THEORY OF COMPUTING, Volume 22 (2), 2026, pp. 1–38
www.theoryofcomputing.org

SPECIAL ISSUE: CCC 2021

On the Power and Limitations of Branch and Cut

Noah Fleming* Mika Göös Russell Impagliazzo

Toniann Pitassi Robert Robere† Li-Yang Tan‡

Avi Wigderson§

Received August 15, 2021; Revised February 16, 2026; Published June 3, 2026

Abstract. The Stabbing Planes proof system was introduced by Beame et al. (ITCS'18) to model the reasoning carried out in practical mixed integer programming solvers. As a proof system, it is powerful enough to simulate Cutting Planes and to refute the Tseitin formulas — certain unsatisfiable systems of linear equations

A preliminary version of this paper appeared in the Proceedings of the 36th Computational Complexity Conference (CCC'21)

*Supported by NSERC and Swedish Research Council grant 2025-06762.

†Supported by NSERC

‡Supported by NSF awards 1942123, 2211237, 2224246, a Sloan Research Fellowship, and a Google Research Award.

§Partially supported by NSF grant CCF-1900460.

ACM Classification: F.1.3, F.1.2, F.2.3

AMS Classification: 03F20,68Q25

Key words and phrases: proof complexity, lower bounds, integer programming, branch-and-cut, cutting planes

 DOI: 10.4086/toc.2026.v022a002

mod2 — which are canonical hard examples for many algebraic proof systems. In a recent (and surprising) result, Dadush and Tiwari (CCC'20) showed that these short refutations of the Tseitin formulas could be translated into quasi-polynomial size and depth Cutting Planes proofs, refuting a long-standing conjecture. This translation raises several interesting questions. First, whether all Stabbing Planes proofs can be efficiently simulated by Cutting Planes. This would allow for the substantial analysis done on the Cutting Planes system to be lifted to practical mixed integer programming solvers. Second, whether the quasi-polynomial depth of these proofs is inherent to Cutting Planes.

In this paper we make progress towards answering both of these questions. First, we show that *any* Stabbing Planes proof with bounded coefficients (SP^*) can be translated into Cutting Planes. As a consequence of the known lower bounds for Cutting Planes, this establishes the first exponential lower bounds on SP^*. Using this translation, we extend the result of Dadush and Tiwari to show that Cutting Planes has short refutations of any unsatisfiable system of linear equations over a prime finite field. Like the Cutting Planes proofs of Dadush and Tiwari, our refutations also incur a quasi-polynomial blow-up in depth, and we conjecture that this is inherent. As a step towards this conjecture, we develop a new *geometric* technique for proving lower bounds on the depth of Cutting Planes proofs. This allows us to establish the first lower bounds on the depth of *Semantic* Cutting Planes proofs of the Tseitin formulas.

1 Introduction

An effective method for analyzing classes of algorithms is to formalize the techniques used by the class into a *formal proof system*, and then analyze the formal proof system instead. By doing this, theorists are able to hide many of the practical details of implementing these algorithms, while preserving the class of methods that the algorithms can feasibly employ. Indeed, this approach has been applied to studying many different families of algorithms, such as

- *Conflict-driven clause-learning* algorithms for SAT [8, 51, 52], which can be formalized using *resolution* proofs [29].
- Optimization algorithms using *semidefinite programming* [37, 54], which can often be formalized using *Sums-of-Squares proofs* [41, 6].
- The classic *cutting planes* algorithms for integer programming [38, 20], which are formalized by *cutting planes proofs* [20, 21, 25].

In the present paper we continue the study of formal proof systems corresponding to modern integer programming algorithms. Recall that in the integer programming problem, we are given a polytope $P \subseteq \mathbb{R}^n$ and a vector $c \in \mathbb{R}^n$, and our goal is to find a point $x \in P \cap \mathbb{Z}^n$ maximizing $c \cdot x$. The classic approach to solving this problem — pioneered by Gomory [38] — is to add[1] *cutting*

[1]Throughout, we will say that a cutting plane, or an inequality is *added* to a polytope P to mean that it is added to the set of inequalities defining P.

planes to P. A *cutting plane* for P is any inequality of the form $ax \le \lfloor b \rfloor$, where a is an integral vector, b is rational, and *every* point of P is satisfied by $ax \le b$. By the integrality of a, it follows that cutting planes *preserve* the integral points of P, while potentially *removing* non-integral points from P. The cutting planes algorithms then proceed by heuristically choosing "good" cutting planes to add to P to try and locate the integral hull of P as quickly as possible.

As mentioned above, these algorithms can be naturally formalized into a proof system — the *Cutting Planes proof system*, denoted CP — as follows [20]. Initially, we are given a polytope P, presented as a list of integer-linear inequalities $\{a_i x \le b_i\}$. From these inequalities we can then deduce new inequalities using two deduction rules:

- *Linear Combination.* From inequalities $ax \le b, cx \le d$, deduce any non-negative linear combination of these two inequalities with integer coefficients.
- *Division Rule.* From an inequality $ax \le b$, if $d \in \mathbb{Z}$ with $d > 0$ divides all entries of a then deduce $(a/d)x \le \lfloor b/d \rfloor$.

A Cutting Planes *refutation* of P is a proof of the trivially false inequality $1 \le 0$ from the inequalities in P; clearly, such a refutation is possible only if P does not contain any integral points. While Cutting Planes has grown to be an influential proof system in propositional proof complexity, the original cutting planes algorithms suffered from numerical instabilities, as well as difficulties in finding good heuristics for the next cutting planes to add [38].

The modern algorithms in integer programming improve on the classical cutting planes method by combining them with a second technique, known as *branch-and-bound*, resulting in a family of optimization algorithms broadly referred to as *branch-and-cut algorithms*. These algorithms search for integer solutions in a polytope P by recursively repeating the following two procedures: First, P is split into smaller polytopes $P_1, \dots, P_k$ such that $P \cap \mathbb{Z}^n \subseteq \bigcup_{i \in [k]} P_i$ (i. e., *branching*). Next, cutting planes deductions are made in order to further refine the branched polytopes (i. e., *cutting*). In practice, branching is usually performed by selecting a variable x_i and branching on all possible values of x_i; that is, recursing on $P \cap \{x_i = t\}$ for each feasible integer value t. More complicated branching schemes have also been considered, such as branching on the hamming weight of subsets of variables [33], branching using basis-reduction techniques [2, 47, 1], and more general linear inequalities [53, 49, 44].

However, while these branch-and-cut algorithms are much more efficient in practice than the classical cutting planes methods, they are no longer naturally modelled by Cutting Planes proofs. So, in order to model these solvers as proof systems, Beame et al. [10] introduced the *Stabbing Planes* proof system. Given a polytope P containing no integral points, a *Stabbing Planes* refutation of P proceeds as follows. We begin by choosing an integral vector a, an integer b, and replacing P with the two polytopes $P \cap \{ax \le b-1\}$ and $P \cap \{ax \ge b\}$. Then, we recurse on these two polytopes, continuing until all descendant polytopes are empty (that is, they do not even contain any *real* solutions). The majority of branching schemes used in practical branch-and-cut algorithms (including all of the concrete schemes mentioned above) are examples of this general branching rule.

It is now an interesting question how the two proof systems — Cutting Planes and Stabbing Planes — are related. By contrasting the two systems we see at least three major differences:

– *Top-down vs. Bottom-up.* Stabbing Planes is a *top-down* proof system, formed by performing arbitrary queries on the polytope and recursing; while Cutting Planes is a *bottom-up* proof system, formed by deducing new inequalities from old ones.
– *Polytopes vs. Halfspaces.* Individual "lines" in a Stabbing Planes proof are *polytopes*, while individual "lines" in a Cutting Planes proof are *halfspaces*.
– *Tree-like vs. DAG-like.* The graphs underlying Stabbing Planes proofs are trees, while the graphs underlying Cutting Planes proofs are general DAGs: intuitively, this means that Cutting Planes proofs can reuse their intermediate steps, while Stabbing Planes proofs cannot.

When taken together, these facts suggest that Stabbing Planes and Cutting Planes could be incomparable in power, as polytopes are more expressive than halfspaces, while DAG-like proofs offer the power of line-reuse. Going against this intuition, Beame et al. proved that Stabbing Planes *can* actually efficiently simulate Cutting Planes [10] (see Figure 1) — this simulation was later extended by Basu et al. [7] to almost all types of cuts used in practical integer programming, including split cuts. Furthermore, Beame et al. proved that Stabbing Planes is *equivalent* to the proof system *tree-like* $\mathsf{R}(\mathsf{CP})$, denoted $\mathsf{TreeR}(\mathsf{CP})$, which was introduced by Krajíček [46], and whose relationship to Cutting Planes was previously unknown.

This leaves the converse problem — of whether Stabbing Planes can also be simulated by Cutting Planes — as an intriguing open question. Beame et al. conjectured that such a simulation was impossible, and furthermore that the *Tseitin formulas* provided a separation between these systems [10]. For any graph G and any $\{0,1\}$-labelling ℓ of the vertices of G, the *Tseitin formula* of (G,ℓ) is the following system of $\mathbb{F}_2$-linear equations: for each edge e we introduce a variable x_e , and for each vertex v we have an equation

$$\bigoplus_{u:uv\in E} x_{uv} = \ell(v)$$

asserting that the sum of the edge variables incident with v must agree with its label $\ell(v)$ (note such a system is unsatisfiable as long as $\sum_v \ell(v)$ is odd). On the one hand, Beame et al. proved that there are *quasi-polynomial size* Stabbing Planes refutations of the Tseitin formulas [10]. On the other hand, Tseitin formulas had long been conjectured to be exponentially hard for Cutting Planes [25], as they form one of the canonical families of hard examples for algebraic and semi-algebraic proof systems, including Nullstellensatz [40], Polynomial Calculus [19], and Sum-of-Squares [41, 62].

In a recent breakthrough, the long-standing conjecture that Tseitin was exponentially hard for Cutting Planes was *refuted* by Dadush and Tiwari [27], who gave *quasi-polynomial size* Cutting Planes refutations of Tseitin instances. Moreover, to prove their result, Dadush and Tiwari showed how to *translate* the quasipolynomial-size Stabbing Planes refutations of Tseitin into Cutting Planes refutations. This translation result is interesting for several reasons. First, it brings up the possibility that Cutting Planes *can actually* simulate Stabbing Planes. If possible, such a simulation would allow the significant analysis done on the Cutting Planes system to be lifted directly to branch-and-cut solvers. In particular, this would mean that the known exponential-size lower bounds for Cutting Planes refutations would immediately imply the first exponential lower bounds for these algorithms for arbitrary branching heuristics. Second, the translation converts *shallow* Stabbing Planes proofs into *very deep* Cutting Planes proofs: the Stabbing Planes refutation of Tseitin has depth $O(\log^2 n)$ and quasi-polynomial size, while the Cutting Planes refutation has quasipolynomial size *and* depth. This is quite unusual since simulations between proof systems typically preserve the structure of the proofs, and thus brings up the possibility that the Tseitin formulas yield a *supercritical* size/depth tradeoff – formulas with short proofs, requiring *superlinear* depth. For contrast: another simulation from the literature which emphatically does *not* preserve the structure of proofs is the simulation of *bounded-size* resolution by *bounded-width* resolution by Ben-Sasson and Wigderson [12]. In this setting, it is known that this simulation is tight [16], and even that there exist formulas refutable in resolution width w requiring maximal size $n^{\Omega(w)}$ [5]. Furthermore, under the additional assumption that the proofs are *tree-like*, Razborov [59] proved a supercritical trade-off between width and size.

1.1 Our results

1.1.1 A new characterization of Cutting Planes

Our first main result gives a *characterization* of Cutting Planes proofs as a natural subsystem of Stabbling Planes that we call *Facelike* Stabbing Planes. A Stabbing Planes query is *facelike* if one of the sets $P \cap \{ax \leq b-1\}$ or $P \cap \{ax \geq b\}$ is either empty or is a face of the polytope P, and a Stabbing Planes proof is said to be facelike if it only uses facelike queries. Our main result is the following theorem.

Theorem 1.1. *The proof systems* CP *and Facelike* SP *are polynomially equivalent.*

The proof of this theorem is a generalization of Dadush and Tiwari's upper bound for the Tseitin formulas. Indeed, the key tool underlying both their proof and ours is a lemma due to Schrijver [63] which allows us to simulate CP refutations of faces of a polytope, when beginning from P itself.

Using this equivalence we prove the following surprising simulation (see Figure 1), stating that Stabbing Planes proofs with relatively small coefficients (quasi-polynomially bounded in magnitude) can be quasi-polynomially simulated by Cutting Planes, provided the *diameter* $d(P)$ — that is, the maximum Euclidean distance between any two points in the polytope — is small.

Theorem 1.2. *Let P be a polytope and suppose there is an* SP *refutation of P with size s and maximum coefficient magnitude c. Then there is a Facelike* SP *refutation of P in size*

$$s(c \cdot d(P)\sqrt{n})^{\log s}.$$

It is interesting to contrast this with the work of Dadush and Tiwari [27], who show that any SP proof of size s of a polytope P can be assumed to have coefficients of magnitude $(nd(P))^{O(n^2)}$. An important corollary of the previous theorem is the following, which states that if the polytope P comes from an *unsatisfiable CNF formula* (which is the typical situation

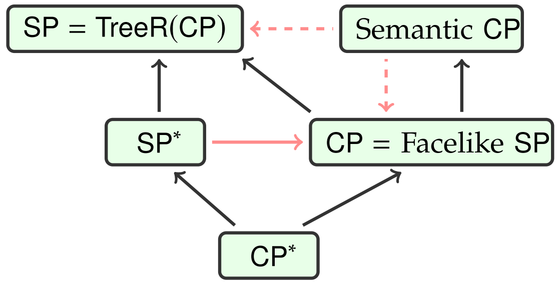


Figure 1: Known relationships between proof systems considered in this paper. A solid black (red) arrow from proof system P_1 to P_2 indicates that P_2 can polynomially (quasi-polynomially) simulate P_1. A dashed arrow indicates that this simulation cannot be done.

in proof complexity) then, since $P \subseteq [0,1]^n$, we have $d(P) \leq \sqrt{n}$ and thus we obtain a true quasi-polynomial simulation, even with coefficients of quasi-polynomial magnitude.

Let F be any unsatisfiable CNF formula on n variables, and suppose that there is a SP refutation of F in size s and maximum coefficient magnitude c. Then there is a CP refutation of F in size $s(cn)^{\log s}$.

As a second application of Theorem 1.1, we generalize the Dadush–Tiwari upper bound for Tseitin to show that Cutting Planes can refute any unsatisfiable system of linear equations over a prime finite field. This follows by showing that, like Tseitin, we can refute such systems of linear equations in quasi-polynomial-size Facelike SP.

Theorem 1.3. *Let F be the CNF encoding of an unsatisfiable system of m linear equations over a prime finite field. There is a* CP *refutation of F of size $|F|^{O(\log m)}$.*

This should be contrasted with the work of Filmus, Hrubeš, and Lauria [32], which gives several unsatisfiable systems of linear equations over $\mathbb{R}$ that require *exponential size* refutations in Cutting Planes (see Figure 1).

