Random $\Theta(\log n)$-CNFs are Hard for Cutting Planes

NOAH FLEMING, University of Toronto, Canada
DENIS PANKRATOV, Concordia University, Canada
TONIANN PITASSI, University of Toronto, Canada and Institute for Advanced Study, USA
ROBERT ROBERE, Institute for Advanced Study, USA

The random k-SAT model is one of the most important and well-studied distributions over k-SAT instances. It is closely connected to statistical physics and is a benchmark for satisfiability algorithms. We show that when $k = \Theta(\log n)$, any Cutting Planes refutation for random k-SAT requires exponential length in the regime where the number of clauses guarantees that the formula is unsatisfiable with high probability.

CCS Concepts: • **Theory of computation → Proof complexity**;

Additional Key Words and Phrases: Random k-SAT, Cutting Planes

ACM Reference Format:
Noah Fleming, Denis Pankratov, Toniann Pitassi, and Robert Robere. 2020. Random $\Theta(\log n)$-CNFs are Hard for Cutting Planes. 1, 1 (April 2020), 31 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

The Satisfiability (SAT) problem — that is, the problem of finding a satisfying assignment for a given boolean formula — is one of the central problems studied in theoretical computer science. As it is one of the classical NP-Complete problems, there is no efficient algorithm that solves SAT on all instances unless P = NP. Furthermore, since any polynomial-time algorithm which solves SAT must also correctly classify all *unsatisfiable* boolean formulas, it follows that the complexity of the SAT problem is also intimately connected with the study of *refuting* unsatisfiable formulas.

In this paper, we study the problem of refuting *randomly generated* SAT instances. The most well-studied random SAT distribution is the *random k-SAT model* $\mathcal{F}(m, n, k)$ where a random k-CNF over n variables is chosen by uniformly and independently selecting m clauses from the set of all possible clauses on k distinct variables. This is an intrinsically natural distribution of instances similar to the Erdős-Rényi random graph model, and it is closely related to phase transitions and structural phenomena occurring in statistical physics (e.g. [30, 42]). Further, the model has close connections with complexity theory through *Feige's Hypothesis*: if $\mathcal{F}(m, n, k)$ is hard to refute on average for the "right" choice of m, n, k then worst-case inapproximability results follow for many NP-Hard optimization problems [17].

We study refuting random k-SAT instances through the lens of *propositional proof complexity*. Proof complexity studies the difficulty of refuting unsatisfiable SAT instances in propositional proof systems of various strengths. In this area, theorists have proven strong lower bounds for refuting random k-SAT formulas in many weak proof systems. For instance, in the *Resolution* proof

Authors' addresses: Noah Fleming, University of Toronto, 10 King's College Road, Toronto, ON, M5S 3G4, Canada, noahfleming@cs.toronto.edu; Denis Pankratov, Concordia University, 1455 De Maisonneuve Blvd. W., Montreal, QC, H3G 1M8, Canada, denis.pankratov@concordia.ca; Toniann Pitassi, University of Toronto, 10 King's College Road, Toronto, ON, M5S 3G4, Canada, Institute for Advanced Study, 1 Einstein Drive, Princeton, NJ, 08540, USA, toni@cs.toronto.edu; Robert Robere, Institute for Advanced Study, 1 Einstein Drive, Princeton, NJ, 08540, USA, robere@cs.toronto.edu.

XXXX-XXXX/2020/4-ART $15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

system — which forms the basis of essentially all modern SAT solvers — a classic result of Chvátal and Szemerédi [12] is that random k-SAT instances require length $\exp(\Omega(n))$ refutations with high probability. Superpolynomial lower bounds for random k-SAT formulas are also known for other proof systems such as Polynomial Calculus, Res(k), and Sum-of-Squares proof systems [1, 2, 6, 41].

In the present work we focus on Cutting Planes[1] refutations of $\mathcal{F}(m, n, k)$. The Cutting Planes technique was introduced in [21] in the context of linear programming, and was shown [10] to be a canonical way of proving that every integral solution to a set of linear inequalities satisfies another inequality. It was introduced as a proof system in [13], and is one of the most well-studied proof system from both the theoretical as well as from the algorithmic side. A Cutting Planes proof begins with a set of unsatisfiable linear integral inequalities — that is, inequalities of the form $a^T x \geq b$ for $a \in \mathbb{Z}^n$ and $b \in \mathbb{Z}$ — and seeks to derive the "false" inequality $0 \geq 1$ with as few derivation steps as possible. New integral inequalities (also called lines) can be derived from old ones by either (i) taking nonnegative linear combinations of previous lines, or (ii) dividing a previous inequality through by d (as long as all coefficients on the left-hand side are divisible by d) and then rounding up the constant term on the right-hand side.

It is a well-known open problem to prove superpolynomial lower bounds for Cutting Planes refutations of random k-SAT formulas (see, for example, [5]), especially because superpolynomial lower bounds for other formulas have been shown [9, 37]. Our main contribution is the first such lower bound on refutations of random k-SAT instances in this system, provided k is large enough.

Theorem 1.1. *There exist constants c, d such that the following holds. Let n be a sufficiently large positive integer, $k = c \log n$ and $m = n2^{dk}$. Then with high probability, any Cutting Planes refutation of a random k-CNF formula $F \sim \mathcal{F}(m, n, k)$ requires $2^{\tilde{\Omega}(n)}$ lines*[2].

In fact, our exponential lower bounds even apply to some stronger proof systems than Cutting Planes — see Section 2 for details. This lower bound has been independently obtained by Pavel Hrubeš and Pavel Pudlák [26] using similar techniques.

Proof Overview. To obtain the new lower bound we introduce a new technique for proving Cutting Planes lower bounds. Our new technique is a generalization of the classic (and, prior to this paper, only) lower bound technique for Cutting Planes proofs: the method of *feasible interpolation* [9, 31, 32, 37, 40]. As our technique generalizes it, let us first describe feasible interpolation. Suppose we are given an unsatisfiable CNF formula $F(\vec{x}, \vec{y}, \vec{z})$ on three sets of variables $\vec{x}, \vec{y}, \vec{z}$ of the following "split" form

$$F(\vec{x}, \vec{y}, \vec{z}) = A(\vec{x}, \vec{z}) \wedge B(\vec{y}, \vec{z}).$$

Then, given an assignment α to the z variables it follows that either the formula $A(\vec{x}, \alpha)$ is unsatisfiable or the formula $B(\vec{y}, \alpha)$ is unsatisfiable. Generally speaking, a feasible interpolation argument shows that the complexity of *computing* the *interpolant function*

$$I(\alpha) = \begin{cases} 1 & \text{if } A(\vec{x}, \alpha) \text{ is unsatisfiable} \\ 0 & \text{otherwise.} \end{cases}$$

is a lower bound on the complexity of *refuting* F — or, said contrapositively, from an efficient refutation of $F(\vec{x}, \vec{y}, \vec{z})$ in some proof system P we can construct an efficient algorithm computing I in some algorithmic model. Feasible interpolation was introduced at this level of generality in a classic work of Krajíček [31] where it was shown, for instance, that lower bounds on *monotone circuit complexity* of I can be used to show *Resolution* proof length lower bounds for the formula F (provided that the split formula F, and therefore I, is "monotone" in a certain technical sense).

[1] More specifically, we focus on Cutting Planes utilizing Chvátal-Gomory cuts.

[2] The notation $\tilde{\Omega}$ ignores factors of $\log n$.

Instantiations of Krajíček's general interpolation method led to exponential length lower bounds for some proof systems where previously no lower bounds were known. First, Razborov [40] proved lower bounds for certain systems of Bounded Arithmetic from monotone circuit lower bounds. Following this, Bonet, Pitassi and Raz [9] gave exponential lower bounds for "low-weight" Cutting Planes proofs (which have all coefficients bounded by $\mathrm{poly}(n)$), as well as for other variants such as CC-proofs where lines are computed by low-depth communication protocols [31]. In particular they proved that polynomial-length low-weight Cutting Planes refutations of monotone split formulas $F(\vec{x}, \vec{y}, \vec{z})$ (in the above sense) imply polynomial monotone circuits for computing the associated monotone interpolant, I. Then, by constructing a split formula associated with the clique function, they reduced lower bounds for low-weight Cutting Planes proofs and CC proofs to the celebrated monotone circuit lower bounds for the clique function [39].

In [37] Pudlák proved the first exponential lower bounds for *general* Cutting Planes refutations. To obtain this result, he first showed that small Cutting Planes refutations for monotone split formulas $F(\vec{x}, \vec{y}, \vec{z})$ imply small monotone *real* circuits for computing the associated monotone interpolant, I; thus, reducing lower bounds for Cutting Planes proofs of monotone split formulas to monotone *real* circuit lower bounds. Secondly, Pudlák [37] strengthened Razborov's clique lower bound to apply to the larger family of monotone real circuits. Lower bounds on monotone real circuits were also independently proved by Cook and Haken for the broken mosquito screen formulas [24]. Taken together these imply exponential Cutting Planes lower bounds. Pudlák's result was later improved to hold for *semantic* Cutting Planes proofs by Filmus, Hrubeš and Lauria [18].

Despite the success of feasible interpolation, it limits the lower bounds to split formulas; in particular, the only family of formulas which are known to be hard for (unrestricted) Cutting Planes are the clique-coclique formulas [9, 37] and the broken mosquito screen formulas [24].

To prove Theorem 1.1, we generalize Pudlák's feasible interpolation theorem for Cutting Planes so that it can be applied to *any* unsatisfiable CNF formula instead of only "split" formulas. That is, we show that for any unsatisfiable CNF formula F, if there is polynomial-length Cutting Planes refutation of F, then there is a polynomial-size monotone real circuit for computing a corresponding monotone (partial) function, mCSP-SAT_F, where mCSP-SAT_F is a monotone encoding of the CSP-SAT problem whose definition depends on F. In fact, we provide a more general connection that holds not just for Cutting Planes, but for the stronger RCC proof system (cf. Section 2). The next theorem characterizes the length of RCC refutations for any formula F by the size of monotone real circuits computing mCSP-SAT.

Theorem 1.2 (Informal). *Let F be any unsatisfiable CNF formula. If there is an* RCC *refutation of F of length ℓ, then there is a monotone real circuit with* $\mathrm{poly}(\ell)$ *gates computing* mCSP-SAT. *Conversely, if there is a monotone real circuit computing* mCSP-SAT *of size ℓ then there is an* RCC *refutation of F of length* $\mathrm{poly}(\ell)$.

The proof of this theorem is inspired by the seminal Karchmer-Wigderson connection between *circuit complexity* and *communication complexity*, and generalizes some earlier results [9, 31, 40]. In more detail: Karchmer and Wigderson [29] proved that the *depth* of a boolean circuit computing a boolean function $f : \{0,1\}^n \to \{0,1\}$ is exactly the communication complexity of solving a certain communication game related to f. Razborov generalized this result, proving a non-trivial equivalence between *size* of certain dag-like communication protocols and boolean circuit *size* [40].

Razborov's work played a key role in inspiring Krajíček's feasible interpolation result [31] discussed above. Using Razborov's equivalence, Krajíček generalized the result of Bonet, Pitassi, and Raz [9] to obtain a general interpolation theorem, showing that circuit lower bounds for computing interpolant functions imply lower bounds on the powerful CC proof system mentioned above, whose lines consist of any boolean function computed by a low-depth communication protocol. A similar result was shown by Bonet, Pitassi, and Raz [9] showing that CC proofs of the clique-coloring formula imply similarly sized monotone circuits computing the clique function. In Section 4 we first observe that Krajíček's result can be generalized: the complexity of refuting *any* unsatisfiable formula F in the CC proof system is actually *characterized* by the circuit complexity of the mCSP-SAT$_F$ function. As the use of communication protocols introduces unnecessary complications, we give a direct proof of this characterization that is partly inspired by Sokolov's [43] recent simplification of Razborov's result. This observation is already strong enough to give lower bounds on cutting planes with polynomially-bounded coefficients.

Theorem 1.2 provides a similar characterization, but this time for RCC proofs. For this, Razborov's equivalence between dag-like communication protocols and boolean circuit size is insufficient. We instead employ a recent (and beautiful) generalization of Razborov's result due to Hrubeš and Pudlák [27], characterizing the size of monotone *real* circuits in terms of dag-like *real* communication protocols. In Appendix A we discuss how Theorem 1.2 follows from their characterization; again, for the sake of simplicity of presentation, we employ the techniques of [27] to give a direct and streamlined proof of Theorem 1.2 avoiding communication protocols completely. Finally, to deduce Theorem 1.1 from Theorem 1.2, we need to prove lower bounds for monotone real circuits computing the mCSP-SAT problem obtained from a random k-SAT instance; this turns out to be possible by using standard techniques (the symmetric method of approximations [8, 25, 28]). Theorem 1.1 follows because RCC proofs generalize Cutting Planes proofs.

As stated above, Hrubeš and Pudlák have independently proved Theorem 1.1 using nearly identical techniques [26]. Given any unsatisfiable CNF F they show how to obtain a partial monotone boolean function which they call an *unsatisfiability certificate* for F, and then show that the complexity of computing an unsatisfiability certificate by a monotone real circuit implies lower bounds for Cutting Planes by directly reducing these certificates to the feasible interpolation lower bounds. As boolean functions, the unsatisfiability certificates are exactly the same as our mCSP-SAT problem, and their lower bounds for random k-SAT are also obtained by using the symmetric method of approximations [8, 25] in a nearly identical proof to ours. Further, they use this technique to give lower bounds for other problems: a generalization of the Pigeonhole Principle called the *Weak Bit Pigeonhole Principle*, and a function related to Feige's hypothesis.

It is natural to wonder whether or not the new lower bound techniques could be pushed to obtain lower bounds for k-SAT instances when k is bounded. By being a bit more careful, one can obtain superpolynomial lower bounds when $k \gg \log\log n$, but when $k = \Theta(1)$ the method of approximations fails to give superpolynomial lower bounds on the CSP problem. Thus, it appears that we will not be able to push the lower bounds any further via this technique without improving the underlying monotone circuit lower bound techniques.

The Random SAT model. In the random k-SAT model $\mathcal{F}(m, n, k)$ the *unsatisfiability* of a random formula $F \sim \mathcal{F}(m, n, k)$ is controlled by the *clause-density* $\Delta = m/n$. For instance, it is easy to show that if $\Delta > 2^k \ln 2$ then $F \sim \mathcal{F}(m, n, k)$ is unsatisfiable with high probability. The *Satisfiability Threshold Conjecture* states that this control exhibits a threshold phenomena: for all k there exists a fixed constant c_k such that random k-SAT formulas with density $\Delta > c_k$ are almost surely unsatisfiable, while formulas with density $< c_k$ are almost surely satisfiable. For $k = 2$, the conjecture was known to be true since the early 1990s [11, 14, 20]. In a recent breakthrough this conjecture was resolved for large values of k by appealing to arguments in statistical mechanics [16].

