edge from $a_i^T X \ge b_i$ to $c^T X \ge d$. We have argued above that j is in $E(x)$ but not in $E(y)$ and therefore X_j is distinct from S. Thus altogether we have $r+1$ distinct variables from $E(x)$ that are mentioned along this long path, completing the inductive step.

3.2 A tight trade-off for rank and tree-size

THEOREM 3.2. *For any set of inequalities I with no $0/1$ solution, in each of the systems LS_0, LS, and LS_+, $rank(I) \le 2\sqrt{2n \ln S_T(I)}$.*

We will need the following two preliminary lemmas.

LEMMA 3.1. *(Proof in full version) Let I be a system of inequalities over variables X_i, $i \in [n]$. For every $i \in [n]$, if there is a refutation of $I \restriction_{X_i=0}$ of rank r, then there is $\epsilon > 0$ and a derivation of $X_i \ge \epsilon$ from I of rank at most r. Similarly, if there is a refutation of $I \restriction_{X_i=1}$ of rank r, then there is $\epsilon > 0$ and a derivation of $(1 - X_i) \ge \epsilon$ from I of rank at most r.*

LEMMA 3.2. *(Proof in full version) For all systems of inequalities I, all positive integers r, and all $\epsilon, \delta > 0$, if there is a rank $\le r-1$ derivation from I of $X_i \ge \epsilon$ and a rank $\le r$ derivation from I of $1 - X_i \ge \delta$, then there is a rank $\le r$ refutation of I. If there is a rank $\le r-1$ derivation from I of $1 - X_i \ge \epsilon$ and a rank $\le r$ deriation from I of $X_i \ge \delta$, then there is a rank $\le r$ refutation of I.*

Proof. (of Theorem 3.2) Let $S \in \mathbb{N}$ be given. Let $d = \sqrt{2n \ln S}$, and let $a = (1 - d/2n)^{-1} = (1 - \sqrt{\ln S/2n})^{-1}$.

Let I be a set of inequalities in n variables, and let Γ be a refutation of I. Let F be the set of long paths in Γ of *variable rank* at least d. We prove by induction on n and b that if I is a system of inequalities in at most n variables that has a refutation with at most a^b long paths, then $\text{rank}(I) \le d + b$.

The claim trivially holds for all b when $d \ge n$, because every refutation that uses at most n variables has rank at most n. In the base case, $b = 0$ and there are no paths in Γ of variable rank more than d, and thus by Theorem 3.1, $\text{rank}(I) \le d$. For the induction step, suppose that $|F| < a^b$. Because there are $2n$ literals making at least $d|F|$ appearances in the $|F|$ many long paths, there is a literal X (here X is X_i or $1 - X_i$ for some $i \in [n]$) that appears in at least $\frac{d}{2n}|F|$ of the long paths. Setting $X = 0$, $\Gamma \restriction_{X=0}$ is a refutation of $I \restriction_{X=0}$ with at most $\left(1 - \frac{d}{2n}\right)|F| < a^{b-1}$ many long paths. By the induction hypothesis, $\text{rank}(I \restriction_{X=0}) \le d + b - 1$. By Lemma 3.1, there is $\epsilon \ge 0$ and a derivation of $1 - X \ge \epsilon$ from I of rank at most $d + b - 1$. On the other hand, $\Gamma \restriction_{X=1}$ is a refutation with at most $|F| < a^b$ many long paths, and in $n - 1$ many variables. By induction on the number of variables, $\text{rank}(I \restriction_{X=1}) \le d + b$. By Lemma 3.1, there is $\delta \ge 0$ and a derivation of $X \ge \delta$ from I of rank at most $d + b$. Therefore by Lemma 3.2, $\text{rank}(I) \le d + b$. This concludes the proof that if $|F| < a^b$, then $\text{rank}(I) \le d + b$.

Because $|F| < |\Gamma| = a^{\log_a(S)}$, we set $b = \log_a S$ which can be seen to be equal to $\sqrt{2n \ln S}$. Thus $\text{rank}(I) \le 2\sqrt{2n \ln S}$ as desired.

COROLLARY 3.1. *For the LS_0, LS and LS_+ systems, for any set of inequalities I in n variables with no $0/1$ solution, $S_T(I) \ge e^{(rank(I))^2/9n}$.*

It is interesting to note that we actually prove a stronger lower bound where size is measured to be the number of inequalities in the proof, and not just the bit size.