1.1.2 Lower Bounds

An important open problem is to prove superpolynomial size lower bounds for Stabbing Planes proofs. We make significant progress toward this goal by proving the first superpolynomial lower bounds on the size of low-weight Stabbing Planes proofs. Let SP* denote the family of Stabbing Planes proofs in which each coefficient has at most quasipolynomial ($n^{\log^{O(1)} n}$) magnitude.

Theorem 1.4. *There exists a family $\{F_n\}$ of unsatisfiable CNF formulas such that any* SP* *refutation of F requires size at least $2^{n^{\varepsilon}}$ for some constant $\varepsilon > 0$. In fact, we can obtain such lower bounds for Stabbing Planes proofs with coefficients bounded by $2^{n^{\delta}}$ for some constant $\delta > 0$.*

Our proof follows in a straightforward manner from Theorem 1.1.1 together with known Cutting Planes lower bounds. We view this as a step toward proving SP lower bounds (with no restrictions on the weight). Indeed, lower bounds for CP* (low-weight Cutting Planes) [17] were first established, and led to (unrestricted) CP lower bounds in [57].

Our second lower bound is a new linear depth lower bound for *semantic* Cutting Planes proofs. (In a semantic Cutting Planes proof the deduction rules for CP are replaced by a simple and much stronger *semantic deduction rule*).

Theorem 1.5. *For all sufficiently large n there is a graph G on n vertices and a labelling ℓ such that the Tseitin formula for (G, ℓ) requires $\Omega(n)$ depth to refute in Semantic Cutting Planes.*

We note that depth lower bounds for Semantic Cutting Planes have already been established via communication complexity arguments [43]. However, since Tseitin formulas have short communication protocols, our depth bound for semantic Cutting Planes proofs of Tseitin is new.

Theorem 1.5 is established via a new technique for proving lower bounds on the depth of semantic Cutting Planes proofs. Our technique is inspired by the result of Buresh-Oppenheim et al. [18], who proved lower bounds on the depth of Cutting Planes refutations of Tseitin by studying the *Chátal rank* of the associated polytope P. Letting $P^{(d)}$ be the polytope defined by all inequalities which can be derived in depth d in Cutting Planes. The Chátal rank of P is the minimum d such that $P^{(d)} = \emptyset$. Thus, in order to establish a depth lower bound of depth d, one would like to show the existence of a point $p \in P^{(d)}$. To do so, they give a sufficient criterion for a point p to be in $P^{(i)}$ in terms of the points in $P^{(i-1)}$. This criterion relies on a careful analysis of the specific rules of Cutting Planes, and is no longer sufficient for semantic CP. Instead, we develop an analogous criterion for semantic CP by using novel *geometric* argument (Theorem 5.10) which we believe will be of independent interest.

Our main motivation behind this depth bound is as a step towards proving a *supercritical* tradeoff in CP for Tseitin formulas. A supercritical tradeoff for CP, roughly speaking, states that small size CP proofs must sometimes necessarily be very deep — that is, beyond the trivial depth upper bound of $O(n)$ [59, 13]. (Observe that Dadush and Tiwari's quasipolynomial-size CP refutations of Tseitin are quasipolynomially deep; this is preserved by our simulation of Facelike Stabbing Planes by Cutting Planes in Theorem 1.1.) Establishing supercritical tradeoffs is a major challenge, both because hard examples witnessing such a tradeoff are rare, and because current methods seem to fail beyond the critical regime. In fact, to date the only supercritical tradeoffs between size and depth for known proof systems are due to Razborov, under the additional assumption that the proofs have *bounded width*. Namely, Razborov exhibited a supercritical size-depth tradeoff for bounded-width tree-like resolution [59], and then extended this result to CP proofs in which each inequality has a bounded number of distinct variables [60].

How could one prove a supercritical depth lower bound for Cutting Planes? All prior depth lower bounds for Cutting Planes proceed by either reducing to communication complexity, or by using so-called *protection lemmas* (e. g., [18]). Since communication complexity is always at most n, it will be useless for proving supercritical lower bounds directly. It therefore stands to reason that we should focus on improving the known lower bounds using protection lemmas and, indeed, our proof of Theorem 1.5 is a novel geometric argument which generalizes the top-down "protection lemma" approach [18] for syntactic CP. At this point in time we are currently unable to use protection lemma techniques to prove size-depth tradeoffs, so, we leave this as an open problem.

Conjecture 1.6. *There exists a family of unsatisfiable formulas $\{F_n\}$ such that F_n has quasipolynomial-size* CP *proofs, but any quasipolynomial-size proof requires superlinear depth.*

1.2 Related work

1.2.1 Lower bounds on SP and TreeR(CP)

Several lower bounds on subsystems of SP and TreeR(CP) have already been established: Krajíček [46] proved exponential lower bounds on the size of R(CP) proofs in which both the *width* of the clauses and the magnitude of the coefficients of each line in the proof are bounded. Concretely, let these bounds be w and c, respectively. The lower bound that he obtains is $2^{n^{\Omega(1)}}/c^{w \log^2 n}$. Kojevnikov [45] removed the dependence on the coefficient size for TreeR(CP) proofs, obtaining a bound of $\exp(\Omega(\sqrt{n/w \log n}))$. Beame et al. [10] provide a size-preserving simulation of Stabbing Planes by TreeR(CP) which translates a depth d Stabbing Planes proof into a width d TreeR(CP) proof, and therefore this implies lower bounds on the size of SP proofs of depth $o(n/\log n)$. Beame et al. [10] exhibit a function for which there are no SP refutations of depth $o(n/\log^2 n)$ via a reduction to the communication complexity of the CNF search problem.

1.2.2 Supercritical tradeoffs

Following Razborov's initial supercritical tradeoff [59], a number of further supercritical tradeoffs have been observed in proof complexity. Perhaps most relevant for our work, Razborov [60] proved a supercritical tradeoff for Cutting Planes proofs under the assumption that each inequality has a bounded number of distinct variables (mimicking the bound on the width of each clause in the supercritical tradeoff of [59]).

A number of supercritical tradeoffs are also known between proof width and proof *space*. Beame et al. [9] and Beck et al. [11] exhibited formulas which admit polynomial size refutations in Resolution and the Polynomial Calculus, respectively, and such that any refutation of sublinear space necessitates a superpolynomial blow-up in size. Recently, Berkholz and Nordström [13] gave a supercritical trade-off between width and space for Resolution.

1.2.3 Depth in Cutting Planes and Stabbing Planes

It is widely known (and easy to prove) that any unsatisfiable family of CNF formulas can be refuted by exponential size and *linear* depth Cutting Planes. It is also known that neither Cutting Planes nor Stabbing Planes can be *balanced*, in the sense that a depth-d proof can always be transformed into a size $2^{O(d)}$ proof [10, 18]. This differentiates both of these proof systems from more powerful proof systems like Frege, for which it is well-known how to balance arbitrary proofs [24]. Furthermore, even though both the Tseitin principles and systems of linear equations in prime finite fields can be proved in both quasipolynomial-size *and* $O(\log^2 n)$ depth in Facelike SP, the simulation of Facelike SP by CP *cannot* preserve both size and depth, as the Tseitin principles are known to require depth $\Theta(n)$ to refute in CP [18].

We first recall the known depth lower bound techniques for Cutting Planes, semantic Cutting Planes, and Stabbing Planes proofs. In all of these proof systems, arguably the primary method for proving depth lower bounds is by reducing to *real communication complexity* [43, 10]; however, communication complexity is always trivially upper bounded by n, and it is far from clear how to use the assumption on the size of the proof to boost this to superlinear.

A second class of methods have been developed for *syntactic* Cutting Planes, which lower bound *rank measures* of a polytope, such as the Chvátal rank. In this setting, lower bounds are typically proven using so-called *protection lemmas* [18], which seems much more amenable to applying a small-size assumption on the proof. We also remark that for many formulas (such as the Tseitin formulas!) it is known how to achieve $\Omega(n)$-depth lower bounds in Cutting Planes via protection lemmas, while proving even $\omega(\log n)$ lower bounds via communication complexity is impossible, due to a known folklore upper bound.

The first lower bound on the Chvátal rank was established by Chvátal et al. [22], who proved a linear bound for a number of polytopes in $[0,1]^n$. Much later, Pokutta and Schulz [56] characterized the polytopes $P \subseteq [0,1]$ with $P \cap \mathbb{Z}^n = \emptyset$ which have Chvátal rank exactly n. However, unlike most other cutting planes procedures, the Chvátal rank is not of polytopes $P \cap [0,1]^n$ with $P \cap \mathbb{Z}^n = \emptyset$ is not upper bounded by n. Eisenbrand and Schulz [31] showed that the Chvátal rank of any polytope $P \subseteq [0,1]^n$ is at most $O(n^2 \log n)$ and gave examples where it is $\Omega(n)$; a nearly-matching quadratic lower bound was later established by Rothvoß and Sanita [61]. For CNF formulas, the Chvátal rank is (trivially) at most n. Buresh-Oppenheim et al. [18] gave the first lower bounds on the Chvátal rank a number of CNF formulas, including an $\Omega(n)$ lower bound for the Tseitin formulas.

The rank of a number of generalizations of Cutting Planes has been studied as well. However, none of these appear to capture the strength of semantic Cutting Planes. Indeed, semantic Cutting Planes is able to refute Knapsack in a single cut, which can be generated in polynomial time, and therefore semantic Cutting Planes is not polynomially verifiable unless $\mathsf{P} = \mathsf{NP}$ [32]. Lower bounds on the rank when using split cuts and mixed integer cuts, instead of CG cuts, was established in [26]. Pokutta and Schulz [55] obtained $\Omega(n/\log n)$ rank lower bounds on the complete tautology (which includes every clause of width n) for the broad class of *admissible cutting planes*, which includes syntactic Cutting Planes, split cuts, and many of the lift-and-project operators. Bodur et al. [15] studied the relationship between rank and integrality gaps for another broad generalization of Cutting Planes known as *aggregate cuts*.

Changes from conference version [34] Minor corrections throughout, as well as an expanded discussion of related work and open problems.

2 Preliminaries

We first recall the definitions of some key proof systems.

Resolution. Fix an unsatisfiable CNF formula F over variables $x_1, \dots, x_n$. A *Resolution refutation* P of F is a sequence of clauses $\{C_i\}_{i \in [s]}$ ending in the empty clause $C_s = \emptyset$ such that each C_i is in

F or is derived from earlier clauses C_j, C_k with $j, k < i$ using one of the following rules:

– *Resolution.* $C_i = (C_j \setminus \{x_k\}) \cup (C_k \setminus \{\overline{x}_k\})$ where $x_k \in C_j$, $\overline{x}_k \in C_k$.
– *Weakening.* $C_i \supseteq C_j$.

The *size* of the resolution proof is s, the number of clauses. It is useful to visualize the refutation P as a directed acyclic graph; with this in mind the *depth* of the proof (denoted $\mathrm{depth}_{\mathsf{Res}}(P)$) is the length of the longest path in the proof DAG. The *resolution depth* $\mathrm{depth}_{\mathsf{Res}}(F)$ of F is the minimal depth of any resolution refutation of F.

Cutting Planes and Semantic Cutting Planes. A *Cutting Planes* (CP) *proof* of an inequality $cx \geq d$ from a system P of linear inequalities is given by a sequence of inequalities

$$a_1 x \geq b_1, a_2 x \geq b_2, \ldots, a_s x \geq b_s$$

such that $a_s = c$, $b_s = d$, and each inequality $a_i x \geq b_i$ is either in P or is deduced from earlier inequalities in the sequence by applying one of the two rules *Linear Combination* or *Division Rule* described at the beginning of Section 1. We will usually be interested in the case that the list P of inequalities defines a polytope, i. e., the set of solutions is bounded.

An alternative characterization of Cutting Planes uses *Chvátal–Gomory cuts* (or just *CG cuts*) [25, 20]. Let P be a polytope. A hyperplane $ax = b$ is *supporting* for P if $b = \max\{ax : x \in P\}$, and if $ax = b$ is a supporting hyperplane then the set $P \cap \{x \in \mathbb{R}^n : ax = b\}$ is called a *face* of P. An inequality $ax \leq b$ is *valid* for P if every point of P satisfies the inequality and $ax = b$ is a supporting hyperplane of P.

Definition 2.1. Let $P \subseteq \mathbb{R}^n$ be a polytope, and let $ax \geq b$ be any valid inequality for P such that all coefficients of a are relatively prime integers. The halfspace $\{x \in \mathbb{R}^n : ax \geq \lceil b \rceil\}$ is called a *CG cut* for P. (We will sometimes abuse notation and refer to the inequality $ax \geq \lceil b \rceil$ also as a CG cut.)

If $ax \geq \lceil b \rceil$ is a CG cut for the polytope P, then we can derive $ax \geq \lceil b \rceil$ from P in $O(n)$ steps of Cutting Planes by the Farkas Lemma (note that the inequality $ax \geq b$ is valid for P by definition, so we can deduce $ax \geq b$ as a linear combination of the inequalities of P and then apply the division rule). If P is a polytope and H is a CG cut, then we will write $P \vdash P \cap H$, and say that $P \cap H$ is *derived* from P.

Given a CNF formula F, we can translate F into a system of linear inequalities in the following natural way. First, for each variable x_i in F add the inequality $0 \leq x_i \leq 1$. If $C = \bigvee_{i \in P} x_i \vee \bigvee_{i \in N} \neg x_i$ is a clause in F, then we add the inequality

$$\sum_{i \in P} x_i + \sum_{i \in N} (1 - x_i) \geq 1.$$

It is straightforward to see that the resulting system of inequalities will have no integral solutions if and only if the original formula F is unsatisfiable. With this translation we consider Cutting

Planes refutations (defined in the introduction) of F to be refutations of the translation of F to linear inequalities.

The *semantic Cutting Planes* proof system (denoted sCP or Semantic CP) is a strengthening of Cutting Planes proofs to allow *any deduction* that is sound over Boolean points [17]. Like Cutting Planes, an sCP proof is given by a sequence of halfspaces $\{a_i x \geq c_i\}_{i \in [s]}$, but now we can use the following very powerful *semantic deduction rule*:

– *Semantic Deduction.* From $a_j x \geq c_j$ and $a_k x \geq c_k$ deduce $a_i x \geq c_i$ if every $\{0, 1\}$ assignment satisfying both $a_j x \geq c_j$ and $a_k x \geq c_k$ also satisfies $a_i x \geq c_i$.

Filmus et al. [32] showed that sCP is extremely strong: there are instances for which any refutation in CP requires exponential size, and yet these instances admit polynomial-size refutations in semantic sCP.

The size of a Cutting Planes proof is the number of lines. (It is known that for unsatisfiable CNF formulas, this measure is polynomially related to the length of the bit-encoding of the proof [25].) As with Resolution, it is natural to arrange Cutting Planes proofs into a proof DAG. With this in mind we analogously define $\mathrm{depth}_{\mathsf{CP}}(F)$ and $\mathrm{depth}_{\mathsf{sCP}}(F)$ to be the smallest depth of any (semantic) Cutting Planes proof of F.