The density parameter Δ also plays a role in lower bounds for refuting $\mathcal{F}(m, n, k)$ in propositional proof systems. Our main theorem holds for $\Delta = \Theta(2^{(1+\tau)k})$ for some $0 < \tau < 1$, and furthermore the interval of τ for which our lower bounds hold seems to be relatively narrow (for instance, it seems impossible to choose $\tau \approx 0$ or $\tau \gtrapprox 1$). In contrast, the classic lower bounds by Chvátal and Szemerédi [12] show for any fixed $\Delta > 2^k \ln 2$ there is a constant $\kappa(\Delta)$ such that random k-SAT requires length $\exp(\kappa(\Delta)n)$ with high probability. In their result, κ decays doubly-exponentially as Δ increases, which makes their lower bound trivial when $m \gg n \log^{1/4} n$. Later lower bounds by Beame et al [4] reduce the decay in κ to polynomial in Δ and, in particular, show that a random k-SAT formula with at most $n^{(k+2)}/4$ clauses requires exponential-length Resolution refutations. Beame et al also give asymptotically matching upper bounds, showing tree-like Resolution refutations for random k-SAT of length $\exp(n/\Delta^{1/(k-2)})$. Similar dependencies on the density exist in lower bounds for random k-SAT in other proof systems, such as Polynomial Calculus [7], Res(k) [1], and Sum-of-Squares [41].

2 DEFINITIONS AND PRELIMINARIES

If $x, y \in \{0,1\}^n$ and for all i we have $x_i \leq y_i$ then we write $x \leq y$. A function $f : \{0,1\}^n \rightarrow \{0,1\}$ is *monotone* if $f(x) \leq f(y)$ whenever $x \leq y$. More generally, if $f(x) = 1$ we call x an *accepting instance* or a *yes instance*, while if $f(x) = 0$ then we call x a *rejecting instance* or a *no instance.* If x is any yes instance of f and y is any no instance of f then there exists an index $i \in [n]$ such that $x_i = 1, y_i = 0$, as otherwise we would have $x \leq y$, contradicting the fact that f is monotone.

A *monotone circuit* is a boolean circuit in which all gates are either $\wedge$ or $\vee$ gates. Motivated by proof complexity, Pudlák [37] introduced *monotone real circuits.* In these circuits each internal gate has two inputs and computes any function $\phi(x, y) : \mathbb{R}^2 \rightarrow \mathbb{R}$ which is monotone nondecreasing in its arguments.

Definition 2.1. *A* linear integral inequality *in variables $x = (x_1, \ldots, x_n)$ with coefficients $a = (a_1, \ldots, a_n) \in \mathbb{Z}^n$ and constant term $c \in \mathbb{Z}$ is an expression $a^T x \geq c$.*

Definition 2.2. *Given a system of linear integral inequalities $Ax \geq b$, where $A \in \mathbb{Z}^{m \times n}$ and $b \in \mathbb{Z}^m$, a* Cutting Planes proof *of an inequality $a^T x \geq c$ is a sequence of inequalities ${a_1}^T x \geq c_1, {a_2}^T x \geq c_2, \ldots, {a_\ell}^T x \geq c_\ell$, such that $a_\ell = a$, $c_\ell = c$ and every inequality $i \in [\ell]$ satisfies either*

- *${a_i}^T x \geq c_i$ appears in $Ax \geq b$,*
- *${a_i}^T x \geq c_i$ is a Boolean axiom, i.e., $x_j \geq 0$ or $-x_j \geq -1$ for some j,*
- *there exists $j, k < i$ such that ${a_i}^T x \geq c_i$ is a non-negative linear combination of the linear inequalities ${a_j}^T x \geq c_j$ and ${a_k}^T x \geq c_k$,*
- *there exists $j < i$ and a positive integer d dividing every coefficient in a_j such that $a_i = a_j/d$ and $c_i = \lceil c_j/d \rceil$.*

The length *of the proof is ℓ, the number of lines. If all coefficients appearing in the Cutting Planes proof are bounded by $O(\text{poly}(n))$, then the proof is said to be of* low weight.

Let $F = C_1 \wedge \ldots \wedge C_m$ be an unsatisfiable CNF formula over variables $x_1, \ldots, x_n$. For any clause C let C^- denote the variables that are negated in C and let C^+ denote variables that are not negated in C. Each clause C in F can be encoded as a linear integral inequality as $\sum_{x_i \in C^+} x_i + \sum_{x_i \in C^-} (1 - x_i) \geq 1$. Thus, each unsatisfiable CNF can be translated into a system of linear integral inequalities $Ax \geq b$ with no 0/1 solutions. A *Cutting Planes (CP) refutation* of this system is a Cutting Planes proof of the inequality $0 \geq 1$ from $Ax \geq b$.

We will also be interested in *semantic* proof systems in which the lines are restricted but we allow any sound deduction. If $f, g, h : \{0,1\}^n \rightarrow \{0,1\}$ are boolean functions on the same domain then write $f, g \vDash h$ if for all $x \in \{0,1\}^n$ we have $f(x) \wedge g(x) \implies h(x)$.

Definition 2.3. *Let $F = C_1 \wedge \ldots \wedge C_m$ be an unsatisfiable k-CNF and let $\mathcal{L} \supseteq \{C_1, C_2, \ldots, C_m\}$ be any collection of boolean functions. An $\mathcal{L}$-*semantic refutation *of F is a sequence $L_1, L_2, \ldots, L_\ell$ of boolean functions $L_i \in \mathcal{L}$ such that*

(1) *$L_i = C_i$ for all $i = 1, 2, \dots, m$.*
(2) *$L_\ell = 0$, the constant 0 function.*
(3) *For all $i > m$ there exists $j, k < i$ such that $L_j, L_k \vDash L_i$.*

The length *of the refutation is ℓ.*

When $\mathcal{L}$ is the set of linear integral inequalities then the resulting proof system is called *semantic Cutting Planes*, and has been previously studied in earlier works [9, 18, 34]. We will be particularly interested in semantic refutations where the lines are computed by efficient communication protocols. We quickly review the framework of communication complexity; for a more detailed introduction, we recommend the excellent exposition by Kushilevitz and Nisan [33].

Definition 2.4. *A d-round* communication protocol *P consists of two players, Alice, who receives an input $x \in \mathcal{X}$, and Bob, who receives an input $y \in \mathcal{Y}$. At each round one of the players, determined by the communication so far, sends a bit, depending on his or her input as well as the bits communicated thus far, to the other. After d rounds, the players output a bit b. The protocol computes a function $F : \mathcal{X} \times \mathcal{Y} \to \{0, 1\}$ if for all $(x, y) \in \mathcal{X} \times \mathcal{Y}$ the protocol outputs $F(x, y)$.*

A d-round communication protocol can be imagined as a full binary tree (known as a *protocol tree*) of depth at most d, where each node corresponds to one of the players speaking, and the two outgoing edges of that node are labelled with 0 and 1. Each root-to-leaf path (equivalently, transcript of bits sent by Alice and Bob) is known as a *history* of the communication protocol. Of course, a d-round protocol can have at most 2^d leaves, and therefore histories. The leaves are labelled with the bit b output by Alice and Bob when communicating according to the history that takes them to this leaf.

For any communication protocol P, it is useful to think of an associated matrix M (known as a *communication matrix*), with rows indexed by $x \in \mathcal{X}$ and columns indexed by $y \in \mathcal{Y}$. The entry at index (x, y) is the outcome of the protocol $P(x, y)$. Initially, before communication begins, Alice and Bob each hold a copy of M. Each bit sent by Alice partitions the rows of the matrix M into two sets, one consistent with Alice sending the bit 0 and the other with Alice sending 1. Similarly, the columns of the matrix are partitioned when Bob sends a bit. Therefore, at every round, Alice and Bob hold a subset $R \subseteq \mathcal{X} \times \mathcal{Y}$ of the indices of M. This subset is known as a *(combinatorial) rectangle* because it satisfies if $(x, y) \in R$ and $(x', y') \in R$, then $(x', y), (x, y') \in R$. The protocol ends when Alice and Bob hold a *monochromatic rectangle*, a rectangle R such that for every $(x, y) \in R$, the outcome of $P(x, y)$ is b, for some $b \in \{0, 1\}$; we call such a rectangle b-monochromatic. Because the protocol P outputs a bit b on every input (x, y), the set of histories and the set of monochromatic rectangles are in 1-1 correspondence. Therefore, every history h has a corresponding monochromatic rectangle $R(h)$ of M. Furthermore, if the players output b on history h, then $R(h)$ is b-monochromatic.

Semantic refutations where the lines are computed by low-depth communication protocols were introduced by Krajíček in the study of feasible interpolation [31], and are defined next.

Definition 2.5. *Let F be an unsatisfiable CNF and let (X, Y) be any partition of the variables of F. A CC_d-refutation of F with respect to the partition (X, Y) is a semantic refutation $L_1, \dots, L_\ell$ of F such that each function L_i in the proof can be computed by a d-bit communication protocol with respect to the partition (X, Y).*

Observe that since any linear integral inequality $a^T x + b^T y \geq c$ with polynomially bounded weights can be evaluated by a trivial $O(\log n)$-bit communication protocol (just by having Alice evaluating $a^T x$ and sending the result to Bob), it follows that low-weight Cutting Planes proofs are also $\mathrm{CC}_{O(\log n)}$-proofs. By strengthening the the underlying communication protocol we can simulate any Cutting Planes proof; this type of protocol was also introduced by Krajíček [32].

Definition 2.6. *A d-round* real communication protocol *is a communication protocol between two players, Alice and Bob, where Alice receives $x \in \mathcal{X}$ and Bob receives $y \in \mathcal{Y}$. In each round, Alice and Bob each send real numbers α, β to a "referee", who responds with a single bit b which is* 1 *if $\alpha \leq \beta$ and* 0 *otherwise. After d rounds of communication, the players output a bit b. The protocol computes a function $F : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ if for all $(x, y) \in \mathcal{X} \times \mathcal{Y}$ the protocol outputs $F(x, y)$.*

Definition 2.7. *Let F be an unsatisfiable CNF and let (X, Y) be any partition of the variables of F. An* RCC_d*-refutation of F is a semantic refutation $L_1, L_2, \dots, L_\ell$ in which each function L_i can be computed by a d-round real communication protocol with respect to the partition (X, Y).*

It is clear that *any* linear integral inequality $a^T x + b^T y \geq c$ can be evaluated by a 1-round real communication protocol: Alice sends $a^T x$ to the referee and Bob sends $c - b^T y$. It follows that a Cutting Planes refutation of F is also an RCC_1-refutation of F. We record each of these observations in the next proposition.

Proposition 2.1. *Let F be an unsatisfiable CNF and let (X, Y) be any partition of the variables into two sets. Any length-ℓ low-weight Cutting Planes refutation of F is a length-ℓ* $\mathrm{CC}_{O(\log n)}$*-refutation of F. Similarly, any length-ℓ Cutting Planes refutation of F is a length-ℓ* RCC_1*-refutation of F.*

Although one only needs to establish the equivalence between RCC_1-proofs and monotone real circuits in order to obtain lower bounds for Cutting Planes proofs, we believe that the equivalence between CC-proofs and monotone circuits is interesting in its own right.

3 UNSATISFIABLE FORMULAS AND MONOTONE CSP-SAT

In this section we introduce mCSP-SAT, which is a monotone version of SAT that plays a central role in our results. Given any unsatisfiable CNF formula F and any partition (X, Y) of F's variables we then show how to produce a corresponding collection of instances of mCSP-SAT. More precisely: for each assignment $X \to \{0,1\}$ to the X variables we will obtain an accepting instance of mCSP-SAT, and for each assignment $Y \to \{0,1\}$ to the Y variables we will obtain a rejecting instance of mCSP-SAT. In the next section, we will show that separating these mCSP-SAT instances by a monotone boolean circuit is *equivalent* to refuting F in the CC proof system with respect to the partition (X, Y) (and we show a similar result for real circuits and RCC_1 refutations). The mCSP-SAT problem has appeared in many different guises in different works — the function essentially appears in the work of Raz and McKenzie [38] under a different name, and it has re-appeared in recent work on lifting theorems in communication complexity [23, 36].

In order to define mCSP-SAT we first introduce a very general form of the boolean constraint satisfaction problem.

Definition 3.1. *A* constraint satisfaction problem *(CSP) $\mathcal{H}$ is defined as follows. Let $H = (L \cup R, E)$ be a bipartite graph and let $n = |R|$. The vertices in L represent the* constraints *of the CSP $\mathcal{H}$, and the vertices in R represent boolean valued* variables. *For each $i \in L$ we let* $\mathrm{vars}(i) \subseteq R$ *denote the neighbourhood of i and we associate a boolean function* $\mathrm{TT}_i : \{0,1\}^{\mathrm{vars}(i)} \to \{0,1\}$ *called the* truth table of i *that encodes the set of satisfying assignments to the ith constraint. The CSP $\mathcal{H}$* accepts *an assignment $\rho \in \{0,1\}^R$ if $TT_i(\rho \restriction \mathrm{vars}(i)) = 1$ for all i, and it is* satisfiable *if it accepts some assignment.*

The mCSP-SAT problem is then defined by simply fixing the underlying constraint graph H and letting the input string specify each of the truth tables TT_i.

Definition 3.2. *Let $H = (L \cup R, E)$ be a bipartite graph and let $N = \sum_{i \in L} 2^{|\mathrm{vars}(i)|}$. The boolean function* $\text{mCSP-SAT}_H : \{0,1\}^N \to \{0,1\}$ *is defined as follows. An input $z \in \{0,1\}^N$ encodes a CSP*

$\mathcal{H}_z$ by specifying for each vertex $i \in L$ its truth table $\mathrm{TT}_i : \{0,1\}^{\mathsf{vars}(i)} \to \{0,1\}$*. For any $z \in \{0,1\}^N$,* mCSP-SAT$_H(z) = 1$ *if and only if the CSP $\mathcal{H}_z$ encoded by z is satisfiable.*

Observe that this is a monotone boolean function since for any $z, z' \in \{0,1\}^N$ with $z \le z'$ (that is, $z_i \le z'_i$ for every $i \in [N]$), any satisfying assignment for the CSP $\mathcal{H}_z$ is also a satisfying assignment for the CSP $\mathcal{H}_{z'}$. This is because z and z' both encode sets of truth tables, and so flipping any bit from 0 to 1 simply makes one of the constraints easier to satisfy.

Next, we show how to take any unsatisfiable k-CNF formula F and any partition of Fs variables and produce a collection of accepting and rejecting instances of mCSP-SAT. This reduction provides the key link between *refutations* of F and *computations* of mCSP-SAT.