Up to logarithmic factors, the trade-off for rank and tree-size is asymptotically tight for LS_0 and LS refutations. This follows from well-known bounds for the propositional pigeonhole principle: On the one hand, it is shown in [16] that LS refutations of PHP_n^{n+1} require LS rank $\Omega(n)$, but on the other hand, there are tree-like LS_0 refutations of PHP_n^{n+1} of size $n^{O(1)}$ (this seems to be a folklore result).

THEOREM 3.3. *For each $n \in \mathbb{N}$, there is is a CNF F on $N = \Theta(n^2)$ many variables such that rank$(F) = \Omega\left(\sqrt{(N/\log N) \cdot \ln S_T(F)}\right)$.*

The propositional pigeonhole principle has a LS_+ refutation of rank one [16], so that example does not show the trade-off to be asymptotically tight for LS_+. Determining whether or not the trade-off is asymptotically tight for LS_+ is an interesting open question.

3.3 No trade-off for arbitrary derivations in LS_0 and LS

Theorem 3.2 shows that for LS or LS_+ *refutations*, strong enough rank lower bounds automatically imply tree-size lower bounds. But what about derivations of arbitrary inequalities? Somewhat counter-intuitively, a similar trade-off does not apply for LS or LS_0 derivations of arbitrary inequalities, nor for the elimination of points from a polytope. It is an interesting open problem to determine whether or not such a tree-size/rank tradeoff for arbitrary derivations holds for LS_+.

THEOREM 3.4. *For sufficiently large n, there exists a system of inequalities I over the variables $\{X_1, \dots X_n\}$ and an inequality $a^TX \le b$ such that: (1) Any LS derivation of $a^TX \le b$ from I requires rank $\Omega(n)$, and (2) There is a tree-like LS_0 derivation of $a^TX \le b$ from I of polynomial size.*

Proof. Let I be the following system of inequalities: For each $1 \le i < j \le n$, there is $X_i + X_j \le 1$. Let $a^TX \le b$ be the inequality $\sum_{i=1}^n X_i \le 1$. We show that deriving $a^TX \le b$ from I requires rank $\Omega(n)$. This is just a reduction from the well-known rank lower bound for LS refutations of PHP_{n-1}^n [16]. Let r be the minimum rank derivation of $\sum_{i=1}^n X_i \le 1$ from I. In the n to $n-1$ pigeonhole principle, there are clauses $X_{i,j} + X_{i',j} \le 1$ (for all $i, i' \in [n]$ with $i \ne i'$, and all $j \in [n-1]$), and $\sum_{j=1}^{n-1} X_{i,j} \ge 1$ (for all $i \in [n]$). In rank r we can derive $\sum_{i=1}^n X_{i,j} \le 1$ for each $j \in [n-1]$. Summing up over all j gives $\sum_{j=1}^{n-1}\sum_{i=1}^n X_{i,j} \le n-1$. On the other hand, there is a rank zero derivation of $\sum_{i=1}^n \sum_{j=1}^{n-1} X_{i,j} \ge n$ from the inequalities of PHP_{n-1}^n. Thus we have a rank r refutation of PHP_{n-1}^n. Because the LS rank of PHP_{n-1}^n is $\Omega(n)$, it follows that $r = \Omega(n)$. Lastly, it is not hard to show by induction on k that there is a polynomial tree-size LS_0 derivation of $\sum_{i=1}^k X_i \le 1$ from I.

It is interesting to note that for any ϵ, the system $I \cup \{\sum_{i=1}^n X_i \ge 1+\epsilon\}$ has a rank one LS_0 refutation. Finally, known bounds for the pigeonhole principle show that for LS_0 and LS, there is no tree-size/rank trade-off for eliminations of points.