It is known that *any* system of linear inequalities in the unit cube has CP depth at most $O(n^2 \log n)$, and moreover there are examples requiring CP-depth more than n [31]. However for unsatisfiable CNF formulas, the CP-depth is at most n [14].

Stabbing Planes. Let F be an unsatisfiable system of linear inequalities, meaning that F has no integral solutions. A *Stabbing Planes (*SP*) refutation* of F is a directed binary tree, T, where each edge is labelled with a linear integral inequality satisfying the following *consistency conditions*:

– *Internal Nodes.* For any internal node u of T, if the right outgoing edge of u is labelled with $ax \geq b$, then the left outgoing edge is labelled with its *integer negation* $ax \leq b - 1$.
– *Leaves.* Each leaf node v of T is labelled with a non-negative linear combination of inequalities in F with inequalities along the path leading to v that yields $0 \geq 1$.

For an internal node u of T, the pair of inequalities $(ax \leq b-1, ax \geq b)$ is called the *query* corresponding to the node. Every node of T has a polytope P associated with it, where P is the intersection of the halfspaces defined by the inequalities in F together with the inequalities labelling the path from the root to this node. We will say that the polytope P *corresponds* to this node. The *slab* corresponding to the query is $\{x^* \in \mathbb{R}^n \mid b - 1 < ax^* < b\}$, which is the set of points ruled out by this query. The *width* of the slab is the minimum distance between $ax \leq b - 1$ and $ax \geq b$, which is $1/\|a\|_2$. The *size* of a refutation is the bit-length needed to encode a description of the entire proof tree, which, for CNF formulas as well as sufficiently bounded systems of inequalities, is polynomially equivalent to the number of queries in the refutation [27]. As well, the *depth* of the refutation is the depth of the binary tree. The proof system SP* is the subsystem of Stabbing Planes obtained by restricting all coefficients of the proofs to have magnitude at most quasipolynomial ($n^{\log^{O(1)} n}$) in the number of input variables.

The Stabbing Planes proof system was introduced by Beame et al. [10] as a generalization of Cutting Planes that more closely modelled query algorithms and branch-and-bound solvers. Beame et al. proved that SP is equivalent to the proof system TreeR(CP) introduced by Krajíček [46] which can be thought of as a generalization of Resolution where the literals are replaced with integer-linear inequalities.

3 Translating Stabbing Planes into Cutting Planes

3.1 Equivalence of CP with subsystems of SP

In this section we prove Theorem 1.1, restated below, which characterizes Cutting Planes as a non-trivial subsystem of Stabbing Planes.

Theorem 1.1. *The proof systems* CP *and Facelike* SP *are polynomially equivalent.*

We begin by formally defining Facelike SP.

Definition 3.1. A Stabbing Planes query $(ax \leq b-1, ax \geq b)$ at a node P is *facelike* if one of the sets $P \cap \{x \in \mathbb{R}^n : ax \leq b-1\}$, $P \cap \{x \in \mathbb{R}^n : ax \geq b\}$ is empty or a face of P (see Figure 2b). An SP refutation is facelike if every query in the refutation is facelike.

Enroute to proving Theorem 1.1, it will be convenient to introduce the following further restriction of Facelike Stabbing Planes.

Definition 3.2. A Stabbing Planes query $(ax \leq b-1, ax \geq b)$ at a node corresponding to a polytope P is *pathlike* if at least one of $P \cap \{x \in \mathbb{R}^n : ax \leq b-1\}$ and $P \cap \{x \in \mathbb{R}^n : ax \geq b\}$ is empty (see Figure 2a). A Pathlike SP refutation is one in which every query is pathlike.

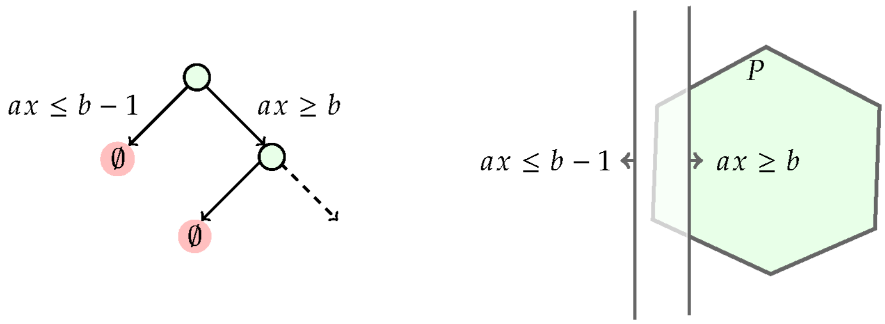
The name "pathlike" stems from the fact that the underlying graph of a pathlike Stabbing Planes proof is a path, since at most one child of every node has any children (see Figure 2). In fact, we have already seen (nontrivial) pathlike SP queries under another name: Chvátal–Gomory cuts.

Theorem 3.3. *Let P be a polytope and let $(ax \leq b-1, ax \geq b)$ be a pathlike Stabbing Planes query for P. Assume that $P \cap \{x \in \mathbb{R}^n : ax \leq b-1\} = \emptyset$ and that $P \cap \{x \in \mathbb{R}^n : ax \geq b\} \subsetneq P$. Then $ax \geq b$ is a CG cut for P.*

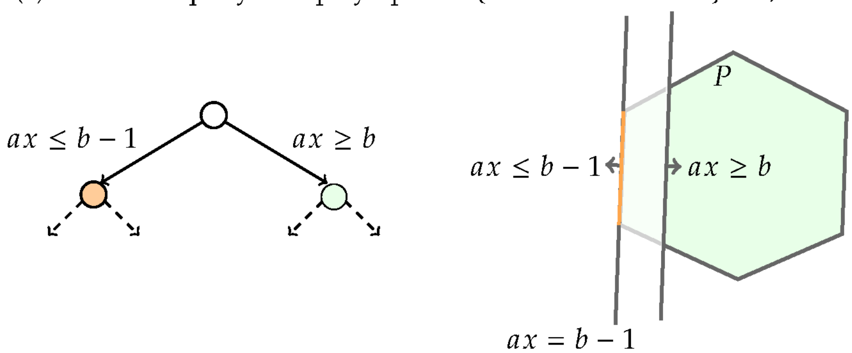
Proof. Since $ax \geq b$ is falsified by some point in P, it follows that there exists some $0 < \varepsilon < 1$ such that $ax \geq b - \varepsilon$ is valid for P — note that $\varepsilon < 1$ since otherwise $ax \leq b-1$ would not have empty intersection with P. This immediately implies that $ax \geq b$ is a CG cut for P. □

With this observation we can easily prove that Pathlike SP is equivalent to CP. Throughout the remainder of the section, for readability, we will use the abbreviation $P \cap \{ax \geq b\}$ for $P \cap \{x \in \mathbb{R}^n : ax \geq b\}$, for any polytope P and linear inequality $ax \geq b$.

Theorem 3.4. *Pathlike* SP *is polynomially equivalent to* CP*.*



(a) A Pathlike query. The polytope $P \cap \{x \in \mathbb{R}^n : ax \le b-1\} = \emptyset$, and $ax \ge b$ is a CG cut for P.



(b) A Facelike query. The polytope $P \cap \{x \in \mathbb{R}^n : ax \le b-1\} = P \cap \{x \in \mathbb{R}^n : ax = b-1\}$ is a face of P.

Figure 2: Pathlike and Facelike SP queries on a polytope P. On the left are the proofs and on the right are the corresponding effects on the polytope.

Proof. First, let $a_1x \ge b_1, a_2x \ge b_2, \ldots, a_sx \ge b_s$ be a CP refutation of an unsatisfiable system of linear inequalities $Ax \ge b$. Consider the sequence of polytopes $P_0 = \{Ax \ge b\}$ and $P_i = P_{i-1} \cap \{a_ix \ge b_i\}$. By inspecting the rules of CP, it can observed that $P_i \cap \{a_ix \le b_i - 1\} = \emptyset$ and thus P_{i+1} can be deduced using one pathlike SP query from P_i for all $0 \le i \le s$.

Conversely, let P be any polytope and let $(ax \le b-1, ax \ge b)$ be any pathlike SP query to P (so, suppose w.l.o.g. that the halfspace defined by $ax \le b-1$ has empty intersection with P). By Theorem 3.3, $ax \ge b$ is a CG cut for P, and so can be deduced in Cutting Planes from the inequalities defining P in length $O(n)$ (see Section 2). Applying this to each query in the Pathlike SP proof yields the theorem. □

Next, we show how to simulate Facelike SP proofs by Pathlike SP proofs of comparable size. The proof of Theorem 3.6 is inspired by Dadush and Tiwari [27], and will use the following lemma due to Schrijver [63] (although, we use the form appearing in [25]). Recall that we write $P \vdash P'$ for polytopes P, P' to mean that P' can be obtained from P by adding a single CG cut to P.

Theorem 3.5 (Lemma 2 in [25])**.** *Let P be a polytope defined by a system of integer linear inequalities and let F be a face of P. If $F \vdash F'$ then there is a polytope P' such that $P \vdash P'$ and $P' \cap F \subseteq F'$.*

Theorem 3.6. *Facelike* SP *is polynomially equivalent to Pathlike* SP.

Proof. That Facelike SP simulates Pathlike SP follows by the fact that any Pathlike SP query is a valid query in Facelike SP. For the other direction, consider an SP refutation π of size t. We describe a recursive algorithm for generating a Pathlike SP proof from π. The next claim will enable our recursive case.

Claim 3.7. *Let P be a polytope and suppose $ax \geq b$ is valid for P. Assume that $P \cap \{ax = b\}$ has a Pathlike* SP *refutation using s queries. Then there is a polytope contained in $P \cap \{ax \geq b+1\}$ which can be derived from P in Pathlike* SP *using $s+1$ queries.*

Proof of Claim. Since $ax \geq b$ is valid for P it follows that $F = P \cap \{ax = b\}$ is a face of P or empty by definition. Consider the Pathlike SP refutation $F_0, F_1, \ldots F_s = \emptyset$, where $F_0 = F$ and the ith polytope F_i for $i \geq 1$ is obtained from F_{i-1} by applying a pathlike SP query and proceeding to the non-empty child. Without loss of generality we may assume that $F_i \subsetneq F_{i-1}$ for all i, and so applying Theorem 3.3 we have that $F_{i-1} \vdash F_i$ for all i. Thus, by applying Theorem 3.5 repeatedly, we get a sequence of polytopes $P = P_0 \vdash P_1 \vdash \cdots \vdash P_s$ such that $P_i \cap F = P_i \cap \{ax = b\} \subseteq F_i$. This means that $P_s \cap \{ax = b\} \subseteq F_s = \emptyset$, and so $(ax \leq b,\ ax \geq b+1)$ is Pathlike SP query for P_s. This means that $P_s \vdash P_s \cap \{ax \geq b+1\} \subseteq P \cap \{ax \geq b+1\}$. Since any CG cut can be implemented as a Pathlike SP query the claim follows by applying the s CG cuts as pathlike queries, followed by the query $(ax \leq b,\ ax \geq b+1)$. ◻

We generate a Pathlike SP refutation by the following recursive algorithm, which performs an *in-order* traversal of π. At each step of the recursion (corresponding to a node in π) we maintain the current polytope P we are visiting and a Pathlike SP proof Π — initially, P is the initial polytope and $\Pi = \emptyset$. We maintain the invariant that when we finish the recursive step at node P, the Pathlike SP refutation Π is a refutation of P. The algorithm is described next:

1. Let $(ax \leq b-1,\ ax \geq b)$ be the current query. Because the query is pathlike, either $ax \leq b$ or $ax \geq b-1$ is valid for P; suppose that it is $ax \geq b-1$.
2. Recursively refute $P \cap \{ax \leq b-1\} = P \cap \{ax = b-1\}$, obtaining a Pathlike SP refutation Π with t queries.
3. Apply the above Claim to deduce $P \cap \{ax \geq b\}$ from P in $t+1$ queries.
4. Refute $P \cap \{ax \geq b\}$ by using the SP refutation for the right child.

Correctness follows immediately from the Claim, and also since the size of the resulting proof is the same as the size of the SP refutation. ◻

Theorem 1.1 then follows by combining Theorem 3.4 with Theorem 3.6.

3.2 Simulating SP* by CP

In this section we prove Theorem 1.2, restated below for convenience. Recall that if P is a polytope $d(P)$ is the *diameter* of P, which is the maximum Euclidean distance between any two points in P.

Theorem 1.2. *Let P be a polytope and suppose there is an* SP *refutation of P with size s and maximum coefficient magnitude c. Then there is a Facelike* SP *refutation of P in size*

$$s(c \cdot d(P)\sqrt{n})^{\log s}.$$

To prove this theorem, we will show that *any* low coefficient SP proof can be converted into a Facelike SP proof with only a quasi-polynomial loss.

Proof. The theorem is by induction on s. Clearly, if $s = 1$ then the tree is a single leaf and the theorem is vacuously true.

We proceed to the induction step. Let P be the initial polytope and π be the SP proof. Consider the first query $(ax \leq b,\ ax \geq b + 1)$ made by the proof, and let π_L be the SP proof rooted at the left child (corresponding to $ax \leq b$) and let π_R be the SP proof rooted at the right child. Let P_L denote the polytope at the left child and P_R denote the polytope at the right child. By induction, let π'_L and π'_R be the Facelike SP refutations for P_L and P_R guaranteed by the statement of the theorem.

Suppose w.l.o.g. that $|\pi_L| \leq |\pi|/2$. Let b_0 be the largest integer such that $ax \geq b_0$ is satisfied for any point in P. The plan is to replace the first query $(ax \leq b,\ ax \geq b + 1)$ with a sequence of queries $q_0, q_1, \ldots, q_{t-1}$ such that

- For each i, $q_i = (ax \leq b_0 + i,\ ax \geq b_0 + i + 1)$.
- The query q_0 is the root of the tree and q_i is attached to the right child of q_{i-1} for $i \geq 1$.
- $q_{t-1} = (ax \leq b,\ ax \geq b + 1)$.

After doing this replacement, instead of having two child polytopes P_L, P_R below the top query, we have $t + 1$ polytopes $P_0, P_1, \ldots, P_{t+1}$ where $P_i = P \cap \{ax = b_0 + i\}$ and $P_{t+1} = P_R$. To finish the construction, for each $i \leq t$ use the proof π'_L to refute P_i and the proof π'_R to refute P_{t+1}.

We need to prove three statements: this new proof is a valid refutation of P, the new proof is facelike, and that the size bound is satisfied.

First, it is easy to see that this is a valid proof, since for each $i \leq t$ the polytope $P_i \subseteq P_L$ and $P_{t+1} \subseteq P_R$ — thus, the refutations π'_L and π'_R can be used to refute the respective polytopes.