Definition 3.3. *Let F be an unsatisfiable k-CNF and let (X, Y) be any partition of the variables of F into two sets. Let $H = H(F, X)$ denote the constraint graph of F* restricted to the X variables, *and consider* mCSP-SAT$_H$*, which is a boolean function on N boolean variables. Define sets of accepting and rejecting instances of* mCSP-SAT$_H$ *from F as follows.*

- **Accepting Instances $\mathcal{U}$.** *For any $x \in \{0,1\}^X$ define $\mathcal{U}(x) \in \{0,1\}^N$ as follows. For each $i \in [m]$ and each $\alpha \in \{0,1\}^{\mathsf{vars}(i)}$ set $TT_i(\alpha) = 1$ iff $x \restriction \mathsf{vars}(i) = \alpha$.*
- **Rejecting Instances $\mathcal{V}$.** *For any $y \in \{0,1\}^Y$ define $\mathcal{V}(y)$ as follows. For each $i \in [m]$ and each $\alpha \in \{0,1\}^{\mathsf{vars}(i)}$ set* $\mathrm{TT}_i(\alpha) = 1$ *iff $C_i(\alpha, y) = 1$.*

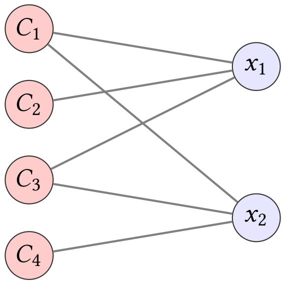
When the underlying unsatisfiable CNF F is clear from context, we will write mCSP-SAT$_F$ *to mean the partial monotone boolean function corresponding to the above set of accepting and rejecting instances.*

Note that accepting and rejecting inputs to mCSP-SAT$_F$ have the following structure. The CSP $\mathcal{H}_{\mathcal{U}(x)}$ corresponding to $\mathcal{U}(x)$ has each truth table TT_i to be 0 everywhere except for exactly one 1 corresponding to x, and it follows that the corresponding CSP $\mathcal{H}_z$ has x as its unique satisfying assignment. In particular, $\mathcal{H}_{\mathcal{U}(x)}$ is satisfiable and so it is an accepting instance of mCSP-SAT. On the other hand, the CSP $\mathcal{H}_{\mathcal{V}(y)}$ corresponding to $\mathcal{V}(y)$ is exactly $F(x, y)$ (note the y variables are fixed); since F is an unsatisfiable CNF formula it follows that $\mathcal{H}_{\mathcal{V}(y)}$ is also unsatisfiable and so a rejecting instance of mCSP-SAT. We give a detailed example next.

Example 3.4. Consider the unsatisfiable CNF formula

$$F = (x_1 \vee x_2 \vee y_1) \wedge (\bar{x}_1) \wedge (x_1 \vee \bar{x}_2) \wedge (x_2 \vee \bar{y}_1)$$

with the obvious partition into x- and y-variables. The underlying constraint graph of mCSP-SAT$_F$ is depicted below — note that we only keep the x variables from the underlying CNF formula.



Consider the truth assignment $x = (1,1)$ and $y = (1)$. The mCSP-SAT$_F$ input $\mathcal{U}(x)$ has $TT_i(\alpha) = 1$ if and only if $\alpha = (1,1)$; equivalently, each constraint TT_i in the CSP is just the AND function $x_1 \wedge x_2$. On the other hand, the mCSP-SAT$_F$ input encoded by $\mathcal{V}(y)$ is obtained by substituting $y = 1$ into each constraint of F, yielding the constraints $TT_1 = 1, TT_2 = \neg x_1, TT_3 = x_1 \vee \neg x_2, TT_4 = x_2$; these constraints are easily seen to be unsatisfiable.

4 RELATING PROOFS AND CIRCUITS

In this section we prove the equivalence between CC_d-proofs and monotone circuits, as well as RCC_1-proofs and monotone *real* circuits. Our argument relating CC_d and monotone circuits is a direct generalization of the main theorem of Bonet, Pitassi, and Raz [9], which establishes the equivalence for the special case of the clique-coclique formulas. A similar argument of this type also appears in the work of Razborov [40]; Razborov's work was recently simplified by Sokolov [43].

Theorem 4.1. *Let F be an unsatisfiable CNF formula on n variables and let (X, Y) be any partition of the variables. Let d be a positive integer. If there is a CC_d-refutation of F with respect to the partition (X, Y) of length ℓ, then there is a monotone circuit separating the accepting and rejecting instances $\mathcal{U}(\{0,1\}^{|X|}), \mathcal{V}(\{0,1\}^{|Y|})$ of $\mathrm{mCSP\text{-}SAT}_F$ of size $O(2^{3d}\ell)$.*

First, we give a high-level sketch of the argument. From a CC_d-proof we will construct a monotone circuit inductively starting with the input clauses of the proof and progressing to the final line. Roughly, for each line L we will construct a "cluster" of circuits C^L satisfying the following property: if L is falsified by a truth assignment (x, y), then C^L "separates" $\mathcal{U}(x)$ and $\mathcal{V}(y)$, meaning that $C^L(\mathcal{U}(x)) = 1$ and $C^L(\mathcal{V}(y)) = 0$. To construct C^L we will use the soundness of the proof. If L was derived from L' and L'' in the proof, then by induction we will have constructed circuits $C^{L'}$ and $C^{L''}$. By soundness, every assignment (x, y) that falsifies L will falsify at least one of L' and L'', and so at least one of the corresponding circuits $C^{L'}$ and $C^{L''}$ will separate $\mathcal{U}(x)$ and $\mathcal{V}(y)$. Using this, we will construct C^L from the circuits $C^{L'}$ and $C^{L''}$. Once we arrive at the final line of the proof, because every truth assignment falsifies $0 \geq 1$, the corresponding circuit will separate $\mathcal{U}(\{0,1\}^{|X|})$ and $\mathcal{V}(\{0,1\}^{|Y|})$.

More concretely, because each line in the CC_d-proof can be computed by a small communication protocol, this induces a partition of the truth assignments to L into at most 2^d monochromatic rectangles. Instead of constructing only a single circuit for each line L, we will actually construct one for every 0-monochromatic rectangle (containing inputs that falsify L) R of L, which will separate $\mathcal{U}(x)$ and $\mathcal{V}(y)$ for every $(x, y) \in R$.

Proof. Let $F = C_1 \wedge \ldots \wedge C_m$ be an unsatisfiable CNF formula over variables $x_1, \ldots, x_{n_1}$, $y_1, \ldots, y_{n_2}$. Let P be a CC_d-proof for F with ℓ lines. Order the lines in P as $L_1, L_2, \ldots, L_\ell$, where each line is either an input clause, or follows semantically from two earlier lines.

We build the circuit for $\mathrm{mCSP\text{-}SAT}_F$ that separates $\mathcal{U}, \mathcal{V}$ by induction on ℓ, the number of lines. By definition, each line L can be computed by a d-round communication protocol. Therefore, for each line L there are at most 2^d possible histories h, each with an associated monochromatic rectangle $R_L(h)$. Recall that each monochromatic rectangle is a subset of truth assignments that evaluate the same under L. We call a history h *good* for L if $R_L(h)$ is 0-monochromatic. Therefore, a good history is one for which every assignment in the associated monochromatic rectangle falsifies L. For every line L and each good history h for L, we will build a monotone circuit C^L_h that correctly "separates" x and y for each $(x, y) \in R_L(h)$. By this, we mean that the circuit C^L_h outputs 1 on $\mathcal{U}(x)$ (the accepting-input associated with x for $\mathrm{mCSP\text{-}SAT}_F$) and outputs 0 on $\mathcal{V}(y)$ (the rejecting-input associated with y). Because every assignment falsifies the final line $0 \geq 1$, the associated monotone circuit will separate $\mathcal{U}$ from $\mathcal{V}$.

For each leaf in the proof, the associated line L is a clause C_i of F. The communication protocol for C_i is a two-bit protocol where Alice/Bob each send 0 iff their inputs α, β are such that $C_i(\alpha, \beta) = 0$. Thus, there is only one good (0-monochromatic) rectangle with history $h = 00$ for $L = C_i$. This pair α, β corresponds to the variable $\mathrm{TT}_i(\alpha)$ of $\mathrm{mCSP\text{-}SAT}_F$, and we define the circuit C^L_h corresponding to line $L = C_i$ and good history $h = 00$ to be the variable $\mathrm{TT}_i(\alpha)$.

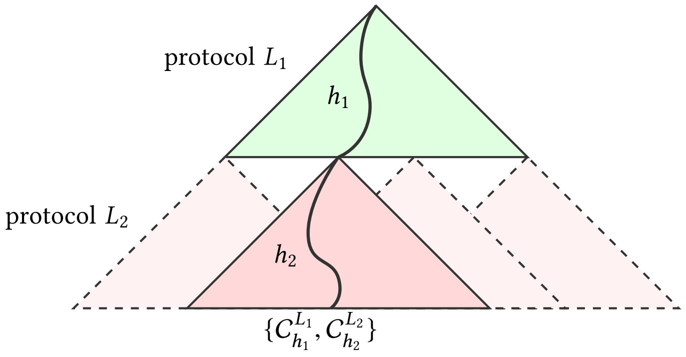


Fig. 1. Protocol tree T.

Now suppose that L is derived from L_1 and L_2, and inductively we have circuits $C^{L_1}_{h'}, C^{L_2}_{h''}$ for each history h' good for L_1 and h'' good for L_2. Given a good history h for L, we will show how to build the circuit C^L_h. It will use all of the circuits that were built for L_1 and L_2 ($\{C^{L_1}_{h'}, C^{L_2}_{h''}\}$ for all good h' and h'') and an additional 2^d gates. To build C^L_h we will construct a *stacked* protocol tree for L, corresponding to first running the communication protocol for L_1 and then running the communication protocol for L_2. This will give us a height $2d$ (full) binary tree, T, where the top part is the communication protocol tree for L_1, with protocol trees for L_2 hanging off of each of the leaves (Figure 1). We label each of the leaves of this stacked tree with a circuit from $\{C^{L_1}_{h'}, C^{L_2}_{h''}\}$ as follows. Consider a path labelled $h_1 h_2$ in T, where h_1 is the history from running L_1 and h_2 is the history from running L_2. Because L is derived by a sound inference from L_1 and L_2, any assignment that falsifies L must falsify at least one of L_1 or L_2. Since $R_L(h)$ is 0-monochromatic (with respect to the communication matrix for L), for every $(x', y') \in R_L(h)$ there is some $i \in \{1, 2\}$ such that $L_i(x', y') = 0$. Therefore, because $R_{L_1}(h_1)$ and $R_{L_2}(h_2)$ are monochromatic rectangles, it follows that either

(i) the rectangle $R_{L_1}(h_1) \cap R_L(h)$ is 0-monochromatic (with respect to the communication matrix of L_1), or
(ii) the rectangle $R_{L_2}(h_2) \cap R_L(h)$ is 0-monochromatic (with respect to the communication matrix of L_2).

In the first case, we will label this leaf with $C^{L_1}_{h_1}$ and otherwise we will label this leaf with $C^{L_2}_{h_2}$. Now we will label the internal vertices of the stacked tree with a gate: if a node corresponds to Alice speaking, then we label the node with an $\vee$ gate, and otherwise if the node corresponds to Bob speaking, then we label the node with an $\wedge$ gate. The resulting monotone circuit[3] for this history h has size 2^{2d} plus the sizes of the sub-circuits, and thus performing the construction for each of the 2^d histories increases circuit size by a factor of 2^{3d}. With this, the theorem is immediately implied by the following claim.

Claim. The monotone circuit resulting from the above construction satisfies: for each line L in P, and for each good history h for L, C^L_h will be correct for all $(x, y) \in R_L(h)$. That is, $C^L_h(\mathcal{U}(x)) > C^L_h(\mathcal{V}(y))$ for every $(x, y) \in R_L(h)$.

[3] The resulting circuit is monotone because the only gates used are $\vee$ and $\wedge$, each of which is a monotone function.

Proof of Claim. If L is an axiom, then L is a clause C_i. The communication protocol for C_i is a two-bit protocol where Alice and Bob each send 0 iff their part of C_i evaluates to 0. There is only one good (0-monochromatic) history, $h = 00$. If $(x, y) \in R_L(h)$ then $C_i(x, y) = 0$ by definition. Let $\alpha = x \upharpoonright \text{vars}(C_i)$. In our construction the circuit corresponding to C^L_h is labelled by the variable $\text{TT}_i(\alpha)$, and it is easy to check that $\mathcal{U}(x)$ sets $\text{TT}_i(\alpha)$ to true, and $\mathcal{V}(y)$ sets $\text{TT}_i(\alpha)$ to false.

If L is not an axiom, then we will prove the claim by proving the following stronger statement by induction: for each line L (derived from previous lines L_1 and L_2), and for each node v in the stacked protocol tree for L, with corresponding (sub)history $h' = h_1h_2$, the subcircuit $C^L_{h'}$ associated with vertex v is correct on all $(x, y) \in R_L(h) \cap R_{L_1}(h_1) \cap R_{L_2}(h_2)$. The claim follows, because once we reach $h' = \emptyset$, then C^L_h will be correct on $(x, y) \in R_L(h) \cap R_{L_1}(h_1) \cap R_{L_2}(h_2) = R_L(h)$. This follows because if $h_i = \emptyset$, then $R_{L_i}(h_i) = \{0, 1\}^{|X|} \times \{0, 1\}^{|Y|}$, that is, if Alice and Bob haven't communicated anything in history h_i, then the corresponding rectangle is the entire communication matrix.

Fix a line L that is not an axiom. For the base case, suppose that v is a leaf of the stacked protocol tree for L with history $h' = h_1h_2$. Then by soundness either

(i) $R_{L_1}(h_1) \cap R_L(h)$ is 0-monochromatic (with respect to the communication matrix of L_1), or
(ii) $R_{L_2}(h_2) \cap R_L(h)$ is 0-monochromatic (with respect to the communication matrix of L_2).

In case (i) we labelled v by $C^{L_1}_{h_1}$. Since $R_{L_1}(h_1) \cap R_L(h)$ is 0-monochromatic, and because $R_{L_1}(h_1)$ is a monochromatic rectangle, $R_{L_1}(h_1)$ is 0-monochromatic. By induction $C^{L_1}_{h_1}$ is defined and is correct on all $(x, y) \in R_{L_1}(h_1)$, so it is correct on all $(x, y) \in R_L(h) \cap R_{L_1}(h_1) \cap R_{L_2}(h_2)$. A similar argument holds in case (ii).