THEOREM 3.5. *For sufficiently large $n \in \mathbb{N}$, there exists a set of inequalities I_n over $X_1, \dots, X_n$ and a point $x \in [0,1]^n$ such that there is a polynomial size tree-like LS_0 derivation of x from I_n, but any LS elimination of x requires rank $\Omega(n)$.*

Proof. As in the proof of Theorem 3.4, let I be the following system of inequalities: For each $1 \le i < j \le n$, there is $x_i + x_j \le 1$. By the argument of the proof of Theorem 3.4, all derivations of $\sum_{i=1}^n x_i \le 1$ from I require rank $r_0 = \Omega(n)$. Therefore, by the affine Farkas Lemma, Lemma 2.4, for all $r < r_0$ there exists $z \in N^r(P_I)$ such that $\sum_{i=1}^n z_i > 1$. Let x be such a point belonging to $N^{(r_0-1)}(P_I)$. On the other hand, there is a tree-like LS_0 derivation of $\sum_{i=1}^n x_i \le 1$ from I of size $n^{O(1)}$. Upon deriving $\sum_{i=1}^n x_i \le 1$, the point x is eliminated.

4 Tree-size Lower Bounds and Integrality Gaps

The tree-size/rank trade-off of Theorem 3.2 allows us to quickly deduce tree-size bounds from previously known rank bounds for LS_+ refutations of prominent "sparse and expanding" unsatisfiable formulas. Specifically, we derive exponential tree size lower bounds for the Tseitin principles, random 3CNF formulas, and random mod 2 linear equations.

DEFINITION 4.1. *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. Finally, the Tseitin formula on an odd-sized graph $G = (V,E)$, $P_{TS(G)}$, has variables x_e for all edges $e \in E$. For each $v \in V$ there is one corresponding equation: $\sum_{e, v \in e} x_e = 1$ mod 2.*

Our tree-size tradeoff together with the rank lower bounds from [8] immediately give the following theorem.

THEOREM 4.1. *1. For all odd n sufficiently large, there exists a G on n nodes and degree Δ such* *that any LS_+ refutation of $P_{TS(G)}$ require tree-size $2^{\Omega(n/\Delta)}$.*

2. *Let $k \geq 5$. There exists c such that for all constants $\Delta > c$, for $F \sim \mathcal{M}^{k,n}_{\Delta n}$, with probability $1 - o(1)$, all LS_+ refutations of P_F require tree-size $2^{\Omega(n)}$.*

3. *Let $k \geq 5$. There exists c such that for all constants $\Delta > c$, for $C \sim \mathcal{N}^{k,n}_{\Delta n}$, with probability $1 - o(1)$, all LS_+ refutations of P_C require tree-size $2^{\Omega(n)}$.*

The above proofs rely on the fact that for $k \geq 5$, the boundary expansion is greater than 2. In a subsequent paper, Alekhnovich, Arora and Tourlakis prove linear rank for random 3-CNFs [2]. This immediately yields the corresponding exponential tree-size lower bounds for random 3CNF formulas.

As discussed in Subsection 3.3, we cannot appeal to Theorem 3.2 to obtain tree-size based integrality gaps because this theorem holds for refutations but not for more general derivations. Nonetheless, we can obtain integrality gaps for sub-exponential tree-size LS and LS_+ relaxations by using similar ideas.

For max-k-SAT and max-k-LIN, we will actually manage to use Theorem 3.2 directly to prove integrality gaps. For vertex cover, we will demonstrate how to use the ideas behind the proof of 3.2 to obtain size-based integrality gaps based on rank-based integrality gaps using a more hand-tailored approach. This is completely analogous to using a hand-tailored random restriction argument to prove Resolution lower bounds, in cases where the general size-width tradeoff for Resolution cannot be applied.

Recall that the high level idea of the proof of Theorem 3.2 is to hit an alleged small proof with a restriction to kill off all high rank paths, and then figure out how to patch together the low-rank derivations (one where $x_i = 1$ and one where $x_i = 0$.) in a low-rank way. For derivations it is no longer possible to argue that we can patch together the low rank derivations, but we can bypass this step as follows: Begin with an alleged small-size derivation of some inequality g from I. Find a "nice" restriction ρ such that: (i) ρ kills off all high rank paths, and (ii) ρ has the property that $g \restriction_\rho$ still requires high rank.

Max-k-SAT and Max-k-LIN. 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. 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: $\sum_{j \in I_i} X_j \equiv a \pmod 2$, let f'_i be the equation $Y_i + \sum_{j \in I_i} X_j \equiv a + 1 \pmod 2$. 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 optimize the linear function $\sum_{i=1}^m Y_i$ subject to the constraints F'. Call this linear program L_F. In the same way, we can obtain a maximization problem, L_C, corresponding to a set of k clauses C.