Second, to see that the proof is facelike, first observe that all the queries in the subtrees π'_L, π'_R are facelike queries by the inductive hypothesis. So, we only need to verify that the new queries at the top of the proof are facelike queries, which can easily be shown by a quick induction. First, observe that the query q_0 is a facelike query, since b_0 was chosen so that $ax \geq b_0$ is valid for the polytope P. By induction, the query $q_i = (ax \leq b_0 + i,\ ax \geq b_0 + i + 1)$ is a facelike query since the polytope P_i associated with that query is $P \cap \{ax \geq b_0 + i\}$ by definition. Thus $ax \geq b_0 + i$ is valid for the polytope at the query.

Finally, we need to prove the size upper bound. Let s be the size of the original proof, s_L be the size of π_L and s_R be the size of π_R. Observe that the size of the new proof is given by the recurrence relation

$$f(s) = t \cdot f(s_L) + f(s_R).$$

where $f(1) = 1$. Since the queries $q_0, q_1, \ldots, q_{t-1}$ cover the polytope P_L with slabs of width $1/\|a\|_2$, it follows that

$$t \leq d(P_L)\|a\|_2 \leq d(P)\sqrt{n}\|a\|_\infty = d(P)c\sqrt{n}$$

where we have used that the maximum coefficient size in the proof is c. Thus, by induction, the previous inequality, and the assumption that $s_L \leq s/2$, we can conclude that the size of the proof is

$$\begin{aligned} f(s) &\leq s_L(c \cdot d(P)\sqrt{n})(c \cdot d(P_L)\sqrt{n})^{\log s_L} + s_R(c \cdot d(P_R)\sqrt{n}+)^{\log s_R} \\ &\leq s_L(c \cdot d(P)\sqrt{n})(c \cdot d(P)\sqrt{n})^{\log(s/2)} + s_R(c \cdot d(P)\sqrt{n})^{\log s} \\ &\leq s_L(c \cdot d(P)\sqrt{n})^{\log s} + s_R(c \cdot d(P)\sqrt{n})^{\log s} \\ &\leq s(c \cdot d(P)\sqrt{n})^{\log s}. \end{aligned}$$

□

Theorem 1.1.1 follows immediately, since for any CNF formula F the encoding of F as a system of linear inequalities is contained in the n-dimensional cube $[0,1]^n$, which has diameter $\sqrt{n}$. We may also immediately conclude Theorem 1.4 by applying the known lower bounds on the size of Cutting Planes proofs [57, 35, 42, 36].

As a consequence of Theorem 1.1.1 and the NP-hardness of automating Cutting Planes [39], we can conclude that SP* proofs are NP-hard to find.

Corollary 3.8. *It is* NP*-hard to automate* SP*.

This follows by observing that the argument in [39] does not require large coefficients.

4 Refutations of linear equations over a prime finite field

In this section we prove Theorem 1.3. To do so, we will extend the approach used by Beame et al. [10] to prove quasi-polynomial upper bounds on the Tseitin formulas to work on any unsatisfiable set of linear equations over any prime finite field.

If $ax = b$ is a linear equation we say the *width* of the equation is the number of non-zero variables occurring in it. Any width-d linear equation over a prime finite field of size q, denoted $\mathbb{F}_q$, can be represented by a CNF formula with q^{d-1} width-d clauses — one ruling out each falsifying assignment. For a width-d system of m linear equations F over $\mathbb{F}_q$, we will denote by $|F| := mq^{d-1}$ the size of the CNF formula encoding F.

Theorem 4.1. *Let $F = \{f_1 = b_1, \ldots, f_m = b_m\}$ be a width-d, unsatisfiable set of linear equations over $\mathbb{F}_q$. There is an* SP *refutation of (the CNF encoding of) F in size $(mqd)^{O(\log m)}q^d = |F|^{O(\log m)}$.*

First we sketch the idea for $\mathbb{F}_2$, i. e., a system of XOR equations. In this case the SP proof corresponds to a branch decomposition procedure which is commonly used to solve SAT (see, e. g., [48, 3, 30, 28]). View the system F as a hypergraph over n vertices (corresponding to the variables) and with a d-edge for each equation. Partition the set of hyperedges into two sets $E = E_1 \cup E_2$ of roughly the same size, and consider the *cut* of vertices that belong to both an edge in E_1 and in E_2. Using the SP rule we branch on all possible values of the sum of the cut variables in order to isolate E_1 and E_2. Once we know this sum, we are guaranteed that either E_1 is unsatisfiable or E_2 is unsatisfiable depending on the parity of the of the sum of the cut variables. This allows us to recursively continue on the side of the cut (E_1 or E_2) that is unsatisfiable. Since there are n Boolean variables, each cut corresponds to at most $n+1$ possibilities for the sum, and if we maintain that the partition of the hyper edges defining the cut is balanced, then we will recurse at most $O(\log m)$ times. This gives rise to a tree decomposition of fanout $O(n)$ and height $O(\log n)$.

Over a prime finite field of size q the proof will proceed in much the same way. Instead of a subgraph, at each step we will maintain a subset of the equations $I \subseteq [m]$ such that $\{f_i = b_i\}_{i \in I}$ must contain a constraint that is violated by the SP queries made so far. We partition I into two sets I_1 and I_2 of roughly equal size and query the values a and b of $\sum_{i \in I_1} f_i$ and $\sum_{i \in I_2} f_i$. Because F is unsatisfiable, at least one of $a - \sum_{i \in I_1} b_i \not\equiv 0$ or $b - \sum_{i \in I_2} b_i \not\equiv 0$, meaning that that it is unsatisfiable, and we recurse on it.

In the following, we will let z stand for a vector of $\mathbb{F}_q$-valued variables z_i. When we discuss any form $f := az$ where $a \in \mathbb{F}_q^n$ and z is a vector of n variables z_i , we will implicitly associate it with the linear form $\sum_{i \in [n]} a_i (\sum_{j \in [\log q]} x_{i,j})$ where $x_{i,j}$ are the $\log q$ many Boolean variables encoding z_i in the CNF encoding of F.

Proof of Theorem 4.1. Let $F = \{f_1 = b_1, \ldots, f_m = b_m\}$ be a system of unsatisfiable linear equations over $\mathbb{F}_q$, where each $f_i = a_i z$ for $a_i \in \mathbb{F}_q^n$, and $b_i \in \mathbb{F}_q$. Because F is unsatisfiable, there exists a $\mathbb{F}_q$ linear combination of the equations in F witnessing this; formally, there exists $\alpha \in \mathbb{F}_q^n$ such that $\sum_{i \in [m]} \alpha_i f_i \equiv 0 \mod q$, but $\sum_{i \in [m]} \alpha_i b_i \not\equiv 0 \mod q$.

Stabbing Planes will implement the following binary search procedure for a violated equation; we describe the procedure first, and then describe how to implement it in Stabbing Planes. In each round we maintain a subset $I \subseteq [m]$ and an integer k_I representing the value of $\sum_{i \in I} \alpha_i f_i$. Over the algorithm, we maintain the invariant that $k_I - \sum_{i \in I} \alpha_i b_i \not\equiv 0 \bmod q$, which implies that there must be a contradiction to F inside of the constraints $\{f_i = b_i\}_{i \in I}$.

Initially, $I = [m]$ and we obtain k_I by querying the value of the sum $\sum_{i \in [m]} \alpha_i f_i$. If $k_I \not\equiv 0 \bmod q$ then this contradicts the fact that $\sum_{i \in I} \alpha_i f_i \equiv 0 \mod q$; thus, the invariant holds. Next, perform the following algorithm.

1. Choose a balanced partition $I = I_1 \cup I_2$ (so that $||I_1| - |I_2|| \leq 1$).
2. Query the value of $\sum_{i \in I_1} \alpha_i f_i$ and $\sum_{i \in I_2} \alpha_i f_i$; denote these values by a and b, respectively.
3. If $a - \sum_{i \in I_1} \alpha_i b_i \not\equiv 0 \bmod q$ then recurse on I_1 with $k_{I_1} := a$. Otherwise, if $b - \sum_{i \in I_2} \alpha_i b_i \not\equiv 0 \bmod q$ then recurse on I_2 with $k_{I_2} := b$.
4. Otherwise (if $a - \sum_{i \in I_1} \alpha_i b_i \equiv b - \sum_{i \in I_2} \alpha_i b_i \equiv 0 \bmod q$), then this contradicts the invariant:

$$\begin{aligned} 0 \not\equiv k_I - \sum_{i \in I} \alpha b_i &= \sum_{i \in I} \alpha_i (f_i - b_i) \\ &= \sum_{i \in I_1} \alpha_i (f_i - b_i) + \sum_{i \in I_2} \alpha_i (f_i - b_i) \\ &= (a - \sum_{i \in I_1} \alpha_i b_i) + (b - \sum_{i \in I_1} \alpha_i b_i) \equiv 0 \mod q. \end{aligned}$$

This recursion stops when $|I| = 1$, at which point we have an immediate contradiction between k_I and the single equation indexed by I.

It remains to implement this algorithm in SP. First, we need to show how to perform the queries in step 2. Querying the value of any sum $\sum_{i\in I}\alpha_i f_i$ can be done in a binary tree with at most q^2md leaves, one corresponding to every possible query outcome. Internally, this tree queries all possible integer values for this sum (e. g., $(\sum_{i\in I}\alpha_i f_i \le 0, \sum_{i\in I}\alpha_i f_i \ge 1), (\sum_{i\in I}\alpha_i f_i \le 1, \sum_{i\in I}\alpha_i f_i \ge 2), \ldots$). For the leaf where we have deduced $\sum_{i\in[m]}\alpha_i f_i \le 0$ we use the fact that each variable is non-negative to deduce that $\sum_{i\in[m]}\alpha_i f_i \ge 0$ as well. Note that q^2md is an upper bound on this sum because there are m equations, each containing at most d variables, each taking value at most $(q-1)$ [2]. Thus, step 2 can be completed in $(q^2md)^2$ queries.

Finally, we show how to derive refutations in the following cases: (i) when we deduced that $\sum_{i\in[m]}\alpha_i f_i \not\equiv 0 \mod q$ at the beginning, (ii) in step 4, (iii) when $|I| = 1$.

(i) Suppose that we received the value $a \not\equiv 0 \mod q$ from querying $\sum_{i\in[m]}\alpha_i f_i$. Note that the coefficient of every variable in $\sum_{i\in[m]}\alpha_i f_i$ is a multiple of q. Query

$$\Big(\sum_{i\in[m]}\alpha_i f_i/q \le \lceil a/q\rceil - 1,\ \sum_{i\in[m]}\alpha_i f_i/q \ge \lceil a/q\rceil\Big).$$

At the leaf that deduces $\sum_{i\in[m]}\alpha_i f_i/q \le \lceil a/q\rceil - 1$, we can derive $0 \ge 1$ as a non-negative linear combination of this inequality together with $\sum_{i\in[m]}\alpha_i f_i \ge a$. Similarly, at the other leaf $\sum_{i\in[m]}\alpha_i f_i/q \ge \lceil a/q\rceil$ can be combined with $\sum_{i\in[m]}\alpha_i f_i \le a$ to derive $0 \ge 1$.

(ii) Suppose that $a - \sum_{i\in I_1}\alpha_i b_i \equiv b - \sum_{i\in I_2}\alpha_i b_i \equiv 0 \bmod q$. Then $0 \ge 1$ is derived by summing $\sum_{i\in I_1}\alpha_i f_i \ge a$, $\sum_{i\in I_2}\alpha_i f_i \ge b$ and $\sum_{i\in I}\alpha_i f_i \le k_I$, all of which have already been deduced.

(iii) When $|I| = 1$ then we deduced that $a_I z = k_I$ for $k_I \not\equiv b_I \mod q$ and we would like to derive a contradiction using the axioms encoding $a_I z \equiv b_I$. These axioms are presented to SP as the linear-inequality encoding of a CNF formula, and while there are no integer solutions satisfying both these axioms and $a_I z = k_I$, there could in fact be *rational* solutions. To handle this, we simply force that each of the at most d variables in $a_I z$ takes an integer value by querying the value of each variable one by one. As there are at most d variables, each taking an integer value between 0 and $q-1$, this can be done in a tree with at most q^d many leaves. At each leaf of this tree we deduce $0 \ge 1$ by a non-negative linear combination with the axioms, the integer-valued variables, and $a_I z \equiv b_I$.

The recursion terminates in at most $O(\log m)$ many rounds because the number of equations under consideration halves every time. Therefore, the size of this refutation is $(qmd)^{O(\log m)}q^d$. Note that by making each query in a balanced tree, this refutation can be carried out in depth $O(\log^2(mqd))$. □

Finally, we conclude Theorem 1.3.

Proof of Theorem 1.3. Observe that the SP refutation from Theorem 4.1 is facelike. Indeed, to perform step 2 we query $(\sum_{i\in I}\alpha_i f_i \le t-1, \sum_{i\in I}\alpha_i f_i \ge t)$ from $t = 1, \ldots, q^2md$. For $t = 1$, the

[2] Note that instead of querying the value of $\sum_{i\in I}\alpha_i f_i$ we could have queried $\sum_{i\in I}\alpha_i f_i \pmod q$ to decrease the number of leaves to qmd.

halfspace $\sum_{i \in I} \alpha_i f_i \geq 0$ is valid for the current polytope because the polytope belongs to the $[0,1]^n$ cube. For each subsequent query, $\sum_{i \in I} \alpha_i f_i \geq t - 1$ is valid because the previous query deduced $\sum_{i \in I} \alpha_i f_i \geq t - 1$. Similar arguments show that the remaining queries are also facelike. Thus, Theorem 3.6 completes the proof. □

We note that the CP refutations that result from Theorem 1.3 have a very particular structure: they are extremely long and narrow. Indeed, they have depth $n^{O(\log m)}$. We give a rough sketch of the argument: it is enough to show that most lines L_i in the CP refutation are derived using some previous line L_j with $j = O(i)$. This is because the final line would have depth proportional to the size of the proof. To see that the CP refutation satisfies this property, observe that for each node visited in the in-order traversal, the nodes in the right subproof π_R depend on the halfspace labelling the root, which in turn depends on the left subproof π_L.

5 Lower bound on the depth of semantic CP refutations

Our results from Section 3 suggest an interesting interplay between depth and size of Cutting Planes proofs. In particular, we note that there is a *trivial* depth n and exponential size refutation of any unsatisfiable CNF formula in Cutting Planes; however, it is easy to see that the Dadush–Tiwari proofs and our own quasipolynomial size CP proofs of Tseitin are also extremely deep (in particular, they are *superlinear*). Even in the stronger *Semantic* CP it is not clear that the depth of these proofs can be decreased. However, this does not hold for SP, which has quasi-polynomial size and poly-logarithmic depth refutations. This motivates Theorem 1.6, regarding the existence of a "supercritical" trade-off between size and depth for Cutting Planes [59, 13]. The Tseitin formulas are a natural candidate for resolving this conjecture.

In this section we develop a new method for proving depth lower bounds which we believe should be more useful for resolving this conjecture. Our method works not only for CP but also for semantic CP. Using our technique, we establish the first linear lower bounds on the depth of Semantic CP refutations of the Tseitin formulas.