For the inductive step, let v be a non-leaf node in the protocol tree with history h'. Assume that Alice owns v, and so v is labelled with an $\vee$ gate. The rectangle $R_L(h) \cap R_{L_1}(h_1) \cap R_{L_2}(h_2) = A \times B$ is partitioned into $A_0 \times B$ and $A_1 \times B$, where

(1) $A = A_0 \cup A_1$,
(2) $A_0 \times B$ is the rectangle with history $h'0$,
(3) $A_1 \times B$ is the rectangle with history $h'1$.

Given $(x, y) \in R_L(h) \cap R_{L_1}(h_1) \cap R_{L_2}(h_2)$, since by induction $C^L_{h'0}$ is correct on all $(x, y) \in A_0 \times B$ and $C^L_{h'1}$ is correct on all $(x, y) \in A_1 \times B$, it follows that $C^L_{h'} = C^L_{h'0} \vee C^L_{h'1}$ is correct on all $(x, y) \in A \times B$. To see this, observe that if $x \in A_0$, then $C^L_{h'0}(\mathcal{U}(x)) = 1$ and therefore

$$C^L_{h'}(\mathcal{U}(x)) = C^L_{h'0}(\mathcal{U}(x)) \vee C^L_{h'1}(\mathcal{U}(x)) = 1.$$

The same applies when $x \in A_1$, as then $C^L_{h'1}(\mathcal{U}(x)) = 1$. Finally if $y \in B$ then both $C^L_{h'0}(\mathcal{V}(y)) = C^L_{h'1}(\mathcal{V}(y)) = 0$ and therefore

$$C^L_{h'}(\mathcal{V}(y)) = C^L_{h'0}(\mathcal{V}(y)) \vee C^L_{h'1}(\mathcal{V}(y)) = 0.$$

A similar argument holds if v is an internal node in the protocol tree that Bob owns (and is therefore labelled by an $\wedge$ gate). □

The converse direction is much easier. Although the converse is not necessary in order to establish Cutting Planes lower bounds, we believe the equivalence between monotone circuits and $\text{CC}_{O(\log n)}$-proofs to be of independent interest.

Theorem 4.2. *If there is a monotone circuit separating the inputs of* mCSP-SAT$_F$ *of size ℓ, then there is a* CC_2*-refutation of F of length ℓ with respect to this variable partition.*

Proof. We show that from a small monotone circuit C for mCSP-SAT$_F$ that separates $\mathcal{U}(\{0, 1\}^{|X|})$ and $\mathcal{V}(\{0, 1\}^{|Y|})$, we can construct a small CC_2-proof for F, where Alice gets $x \in \{0, 1\}^{|X|}$ and Bob gets $y \in \{0, 1\}^{|Y|}$. The lines/vertices of the refutation will be in 1-1 correspondence with the gates of C. The protocol is constructed inductively from the leaves of C to the root. For a gate g of C, let U_g be those inputs $u \in \mathcal{U}(\{0,1\}^{|X|})$ such that $g(u) = 1$, and let V_g be those inputs $v \in \mathcal{V}(\{0,1\}^{|Y|})$ such that $g(v) = 0$. At each gate g we will prove that for every pair $(u, v) \in U_g \times V_g$ and for every (x, y) such that $u = \mathcal{U}(x), v = \mathcal{V}(y)$, the protocol R_g on input (x, y) will output 0. Since the output gate of C is correct for all pairs, this will achieve our desired protocol.

If g is a leaf of the circuit labeled by some variable $\mathrm{TT}_j(\alpha)$, the pairs associated with this leaf must have $\mathrm{TT}_j(\alpha) = 1$ in u and 0 in v, and thus we can define $R_g(x, y)$ to be 0 if and only if x is consistent with α and the clause C_j evaluates to false on (x, y). This is a 2-bit protocol, and by definition of the accepting and rejecting instances we have for all (x, y) satisfying $u = \mathcal{U}(x), v = \mathcal{V}(y)$ that $x \restriction \mathsf{vars}(j) = \alpha$ and $C_j(\alpha, y) = 0$.

Now suppose that g is an $\vee$ gate of C, with inputs g_1, g_2, and let C_{g_1}, C_{g_2} be the sub-circuits of C rooted at g_1 and g_2 respectively. The protocol R_g on (x, y) is as follows. Alice privately simulates $C_{g_1}(\mathcal{U}(x))$ and $C_{g_2}(\mathcal{U}(x))$, and Bob simulates $C_{g_1}(\mathcal{V}(y))$ and $C_{g_2}(\mathcal{V}(y))$. If (i) either $C_{g_1}(\mathcal{U}(x)) = 1$ or $C_{g_2}(\mathcal{U}(x)) = 1$ and (ii) both $C_{g_1}(\mathcal{V}(y)) = 0$ and $C_{g_2}(\mathcal{V}(y)) = 0$, then they output 0, and otherwise they output 1. This is a 2-bit protocol, with Alice sending one bit to report whether or not condition (i) is satisfied, and Bob sending one bit to report if (ii) is satisfied.

Now, we want to show that for all (x, y) such that $C_g(\mathcal{U}(x)) = 1$ and $C_g(\mathcal{V}(y)) = 0$ we have that $R_g(x, y) = 0$. This is easy — since $g = g_1 \vee g_2$ we have that $C_g(\mathcal{U}(x)) = 1$ and $C_g(\mathcal{V}(y)) = 0$ implies that either $C_{g_1}(\mathcal{U}(x)) = 1$ or $C_{g_2}(\mathcal{U}(x)) = 1$ and $C_{g_1}(\mathcal{V}(y)) = 0$ and $C_{g_2}(\mathcal{V}(y)) = 0$, implying that the protocol will output 0 on (x, y) by definition.

Similarly, if g is an $\wedge$ gate, then again Alice privately simulates $C_{g_1}(\mathcal{U}(x))$ and $C_{g_2}(\mathcal{U}(x))$ and Bob privately simulates $C_{g_2}(\mathcal{V}(y))$ and $C_{g_2}(\mathcal{V}(y))$. If (i) $C_{g_1}(\mathcal{U}(x)) = 1$ and $C_{g_2}(\mathcal{U}(x)) = 1$ and (ii) either $C_{g_2}(\mathcal{V}(y)) = 0$ or $C_{g_2}(\mathcal{V}(y)) = 0$, then they ouput 0, and otherwise they output 1. By an analogous argument to the $\vee$ case, it's easy to see that the protocol will output 0 whenever $C_g(\mathcal{U}(x)) = 1$ and $C_g(\mathcal{V}(y)) = 0$. □

The next theorem relates RCC_1 proofs and monotone real circuits. The proof (which is in Appendix A) crucially uses a recent technical result regarding real monotone circuits due to Pavel Hrubeš and Pavel Pudlák [27].

Theorem 4.3 (cf. Theorem 1.2). *Let F be an unsatisfiable CNF formula and let (X, Y) be any partition of the variables. If there is a RCC_1-refutation of F with respect to the partition (X, Y) of length ℓ, then there is a monotone real circuit separating the accepting and rejecting instances $\mathcal{U}(\{0,1\}^{|X|}), \mathcal{V}(\{0,1\}^{|Y|})$ of $\mathrm{mCSP\text{-}SAT}_F$ of size ℓ. Conversely, a monotone real circuit separating the inputs of $\mathrm{mCSP\text{-}SAT}_F$ implies a RCC_1-refutation of F of the same length.*

Because every Cutting Planes line can be computed by a single-round real communication protocol (Proposition 2.1), the above theorem implies that for any family of formulas F and for any partition of the underlying variables into (X, Y), a Cutting Planes refutation of F of length ℓ implies a similar size monotone real circuit for separating the accepting and rejecting instances $\mathcal{U}(\{0,1\}^{|X|}), \mathcal{V}(\{0,1\}^{|Y|})$ of $\mathrm{mCSP\text{-}SAT}_F$. Thus, lower bounds on the size of monotone real circuits give lower bounds on the length of Cutting Planes proofs.

5 LOWER BOUNDS FOR RANDOM CNFS

In this section we use Theorem 4.3 to prove Theorem 1.1. In particular, we prove lower bounds for RCC_1-refutations (and therefore Cutting Planes refutations) of uniformly random k-CNFs with sufficient clause density.

Definition 5.1. *Let $\mathcal{F}(m, n, k)$ denote the distribution of random k-CNFs on n variables obtained by sampling m clauses (out of the $\binom{n}{k}2^k$ possible clauses) uniformly at random.*

The proof of Theorem 1.1 is delayed to Section 5.2; to get a feeling for the argument, we first prove an easier lower bound for a simpler distribution of *balanced* random CNFs.

5.1 Balanced Random CNFs

Definition 5.2. *Let $X = \{x_1, \dots, x_n\}$ and $Y = \{y_1, \dots, y_n\}$ be two disjoint sets of variables, and let $\mathcal{F}(m, n, k)^{\otimes 2}$ denote the following distribution over $2k$-CNFs: first sample $F^1 = C^1_1 \wedge C^1_2 \wedge \cdots \wedge C^1_m$ from $\mathcal{F}(m, n, k)$ on the X variables, and then $F^2 = C^2_1 \wedge C^2_2 \wedge \cdots \wedge C^2_m$ from $\mathcal{F}(m, n, k)$ on the Y variables independently. Then output*

$$F = (C^1_1 \vee C^2_1) \wedge (C^1_2 \vee C^2_2) \wedge \cdots \wedge (C^1_m \vee C^2_m).$$

This distribution shares the well-known property with $\mathcal{F}(m, n, k)$ that dense enough formulas are unsatisfiable with high probability.

Lemma 5.1. *Let $c > 2/\log e$ and let n be any positive integer. If $k \in [n]$ and $m \geq cn2^{2k}$ then $F \sim \mathcal{F}(m, n, k)^{\otimes 2}$ is unsatisfiable with high probability.*

Proof. Fix any assignment (x, y) to the variables of F. The probability that the ith clause is satisfied by the joint assignment is $1 - 1/2^{2k}$, and so the probability that *all* clauses are satisfied by the joint assignment is $(1 - 1/2^{2k})^m \leq e^{-m/2^{2k}}$, since the clauses are sampled independently. By the union bound, the probability that some joint assignment satisfies the formula is at most $2^{2n}e^{-m/2^{2k}} = 2^{2n-(\log e)m/2^{2k}} \leq 2^{2n-(\log e)cn} \leq 2^{-\Omega(n)}$. Thus, the probability that the formula is unsatisfiable is at least $1 - 2^{-\Omega(n)}$. □

The main theorem of this section is that $F \sim \mathcal{F}(m, n, k)^{\otimes 2}$ requires large RCC_1-proofs, which is obtained by using Theorem 4.3 and applying the well-known method of symmetric approximations [8, 25] to obtain lower bounds on monotone circuits computing mCSP-SAT$_F$. We use the following formalization of the method which is exposited in Jukna's excellent book [28]. First we introduce some notation: if $U \subseteq \{0, 1\}^N$, then for $r \in [N]$ and $b \in \{0, 1\}$ let

$$A_b(r, U) = \max_{I \subseteq [N]:|I|=r} |\{u \in U \mid \forall i \in I : u_i = b\}|.$$

Theorem 5.2 (Theorem 9.19 in Jukna). *Let $f : \{0, 1\}^N \longrightarrow \{0, 1\}$ be a monotone boolean function and let $1 \leq r, s \leq N$ be any positive integers. Let $U \subseteq f^{-1}(1)$ and $V \subseteq f^{-1}(0)$ be arbitrary subsets of accepting and rejecting inputs of f. Then every monotone real circuit that outputs 1 on all inputs in U and 0 on all inputs in V has size at least*

$$\min\left\{\frac{|U| - (2s)A_1(1, U)}{(2s)^{r+1}A_1(r, U)}, \frac{|V|}{(2r)^{s+1}A_0(s, V)}\right\}.$$

Next, we state the main theorem of this section.

Theorem 5.3. *Let $k = 4 \log n$ and $m = cn2^{2k}$ where $c > 2/\log e$ is some constant. Let $F \sim \mathcal{F}(m, n, k)^{\otimes 2}$ with variable partition (X, Y), and let $U = \mathcal{U}(\{0, 1\}^{|X|})$, $V = \mathcal{V}(\{0, 1\}^{|Y|})$. Then with high probability any monotone real circuit separating U and V has at least $2^{\tilde{\Omega}(n)}$ gates.*

Corollary 5.4. *Let n be a sufficiently large positive integer, and let $k = 4 \log n$ and $m = cn^9$ where $c > 2/\log e$ is some constant. If $F \sim \mathcal{F}(m, n, k)^{\otimes 2}$ then with high probability every RCC_1-refutation (and therefore, Cutting Planes refutation) of F has at least $2^{\tilde{\Omega}(n)}$ lines.*

Proof. Immediate consequence of Theorems 4.3 and 5.3. □

The proof of Theorem 5.3 comes down to the essential property that random k-CNFs are good expanders. The next lemma records the expansion properties we require of random CNFs; the proof is adapted from the notes of Salil Vadhan [44]. For any subset $S \subseteq F$ of clauses of a CNF F let $\mathrm{vars}(S)$ denote the subset of variables appearing in clauses S.

Lemma 5.5. *Let n be any sufficiently large positive integer. Let k, m be positive integers and sample $F \sim \mathcal{F}(m, n, k)$. Suppose for some $0 < \delta < 1$ we have*

$$\log m \leq \delta \cdot \frac{k}{2} \log\left(\frac{k}{2}\right).$$

Then every set $S \subseteq F$ of size $s \leq n/ek^2$ satisfies $|\mathrm{vars}(S)| \geq ks/2$ with probability at least $1 - 2^{-(1-\delta)(ks/2)\log(k/2)}$.

Proof. Fix any set $S \subseteq F$ of size s, and for each clause $C \in S$ sample the variables in C one at a time without replacement. Let $v_1, v_2, \ldots, v_{ks}$ denote the concatenation of all sequences of sampled variables over all $C \in S$. We say that variable v_i is a repeat if it has already occurred among $v_1, \ldots, v_{i-1}$. In order for $|\mathrm{vars}(S)| < ks/2$ the concatenated sequence must have at least $ks/2$ repeats, and the probability that variable v_i is a repeat is at most $(i-1)/n \leq ks/n$. This implies that

$$\Pr[|\mathrm{vars}(S)| < ks/2] \leq \binom{ks}{ks/2}\left(\frac{ks}{n}\right)^{ks/2} \leq \left(\frac{2eks}{ks}\right)^{ks/2}\left(\frac{ks}{n}\right)^{ks/2} \leq \left(\frac{2}{k}\right)^{ks/2}$$

using standard bounds on binomial coefficients and the fact that $s \leq n/ek^2$. Thus

$$\Pr[\exists S : |S| = s, |\mathrm{vars}(S)| < ks/2] \leq m^s \left(\frac{2}{k}\right)^{ks/2},$$

and by assumption $\log m \leq \delta \cdot \frac{k}{2} \log\left(\frac{k}{2}\right)$, finishing the proof of the lemma. □

Using the expansion lemma we are ready to prove Theorem 5.3.