THEOREM 4.2. *Let $k \geq 5$. For any constant $\epsilon > 0$, there are constants $\Delta, \beta > 0$ such that if $F \sim \mathcal{M}^{k,n}_{\Delta n}$ then the integrality gap of any size $s \leq 2^{\beta n}$ tree-like LS_+ relaxation of L_F is at least $2 - \epsilon$ with high probability. Similarly, for any $k \geq 5$ and any $\epsilon > 0$, there exists $\Delta, \beta > 0$ such that if $C \sim \mathcal{N}^{k,n}_{\Delta n}$, then the integrality gap of any size $s \leq 2^{\beta n}$-round relaxation of L_C is at least $\frac{2^k}{2^k - 1}$ with high probability.*

$LS+$ **Integrality Gap for Vertex Cover.** Given a 3XOR instance F over $\{X_1, \ldots X_n\}$ with $m = \Delta n$ equations, we define the FGLSS graph G_F as follows. G_F has $N = 4m$ vertices, one for each equation of F and for each assignment to the three variables that satisfies the equation. to three variables. Two vertices u and v are connected if and only if the partial assignments corresponding to u and v are inconsistent. The optimal integral solution for F is equal to the largest independent set in G_F. Note that $N/4$ is the largest possible independent set in G_F, where we choose one node from each 4-clique. The vertex cover and independent set problems on G_F is encoded in the usual way, with a variable $Y_{C,\eta}$ for each node (C, η) of G_F, where C corresponds to a 3XOR equation in F, and η is a satisfying assignment for C. Its polytope is denoted $VC(G_F)$. Our final result in this subsection is a generalization of the rank bound of [25] to a tree-size bound.

THEOREM 4.3. *For all $\epsilon > 0$, there exists $\Delta, c > 0$ such that for sufficiently large n, there exists F, a system of at most Δn many 3XOR equations over $\{X_1, \ldots X_n\}$ such that any tree-like LS_+ tightening of $VC(G_F)$ with integrality gap at most $7/6 - \epsilon$ has size at least 2^{cn}.*

5 Separations between proof systems

In this section, we show that tree-like LS_+ refutations can require an exponential-size increase to simulate several other proof systems. Our first theorem shows that tree-like LS_+ cannot efficiently simulate *Gomory-Chvatal* (GC) cutting planes, and our second theorem below shows in particular that small rank LS_+ cannot simulate resolution. The proofs of the following theorems (in the full version) first prove new rank bounds, and then use the machinery developed in Sections 3 and 4 to obtain size bounds from rank bounds.

THEOREM 5.1. *Tree-like LS_+ does not polynomially simulate GC cutting planes.*

THEOREM 5.2. *Tree-like LS_+ refutations cannot p-simulate either DAG-like resolution, or DAG-like LS_+.*

6 Tradeoffs beyond LS

We note that rank-size tradeoffs for Lassere and Shirali Adams proof systems are easier to obtain and moreover they are stronger. We will prove that for these systems, linear rank bounds imply exponential size bounds (and not just tree-size bounds as was the case for the LS systems.) For this section, let R be either Lassere or Shirali Adams.

THEOREM 6.1. *(Proof in full version) Let I be a system of inequalities with n underlying variables, and suppose that I has an R refutation of size S. Then I has a rank $O\sqrt{n \log S}$ R refutation.*

The idea behind the above theorem is very similar to Theorem 3.2. The first step is to show that any R proof can be *multilinearized* without increasing the rank. That is, if I is an system of inequalities with an R refutation, then the refutation can be converted into one where all inequalities are multilinear. This can be accomplished straightforwardly by adding appropriate quantities of $(x_i^2 - x_i)$. The second step is to reprove Lemmas 3.1 and 3.2 for R. These lemmas allow us to combine a rank $r-1$ refutation of $I_{x=0}$ with a rank r refutation of $I_{x=1}$ in order to obtain a rank r R-refutation of I. Finally, with these lemmas at hand, Theorem 6.1 can be proven analogously to the proof of Theorem 3.2.