Lower bounds on the depth of *syntactic* CP refutations of Tseitin formulas were established by Buresh-Oppenheim et al. [18] using a rank-based argument. Our proof is inspired by their work, and so we describe it next. Briefly, their proof proceeds by considering a sequence of polytopes $P^{(0)} \supseteq \ldots \supseteq P^{(d)}$ where $P^{(i)}$ is the polytope defined by all inequalities that can be derived in depth i from the axioms in F. The goal is to show that $P^{(d)}$ is not empty. To do so, they show that a point $p \in P^{(i)}$ is also in $P^{(i+1)}$ if for every coordinate j such that $0 < p_j < 1$, there exists points $p^{(j,0)}, p^{(j,1)} \in P^{(i)}$ such that $p_k^{(j,b)} = b$ if $k = j$ and $p_k^{(j,b)} = p_k$ otherwise. The proof of this fact is syntactic: it relies on the careful analysis of the precise rules of CP.

When dealing with Semantic CP, we can no longer analyze a finite set of syntactic rules. Furthermore, it is not difficult to see that the aforementioned criterion for membership in $P^{(i+1)}$ is no longer sufficient for Semantic CP. We develop an analogous criterion for Semantic CP given later in this section. As well, we note that the definition of $P^{(i)}$ is not well-suited to studying the depth of bounded-size CP proofs like those in Theorem 1.6 — there does not appear to be a useful way to limit $P^{(i)}$ to be a polytope derived by a bounded number of halfspaces. Therefore we develop our criterion in the language of lifting, which is more amenable to supercritical tradeoffs [59, 13].

Through this section we will work with the following *top-down* definition of Semantic CP.

Definition 5.1. Let F be an n-variate unsatisfiable CNF formula. An sCP refutation of F is a directed acyclic graph of fan-out ≤ 2 where each node v is labelled with a halfspace $H_v \subseteq \mathbb{R}^n$ (understood as a set of points satisfying a linear inequality) satisfying the following:

1. *Root.* There is a unique source node r labelled with the halfspace $H_v = \mathbb{R}^n$ (corresponding to the trivially true inequality $1 \geq 0$).
2. *Internal-Nodes.* For each non-leaf node u with children v, w, we have
$$H_u \cap \{0,1\}^n \subseteq H_v \cup H_w.$$
3. *Leaves.* Each sink node u is labeled with a unique clause $C \in F$ such that $H_v \cap \{0,1\}^n \subseteq C^{-1}(0)$.

The above definition is obtained by taking a (standard) sCP proof and *reversing all inequalities*: now, a line is associated with the set of assignments *falsified* at that line, instead of the assignments *satisfying* the line.

To prove the lower bound we will need to find a long path in the proof. To find this path we will be taking a root-to-leaf walk down the proof while constructing a partial restriction $\rho \in \{0,1,*\}^n$ on the variables. For a partial restriction ρ, denote by $\mathrm{free}(\rho) := \rho^{-1}(*)$ and $\mathrm{fix}(\rho) := [n] \setminus \mathrm{free}(\rho)$. Let the *restriction* of H by ρ be the halfspace
$$H \upharpoonright \rho := \{x \in \mathbb{R}^{\mathrm{free}(\rho)} : \exists \alpha \in H,\ \alpha_{\mathrm{fix}(\rho)} = \rho_{\mathrm{fix}(\rho)},\ \alpha_{\mathrm{free}(\rho)} = x\}.$$
It is important to note that $H \upharpoonright \rho$ is itself a halfspace on the *free* coordinates of ρ.

One of our key invariants needed in the proof is the following.

Definition 5.2. A halfspace $H \subseteq \mathbb{R}^n$ is *good* if it contains the all-$\frac{1}{2}$ vector, that is, $(\frac{1}{2})^n = (\frac{1}{2}, \frac{1}{2}, \ldots, \frac{1}{2}) \in H$.

We will need two technical lemmas to prove the lower bounds. The first lemma shows that if a good halfspace H has its boolean points covered by halfspaces H_1, H_2, then one of the two covering halfspaces is also good modulo restricting a small set of coordinates.

Lemma 5.3. *Let $H \subseteq \mathbb{R}^n$ be any good halfspace, and suppose $H \cap \{0,1\}^n \subseteq H_1 \cup H_2$ for halfspaces H_1, H_2. Then there is a restriction ρ and an $i = 1, 2$ such that $|\mathrm{fix}(\rho)| \leq 2$ and $H_i \upharpoonright \rho$ is good.*

The second lemma shows that good halfspaces are *robust*, in the sense that we can restrict a good halfspace to another good halfspace while also satisfying any mod-2 equation.

Lemma 5.4. *Let $n \geq 2$ and $H \subseteq \mathbb{R}^n$ be a good halfspace. For any $I \subseteq [n]$ with $|I| \geq 2$ and $b \in \{0,1\}$, there is a partial restriction $\rho \in \{0,1,*\}^n$ with* $\mathrm{fix}(\rho) = I$ *such that*

– $\bigoplus_{i \in I} \rho(x_i) = b$ *and*
– $H \restriction \rho \subseteq \mathbb{R}^{\mathrm{free}(\rho)}$ *is good.*

With these two lemmas one can already get an idea of how to construct a long path in the proof. Suppose we start at the root of the proof; the halfspace is $1 \geq 0$ (which is clearly good) and the restriction we maintain is $\rho = *^n$. We can use the first lemma to move from the current good halfspace to a good child halfspace while increasing the number of fixed coordinates by at most 2. However, we have no control over the two coordinates which are fixed by this move, and so we may fall in danger of falsifying an initial constraint. Roughly speaking, we will use the second lemma to satisfy constraints that are in danger of being falsified.

We delay the proofs of these technical lemmas to the end of the section, and first see how to prove the depth lower bounds.

5.1 Lifting decision-tree depth to semantic CP depth

As a warm-up, we show how to lift lower bounds on Resolution depth to Semantic CP depth by composing with a constant-width XOR gadget. If F is a CNF formula then we can create a new formula by replacing each variable z_i with an XOR of 4 new variables $x_{i,1}, \ldots, x_{i,4}$:

$$z_i := \mathsf{XOR}_4(x_{i,1}, \ldots, x_{i,4}) = x_{i,1} \oplus \cdots \oplus x_{i,4}.$$

We call z_i the *unlifted* variable associated with the output of the XOR_4 *gadget* applied to the i-th *block* of variables. Formally, let $\mathsf{XOR}_4^n : \{0,1\}^{4n} \to \{0,1\}^n$ be the application of XOR_4 to each 4-bit block of a $4n$-bit string. Let $F \circ \mathsf{XOR}_4^n$ denote the formula obtained by performing this substitution on F and transforming the result into a CNF formula in the obvious way.

The main result of this section is the following.

Theorem 5.5. *For any unsatisfiable CNF formula F,*

$$\mathrm{depth}_{\mathsf{sCP}}(F \circ \mathsf{XOR}_4^n) \geq \frac{1}{2}\mathrm{depth}_{\mathsf{Res}}(F).$$

Key to our lower bound will be the following characterization of Resolution depth by *Prover–Adversary* games.

Definition 5.6. The *Prover–Adversary* game associated with an n-variate formula F is played between two competing players, Prover and Adversary. The game proceeds in rounds, where in each round the state of the game is recorded by a partial assignment $\rho \in \{0,1,*\}^n$ to the variables of F.

Initially the state is the empty assignment $\rho = *^n$. Then, in each round, the Prover chooses an $i \in [n]$ with $\rho_i = *$, and the Adversary chooses $b \in \{0,1\}$. The state is updated by $\rho_i \leftarrow b$ and play continues. The game ends when the state ρ falsifies an axiom of F.

It is known [58] that $\mathrm{depth}_{\mathsf{Res}}(F)$ is exactly the smallest d for which there is a Prover strategy that ends the game in d rounds, regardless of the strategy for the Adversary.

The proof of Theorem 5.5 will follow by using an optimal Adversary strategy for F to construct a long path in the Semantic CP proof of $F \circ \mathsf{XOR}_4^n$. Crucially, we need to understand how halfspaces H transform under XOR_4^n:

$$\mathsf{XOR}_4^n(H) := \{z \in \{0,1\}^n : \exists x \in H \cap \{0,1\}^{4n}, \mathsf{XOR}_4^n(x) = z\}.$$

As we have already stated, we will maintain a partial assignment $\rho \in \{0,1,*\}^{4n}$ on the $4n$ *lifted* variables. However, in order to use the Adversary, we will need to convert ρ to a partial assignment on the n *unlifted* variables. To perform this conversion, for any $\rho \in \{0,1,*\}^{4n}$ define $\mathsf{XOR}_4^n(\rho) \in \{0,1,*\}^n$ as follows: for each block $i \in [n]$, define

$$\mathsf{XOR}_4^n(\rho)_i = \begin{cases} \mathsf{XOR}_4(\rho(x_{i,1}), \ldots, \rho(x_{i,4})) & \text{if } (i,j) \in \mathrm{fix}(\rho) \text{ for } j \in [4], \\ * & \text{otherwise.} \end{cases}$$

We are now ready to prove Theorem 5.5. Fix any Semantic CP refutation of $F \circ \mathsf{XOR}_4^n$, and suppose that there is a strategy for the Adversary in the Prover–Adversary game of F certifying that F requires depth d. Throughout the walk, we maintain a partial restriction $\rho \in \{0,1,*\}^{4n}$ to the lifted variables satisfying the following three invariants with respect to the current visited halfspace H.

- *Block Closed*. In every block either all variables in the block are fixed or all variables in the block are free.
- *Good Halfspace*. $H \restriction \rho$ is good.
- *Strategy Consistent*. The unlifted assignment $\mathsf{XOR}_4^n(\rho)$ does not falsify any clause in F.

Initially, we set $\rho = *^{4n}$ and the initial halfspace is $1 \geq 0$, so the pair (H, ρ) trivially satisfy the invariants. Suppose we have reached the halfspace H in our walk and ρ is a restriction satisfying the invariants. We claim that H cannot be a leaf. To see this, suppose that H is a leaf, then by definition $H \cap \{0,1\}^{4n} \subseteq C^{-1}(0)$ for some clause $C \in F \circ \mathsf{XOR}_4^n$. By the definition of the lifted formula, this implies that $\mathsf{XOR}_4^n(H) \subseteq D^{-1}(0)$ for some clause $D \in F$. Since (H, ρ) satisfy the invariants, the lifted assignment $\mathsf{XOR}_4^n(\rho)$ does not falsify D, and so by the block-closed property it follows that there must be a variable $z_i \in D$ such that all lifted variables in the block i are free under ρ. But then applying Theorem 5.4 to the block of variables $\{x_{i,1}, x_{i,2}, x_{i,3}, x_{i,4}\}$, we can extend ρ to a partial assignment ρ' such that $z_i = \mathsf{XOR}_4(\rho(x_{i,1}), \rho(x_{i,2}), \rho(x_{i,3}), \rho(x_{i,4}))$ satisfies D. But $H \restriction \rho'$ is a projection of $H \restriction \rho$ and so this contradicts that $\mathsf{XOR}_4^n(H)$ violates D.

It remains to show how to take a step down the proof. Suppose that we have taken $t < d/2$ steps down the Semantic CP proof, the current node is labelled with a halfspace H, and the partial assignment ρ satisfies the invariants. If H has only a single child H_1, then $H \cap \{0,1\}^{4n} \subseteq H_1 \cap \{0,1\}^{4n}$ and ρ will still satisfy the invariants for H_1. Otherwise, if H has two children H_1 and H_2 then applying Theorem 5.3 to the halfspaces $H \restriction \rho, H_1 \restriction \rho, H_2 \restriction \rho$ we can find an $i \in \{1,2\}$ and a restriction τ such that $H_i \restriction (\rho\tau)$ is good and τ restricts at most 2 extra coordinates. Let $i_1, i_2 \in [n]$ be the two blocks of variables in which τ restricts variables, and note that it could be that $i_1 = i_2$.

Finally, we must restore our invariants. We do this in the following three step process.

– Query the Adversary strategy at the state $\mathsf{XOR}_4^n(\rho)$ on variables z_{i_1}, z_{i_2} and let $b_1, b_2 \in \{0,1\}$ be the responses.
– For $i = i_1, i_2$ let I_i be the set of variables free in the block i, and note that $|I_i| \geq 2$. Apply Theorem 5.4 to $H \restriction (\rho\tau)$ and I_i to get new restrictions ρ_{i_1}, ρ_{i_2} so that blocks i_1 and i_2 both take values consistent with the Adversary responses b_1, b_2.
– Update $\rho \leftarrow \rho\tau\rho_{i_1}\rho_{i_2}$.

By Theorem 5.4 the new restriction ρ satisfies the block-closed and the good halfspace invariants. At each step we fix at most two blocks of variables, and thus the final invariant is satisfied as long as $t < d/2$. This completes the proof.

5.2 Semantic CP depth lower bounds for unlifted formulas

Next we show how to prove depth lower bounds directly on *unlifted* families of $\mathbb{F}_2$-linear equations. The strength of these lower bounds will depend directly on the expansion of the underlying constraint-variable graph of F.

Throughout this section, let F denote a set of $\mathbb{F}_2$-linear equations. In a Semantic CP proof, we must encode F as a CNF formula, but while proving the lower bound we will instead work with the underlying system of equations. For a set F of $\mathbb{F}_2$-linear equations let $G_F := (F \cup V, E)$ be the bipartite *constraint-variable* graph defined as follows. Each vertex in F corresponds to an equation in F and each vertex in V correspond to variables x_i. There is an edge $(C_i, x_j) \in E$ if x_j occurs in the equation C_i. For a subset of vertices $X \subseteq F \cup V$ define the *neighbourhood* of X in G_F as $\Gamma(X) := \{v \in F \cup V : \exists u \in X, (u,v) \in E\}$.

Definition 5.7. For a bipartite graph $G = (U \cup V, E)$ the *boundary* of a set $W \subseteq U$ is

$$\delta(W) := \{v \in V : |\Gamma(v) \cap W| = 1\}.$$

The *boundary expansion* of a set $W \subseteq U$ is $|\delta(W)|/|W|$. The graph G is a (r,s)-*boundary expander* if the boundary expansion of every set $W \subseteq U$ with $|W| \leq r$ has boundary expansion at least s.

If F is a system of linear equations then we say that F is an (r,s)-boundary expander if its constraint graph G_F is. The main result of this section is the following theorem, analogous to Theorem 5.5.

Theorem 5.8. *For any system of $\mathbb{F}_2$-linear equations F that is an $(r, s+3)$-boundary expander,*

$$\mathrm{depth}_{\mathsf{sCP}}(F) \geq rs/2.$$

The proof of this theorem follows the proof of Theorem 5.5 with some small changes. As before, we will maintain a partial assignment $\rho \in \{0,1,*\}^n$ that will guide us on a root-to-leaf walk through a given Semantic CP proof; we also require that each halfspace H that we visit is *good* relative to our restriction ρ. Now our invariants are (somewhat) simpler: we will only require that $F \restriction \rho$ is a sufficiently good boundary expander.

We first prove an auxiliary lemma that will play the role of Theorem 5.4 in the proof of Theorem 5.8. We note that it follows immediately from Theorem 5.4 and boundary expansion.