Proof of Theorem 5.3. We shall apply Theorem 5.2 to $U = \mathcal{U}(\{0,1\}^n)$ and $V = \mathcal{V}(\{0,1\}^n)$ (cf. Section 3) with $r = s = n/ek^2$, $k = 4\log n$, and $m = cn2^{2k}$. Recall that $\mathcal{U}$ and $\mathcal{V}$ are the functions mapping x inputs to 1-inputs of mCSP-SAT$_F$ and mapping Y inputs to 0-inputs of mCSP-SAT$_F$, respectively. To finish the argument we need to compute $|U|, A_1(1, U), A_1(r, U), |V|, A_0(s, V)$.

By definition, in the accepting input $\mathcal{U}(x)$ we set $\mathrm{TT}_i(\alpha) = 1$ if and only if $x \upharpoonright \mathrm{vars}(i) = \alpha$; thus, $\mathcal{U}(x) = \mathcal{U}(x')$ for some $x \neq x'$ only if there exists a variable in X that doesn't appear in any clause. However, it is easy to see that with high probability every variable in X participates in some clause, and thus $\mathcal{U}$ is 1-1 with high probability, and therefore $|U| = 2^n$ with high probability.

Recall that the 0-inputs of mCSP-SAT$_F$ correspond to substituting a Y-assignment into F and writing out truth tables of all the clauses. The truth tables corresponding to the clauses that were satisfied by the Y-assignment are identically 1, and the truth tables corresponding to the clauses that were not satisfied by the given Y-assignment contain exactly one 0-entry, because each clause has a unique falsifying assignment to its variables. Given a Y-assignment we call the set of clauses that were not satisfied by the Y assignment the *profile* of Y. The next lemma implies that the profiles of all Y-assignments are distinct with high probability.

Lemma 5.6. *Let n, m, k be positive integers. Let $F \sim \mathcal{F}(m, n, k)$, let $\mathcal{S} \subseteq \{0,1\}^n$ be a collection of boolean assignments, and define the following $|\mathcal{S}| \times m$ matrix M, with the rows labelled by assignments $\alpha \in \mathcal{S}$ and the columns labelled by clauses of F. Namely, for any pair (α, i) set*

$$M[\alpha, i] = \begin{cases} 1 & \text{if the } i\text{th clause is not satisfied by } \alpha, \\ 0 & \text{otherwise.} \end{cases}$$

If $\log |\mathcal{S}| < km/8n2^k$ *then the rows of* M *are distinct with probability at least* $1 - 2^{km/n2^k}$.

Proof. We think of M as generated column by column with the columns sampled independently. Fix two assignments α and $\widehat{\alpha}$ such that $\alpha \neq \widehat{\alpha}$. Let S be the set of indices on which the two assignments differ, i.e., $S = \{i \mid \alpha_i \neq \widehat{\alpha}_i\}$. Set $s = |S|$. Letting C_i denote the ith clause we have

$$\Pr[C_i \text{ unsat by } \widehat{\alpha} \text{ and satisfied by } \alpha] = \frac{1}{2^k}\left(1 - \frac{\binom{n-s}{k}}{\binom{n}{k}}\right)$$

as $\hat{\alpha}$ must falsify C_i and α must differ from $\hat{\alpha}$ on one of the indices in S. Continuing the calculation,

$$\frac{1}{2^k}\left(1 - \frac{\binom{n-s}{k}}{\binom{n}{k}}\right) \geq \frac{1}{2^k}\frac{\binom{n}{k} - \binom{n-1}{k}}{\binom{n}{k}} = \frac{1}{2^k}\frac{\binom{n-1}{k-1}}{\binom{n}{k}} = \frac{k}{2^k n}.$$

Thus the probability that rows α and $\widehat{\alpha}$ agree on column i is at most $1 - \frac{k}{2^k n}$. Since columns are sampled independently, the probability that α and $\widehat{\alpha}$ agree on all columns is at most

$$\left(1 - \frac{k}{n2^k}\right)^m \leq e^{-km/(n2^k)} \leq 2^{-5km/4n2^k}$$

since $\log e > 5/4$. By a union bound over ordered pairs of assignments in $\mathcal{S}$, the probability that there exists a pair of rows that agree on all columns is at most

$$|\mathcal{S}|^2 2^{-5km/4n2^k} \leq 2^{2\log|\mathcal{S}| - 5km/4n2^k} \leq 2^{-km/n2^k}. \qquad \square$$

In our current setting we have $\mathcal{S} = \{0,1\}^n$ and $km/n2^k \geq n \log n$, thus applying the previous lemma yields that all rows of M are distinct with high probability. Since each profile is distinct with high probability, this implies that $\mathcal{V}$ is 1-1 with high probability, and therefore $|V| = 2^n$. It remains to bound the terms $A_1(1,U), A_1(r,U)$, and $A_0(s,V)$.

Bounding $A_1(1,U)$**.** Fixing a single bit of a 1-input in U to mCSP-SAT$_F$ to 1 is the same as selecting a vertex C in the bipartite constraint graph of F and an assignment α to the variables which participate in C, and then setting $\mathrm{TT}_C(\alpha) = 1$. By the definition of $\mathcal{U}$, for any input $x \in \{0,1\}^n$, fixing this bit to 1 determines exactly k out of the n variables of x. Thus the number of $x \in \{0,1\}^n$ that are consistent with this partial assignment is 2^{n-k}, and since $\mathcal{U}$ is one-to-one, we have $A_1(1,U) = 2^{n-k}$.

Bounding $A_1(r,U)$**.** Similar to the previous bound, but now we fix r of the truth table bits to 1. By definition of $\mathcal{U}$, these bits must be chosen from r distinct truth tables in the 1-input in order to be consistent with any $x \in \{0,1\}^n$. With respect to the underlying CNF F, this corresponds to fixing an assignment to the set of variables appearing in an arbitrary set $\mathcal{S}$ of r clauses in F. By Lemma 5.5, with high probability we have $|\mathrm{vars}(\mathcal{S})| \geq rk/2$. Thus fixing these r bits in the definition of $A_1(r,U)$ corresponds to setting at least $rk/2$ of the input variables that participate in the constraints with determined truth tables. The number of x inputs that are consistent with these indices fixed is therefore $\leq 2^{n-rk/2}$, and so $A_1(r,U) \leq 2^{n-rk/2}$.

Bounding $A_0(s,V)$**.** This case is similar to $A_1(r,U)$. We get $A_0(s,V) \leq 2^{n-sk/2}$.

Observe that $(2s)A_1(1,U) = (2s)2^{n-k} = (2s)2^n/n^4 \leq 2^{n-1}$. Putting this altogether we get the following lower bound on monotone circuit size is at least

$$\frac{2^{n-1}}{(2s)^{s+1}2^{n-sk/2}} = 2^{sk/2-(s+1)\log 2s - 1} \geq 2^{s(k/2 - 2\log s)} \geq 2^{\tilde{\Omega}(n)},$$

where the last inequality follows from $s = n/ek^2$ and $k/4 \geq \log n$. $\square$

5.2 Random CNFs

In this section we show how to modify the argument from the previous section to apply to the usual distribution of random CNFs $\mathcal{F}(m, n, k)$. Using the probabilistic method we find a partition of the variables of a random formula $F \sim \mathcal{F}(m, n, k)$ such that many of the clauses in F are balanced with respect to the partition. Ideally, every clause would be balanced, but it turns out that this is too strong — instead, we show that we can balance many of the clauses, and the remaining imbalanced clauses are always satisfied by a large collection of assignments. First, we introduce our notion of "imbalanced" clauses.

Definition 5.3. *Fix $\epsilon > 0$. Given a partition of n variables into x-variables and y-variables, a k-clause is called X-heavy if it contains more than $(1-\epsilon)k$ x-variables. A k-clause C is called Y-heavy if it contains more than $(1-\epsilon)k$ y-variables. A k-clause is called balanced if it is neither X-heavy nor Y-heavy.*

Before continuing, we recall the standard multiplicative Chernoff bound which will be used throughout this section.

Fact 5.7 (Multiplicative Chernoff Bound (Theorems 4.4 and 4.5 in [35])). *Suppose $Z_1, \dots, Z_n$ are independent random variables taking values in $\{0, 1\}$. Let Z denote their sum and let $\mu = \mathbb{E}(Z)$. Then*

- *For any $\delta \geq 0$,*
$$\Pr[Z \geq (1+\delta)\mu] \leq e^{-\delta\mu/3}$$
- *For any $0 \leq \delta \leq 1$*
$$\Pr[Z \leq (1-\delta)\mu] \leq e^{-\delta^2\mu/3}$$

For balanced random CNFs, Lemma 5.6 allowed us to show that the function $\mathcal{V}$ is 1-1. To handle random CNFs from the usual distribution we must modify this lemma to take into account the fact that the number of X and Y variables in each clause may no longer be equal.

Lemma 5.8. *Let n, m, k be positive integers. Fix n variables $Z = \{z_1, z_2, \dots, z_n\}$ and choose a partition $Z = (X, Y)$ by adding each variable to X with probability $1/2$, and adding it to Y otherwise. Now, sample m random clauses independently as follows. Sample a uniformly random k-clause C over Z, then discard all X literals from C; if all literals in C are discarded, then discard the entire clause. Let F' be the resulting formula. Let m' be the number of clauses in F'. Let $\mathcal{S}$ be a collection of assignments to the Y-variables, and define the following $|\mathcal{S}| \times m'$ matrix M as follows: for any pair (α, i) with $\alpha \in \mathcal{S}$ let*

$$M[\alpha, i] = \begin{cases} 1 & \text{if the } i\text{th clause is not satisfied by } \alpha \\ 0 & \text{otherwise.} \end{cases}$$

If $\log|\mathcal{S}| < m/8n2^{k+3}$ then the rows of M are distinct with probability at least $1 - 2^{-m/n2^{k+3}+1}$.

The proof of this lemma will require the following auxiliary lemma.

Lemma 5.9. *Let C be a k-clause over the Y-variables, sampled as in the statement of Lemma 5.8. Then*

$$\Pr[C \text{ is empty}] = 1/2^k.$$

Proof Sketch. Since the variables in C do not repeat, the following distributions on C are identical:

(1) Sample a random partition $Z = (X, Y)$ and then choose a uniformly random k-clause over (X, Y) and discard the X-variables.

(2) Sample a uniformly random k-clause on the variables Z. Choose a partition $Z = (X, Y)$ by first partitioning the variables occurring in C by including each one in X with probability $1/2$ and in Y otherwise. Partition the remaining variables not occurring in C uniformly at random and discard the X-variables from C.

In the latter interpretation it is easy to see that $|C|$ follows a binomial distribution with k trials and probability $1/2$ of success. In Appendix B we include a formal proof of this lemma that confirms that these distributions are identical. □

Proof of Lemma 5.8. We think of M as being generated column-by-column, with each column sampled independently (as described in the statement of the lemma). Fix two assignments α, α' such that $\alpha \neq \alpha'$. Let S be the set of indices on which α and α' differ, and let $s = |S|$. Let C_i denote the ith clause in F' and let w_i denote the width of C_i, and note that w_i is a random variable. Let $t \leq k$ and $n' \leq n$ be integers. First, observe that conditioned on $w_i = t$, $|Y| = n'$, the clause C_i is a uniformly random clause over Y-variables of width t. Thus, if $t \geq 1$,

$$\Pr[C_i \text{ unsat by } \alpha,\ \text{sat by } \alpha' | w_i = t, |Y| = n'] = \frac{1}{2^t}\left(1 - \frac{\binom{n'-s}{t}}{\binom{n'}{t}}\right)$$

as α' falsifies C_i and α must differ from C_i on one of the indices in S. Continuing the calculation:

$$\frac{1}{2^t}\left(1 - \frac{\binom{n'-s}{t}}{\binom{n'}{t}}\right) \geq \frac{1}{2^t}\left(\frac{\binom{n'}{t} - \binom{n'-1}{t}}{\binom{n'}{t}}\right) = \frac{1}{2^t}\frac{\binom{n'-1}{t-1}}{\binom{n'}{t}} = \frac{t}{2^t n'} \geq \frac{1}{2^k n},$$

where the last step holds because $t \leq k$, $n' \leq n$, and $t \geq 1$ since C_i is non-empty. Let E_t denote the event that C_i has width t for $0 \leq t \leq k$. Then

$$\begin{aligned}\Pr[C_i \text{ unsat by } \alpha,\ \text{sat by } \alpha'] &= \sum_{t=0}^{k} \Pr[C_i \text{ unsat by } \alpha,\ \text{sat by } \alpha' \wedge E_t] \\ &= \sum_{t=0}^{k} \Pr[E_t] \Pr[C_i \text{ unsat by } \alpha,\ \text{sat by } \alpha' | E_t] \\ &= \sum_{t=1}^{k} \Pr[E_t] \Pr[C_i \text{ unsat by } \alpha,\ \text{sat by } \alpha' | E_t] \\ &\geq \frac{1 - \Pr[E_0]}{2^k n} = \frac{1 - 1/2^k}{2^k n} \geq \frac{1}{2^{k+1} n} \qquad (1)\end{aligned}$$

where we have used the fact that the events $\{E_t\}$ partition the probability space, and the last equality follows from Lemma 5.9.

We now turn to bounding the probability that α and α' agree on all columns. By Lemma 5.9, $\mathbb{E}[m'] = (1 - 2^{-k})m$, where m' is the size of F'. Let E be the event that $m' \leq (1 - \delta)\mathbb{E}[m']$, where δ

is a parameter to be chosen later. By a Chernoff bound, $\Pr[E] \leq \exp(-\delta^2 \mathbb{E}[m']/3)$. Then

$$\begin{aligned}\Pr[\alpha \text{ and } \alpha' \text{ agree on all columns}] &\leq \Pr[E] + \Pr[\alpha \text{ and } \alpha' \text{ agree on all columns} | \neg E] \\ &< \exp(-\delta^2 \mathbb{E}(m')/3) + \left(1 - \frac{1}{2^{k+1}n}\right)^{(1-\delta)\mathbb{E}[m']} \qquad \text{(By (1))} \\ &\leq \exp(-\delta^2 \mathbb{E}(m')/3) + \exp\left(-\frac{(1-\delta)\mathbb{E}(m')}{2^{k+1}n}\right) \\ &\leq \exp\left(-\frac{\delta^2(1-2^{-k})m}{3}\right) + \exp\left(-\frac{(1-\delta)(1-2^{-k})m}{2^{k+1}n}\right)\end{aligned}$$

Set $\delta = 1/(2^k - 1)^{1/2}$. Continuing the calculation we get

$$\begin{aligned}\exp\left(-\frac{\delta^2(1-2^{-k})m}{3}\right) + \exp\left(-\frac{(1-\delta)(1-2^{-k})m}{2^{k+1}n}\right) &\leq \exp\left(-\frac{m}{3 \cdot 2^k}\right) + \exp\left(-\frac{(1-\delta)(1-2^{-k})m}{2^{k+1}n}\right) \\ &\leq 2\exp\left(-\frac{(1-\delta)(1-2^{-k})m}{2^{k+1}n}\right) \\ &\leq 2\exp\left(-\frac{m}{2^{k+3}n}\right)\end{aligned}$$

where the second line uses the fact that a sum is at most twice the maximum, and the last line follows since $(1-\delta)(1-2^{-k}) \geq 1/4$ holds for sufficiently large n, by the definition of δ and the assumption that $k = 240 \log n$.