References

[1] M. Alekhnovich. Lower bounds for k-DNF resolution on random 3-CNFs. In *Proceedings of the Thirty-Seventh Annual ACM Symposium on the Theory of Computing*, pages 251–256, 2005.

[2] M. Alekhnovich, S. Arora, and I. Tourlakis. Towards strong nonapproximability results in the lovasz-schrijver hierarchy. In *STOC*, pages 294–303, 2005.

[3] M. Alekhnovich, E. Hirsch, and D. Itsykson. Exponential lower bounds for the running times of DPLL algorithms on satisfiable formulas. *Journal of Automated Reasoning*, 35(1-3):51–72, 2005.

[4] M. Alekhnovich and A. Razborov. Lower bounds for the polynomial calculus: Non-binomial case. In *Proceedings of the Forty-Second Annual IEEE Symposium on Foundations of Computer Science*, pages 190–199, 2001.

[5] S. Arora, B. Bollobas, L. Lovasz, and I. Tourlakis. Proving integrality gaps without knowing the linear program. *Theory of Computing*, 2(2):19–51, 2006.

[6] 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*, pages 222–231, 2004.

[7] P. Austrin. Towards sharp inapproximability for any 2csp. In *FOCS*, pages 307–317, 2007.

[8] J. Buresh-Oppenheim, N. Galesi, S. Hoory, A. Magen, and T. Pitassi. Rank bounds and integrality gaps for cutting planes procedures. *Theory of Computing*, 2:65–90, 2006.

[9] S. Buss, D. Grigoriev, R. Impagliazzo, and T. Pitassi. Linear gaps between degrees for the polynomial calculus modulo distinct primes. In *Proceedings of the Thirty-First Annual ACM Symposium on Theory of Computing*, pages 547–556, Atlanta, GA, May 1999.

[10] M. Charikar, K. Makarychev, and Y. Makarychev. Integrality gaps for sherali-adams relaxations, 2007.

[11] 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*, pages 174–183, Philadelphia, PA, May 1996.

[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] W. de la Vega and C. Kenyon-Mathiew. Linear programming relaxations of maxcut. In *SODA*, 2007.

[14] 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.

[15] D. Grigoriev. Linear lower bound on degrees of positivstellensatz calculus proofs for the parity. *Theoretical Computer Science*, 259:613–622, 2001.

[16] D. Grigoriev, E. Hirsch, and D. Pasechnik. Complexity of semialgebraic proofs. *Moscow Mathematical Journal*, 2:647–679, 2002.

[17] D. Grigoriev and E. Hirsh. Algebraic proof systems over formulas. *Theoretical Computer Science*, 303:83–102, 2003.

[18] D. Grigoriev and N. Vorobjov. Complexity of null- and positivestellensatz proofs. *Annals of Pure and Applied Logic*, 113:153–160, 2001.

[19] A. Kojevnikov and D. Itsykson. Lower bounds of statis lovasz-schrijver calculus proofs for tseitin tautologies. In *ICALP*, pages 323–334, 2006.

[20] G. Konstantinos, A. Magen, T. Pitassi, and I. Tourlakis. Optimal integrality gaps for lovasz-schrijver relaxations of vertex cover. In *ECCC*, 2006.

[21] L. Lovász. On the Shannon capacity of a graph. *IEEE Transactions on Information Theory*, 25:1–7, 1979.

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

[23] T. Pitassi and N. Segerlind. Exponential lower bounds and integrality gaps for tree-like Lovasz-Schrijver procedures. Electronic Colloquium on Computational

Complexity, Technical Report TR07-107. Web address: `http://eccc.hpi-web.de`

[24] P. Raghavendra. Optimal algorithms and inapproximability results for every csp? In *STOC*, pages 245–254, 2008.

[25] G. Schoenebeck, L. Trevisan, and M. Tulsiani. A linear round lower bound for lovasz-schrijver sdp relaxations of vertex cover. In *ECCC*, number 98, 2006.

[26] G. Schoenebeck, L. Trevisan, and M. Tulsiani. Tight integrality gaps for lovasz-schrijver lp relaxations of vertex cover and max cut. In *ECCC*, 2006.

[27] I. Tourlakis. New lower bounds for vertex cover in the lovasz-schrijver hierarchy. In *IEEE Conference on Computational Complexity*, 2006.