Lemma 5.9. *Suppose F is a system of $\mathbb{F}_2$-linear equations that is an (r, s)-boundary expander for $s > 1$, and suppose $F' \subseteq F$ with $|F'| \le r$. Let H be a good halfspace. Then there exists a $\rho \in \{0, 1, *\}^n$ with $\operatorname{fix}(\rho) = \Gamma(F')$ such that*

- *F' is satisfied by ρ, and*
- *$H \restriction \rho$ is good.*

Proof. We first use expansion to find, for each constraint $C_i \in F'$, a pair of variables $y_{i,1}, y_{i,2}$ that are in C_i's boundary. To do this, first observe that $|\delta(F')| \ge s|F'| > |F'|$ by the definition of boundary expansion. The pigeonhole principle then immediately implies that there are variables $y_{i,1}, y_{i,2} \in \delta(F')$ and a constraint $C_i \in F'$ such that $y_{i,1}, y_{i,2} \in C_i$. Since $y_{i,1}, y_{i,2}$ do not occur in $F' \setminus \{C_i\}$, it follows that $F' \setminus \{C_i\}$ is still an (r, s)-boundary expander. So, we update $F' = F' \setminus \{C_i\}$ and repeat the above process.

When the process terminates, we have for each constraint $C_i \in F'$ a pair of variables $y_{i,1}, y_{i,2}$ that occur *only* in C_i. Write the halfspace $H = \sum_i w_i x_i \ge c$, and let $I = \Gamma(F') \setminus \bigcup_{i \in I}\{y_{i,1}, y_{i,2}\}$ be the set of variables occurring in F' that were not collected by the above process. We define a partial restriction ρ with $\operatorname{fix}(\rho) = I$ that depends on $|I|$ as follows.

- If $|I| = 0$ then $\rho = *^n$.
- If $I = \{x_i\}$ then define $\rho(x_i) = 1$ if $w_i \ge 0$ and $\rho(x_i) = 0$ otherwise, and for all other variables set $\rho(x) = *$.
- If $|I| > 2$ then apply Theorem 5.4 to generate a partial restriction ρ with $\operatorname{fix}(\rho) = I$ that sets the XOR of I arbitrarily.

Observe that $H \restriction \rho$ is good. The only non-trivial case is when $|I| = 1$, but, in this case we observe that $(H \restriction \rho)((1/2)^{n-1}) = 1$ because

$$w_i \rho(x_i) + \sum_{j \ne i} w_i/2 \ge \sum_i w_i/2 \ge c,$$

where we have used that H is good and the definition of ρ.

Next we extend ρ as follows: for each $i = 1, 2, \ldots, |F'|$ apply Theorem 5.4 to $I_i = \{y_{i,1}, y_{i,2}\}$ to generate a partial restriction ρ_i with $\operatorname{fix}(\rho_i) = I_i$ so that the constraint $C_i \restriction \rho\rho_1 \cdots \rho_{i-1}$ is satisfied by ρ_i. Observe that this is always possible since I_i is in the boundary of C_i. Finally, we update $\rho \leftarrow \rho\rho_1 \cdots \rho_{|F'|}$. It follows by Theorem 5.4 that F' is satisfied by ρ and $H \restriction \rho$ is good. □

We are now ready to prove Theorem 5.8. Fix any Semantic CP refutation of F and let n be the number of variables. We take a root-to-leaf walk through the refutation while maintaining a partial assignment $\rho \in \{0, 1, *\}^n$ and an integer valued parameter $k \ge 0$. Throughout the walk we maintain the following invariants with respect to the current halfspace H:

- *Good Expansion.* $F \restriction \rho$ is a (k, t)-boundary expander with $t > 3$.
- *Good Halfspace.* $H \restriction \rho$ is good.
- *Consistency.* The partial assignment ρ does not falsify any clause of F.

Initially, we set $k = r$, $\rho = *^n$, and $t = s + 3$, so the invariants are clearly satisfied since F is an $(r, s+3)$-expander. So, suppose that we have reached a halfspace H in our walk, and let k, ρ be parameters satisfying the invariants. We first observe that if $k > 0$ then H cannot be a sink node of the proof. To see this, it is enough to show that H contains a satisfying assignment for each equation $C \in F$. Because $H \restriction \rho$ is non-empty (since it is good) there exists a satisfying assignment in H for every equation satisfied by ρ, so, assume that C is not satisfied by ρ. In this case, since $F \restriction \rho$ is a (k, t)-expander for $k > 0$ we can apply Theorem 5.9 to $\{C\}$ and $H \restriction \rho$ and obtain a partial restriction τ with $\mathrm{fix}(\tau) = \Gamma(C)$ such that τ satisfies C. It follows that H is not a leaf.

Next, we show how to take a step down the proof while maintaining the invariants. If H has only a single child H_1, then $H \subseteq H_1$ and we can move to H_1 without changing ρ or k. Otherwise, let the children of H be H_1 and H_2. Applying Theorem 5.3 to $H \restriction \rho, H_1 \restriction \rho, H_2 \restriction \rho$ we get a partial restriction τ and an $i \in \{1, 2\}$ such that $H_i \restriction \rho\tau$ is good and $|\mathrm{fix}(\tau)| \leq 2$. Due to this latter fact, since $F \restriction \rho$ is a (k, t)-expander it follows that $F \restriction \rho\tau$ is a $(k, t-2)$-expander in the worst case. Observe that since $t > 3$ it follows that $F \restriction \rho\tau$ still satisfies the consistency invariant. It remains to restore the expansion invariant.

To restore the expansion invariant, let W be the largest subset of equations such that $|W| \leq k$ and W has boundary expansion at most 3 in $F \restriction \rho\tau$, and note that W has boundary expansion at least $t - 2 > 1$. Applying Theorem 5.9, we can find a restriction ρ' such that $W \restriction \rho\tau\rho'$ is satisfied, and $H \restriction \rho\tau\rho'$ is a good halfspace. Since W is the largest subset with expansion at most 3, it follows that $F \restriction \rho\tau\rho'$ is now a $(k - |W|, t')$-boundary expander with $t' > 3$. Suppose otherwise, then there exists a subset of equations W' which has boundary expansion at most 3 in $F \restriction \rho\tau\rho'$. Then $W \cup W'$ would have had boundary expansion at most 3 in $F \restriction \rho\tau$, contradicting the maximality of W. Now update $\rho \leftarrow \rho\tau\rho'$ and $k \leftarrow k - |W|$. Finally, we halt the walk if $k = 0$.

We now argue that this path must have had depth at least $rs/2$ upon halting. Assume that we have taken t steps down the proof. For each step $i \leq t$ let W_i be the set of equations which lost boundary expansion during the ith cleanup step. Note that $W_i \cap W_j = \emptyset$ for every $i \neq j$. Let $W^* = \cup_{i=1}^{t} W_i$, note that $|W^*| = r$ because at the ith step we decrease k by $|W_i|$. Furthermore, at the end of the walk, W^* has no neighbours and therefore no boundary in $F \restriction \rho$. Before the start of the ith cleanup step, W_i has at most $3|W_i|$ boundary variables. Therefore, at most $3|W^*| = 3r$ boundary variables were removed during the cleanup step. Since F started as an $(r, s+3)$-boundary expander, it follows that W^* had at least $r(s+3)$ boundary variables at the start of the walk. But, since *all* variables have been removed from the boundary by the end, this means that rs variables must have been removed from the boundary during the move step. Thus, as each move step sets at most 2 variables, it follows that $t \geq rs/2$ before the process halted.

5.3 Proof of Theorem 5.3 and Theorem 5.4

In this section we prove our two key technical lemmas: Theorem 5.3 and Theorem 5.4. We begin by proving Theorem 5.4 as it is simpler.

Proof of Theorem 5.4. Let H be represented by $\sum_{i \in [n]} w_i x_i \geq c$ and suppose without loss of generality that $c \geq 0$ and that $I = \{1, \ldots, k\}$. Let the weights of I in H be ordered $|w_1| \geq |w_2| \geq \ldots |w_k|$. Define ρ by setting $\rho(x_i) = *$ for $i \notin I$, for $i \leq k-1$ set $\rho(x_i) = 1$ if $w_i \geq 0$ and $\rho(x_i) = 0$ otherwise, and set $\rho(x_k)$ so that $\bigoplus_{i \in I} \rho(x_i) = b$. Clearly the parity constraint is satisfied, we show that $H \restriction \rho$ is good. This follows by an easy calculation:

$$\begin{aligned} &w_{k-1}\rho(x_{k-1}) + w_k\rho(x_k) + \sum_{i \leq k-2} w_i\rho(x_i) + \sum_{i \geq k+1} w_i/2 \\ &\geq w_{k-1}/2 + w_k/2 + \sum_{i \leq k-2} w_i\rho(x_i) + \sum_{i \geq k+1} w_i/2 \\ &\geq \sum_{i \in [n]} w_i/2 \geq c \end{aligned}$$

where the first inequality follows by averaging since $|w_{k-1}| \geq |w_k|$, and the final inequality follows since H is good. Therefore, $(H \restriction \rho)((1/2)^{[n] \setminus I}) = 1$, and $H \restriction \rho$ is good. □

In the remainder of the section we prove Theorem 5.3. It will be convenient to work over $\{-1, 1\}^n$ rather than $\{0, 1\}^n$, so, we restate it over this set and note that we can move between these basis by using the bijection $v \mapsto (1 - v)/2$.

Lemma 5.10. *Let $H \in \mathbb{R}^n$ be a halfspace such that $0^n \in H$ and suppose that $H \cap \{-1, 1\}^n \subseteq H_1 \cup H_2$. Then one of H_1 or H_2 contains a point $y \in \{-1, 0, 1\}^n$ such that y has at most two coordinates in $\{-1, 1\}$.*

The key ingredient in our proof of Theorem 5.10 is the following simple topological lemma, which will allow us to find a well-behaved point lying on a 2-face of the $\{-1, 1\}^n$ cube

Definition 5.11 (2-face)**.** A 2-face of the n-cube with vertices $\{-1, 1\}^n$ are the 2-dimensional 2-by-2 squares spanned by four vertices of the cube that agree on all but two coordinates. That is, a two face is a set $A \subseteq [-1, 1]^n$ such that there exists $\rho \in \{-1, 1, *\}^n$ with $|\mathrm{free}(\rho)| = 2$ and $A = [-1, 1]^n \restriction \rho$.

Lemma 5.12. *Let $w_1, w_2 \in \mathbb{R}^n$ be any pair of non-zero vectors, then we can find a vector $v \in \mathbb{R}^n$ orthogonal to w_1, w_2, such that v lies on a 2-face.*

Proof. We will construct the vector v iteratively by rounding one coordinate at a time to a $\{-1, 1\}$-value until v contains exactly $n - 2$ coordinates fixed to $\{-1, 1\}$. At each step, we will maintain that $v \in [-1, 1]^n$ and that v is orthogonal to w_1 and w_2. Therefore when the process halts v will lie on a 2-face.

Initially, set $v = 0^n$ and observe that the invariants are satisfied. Suppose that we have constructed a vector v that is orthogonal to w_1 and w_2, all of its coordinates belong to $[-1, 1]$, and exactly $i < n - 2$ of its coordinates belong to $\{-1, 1\}$; suppose w.l.o.g. that they are the first i coordinates. We will show how to "booleanize" an additional coordinate of v. Let u be any non-zero vector that is orthogonal to $\{w_1, w_2, e_1, \ldots, e_i\}$, where e_j is the jth standard basis vector. Begin moving from v in the direction of u and let $\alpha > 0$ be the smallest value such that one of the coordinates $j > i$ of $v + \alpha u$ is in $\{-1, 1\}$. We verify that the following properties hold:

1. The first i coordinates of $v + \alpha u$ are in $\{-1, 1\}$. This follows because we moved in a direction that is orthogonal to $e_1, \ldots, e_i$.
2. $v + \alpha u$ is orthogonal to w_1 and w_2. Let w be either of the vectors w_1 or w_2 and observe that $v_{i+1}w = v_i w + \alpha(uw) = 0$, where the final equality follows because w is orthogonal to v_i by induction and to u by assumption.

Finally, set v to be $v + \alpha u$. $\square$

Proof of Theorem 5.10. Let the children H_1 and H_2 of H be given by the halfspaces $w_1 x \geq b_1$ and $w_2 x \geq b_2$, respectively. By Theorem 5.12 we can find a vector v which is orthogonal to w_1 and w_2, and which lies on some 2-face F of the $[-1, 1]^n$ cube corresponding to some restriction $\rho \in \{0, 1, *\}^n$. Then, v lies in (at least) one of the four 1-by-1 quadrants of the 2-face, $[0, 1]^2$, $[0, 1] \times [-1, 0]$, $[-1, 0] \times [0, 1]$, or $[-1, 0]^2$; suppose that v lies in the $[-1, 0] \times [0, 1]$ quadrant of F, the other cases will follow by symmetry (see Figure 3).

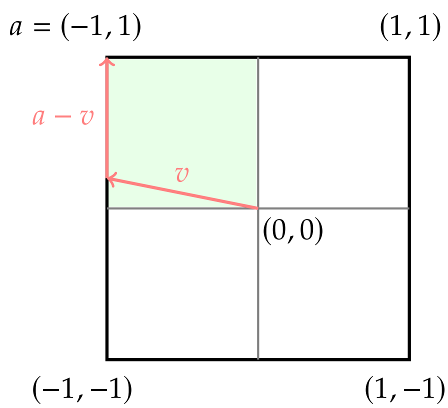


Figure 3: A 2-face of the n-cube together with a depiction of the booleanizing process.

Let $a \in \mathbb{R}^n$ be the vector corresponding to the $(-1, 1)$ corner of F, i. e., a is ρ extended by setting the two free bits to -1 and 1. By symmetry and the fact that H is good (and therefore $0^n \in H$), we can assume that a is contained in H — otherwise, simply exchange a and v for $-a$ and $-v$. Since $H \cap \{-1, 1\}^n \subseteq H_1 \cup H_2$ and $a \in \{-1, 1\}^n$, it follows that a is in one of H_1 or H_2. Assume that $a \in H_1$; that is, $w_1 a \geq b_1$. Our goal is to construct a vector $y \in H_1$ that satisfies the statement of the lemma. Consider the following two cases:

(i) If $w_1(a - v) \leq 0$, then it follows that $y := 0^n \in H_1$. Indeed, $w_1 y = w_1 v \geq w_1 a \geq b_1$, where first equality follows because w_1 and p are orthogonal by assumption, and the final inequality follows because $a \in H_1$.

(ii) Otherwise, we have that $w_1(a - v) > 0$. We construct a point that satisfies the statement of the lemma as follows. First, note that since $a, v \in F$, it follows that the vector $a - v$ has at most two non-zero coordinates. Beginning at the origin 0^n, move in the direction $a - v$ until a free coordinate coordinate becomes fixed to -1 or 1; that is, let $\alpha > 0$ be the minimum value such that $\alpha(a - v)$ has at most one coordinate which is not $\{-1, 1\}$-valued. Since both a and v belong to the same 1×1 quadrant of the 2-face, $\|a - v\|_\infty \leq 1$ and so $\alpha \geq 1$. We can then verify that $\alpha(a - v) \in H_1$, since

$$w_1\alpha(a - v) = \alpha(w_1 a) - 0 \geq w_1 a \geq b_1,$$

where we have used the fact that v is orthogonal to w_1 and $\alpha \geq 1$. Finally, since $\alpha(a-v) \in H_1$ we can round the final non-zero coordinate to -1 or 1; since H_1 is a halfspace one of the two vectors will remain in H_1. ◻

5.4 Applications

We now use the theorems from the previous sections to obtain several concrete lower bounds. First, we give strong depth lower bounds for sCP proofs of Tseitin formulas on expander graphs.