Thus, we can conclude that

$$\Pr[\alpha \text{ and } \alpha' \text{ agree on all columns}] \leq 2\exp\left(-\frac{m}{n2^{k+3}}\right) \leq 2 \cdot 2^{-5m/4n2^{k+3}}$$

where the last step holds since $\log e > 5/4$. By a union bound over all pairs of assignments in $\mathcal{S}$, the probability that there exists a pair of rows that agree on all columns is at most

$$2 \cdot |\mathcal{S}|^2 \cdot 2^{-5m/4n2^{k+3}} \leq 2 \cdot 2^{2\log|\mathcal{S}| - 5m/4n2^{k+3}} \leq 2^{-m/n2^{k+3}+1}. \qquad \square$$

The proof of the next lemma will rely on the following entropy bound on the binomial tail.

Fact 5.10 (Entropy bound on binomial tail (Lemma 6.19 in [19])). *For any $0 < \varepsilon < 1/2$ we have*

$$\frac{2^{H(\varepsilon)n}}{\sqrt{8n\varepsilon(1-\varepsilon)}} \leq \sum_{j=0}^{\lfloor \varepsilon n \rfloor} \binom{n}{j} \leq 2^{H(\varepsilon)n},$$

where $H(\varepsilon) = -\varepsilon \log \varepsilon - (1-\varepsilon)\log(1-\varepsilon)$ is the binary entropy function.

The main lemma below defines a good partition of the variables of a random CNF $F \sim \mathcal{F}(m, n, k)$ and shows that such a good partition exists with high probability. As discussed earlier, the notion of a good partition is supposed to help the rest of the proof in this section mimic the proof for balanced CNFs in the previous section. However, now we have a delicate balance of parameters. In particular, there is tension between Lemma 5.8 which requires m to be large, and the Lovasz Local Lemma (Lemma 5.14) used in Lemma 5.15 which requires m to be small. This is further complicated because we would like to retain all but a constant fraction of assignments in Lemma 5.15. Because of this we need to set our parameters with precision.

Lemma 5.11. *Let $\varepsilon = 1/50$, and let n be a sufficiently large positive integer. Let $k = 240 \log n$, and let $m = n2^{(1+1/16)k}$ $(= n^{256})$. Let $F \sim \mathcal{F}(m, n, k)$ and partition the variables F into two sets (X, Y) by* *adding each variable to X with probability $1/2$, and adding it to Y otherwise. Then with probability $1 - o(1)$ the following holds:*

(1) *The number of variables in X is $n/2 \pm o(n)$.*
(2) *The number of X-heavy clauses and Y-heavy clauses are each upper bounded by $(3/2)n2^{(1/16+H(\varepsilon))k}$.*
(3) *The functions $\mathcal{U}$ and $\mathcal{V}$ are both 1-1 on all $\{0,1\}^X$ and $\{0,1\}^Y$ assignments respectively to F.*
(4) *Each X-variable (Y-variable) occurs in at most $9k2^{(1/16+H(\varepsilon))k}$ X-heavy (Y-heavy) clauses.*

Proof. We will bound the probability that each event occurs and then conclude by a union bound that they hold simultaneously with high probability.

(1) We have $\mathbb{E}[|X|] = n/2$ and since each variable is placed in X independently with probability $1/2$ we have

$$\Pr[|X - n/2| > n^{2/3}] \leq 2\exp(-n^{1/3}/6)$$

by applying the Chernoff bound from Fact 5.7.

(2) For convenience, let $m = n2^{(1+\tau)k}$ where we set $\tau = 1/16$. For each clause C_i in F let T_i be the random variable indicating whether this clause is X-heavy. Using both inequalities in Fact 5.10 we have that

$$\Pr[T_i = 1] = \sum_{j=0}^{\varepsilon k} \binom{k}{j} 2^{-k} \leq 2^{(H(\varepsilon)-1)k}$$

and

$$\Pr[T_i = 1] = \sum_{j=0}^{\varepsilon k} \binom{k}{j} 2^{-k} \geq 2^{-k} \frac{2^{H(\varepsilon)k}}{\sqrt{8k\varepsilon(1-\varepsilon)}} > \frac{2^{(H(\varepsilon)-1)k}}{\sqrt{k}},$$

since $8\varepsilon(1-\varepsilon) < 1$ by our choice of ε. Let $T = \sum_{i=1}^{m} T_i$. Then the above two bounds and linearity of expectation imply that

$$\frac{m2^{(H(\varepsilon)-1)k}}{\sqrt{k}} \leq \mathbb{E}[T] \leq m2^{(H(\varepsilon)-1)k}.$$

Let $m_L := m2^{(H(\varepsilon)-1)k}/\sqrt{k}$ and $m_U := m2^{(H(\varepsilon)-1)k} = n2^{(\tau+H(\varepsilon))k}$. By the Chernoff bound (see Fact 5.7) we have

$$\Pr[T > 3m_U/2] \leq \Pr[T > 3\mathbb{E}[T]/2] \leq \exp(-\mathbb{E}[T]/12) \leq \exp(-m_L/12). \tag{2}$$

Thus, we have that $T < 3m_U/2$ with probability at least $1 - 2\exp(-m_L/12)$, and the same conclusion holds for Y-heavy clauses by symmetry. It follows by a union bound that the partition satisfies both of the properties with high probability.

(3) Recall that $\mathcal{V}$ maps an assignment $y \in \{0,1\}^Y$ to the vector obtained by writing out the truth tables of each of the clauses of F under this assignment y. The truth tables corresponding to clauses that were satisfied by y are identically 1, while the truth tables that were not satisfied by y contain exactly one 0-entry. Let $\mathcal{S} = \{0,1\}^{|Y|}$ and observe that

$$m/8n2^{k+3} = 2^{k/16}/8^2 = n^{15}/8^2 > n \geq \log|\mathcal{S}|.$$

Therefore, Lemma 5.8 implies that the set of clauses not satisfied by each assignment in $\mathcal{S}$ are distinct, and so $\mathcal{V}$ is 1-1, with probability at least $1 - 2^{-m/n2^{k+3}+1} = 1 - 2^{-n^{15}/8+1}$.

By the definition of $\mathcal{U}(x)$, $TT_i(\alpha) = 1$ if and only if $x \restriction \mathrm{vars}(i) = \alpha$. Thus, $\mathcal{U}(x) = \mathcal{U}(x')$ for some $x \neq x'$ only if there exists an X-variable that doesn't appear in any clause. The probability that *any* variable (X or Y) does not occur in any clause is at most

$$n\left(\frac{\binom{n-1}{k}}{\binom{n}{k}}\right)^m = n\left(1-\frac{k}{n}\right)^m \leq ne^{-km/n}.$$

Thus, the probability that every variable in X appears in some clause is at least $1 - ne^{-km/n}$, and so by a union bound the probability that $\mathcal{U}$ and $\mathcal{V}$ are 1-1 is at least $1 - 2^{-n^{15}/8+1} - ne^{-km/n} = 1 - o(1)$.

(4) We will use the same notation as in part **(2)** and we will prove this statement for the X-variables; the Y-variables will follow by symmetry. It is enough to prove this statement for partitions $(X, Y) \in \mathcal{P} := \{(X, Y) \mid ||X| - n/2| \leq n^{2/3}\}$ because, by part **(1)**, the probability of drawing an $(X, Y) \notin \mathcal{P}$ is exponentially small.

Fix a partition $(X, Y) \in \mathcal{P}$. Let x be any fixed variable in X, and let Z_i be the indicator random variable (conditioned on this fixed partition) which is 1 if the variable x occurs in the i^{th} X-heavy clause and 0 otherwise. Let $Z = \sum Z_i$ denote the total number of heavy clauses in which x occurs. We give an upper bound on Z and then conclude the statement via a union bound over the variables in X. One would hope to apply a usual Chernoff + Union bound but in this case Z is a sum of a *random number* of random variables. Fortunately, we can sidestep this issue by conditioning on the number of X-heavy clauses, denoted T.

First we show that the Z_i variables are independent once conditioned on $T = t$. Consider the following method of sampling a random CNF subject to the partition (X, Y).

- Independently for each $i \in [m]$ sample a number $v_i \in \{0, 1, \ldots, k\}$ where $v_i = t$ is chosen with probability
$$\frac{\binom{|X|}{t}\binom{|Y|}{k-t}}{\binom{n}{k}}.$$
- Independently for each $i \in [m]$ sample a random clause by choosing a random set of v_i X-literals and a random set of $(k - v_i)$ Y-literals.

From the definition of the experiment it is clear that the variables Z_j and $Z_{j'}$ with $j \neq j'$ are independent, and will remain independent even after conditioning on any subset of t clauses being heavy.

We need the following two claims. The first claim shows that the bound on the number of X-heavy and Y-heavy clauses from part **(2)** holds even when conditioning on a partition (X, Y). As the proof is similar to the argument in part **(2)** we defer it to Appendix C.

Claim 5.12. *For any fixed $(X, Y) \in \mathcal{P}$, the number of X-heavy and Y-heavy clauses are each upper bounded by $3m_U/2$ and lower bounded by $m_L/2$, except with probability at most $\exp(-\Omega(m_L))$.*

The second claim shows that when $m_L/2 \leq t \leq 3m_U/2$ the probability of Z being large is very small.

Claim 5.13. *For any t such that $m_L/2 \leq t \leq 3m_U/2$,*

$$\Pr[Z > 9km_U/n \mid (X, Y), T = t] \leq \exp(-km_U/n).$$

Proof. Let

$$\mu_t = \mathbb{E}[Z \mid T = t, (X, Y)], \quad \delta_t = \frac{9|X|m_U}{tn} - 1.$$

Observe that

$$(1 - \varepsilon)k/|X| \leq \Pr[Z_i = 1 \mid (X, Y), T = t] \leq k/|X|$$

because the i^{th} heavy clause is generated by picking at most k and at least $(1-\varepsilon)k$ variables from X. It follows that $t(1-\varepsilon)k/|X| \le \mu_t \le tk/|X|$. By a Chernoff bound we have

$$\begin{aligned}\Pr[Z > 9km_U/n \,|\, (X,Y), T=t] &\le \Pr[Z > (1+\delta_t)\mu_t \,|\, (X,Y), T=t] \\ &\le \exp(-\delta_t\mu_t/3) \\ &\le \exp(-3(1-\varepsilon)km_U/n + \mu_t/3) \\ &\le \exp(-3(1-\varepsilon)km_U/n + tk/3|X|) \\ &\le \exp(-3(1-\varepsilon)km_U/n + m_U k/2|X|).\end{aligned}$$

Note that $|X| \ge n/2 - n^{2/3} \ge n/3$ for sufficiently large n. Continuing the calculation

$$\begin{aligned}\exp(-3(1-\varepsilon)km_U/n + m_U k/2|X|) &\le \exp(-3(1-\varepsilon)km_U/n + 3m_U k/2n) \\ &\le \exp(-km_U/n)\end{aligned}$$

where the last step follows since $\varepsilon = 1/50$. □

Now, using the above independence together with Claim 5.12 and Claim 5.13 we can complete the proof. First we bound the probability that there are many X-heavy clauses for a *fixed* partition $(X,Y) \in \mathcal{P}$. Let $\mathcal{R} = \{m_L/2+1, m_L/2+2, \ldots, 3m_U/2-1\}$, then

$$\begin{aligned}\Pr[Z > 9km_U/n \,|\, (X,Y)] &\le \Pr[T \notin \mathcal{R} \,|\, (X,Y)] + |\mathcal{R}| \max_{t\in\mathcal{R}} \Pr[Z > 9km_U/n \,|\, T=t, (X,Y)] \\ &\le \exp(-\Omega(m_L)) + m\exp(-km_U/n) \le \exp(-\Omega(m_L))\end{aligned}$$

where the last step holds for sufficiently large n since $k = O(\log n)$ and $m_L = m_U/\sqrt{k}$. Thus, we can take a union bound over all $x \in X$ and conclude that, for a fixed $(X,Y) \in \mathcal{P}$, the probability that there exists some X-variable which occurs in more than $9km_U/n$ clauses is at most $n\exp(-\Omega(m_L))$.

We can now complete the proof of this part. Let B be the event that there is an X variable occuring in more than $9km_U/n$ X-heavy clauses. Then

$$\Pr[B] \le \Pr[(X,Y) \notin \mathcal{P}] + 2^n \max_{(X,Y)\in\mathcal{P}} \Pr[B \,|\, (X,Y)] \le 2\exp(-n^{1/3}/6) + n\exp(n\ln 2 - \Omega(m_L)) = o(1)$$

where we have used the bound from part **(1)** and the fact that $m_L = \mathrm{poly}(n)$. By symmetry the same bound holds for the Y-heavy clauses, so, taking a union bound finishes the proof of this part.

Finally, a union bound over parts **(1)** – **(4)** finishes the lemma. □

Conditioning on a partition (X,Y) satisfying the main lemma, it remains to show that there exists a large collection of assignments satisfying all heavy clauses. The main technical tool in the proof is the Lovász Local Lemma.

Lemma 5.14 (Lovász Local Lemma (Theorem 5.1.1 in [3])). *Let $\mathcal{E} = \{E_1, \ldots, E_n\}$ be a finite set of events in the probability space Ω. For $E \in \mathcal{E}$ let $\Gamma(E)$ denote the set of events E_i on which E depends. If there is $q \in [0,1)$ such that $\forall E \in \mathcal{E}$ we have $\Pr[E] \le q(1-q)^{|\Gamma(E)|}$, then the probability that none of the events E_i occur is at least $\Pr[\overline{E_1} \wedge \overline{E_2} \wedge \cdots \wedge \overline{E_n}] \ge (1-q)^n$.*

The following lemma shows that for any partition (X,Y) satisfying the conditions of the main lemma, there is a large collection of assignments satisfying all heavy clauses.

Lemma 5.15. *Let $F \sim \mathcal{F}(m,n,k)$ and let (X,Y) be a partition satisfying properties (1)-(4) of Lemma 5.11. There exists a set $\mathcal{A}$ of $2^{|X|}/e^3$ truth assignments to the X-variables that satisfy all X-heavy clauses, and a set $\mathcal{B}$ of $2^{|Y|}/e^3$ truth assignments to the Y-variables satisfying all of the Y-heavy clauses.*

Proof. Consider selecting a random assignment to the X-variables. Let E_i be the event that the ith X-heavy clause is not satisfied by the random assignment, and observe that $\Pr[E_i] \le 2^{-(1-\varepsilon)k}$ since the clause is X-heavy.

We continue using the notation introduced in the proof of Lemma 5.11, namely, $\varepsilon = 1/50$, $\tau = 1/16$, $m_U = n2^{(\tau+H(\varepsilon))k}$. By property (2) of Lemma 5.11, the number of events E_i is at most $(3/2)m_U$. By property (4), we have that for any event E_i the number of events that share any X-variable with E_i is $|\Gamma(E_i)| \le (9km_U/n)\cdot k = 9k^2 m_U/n$.

Set $q = 2^{-\delta k}$ for $\delta = 1/15 + H(\varepsilon)$. Then for each E_i we have

$$\begin{aligned} q(1-q)^{|\Gamma(E_i)|} &\ge q\exp(-2q|\Gamma(E_i)|) = q\exp(-2\cdot 2^{-\delta k}(9k^2/n)n2^{(\tau+H(\varepsilon))k}) \\ &= q\exp(-(18k^2)2^{-k/240}) \ge q/e \ge 2^{-(1-\varepsilon)k}, \end{aligned}$$

where we used the fact that $e^{-2x} \le 1-x$ when $x \in [0,1/2]$ and that $-(18k^2)2^{-k/240} \ge -\mathrm{poly}(\log n)/n \ge -1$ for sufficiently large n.

We have set q such that only a constant fraction of assignments will not satisfy all X-heavy clauses. To see this, observe that for our settings of τ, δ, and k,

$$qm_U = 2^{-\delta k}n2^{(\tau+H(\varepsilon))k} = n2^{-(\delta-(H(\varepsilon)+\tau)k} = n2^{-(1/15-1/16)240\log n} = 1.$$

Applying the Lovász Local Lemma (Lemma 5.14) we get that the probability that an assignment satisfies all X-heavy clauses is at least

$$(1-q)^{3m_U/2} \ge e^{-3qm_U} = e^{-3}.$$

Thus the number of assignments to the X-variables satisfying all heavy clauses is at least $2^{|X|}/e^3$, and an identical calculation applies to the Y-variables by symmetry. □

With this lemma in place, we can proceed more or less as in the last section. Now we perform the whole argument with respect to $U = \mathcal{U}(\mathcal{A})$ and $V = \mathcal{V}(\mathcal{B})$, with $\mathcal{A}$ and $\mathcal{B}$ chosen as in the previous lemma. This allows us to restrict our attention only to the balanced clauses, and the calculations from the previous section work *mutatis mutandis* since many clauses are balanced.

Theorem 5.16. *There exists a constant $c > 0$ such that the following holds. Let $n \ge c$ be any positive integer. Let $F \sim \mathcal{F}(m,n,k)$ for $m = n2^{(1+1/16)k}$ and $k = 240\log n$. With high probability there exists a partition (X,Y) of the variables of F and a $\delta > 0$ such that any monotone real circuit computing* mCSP-SAT$_F$ *requires at least $2^{\tilde{\Omega}(n)}$ gates.*

Proof. Apply Lemma 5.11 to get a partition of the variables (X,Y), and let $\mathcal{A}$, $\mathcal{B}$ denote the set of assignments to the X and Y-variables, respectively, given by Lemma 5.15. If z is an input to mCSP-SAT$_F$, let z' be z restricted to truth tables corresponding to balanced clauses of F with respect to the partition (X,Y); it follows from the lemma that with high probability there are at least $m - 3m2^{-k/2} \ge m/2$ balanced clauses for n sufficiently large. Let $U = \{z' \mid z \in \mathcal{U}(\mathcal{A})\}$ and $V = \{z' \mid z \in \mathcal{V}(\mathcal{B})\}$. Letting $F' \subseteq F$ be the formula containing only balanced clauses of F, then we can think of z' as input to mCSP-SAT$_{F'}$.

Our aim will be to apply Theorem 5.2 to U and V, similar to what we did in the previous section. However in order to do this we will have to show that the existence of a small monotone circuit separating $\mathcal{U}(X)$ and $\mathcal{V}(Y)$ implies the existence of a small circuit that separates the truncated assignments U and V. The strategy of the proof is as follows: given a monotone real circuit C separating $\mathcal{U}(X)$ and $\mathcal{V}(Y)$ (and therefore $\mathcal{U}(\mathcal{A})$ and $\mathcal{V}(\mathcal{B})$) we aim to apply a restriction ρ to C that fixes all of the input gates corresponding to the X-heavy and Y-heavy clauses in such a way that the resulting circuit C_ρ separates U and V. Because F' is balanced, we can then perform the same argument for C_ρ with respect to $\mathcal{U}(\mathcal{A})$ and $\mathcal{V}(\mathcal{B})$ as we did for balanced random CNFs in the previous section. A lower bound on the size of C_ρ then implies a lower bound on the size of the unrestricted circuit C.

We define the restriction ρ setting inputs (i.e. truth tables) corresponding to unbalanced clauses as follows:

- Truth table entries corresponding to an X-heavy clause are all set to 1 except for the entry corresponding to the assignment that does not satisfy the clause.
- Truth table entries corresponding to a Y-heavy clause are all set to 1.

Claim 5.17. *The circuit C_ρ obtained from applying the restriction ρ to C separates U and V.*

Proof of Claim. Let $x \in \mathcal{A}$, and let $z = \mathcal{U}(x)$, then there is a corresponding $z' \in U$. Let $z' \circ \rho$ denote the extension of z' by ρ to an input to mCSP-SAT$_F$. Thus, C_ρ evaluated on z' is the same as the original circuit C evaluated on $z' \circ \rho$. We claim that $z' \circ \rho \geq z$, i.e., $z' \circ \rho$ is z with some entries set to 1. To see this, observe that the truth table corresponding to every balanced clause is given the same assignment by z and $z' \circ \rho$. Clearly, for any Y-heavy clause C_i, the assignment given to TT_i by $z \circ \rho$ is at least the assignment given by z. Now, let C_i be an X-heavy clause, and recall that according to Definition 3.3, z is defined by setting $TT_i(\alpha) = 1$ if and only if $x \restriction_{\text{vars}(i)} = \alpha$. Let α' be the unique assignment to vars(i) (the variables of C_i) that does not satisfy C_i. Because every assignment in $\mathcal{A}$ satisfies every X-heavy clause, it cannot be that $x \restriction_{\text{vars}(i)} = \alpha'$, and so $TT_i(\alpha') = 0$ in both z and $z' \circ \rho$. Therefore, $z' \circ \rho \geq z$. The original circuit C output 1 on z and therefore, by monotonicity, it also outputs 1 on $z' \circ \rho$. This, in turn, means that C_ρ outputs 1 on z'.

Now let $y \in \mathcal{B}$, let $z = \mathcal{V}(y)$, and consider $z' \circ \rho$. We claim that $z' \circ \rho \leq z$, i.e., $z' \circ \rho$ is z with some entries set to 0. Both z and $z' \circ \rho$ assign the same values to balanced clauses. Because every assignment in $\mathcal{B}$ satisfies every Y-heavy clause, the truth tables corresponding to Y-heavy clauses are identically 1 in both z and $z' \circ \rho$ by the definition of mCSP-SAT$_F$. The truth tables corresponding to X-heavy clauses C_i are either the same in z as in $z' \circ \rho$ (if there exists $\alpha \in \{0,1\}^{|X|}$ such that $C_i(x, y) = 0$) or are identically 1 in z and containing a single 0-entry in ρ (if there is no such α). The original circuit C outputs 0 on z therefore, by monotonicity, it also outputs 0 on $z' \circ \rho$. This completes the proof of the claim.

The rest of the proof mirrors the proof of Theorem 5.3 with small changes. We will apply Theorem 5.2 to U and V, and count with respect to the balanced clauses. Because our partition (X, Y) satisfies Lemma 5.11, $\mathcal{U}$ and $\mathcal{V}$ are 1-1 on $\{0,1\}^X$ and $\{0,1\}^Y$ respectively, and are therefore 1-1 on $\mathcal{A}$ and $\mathcal{B}$. This implies that $|U| = |\mathcal{A}| = 2^{|X|-3\log(e)}$ and $|V| = |\mathcal{B}| = 2^{|Y|-3\log(e)}$. We now turn to bounding $A_1(r, U)$, $A_1(1, U)$ and $A_0(s, V)$. For this will use the following immediate corollary of Lemma 5.5.

Lemma 5.18. *Let n be any sufficiently large integer, and k_0, m be positive integers. Let F be a CNF formula on m clauses, where each clause is sampled from $\mathcal{F}(1, n, k')$ for $k' \geq k_0$. Let $s \leq n/ek_0^2$ be a positive integer. If*

$$\log m \leq \delta \cdot \frac{k_0}{2} \log\left(\frac{k_0}{2}\right)$$

for some $0 < \delta < 1$, then every set $S \subseteq F$ of size s satisfies $|\text{vars}(S)| \geq k_0 s/2$ with probability at least $1 - 2^{-(1-\delta)(k_0 s/2)\log(k_0 s/2)}$.

This lemma follows immediately from the proof of Lemma 5.5 with $k_0 = k$ by noting that if each clause contains greater than k variables, then this can only increase the size of vars(S).

Bounding $A_1(r, U)$. Fixing a single bit of an input in U to 1 is the same as selecting a balanced clause C in the constraint graph of F and an assignment α to the variables and setting $TT_C(\alpha) = 1$.

Fixing this bit to 1 determines all variables from X that participate in this clause. By definition, each balanced clause contains at least $k_0 = k/50$ variables from X. Now, to fix r truth table bits to 1, by the definition of $\mathcal{U}$, these bits must be chosen from r distinct truth tables in order to be consistent with any $x \in \{0,1\}^n$. Let $\mathcal{S}$ be an arbitrary set of r balanced clauses from F; we will apply Lemma 5.18. There are at least $m/2$ balanced clauses, and so

$$\log(m/2) = \log\left(n2^{(1+1/16)k-1}\right) = 256\log n - 1 \leq \gamma \cdot \frac{k_0}{2}\log\frac{k_0}{2}$$

for sufficiently large n and some universal constant $\gamma > 0$. We set $r = n/2ek_0^2$; by Lemma 5.18 this implies that each collection $\mathcal{S}$ of r balanced clauses satisfies $|\mathrm{vars}_X(\mathcal{S})| \geq k_0 r/2$ with high probability. Note that we can apply the argument from Lemma 5.18 because conditioned on containing some fixed number $k' \geq k/20 = k_0$ of X-variables, the X-part of a clause is distributed exactly according to $\mathcal{F}(1, |X|, k')$. Thus, fixing these r bits in the definition of $A_1(r, U)$ corresponds to setting at least $k_0 r/2$ of the input variables that participate in the constraints with determined truth tables. The number of x-inputs that are consistent with these indices fixed is at most $2^{|X|-rk_0/2}$, and so $A_1(r, U) \leq 2^{|X|-rk_0/2}$. Using the same argument, we have $A_1(1, U) \leq 2^{|X|-k_0}$.

Bounding $A_0(s, V)$**.** This case is similar to $A_1(r, V)$ and we get $A_0(s, V) \leq 2^{|Y|-sk_0/2}$.

To put everything together, we just follow the calculation at the end of the proof of Theorem 5.3 using our new estimates. Note that our choice of $r = s = n/2ek_0^2$ implies that $2\log(2r) \leq 2\log n \leq k_0/2$ since $k_0 = k/50 > 4\log n$. Applying this,

$$(2s)A_1(1, U) \leq 2^{\log(2r)+|X|-k_0} \leq 2^{|X|-(3/4)k_0}.$$

This yields the following lower bound on the monotone real circuit size of mCSP-SAT$_F$:

$$\begin{aligned}\frac{|U| - (2s)A_1(1, U))}{(2s)^{r+1}A_1(r, U)} &\geq \frac{2^{|X|-3\log(e)-1}}{(2r)^{r+1}2^{|X|-rk_0/2}} \\ &\geq 2^{r(k_0/2-\log(2r))-\log(2r)-3\log(e)-1} \\ &\geq 2^{rk_0/4-\log(2r)-3\log(e)-1} \\ &\geq 2^{rk_0/4-\log(n)-3\log(e)-1} \geq 2^{\tilde{\Omega}(n)}.\end{aligned}$$

□

Corollary 5.19 (Theorem 1.1)**.** *Let F be distributed as above. There exists $\varepsilon > 0$ such that with high probability any* RCC$_1$*-refutation requires $2^{\tilde{\Omega}(n)}$ lines.*

6 CONCLUSION

The obvious problem left open by this paper is to prove lower bounds on other conjectured hard instances for Cutting Planes: perhaps most important is improving the lower bounds for random k-SAT when $k = \Theta(1)$. It seems likely that such lower bounds should hold for some (possibly large) constant k even for CC-proofs, however, as we discussed in the introduction it seems that the symmetric method of approximations is incapable of obtaining strong lower bounds for constant k.

ACKNOWLEDGMENTS

This material is based upon work supported by the National Science Foundation under Grant No. DMS-1638352 and by the Natural Sciences and Engineering Research Council

REFERENCES

[1] Michael Alekhnovich. 2005. Lower bounds for k-DNF resolution on random 3-CNFs. In *Proc. of the 37th STOC*. 251–256. https://doi.org/10.1145/1060590.1060628

[2] Michael Alekhnovich and Alexander A. Razborov. 2001. Lower Bounds for Polynomial Calculus: Non-Binomial Case. In *42nd Annual Symposium on Foundations of Computer Science, FOCS 2001, 14-17 October 2001, Las Vegas, Nevada, USA*. 190–199. https://doi.org/10.1109/SFCS.2001.959893

[3] Noga Alon and Joel Spencer. 1992. *The Probabilistic Method*. John Wiley.

[4] Paul Beame, Richard M. Karp, Toniann Pitassi, and Michael E. Saks. 1998. On the Complexity of Unsatisfiability Proofs for Random k-CNF Formulas. In *Proc. of the 13th STOC*. 561–571. https://doi.org/10.1145/276698.276870

[5] Paul Beame and Toniann Pitassi. 1998. Propositional Proof Complexity: Past, Present and Future. *Bulletin of the EATCS* 65 (1998), 66–89.