Theorem 5.13. *There exists a graph G and labelling $\ell : V \to \{0, 1\}$ such that any* sCP *refutation of* Tseitin(G, ℓ) *requires depth $\Omega(n)$.*

Proof. A graph $G = (V, E)$ is a *γ-vertex expander* if

$$\min\{|\Gamma(W)| : W \subseteq V, |W| \leq |V|/2\} \geq \gamma|W|,$$

where $\Gamma(W)$ is the neighbourhood of W. We claim that if G is a γ-vertex expander then any Tseitin formula over G is a $(n/2, \gamma)$-boundary expander. Fix any subset W of the equations with $|W| \leq n/2$. By the definition of vertex expansion we have that $|\Gamma(W)| \geq \gamma|W|$, and since each variable is contained in exactly two constraints, it follows that the boundary of W in Tseitin(G, ℓ) has size at least $|\delta(W)| \geq \gamma|W|$. The result then follows from Theorem 5.8 and the existence of strong vertex expanders G (e. g., d-regular Ramanujan graphs are at least $d/4$-vertex expanders, and exist for all d and n [50]). ◻

Next, we give lower bounds on the depth of Semantic CP refutations of random k-XOR and random k-CNF formulas for constant k.

Definition 5.14. Let XOR(m, n, k) be the distribution on random k-XOR formulas obtained by sampling m equations from the set of all mod 2 linear equations with exactly k variables.

Theorem 5.15. *The following holds for Semantic* CP :

1. *For any $k \geq 6$ there exists $m = O(n)$ such that $F \sim$ XOR(m, n, k) requires refutations of depth at least $\Omega(n)$ with high probability.*
2. *For any $k \geq 6$ there exists $m = O(n)$ such that $F \sim \mathcal{F}(m, n, k)$ requires refutations of depth at least $\Omega(n)$ with high probability.*

Proof. We first prove (1) and obtain (2) via a reduction. Fix $m = O(n)$ so that F is unsatisfiable with high probability. For any constant k, δ and $m = O(n)$, $F \sim$ XOR(m, n, k) is an $(\alpha n, k-2-2\delta)$-boundary expander for some $\alpha > 0$ (see, e. g., [18, 23]). Thus, setting $k \geq 6$ and ε to be some small constant, the boundary expansion of G_F is at least 3. By Theorem 5.8, F requires depth $\Omega(n)$ to refute in Semantic CP with high probability.

The proof of (2) is via a reduction from $\mathcal{F}(m,n,k)$ to $\mathsf{XOR}(m,n,k)$. Every k-clause occurs in the clausal encoding of exactly one k-XOR constraint. It follows that from any k-CNF formula F we can generate a k-XOR formula whose clausal expansion F' contains F as follows: for each clause $C \in F$, if C contains an even (odd) number of positive literals then add to F' every clause on the variables of C which contains an even (odd) number of positive literals. The resulting F' is the clausal encoding of a set of $|F|$ k-XOR constraints. As there is a unique k-XOR consistent with the clauses of F, we can define the distribution $\mathsf{XOR}(m,n,k)$ equivalently as follows:

1. Sample $F \sim \mathcal{F}(m,n,k)$,
2. Return the k-XOR F' generated from F according to the aforementioned process.

It follows that the complexity of refuting $F \sim \mathcal{F}(m,n,k)$ is at least that of refuting $F' \sim \mathsf{XOR}(m,n,k)$ and (2) follows from (1) with the same parameters. □

Finally, we use Theorem 5.8 to extend the integrality gaps from [18] to sCP by essentially the same argument. For a linear program with constraints given by a system of linear inequalities $Ax \le b$, the *r-round* sCP *relaxation* adds all inequalities that can be derived from $Ax \le b$ by a depth-r sCP proof. We show that the r-round Semantic sCP linear program relaxation cannot well-approximate the number of satisfying assignments to a random k-SAT or k-XOR instance.

First we define our LP relaxations. Suppose that F is a k-CNF formula with m clauses $C_1, C_2, \dots, C_m$ and n variables $x_1, x_2, \dots, x_n$. If $C_i = \bigvee_{i\in P} x_i \vee \bigvee_{i\in N} \overline{x}_i$ then let $E(C_i) = \sum_{i\in P} x_i + \sum_{i\in N} 1 - x_i$. We consider the following LP relaxation of F:

$$\begin{aligned} & \max \sum_{i=1}^{m} y_i \\ \text{subject to}\quad & E(C_i) \ge y_i \quad \forall i \in [m] \\ & 0 \le x_j \le 1 \quad \forall j \in [n] \\ & 0 \le y_i \le 1 \quad \forall i \in [m] \end{aligned}$$

If F is a k-XOR formula with m constraints and n variables then we consider the above LP relaxation obtained by writing F as a k-CNF. Finally, recall that the *integrality gap* is the ratio between the optimal integral solution to a linear program and the optimal solution produced by the LP.

Theorem 5.16. *For any $\varepsilon > 0$ and $k \ge 6$,*

1. *There is $\kappa > 0$ and $m = O(n)$ such that for $F \sim \mathsf{XOR}(m,n,k)$ the integrality gap of the κn-round sCP relaxation of F is at least $(2-\varepsilon)$ with high probability.*
2. *There is $\kappa > 0$ and $m = O(n)$ such that for $F \sim \mathcal{F}(m,n,k)$ the integrality gap of the κn-round sCP relaxation of F is at least $2^k/(2^k-1) - \varepsilon$ with high probability.*

Proof. Let $F \sim \mathsf{XOR}(m, n, k)$ and let Y_i be the event that the ith constraint is falsified by a uniformly random assignment. Let $\delta := \varepsilon/(2-\varepsilon)$, then by a multiplicative Chernoff Bound, the probability that a uniformly random assignment satisfies at least a $1/(2-\varepsilon)$-fraction of F is $\Pr[\sum_{i\in[m]} Y_i \geq (1+\delta)\frac{m}{2}] \leq 2^{-\delta m/6}$. By a union bound, the probability that there exists an assignment satisfying at least a $1/(2-\varepsilon)$ fraction of F is $2^{n-\delta m/6}$ which is exponentially small when $m \geq 7n(2-\varepsilon)/\varepsilon$.

On the other hand, consider the partial restriction to the LP relaxation of F that sets $y_i = 1$ for all $i \in [m]$. Setting $m \geq 7n(2-\varepsilon)/\varepsilon$ large enough, by Theorem 5.15 there some $\kappa > 0$ such that with high probability F requires depth κn. Hence, the κn round Semantic CP LP relaxation is non-empty, and there is a satisfying assignment $\alpha \in \mathbb{R}^n$. Thus $\alpha \cup \{y_i = 1\}$ satisfies all constraints of $\max(F)$.

The second result follows by an analogous argument. □

6 Conclusion

The most obvious question left open by these simulations is whether CP can polynomially simulate SP, or even polynomially simulate SP^*. Similarly, what are the relationships of both SP and CP to (bounded-coefficient) $\mathsf{R(CP)}$, which is the proof system corresponding to dag-like SP. $\mathsf{R(CP)}$ can polynomially simulate DNF resolution, and therefore has polynomial size proofs of the Clique-Colouring formulas, for cliques of size $\Omega(\sqrt{n})$ and colourings of size $o(\log^2 n)$ [4]. Quasi-polynomial lower bounds on the size of CP refutations are known for this range of parameters and this rules out a polynomial simulation by Cutting Planes; however, a quasi-polynomial simulation may be possible. A potential approach to resolving this question is to use the added expressibility of $\mathsf{R(CP)}$ over DNF resolution to extend the upper bound on Clique-Colouring to the range of parameters for which superpolynomial CP lower bounds are known. Another potentially related family of proof systems are the $\mathsf{Res(lin)}$ systems, whose lines are disjuncts of linear *equalities* over the reals. Can we relate tree-like $\mathsf{Res(lin)}$ to SP?

A second direction of interest is regarding the coefficient size in the simulation. The simulation of SP^* by CP presented in this paper incurs a significant blowup in the coefficient size due to the use of Shrijver's lemma. It would be interesting to understand whether SP^* can already be quasi-polynomially simulated by CP^*; that is, whether this blowup in the size of the coefficients is necessary.

Another major open question is to resolve Theorem 1.6. Since essentially all current size lower bound techniques for Cutting Planes are by reduction to monotone circuit lower bounds [57, 35, 42, 36], it is natural to ask if supercritical size-depth tradeoffs can also be established for monotone circuits. As a first step towards both of these, can one prove a supercritical size-depth tradeoff for a weaker proof system such as resolution?

Finally, the quasi-polynomial upper bound on the Tseitin formulas [27] is quite remarkable since these formulas were long conjectured to be hard to refute in Cutting Planes [25]. However, it is not known whether this upper bound is tight; does Cutting Planes admit polynomial size proofs of the Tseitin formulas? A similar, and perhaps even more interesting open problem is whether *tree-like* Cutting Planes admits short proofs of Tseitin.

References

[1] KAREN AARDAL, ROBERT E. BIXBY, COR A. J. HURKENS, ARJEN K. LENSTRA, AND JOB W. SMELTINK: Market split and basis reduction: Towards a solution of the Cornuéjols–Dawande instances. *INFORMS J. Comput*, 12(3):192–202, 2000. [doi:10.1287/ijoc.12.3.192.12635] 3

[2] KAREN AARDAL AND ARJEN K. LENSTRA: Hard equality constrained integer knapsacks. *Math. Oper. Res.*, 29(3):724–738, 2004. [doi:10.1287/moor.1040.0099] 3

[3] MICHAEL ALEKHNOVICH AND ALEXANDER A. RAZBOROV: Satisfiability, branch-width and Tseitin tautologies. *Comput. Complexity*, 20:649–678, 2011. Preliminary version in FOCS'02. [doi:10.1007/s00037-011-0033-1] 16

[4] ALBERT ATSERIAS, MARIA LUISA BONET, AND JUAN LUIS ESTEBAN: Lower bounds for the weak pigeonhole principle and random formulas beyond resolution. *Inform. Comput.*, 176(2):136–152, 2002. [doi:10.1006/inco.2002.3114] 30

[5] ALBERT ATSERIAS, MASSIMO LAURIA, AND JAKOB NORDSTRÖM: Narrow proofs may be maximally long. *ACM Trans. Comput. Logic*, 17(3):19:1–30, 2016. [doi:10.1145/2898435] 5

[6] BOAZ BARAK, FERNANDO G. S. L. BRANDÃO, ARAM WETTROTH HARROW, JONATHAN A. KELNER, DAVID STEURER, AND YUAN ZHOU: Hypercontractivity, sum-of-squares proofs, and their applications. In *Proc. 44th STOC*, pp. 307–326. ACM Press, 2012. [doi:10.1145/2213977.2214006] 2

[7] AMITABH BASU, MICHELE CONFORTI, MARCO DI SUMMA, AND HONGYI JIANG: Complexity of branch-and-bound and cutting planes in mixed-integer optimization. *Math. Programming*, 198(1):787–810, 2022. Preliminary version in IPCO'21. [doi:10.1007/s10107-022-01789-5] 4

[8] ROBERTO J. BAYARDO JR. AND ROBERT C. SCHRAG: Using CSP look-back techniques to solve real-world SAT instances. In BENJAMIN KUIPERS AND BONNIE L. WEBBER, editors, *Proc. 14th National Conf. on Artificial Intelligence and 9th Innovative Applications of Artificial Intelligence Conf., AAAI'97/IAAI'97*, pp. 203–208. AAAI Press / The MIT Press, 1997. Available at AAAI Library. 2

[9] PAUL BEAME, CHRISTOPHER BECK, AND RUSSELL IMPAGLIAZZO: Time-space tradeoffs in resolution: superpolynomial lower bounds for superlinear space. *SIAM J. Comput.*, 45(4), 2016. Preliminary version in STOC'12. [doi:10.1137/130914085] 8

[10] PAUL BEAME, NOAH FLEMING, RUSSELL IMPAGLIAZZO, ANTONINA KOLOKOLOVA, DENIS PANKRATOV, TONIANN PITASSI, AND ROBERT ROBERE: Stabbing planes. In *Proc. 9th Innovations in Theoret. Comp. Sci. Conf. (ITCS'18)*, pp. 10:1–20. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2018. [doi:10.4230/LIPIcs.ITCS.2018.10] 3, 4, 8, 9, 12, 16

[11] CHRIS BECK, JAKOB NORDSTRÖM, AND BANGSHENG TANG: Some trade-off results for polynomial calculus: Extended abstract. In *Proc. 45th STOC*, pp. 813–822. ACM Press, 2013. [doi:10.1145/2488608.2488711] 8

[12] Eli Ben-Sasson and Avi Wigderson: Short proofs are narrow – resolution made simple. *J. ACM*, 48(2):149–169, 2001. [doi:10.1145/375827.375835] 5

[13] Christoph Berkholz and Jakob Nordström: Supercritical space-width trade-offs for resolution. *SIAM J. Comput.*, 49(1):98–118, 2020. [doi:10.1137/16M1109072] 7, 8, 19, 20

[14] Alexander Bockmayr, Friedrich Eisenbrand, Mark E. Hartmann, and Andreas S. Schulz: On the Chvátal rank of polytopes in the 0/1 cube. *Discr. Appl. Math.*, 98(1–2):21–27, 1999. [doi:10.1016/S0166-218X(99)00156-0] 11

[15] Merve Bodur, Alberto Del Pia, Santanu S. Dey, Marco Molinaro, and Sebastian Pokutta: Aggregation-based cutting-planes for packing and covering integer programs. *Math. Programming*, 171(1–2):331–359, 2018. [doi:10.1007/s10107-017-1192-x] 9

[16] Maria Luisa Bonet and Nicola Galesi: Optimality of size-width tradeoffs for resolution. *Comput. Complexity*, 10(4):261–276, 2001. [doi:10.1007/s000370100000] 5

[17] Maria Luisa Bonet, Toniann Pitassi, and Ran Raz: Lower bounds for cutting planes proofs with small coefficients. *J. Symbolic Logic*, 62(3):708–728, 1997. [doi:10.2307/2275569] 7, 11

[18] Joshua Buresh-Oppenheim, Nicola Galesi, Shlomo Hoory, Avner Magen, and Toniann Pitassi: Rank bounds and integrality gaps for cutting planes procedures. *Theory of Computing*, 2(4):65–90, 2006. [doi:10.4086/toc.2006.v002a004] 7, 8, 9, 19, 28, 29

[19] Samuel R. Buss, Dima Grigoriev, Russell Impagliazzo, and Toniann Pitassi: Linear gaps between degrees for the polynomial calculus modulo distinct primes. *J. Comput. System Sci.*, 62(2):267–289, 2001. [doi:10.1006/jcss.2000.1726] 4

[20] Vašek Chvátal: Edmonds polytopes and a hierarchy of combinatorial problems. *Discr. Math.*, 4(4):305–337, 1973. [doi:10.1016/0012-365X(73)90167-2] 2, 3, 10

[21] Vašek Chvátal: Cutting-plane proofs and the stability number of a graph. Technical Report TR-84326, Inst. für Ökonometrie und Operations Research, Rhein. Friedrich-Wilhelms-Univ., 1984. Available on author's webpage. 2

[22] Vašek Chvátal, William Cook, and Mark Hartmann: On cutting-plane proofs in combinatorial optimization. *Lin. Alg. Appl.*, 114–115:455–499, 1989. [doi:10.1016/0024-3795(89)90476-x] 9

[23] Vašek Chvátal and Endre Szemerédi: Many hard examples for resolution. *J. ACM*, 35(4):759–768, 1988. [doi:10.1145/48014.48016] 28

[24] Stephen A. Cook and Robert A. Reckhow: The relative efficiency of propositional proof systems. *J. Symbolic Logic*, 44(1):36–50, 1979. [doi:10.2307/2273702] 8

[25] William J. Cook, Collette R. Coullard, and György Turán: On the complexity of cutting-plane proofs. *Discr. Appl. Math.*, 18(1):25–38, 1987. [doi:10.1016/0166-218X(87)90039-4] 2, 4, 10, 11, 13, 30

[26] GÉRARD CORNUÉJOLS AND YANJUN LI: On the rank of mixed 0, 1 polyhedra. *Math. Programming*, 91(2):391–397, 2002. [doi:10.1007/s101070100250] 9

[27] DANIEL DADUSH AND SAMARTH TIWARI: On the complexity of branching proofs. In *Proc. 35th Comput. Complexity Conf. (CCC'20, virtual conference)*, pp. 34:1–35. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2020. [doi:10.4230/LIPIcs.CCC.2020.34] 4, 5, 11, 13, 30

[28] ADNAN DARWICHE: Recursive conditioning. *Artificial Intelligence*, 126(1–2):5–41, 2001. [doi:10.1016/S0004-3702(00)00069-2] 16

[29] MARTIN DAVIS AND HILARY PUTNAM: A computing procedure for quantification theory. *J. ACM*, 7(3):201–215, 1960. [doi:10.1145/321033.321034] 2

[30] RINA DECHTER: Bucket elimination: A unifying framework for processing hard and soft constraints. *ACM Computing Surveys*, 28(4es):61:1–5, 1996. [doi:10.1145/242224.242302] 16

[31] FRIEDRICH EISENBRAND AND ANDREAS S. SCHULZ: Bounds on the Chvátal rank of polytopes in the 0/1-cube. *Combinatorica*, 23:245–261, 2003. Preliminary version in IPCO'99. [doi:10.1007/s00493-003-0020-5] 9, 11

[32] YUVAL FILMUS, PAVEL HRUBEŠ, AND MASSIMO LAURIA: Semantic versus syntactic cutting planes. In *Proc. 33rd Symp. Theoret. Aspects of Comp. Sci. (STACS'16)*, pp. 35:1–13. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2016. [doi:10.4230/LIPIcs.STACS.2016.35] 6, 9, 11

[33] MATTEO FISCHETTI AND ANDREA LODI: Local branching. *Math. Programming*, 98(1–3):23–47, 2003. [doi:10.1007/s10107-003-0395-5] 3

[34] NOAH FLEMING, MIKA GÖÖS, RUSSELL IMPAGLIAZZO, TONIANN PITASSI, ROBERT ROBERE, LI-YANG TAN, AND AVI WIGDERSON: On the power and limitations of branch and cut. In *Proc. 36th Comput. Complexity Conf. (CCC'21)*, pp. 6:1–30. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2021. [doi:10.4230/LIPIcs.CCC.2021.6] 9

[35] NOAH FLEMING, DENIS PANKRATOV, TONIANN PITASSI, AND ROBERT ROBERE: Random Θ(log n)-CNFs are hard for cutting planes. *J. ACM*, 69(3):19:1–32, 2022. Preliminary version in FOCS'17. [doi:10.1145/3486680] 16, 30

[36] ANKIT GARG, MIKA GÖÖS, PRITISH KAMATH, AND DMITRY SOKOLOV: Monotone circuit lower bounds from resolution. *Theory of Computing*, 16(13):1–30, 2020. Preliminary version in STOC'18. [doi:10.4086/toc.2020.v016a013] 16, 30

[37] MICHEL X. GOEMANS AND DAVID P. WILLIAMSON: .879-approximation algorithms for MAX CUT and MAX 2SAT. In *Proc. 26th STOC*, pp. 422–431. ACM Press, 1994. [doi:10.1145/195058.195216] 2

[38] Ralph E. Gomory: An algorithm for integer solutions to linear programs. In Robert L. Graves and Philip Wolfe, editors, *Recent Advances in Mathematical Programming (1962)*, volume 64, pp. 269–302. McGraw-Hill, 1963. Available at ralphgomory.org, PDF page 286. 2, 3

[39] Mika Göös, Sajin Koroth, Ian Mertz, and Toniann Pitassi: Automating cutting planes is NP-hard. In *Proc. 52nd STOC*, pp. 68–77. ACM Press, 2020. [doi:10.1145/3357713.3384248] 16

[40] Dima Grigoriev: Tseitin's tautologies and lower bounds for Nullstellensatz proofs. In *Proc. 39th FOCS*, pp. 648–652. IEEE Comp. Soc., 1998. [doi:10.1109/SFCS.1998.743515] 4

[41] Dima Grigoriev: Linear lower bound on degrees of Positivstellensatz calculus proofs for the parity. *Theoret. Comput. Sci.*, 259(1–2):613–622, 2001. [doi:10.1016/S0304-3975(00)00157-2] 2, 4

[42] Pavel Hrubeš and Pavel Pudlák: Random formulas, monotone circuits, and interpolation. In *Proc. 58th FOCS*, pp. 121–131. IEEE Comp. Soc., 2017. [doi:10.1109/FOCS.2017.20] 16, 30

[43] Russell Impagliazzo, Toniann Pitassi, and Alasdair Urquhart: Upper and lower bounds for tree-like cutting planes proofs. In *Proc. 9th Ann. ACM/IEEE Symp. on Logic in Computer Science (LICS'94)*, pp. 220–228. IEEE Comp. Soc., 1994. [doi:10.1109/LICS.1994.316069] 7, 9

[44] Miroslav Karamanov and Gérard Cornuéjols: Branching on general disjunctions. *Math. Programming*, 128(1–2):403–436, 2011. [doi:10.1007/s10107-009-0332-3] 3

[45] Arist Kojevnikov: Improved lower bounds for tree-like resolution over linear inequalities. In João Marques-Silva and Karem A. Sakallah, editors, *Proc. 10th Internat. Conf. Theory Appl. Satisfiability Testing (SAT'07)*, volume 4501 of *LNCS*, pp. 70–79. Springer, 2007. [doi:10.1007/978-3-540-72788-0_10] 8

[46] Jan Krajícek: Discretely ordered modules as a first-order extension of the Cutting Planes proof system. *J. Symbolic Logic*, 63(4):1582–1596, 1998. [doi:10.2307/2586668] 4, 8, 12

[47] Bala Krishnamoorthy and Gábor Pataki: Column basis reduction and decomposable knapsack problems. *Discr. Optimization*, 6(3):242–270, 2009. [doi:10.1016/j.disopt.2009.01.003] 3

[48] Neha Lodha, Sebastian Ordyniak, and Stefan Szeider: A SAT approach to branchwidth. *ACM Trans. Comput. Logic*, 20(3):15:1–24, 2019. [doi:10.1145/3326159] 16

[49] Ashutosh Mahajan and Theodore K. Ralphs: Experiments with branching using general disjunctions. In *Operations Research and Cyber-Infrastructure (ORCS)*, pp. 101–118. Springer, 2009. [doi:10.1007/978-0-387-88843-9_6] 3

[50] Adam W. Marcus, Daniel A. Spielman, and Nikhil Srivastava: Interlacing families IV: Bipartite Ramanujan graphs of all sizes. *SIAM J. Comput.*, 47(6):2488–2509, 2018. [doi:10.1137/16M106176X] 28

[51] JOÃO P. MARQUES-SILVA AND KAREM A. SAKALLAH: GRASP: A search algorithm for propositional satisfiability. *IEEE Trans. Computers*, 48(5):506–521, 1999. [doi:10.1109/12.769433] 2

[52] MATTHEW W. MOSKEWICZ, CONOR F. MADIGAN, YING ZHAO, LINTAO ZHANG, AND SHARAD MALIK: Chaff: Engineering an efficient SAT solver. In *Proc. 38th Design Automation Conf. (DAC'01)*, pp. 530–535. ACM Press, 2001. [doi:10.1145/378239.379017] 2

[53] JONATHAN H. OWEN AND SANJAY MEHROTRA: Experimental results on using general disjunctions in branch-and-bound for general-integer linear programs. *Comput. Optim. Appl.*, 20(2):159–170, 2001. [doi:10.1023/A:1011207119557] 3

[54] PABLO A. PARRILO: *Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization*. Ph. D. thesis, CalTech, 2000. Available on author's website. 2

[55] SEBASTIAN POKUTTA AND ANDREAS S. SCHULZ: On the rank of cutting-plane proof systems. In *Proc. 14th Integer Prog. Combinat. Optim. (IPCO'10)*, volume 6080 of *LNCS*, pp. 450–463. Springer, 2010. [doi:10.1007/978-3-642-13036-6_34] 9

[56] SEBASTIAN POKUTTA AND ANDREAS S. SCHULZ: Integer-empty polytopes in the 0/1-cube with maximal Gomory–Chvátal rank. *Oper. Res. Lett.*, 39(6):457–460, 2011. [doi:10.1016/j.orl.2011.09.004] 9

[57] PAVEL PUDLÁK: Lower bounds for resolution and cutting plane proofs and monotone computations. *J. Symbolic Logic*, 62(3):981–998, 1997. [doi:10.2307/2275583] 7, 16, 30

[58] PAVEL PUDLÁK: Proofs as games. *Amer. Math. Monthly*, 107(6):541–550, 2000. [doi:10.1080/00029890.2000.12005233] 21

[59] ALEXANDER A. RAZBOROV: A new kind of tradeoffs in propositional proof complexity. *J. ACM*, 63(2):16:1–14, 2016. [doi:10.1145/2858790] 5, 7, 8, 19, 20

[60] ALEXANDER A. RAZBOROV: On the width of semialgebraic proofs and algorithms. *Math. Oper. Res.*, 42(4):1106–1134, 2017. [doi:10.1287/moor.2016.0840] 7, 8

[61] THOMAS ROTHVOSS AND LAURA SANITÀ: 0/1 polytopes with quadratic Chvátal rank. In *Proc. 16th Integer Prog. Combinat. Optim. (IPCO'13)*, pp. 349–361. Springer, 2013. [doi:10.1007/978-3-642-36694-9_30] 9

[62] GRANT SCHOENEBECK: Linear level Lasserre lower bounds for certain k-CSPs. In *Proc. 49th FOCS*, pp. 593–602. IEEE Comp. Soc., 2008. [doi:10.1109/FOCS.2008.74] 4

[63] ALEXANDER SCHRIJVER: On cutting planes. In PETER L. HAMMER, editor, *Combinatorics 79*, volume 9 of *Ann. Discr. Math.*, pp. 291–296. Elsevier, 1980. [doi:10.1016/S0167-5060(08)70085-2] 5, 13

AUTHORS

Noah Fleming
Assistant Professor
Department of Computer Science
Lund University
Lund, Sweden
noah.fleming@cs.lth.se
https://noahrfleming.github.io/

Mika Göös
Assistant Professor
School of Computer and Communication Sciences
EPFL
Lausanne, Switzerland
mika.goos@epfl.ch
https://ic-people.epfl.ch/~goos/

Russell Impagliazzo
Professor
Department of Computer Science and Engineering
UCSD
San Diego, California, USA
russell@cs.ucsd.edu
https://cseweb.ucsd.edu/~russell/

Toniann Pitassi
Jeffrey L. and Brenda Bleustein Professor of Engineering
Department of Computer Science
Columbia University
New York, New York, USA
tp2684@columbia.edu
https://www.cs.columbia.edu/~toni/

Robert Robere
Assistant Professor
Department of Computer Science
McGill University
Montreal, Quebec, Canada
robert.robere@mcgill.ca
https://www.cs.mcgill.ca/~robere/

Li-Yang Tan
Assistant Professor
Department of Computer Science
Stanford University
Stanford, California, USA
lytan@stanford.edu
https://theory.stanford.edu/~liyang/

Avi Wigderson
Herbert H. Maass Professor
School of Mathematics
Institute for Advanced Study
Princeton, New Jersey, USA
avi@ias.edu
https://www.math.ias.edu/avi/home

ABOUT THE AUTHORS

NOAH FLEMING is an assistant professor at Lund University. Prior to that he was an assistant professor at the Memorial University of Newfoundland, was a postdoctoral researcher at UC San Diego, and a research fellow at the Simons Institute. He completed his Ph. D. under the supervision of Toniann Pitassi at the University of Toronto. He focuses mainly on proof complexity, circuit complexity, TFNP, and related areas.

Mika Göös is an assistant professor at EPFL in the Theory Group. Previously, he was a postdoc at Stanford Theory, the Institute for Advanced Study, and the ToC group at Harvard. He completed his Ph. D. at the University of Toronto under the watchful eye of Toniann Pitassi. He obtained his B. S. from Aalto University, and his M. S. from the University of Oxford. In his spare time, Mika enjoys rock climbing and maintains a viral YouTube climbing channel. However, he is forsaking his dream of climbing a 7C-grade outdoor boulder one day due to now having small children at home.

Russell Impagliazzo is a professor at UC San Diego specializing in computational complexity theory, in particular in proof complexity, the theory of cryptography, computational randomness, structural complexity, and analyzing optimization heuristics and other approaches to solving hard problems.

Toniann Pitassi is a professor at Columbia University. She received a Ph. D. from the University of Toronto in 1992. After that, she spent 2 years as a postdoc at UCSD, and then 2 years as an assistant professor 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 the University of Toronto where she worked until 2021. Her primary research interests are complexity theory, and fairness and privacy in machine learning.

Robert Robere is an Assistant Professor in the School of Computer Science at McGill University. His main research topic is computational complexity theory, with a particular interest in proof complexity and related topics. But this is not prescriptive, and he likes to think about any fun problems that come his way!

Li-Yang Tan is an associate professor of computer science at Stanford. His research is in theoretical computer science, with an emphasis on complexity theory.

Avi Wigderson is the Herbert H. Maass Professor at the School of Mathematics, Institute for Advanced Study, Princeton. His research interests are in Randomness and computation, algorithms and optimization, complexity theory, circuit complexity, proof complexity, quantum computation and communication, and cryptography and distributed computation.