[6] Eli Ben-Sasson and Russell Impagliazzo. 2010. Random CNFs are Hard for the Polynomial Calculus. *Computational Complexity* 19, 4 (2010), 501–519. https://doi.org/10.1007/s00037-010-0293-1

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

[8] Christer Berg and Staffan Ulfberg. 1999. Symmetric Approximation Arguments for Monotone Lower Bounds Without Sunflowers. *Computational Complexity* 8, 1 (1999), 1–20. https://doi.org/10.1007/s000370050017

[9] Maria Luisa Bonet, Toniann Pitassi, and Ran Raz. 1997. Lower Bounds for Cutting Planes Proofs with Small Coefficients. *J. Symb. Log.* 62, 3 (1997), 708–728. https://doi.org/10.2307/2275569

[10] Vasek Chvátal. 1973. Edmonds polytopes and a hierarchy of combinatorial problems. *Discrete Mathematics* 4, 4 (1973), 305–337. https://doi.org/10.1016/0012-365X(73)90167-2

[11] Vasek Chvátal and Bruce A. Reed. 1992. Mick Gets Some (the Odds Are on His Side). In *33rd Annual Symposium on Foundations of Computer Science, Pittsburgh, Pennsylvania, USA, 24-27 October 1992*. 620–627. https://doi.org/10.1109/SFCS.1992.267789

[12] Vasek Chvátal and Endre Szemerédi. 1988. Many Hard Examples for Resolution. *J. ACM* 35, 4 (1988), 759–768. https://doi.org/10.1145/48014.48016

[13] William Cook, Collette Coullard, and György Turán. 1987. On the complexity of cutting-plane proofs. *Discrete Applied Mathematics* 18, 1 (1987), 25 – 38. https://doi.org/10.1016/0166-218X(87)90039-4

[14] Wenceslas Fernandez de la Vega. 1992. On random 2-SAT. (1992). Unpublished Manuscript.

[15] Susanna F. de Rezende, Jakob Nordström, and Marc Vinyals. 2016. How Limited Interaction Hinders Real Communication (and What It Means for Proof and Circuit Complexity). In *IEEE 57th Annual Symposium on Foundations of Computer Science, FOCS 2016, 9-11 October 2016, Hyatt Regency, New Brunswick, New Jersey, USA*. 295–304. https://doi.org/10.1109/FOCS.2016.40

[16] Jian Ding, Allan Sly, and Nike Sun. 2015. Proof of the Satisfiability Conjecture for Large k. In *Proc. of the 47th STOC*. 59–68. https://doi.org/10.1145/2746539.2746619

[17] Uriel Feige. 2002. Relations between average case complexity and approximation complexity. In *Proc. of the 34th STOC*. 534–543. https://doi.org/10.1145/509907.509985

[18] Yuval Filmus, Pavel Hrubeš, and Massimo Lauria. 2016. Semantic Versus Syntactic Cutting Planes. In *33rd Symposium on Theoretical Aspects of Computer Science, STACS 2016, February 17-20, 2016, Orléans, France*. 35:1–35:13. https://doi.org/10.4230/LIPIcs.STACS.2016.35

[19] Jörg Flum and Martin Grohe. 2006. *Parameterized Complexity Theory*. Springer. https://doi.org/10.1007/3-540-29953-X

[20] Andreas Goerdt. 1996. A Threshold for Unsatisfiability. *J. Comput. Syst. Sci.* 53, 3 (1996), 469–486. https://doi.org/10.1006/jcss.1996.0081

[21] Ralph E. Gomory. 1958. Outline of an algorithm for integer solutions to linear programs. *Bull. Amer. Math. Soc.* 64, 5 (09 1958), 275–278. https://projecteuclid.org:443/euclid.bams/1183522679

[22] Mika Göös and Toniann Pitassi. 2014. Communication lower bounds via critical block sensitivity. In *Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 - June 03, 2014*. 847–856. https://doi.org/10.1145/2591796.2591838

[23] Mika Göös, Toniann Pitassi, and Thomas Watson. 2015. Deterministic Communication vs. Partition Number. In *IEEE 56th Annual Symposium on Foundations of Computer Science, FOCS 2015, Berkeley, CA, USA, 17-20 October, 2015*, Venkatesan Guruswami (Ed.). IEEE Computer Society, 1077–1088.

[24] Armin Haken and Stephen A. Cook. 1999. An Exponential Lower Bound for the Size of Monotone Real Circuits. *J. Comput. Syst. Sci.* 58, 2 (1999), 326–335. https://doi.org/10.1006/jcss.1998.1617

[25] Armin Haken and Stephen A. Cook. 1999. An Exponential Lower Bound for the Size of Monotone Real Circuits. *J. Comput. Syst. Sci.* 58, 2 (1999), 326–335. https://doi.org/10.1006/jcss.1998.1617

[26] Pavel Hrubeš and Pavel Pudlák. 2017. Random formulas, monotone circuits, and interpolation. *Electronic Colloquium on Computational Complexity (ECCC)* 24 (2017), 42. https://eccc.weizmann.ac.il/report/2017/042

[27] Pavel Hrubeš and Pavel Pudlák. 2018. A note on monotone real circuits. *Inf. Process. Lett.* 131 (2018), 15–19. https://doi.org/10.1016/j.ipl.2017.11.002

[28] Stasys Jukna. 2012. *Boolean Function Complexity - Advances and Frontiers.* Algorithms and combinatorics, Vol. 27. Springer. https://doi.org/10.1007/978-3-642-24508-4

[29] Mauricio Karchmer and Avi Wigderson. 1990. Monotone Circuits for Connectivity Require Super-Logarithmic Depth. *SIAM J. Discrete Math.* 3, 2 (1990), 255–265. https://doi.org/10.1137/0403021

[30] Scott Kirkpatrick and Bart Selman. 1994. Critical Behavior in the Satisfiability of Random Boolean Expressions. *Science* 264, 5163 (1994), 1297–1301. https://doi.org/10.1126/science.264.5163.1297 arXiv:http://science.sciencemag.org/content/264/5163/1297.full.pdf

[31] Jan Krajíček. 1997. Interpolation Theorems, Lower Bounds for Proof Systems, and Independence Results for Bounded Arithmetic. *J. Symb. Log.* 62, 2 (1997), 457–486. https://doi.org/10.2307/2275541

[32] Jan Krajíček. 1998. Interpolation by a Game. *Math. Log. Q.* 44 (1998), 450–458. https://doi.org/10.1002/malq.19980440403

[33] Eyal Kushilevitz and Noam Nisan. 1997. *Communication complexity.* Cambridge University Press.

[34] Massimo Lauria and Neil Thapen. 2018. On semantic cutting planes with very small coefficients. *Inf. Process. Lett.* 136 (2018), 70–75. https://doi.org/10.1016/j.ipl.2018.04.007

[35] Michael Mitzenmacher and Eli Upfal. 2005. *Probability and computing - randomized algorithms and probabilistic analysis.* Cambridge University Press.

[36] Igor C. Oliveira. 2015. *Unconditional lower bounds in complexity theory.* Ph.D. Dissertation. Columbia University.

[37] Pavel Pudlák. 1997. Lower Bounds for Resolution and Cutting Plane Proofs and Monotone Computations. *J. Symb. Log.* 62, 3 (1997), 981–998. https://doi.org/10.2307/2275583

[38] Ran Raz and Pierre McKenzie. 1999. Separation of the Monotone NC Hierarchy. *Combinatorica* 19, 3 (1999), 403–435. https://doi.org/10.1007/s004930050062

[39] Alexander Razborov. 1985. Lower Bounds for the Monotone Complexity of some Boolean Functions. *Sov. Math. Dokl.* 31 (1985), 354–357.

[40] Alexander Razborov. 1995. Unprovability of Lower Bounds on Circuit Size in Certain Fragments of Bounded Arithmetic. *Izvestiya Mathematics* 59, 1 (1995), 205–227.

[41] Grant Schoenebeck. 2008. Linear Level Lasserre Lower Bounds for Certain k-CSPs. In *49th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2008, October 25-28, 2008, Philadelphia, PA, USA.* 593–602. https://doi.org/10.1109/FOCS.2008.74

[42] Bart Selman, David G. Mitchell, and Hector J. Levesque. 1996. Generating Hard Satisfiability Problems. *Artif. Intell.* 81, 1-2 (1996), 17–29. https://doi.org/10.1016/0004-3702(95)00045-3

[43] Dmitry Sokolov. 2016. Dag-like Communication and Its Applications. *Electronic Colloquium on Computational Complexity (ECCC)* 23 (2016), 202. http://eccc.hpi-web.de/report/2016/202

[44] Salil P. Vadhan. 2012. Pseudorandomness. *Foundations and Trends in Theoretical Computer Science* 7, 1-3 (2012), 1–336. https://doi.org/10.1561/0400000010

Appendix A

In this appendix we prove Theorem 4.3, which is split into two lemmas. Lemma A.2 is the difficult direction, translating RCC_1 refutations of F into monotone real circuits for mCSP-SAT. Lemma A.3 shows a converse, and is a simple direct argument analogous to Theorem 4.2. As mentioned in the Introduction, Lemma A.2 follows from the proof of the following result of Hrubeš and Pudlák [26] relating real monotone circuits and certain "dag-like" real communication protocols.

Theorem A.1 (Theorem 5 in [27]). *Let f be a monotone Boolean function. Given a dag-like real protocol P solving the monotone Karchmer-Wigderson game*[4] *for f, there is a monotone real circuit of the same size computing f.*

The formal definition of dag-like real protocols will not be necessary for our technical results, and so we refer the interested reader to [43] for their definition. Our Lemma A.2 states that from an RCC_1 refutation of an unsatisfiable formula F, we can construct a similarly-sized monotone real circuit for the function mCSP-SAT$_F$.

[4] For a monotone function $f : \{0,1\}^n \to \{0,1\}$ the monotone Karchmer-Wigderson (KW) game asks for Alice and Bob, given $x \in f^{-1}(1)$ and $y \in f^{-1}(0)$ respectively, to agree on an index $i \in [n]$ such that $x_i > y_i$.

Lemma A.2. *Let F be an unsatisfiable CNF formula on n variables and let (X, Y) be any partition of the variables. If there is a RCC_1 refutation of F with respect to the partition (X, Y) of length ℓ, then there is a real monotone circuit separating the accepting and rejecting instances $\mathcal{U}(\{0,1\}^{|X|}), \mathcal{V}(\{0,1\}^{|Y|})$ of $\mathrm{mCSP\text{-}SAT}_F$ with ℓ gates.*

We will give a direct proof of Lemma A.2 which is modelled on the proof of Theorem A.1, but first let us sketch how Lemma A.2 can be obtained using Theorem A.1 as a black box. Let F be an unsatisfiable formula on n variables, let (X, Y) be any partition of these variables, and suppose that F has an RCC_1 refutation P. The *search problem* associated with F and variable partition (X, Y) is the following two-party communication problem: Alice receives an assignment to the variables in X, and Bob receives an assignment to the variables in Y, and they want to find and output a clause of F that is falsified by their joint assignment. From an RCC_1 refutation of F, one can extract a dag-like real protocol for solving this search problem; the proof follows standard ideas in the literature transforming communication lower bounds into proof length lower bounds (e.g. time-space tradeoffs for cutting planes [15]). By combining the reductions appearing in [22, 38], the search problem associated with F is *equivalent* to the monotone Karchmer-Wigderson game associated with $\mathrm{mCSP\text{-}SAT}_F$. Thus, by the above theorem, $\mathrm{mCSP\text{-}SAT}_F$ also has a monotone real circuit of the same size as P.

To prove the other direction of Theorem 4.3 (Lemma A.3), we need to translate monotone real circuits computing $\mathrm{mCSP\text{-}SAT}_F$ into RCC_1 refutations for F. This follows by viewing the monotone real circuit as a dag-like real protocol for solving the monotone KW game associated with $\mathrm{mCSP\text{-}SAT}_F$, along with the equivalence between such protocols and dag-like real protocols solving the search problem associated with F; the latter is exactly an RCC_1 refutation of F.

For completeness, we will now give self-contained proofs of Lemmas A.2 and A.3. The proofs are an adaptation of the argument in [27] to our setting, bypassing the intermediate communication protocols associated with F.

Proof of Lemma A.2. Fix an RCC_1-refutation of F. With each node v of the underlying directed acyclic graph (dag) associate two functions $A_v : \{0,1\}^{|X|} \to \mathbb{R}$ and $B_v : \{0,1\}^{|Y|} \to \mathbb{R}$ that Alice and Bob use to communicate with the referee. We assume without loss of generality that the referee outputs 0 if and only if $A_v(x) > B_v(y)$, and furthermore, that $B_v \geq 0$. Recall that each leaf in this dag is associated with a clause C_i and let α_i be the assignment to the X-variables that does not satisfy the X-part of C_i. Note: we may assume that if v is a leaf then

$$A_v(x) = \mathrm{TT}_i^{\mathcal{U}(x)}(\alpha_i) \text{ and } B_v(y) = \mathrm{TT}_i^{\mathcal{V}(y)}(\alpha_i). \tag{3}$$

Next, we convert the given dag to the real circuit separating $\mathcal{U}(\{0,1\}^{|X|})$ from $\mathcal{V}(\{0,1\}^{|Y|})$ as follows. The topology of the derived circuit is exactly the same as that of the dag. Thus, to finish specifying the circuit we need to label inputs to the circuit and label the internal nodes by monotone real gates. Each leaf labeled by clause C_i in the dag turns into an input variable to the circuit labeled by $\mathrm{TT}_i(\alpha_i)$. With each internal node v of the dag with children u_1 and u_2 we associate the function f_v defined recursively as follows:

$$f_v(z) = \max_{x \in \{0,1\}^{|X|}} \{A_v(x) \mid f_{u_1}(z) \geq A_{u_1}(x) \wedge f_{u_2}(z) \geq A_{u_2}(x)\}.$$

We define $f_v(z)$ to be 0 if the set on the right-hand side is empty. We claim that these functions can be computed by monotone real gates and for every $x \in \{0,1\}^{|X|}$ and every $y \in \{0,1\}^{|Y|}$ we have

$$f_v(\mathcal{U}(x)) \geq A_v(x) \text{ and } f_v(\mathcal{V}(y)) \leq B_v(y). \tag{4}$$

First, let's see how the above properties of f_v imply that the constructed circuit separates $\mathcal{U}(\{0,1\}^{|X|})$ from $\mathcal{V}(\{0,1\}^{|Y|})$. Let r be the root node of the dag. Since we started with a valid RCC_1 refutation of

F we have $A_r(x) > B_r(y)$ for all x and y. Therefore, $f_r(\mathcal{U}(x)) > f_r(\mathcal{V}(y))$ for all x and y. Modifying f_r by composing it with an appropriately chosen threshold function gives us the separating circuit.

It is easy to see that f_v can be computed by a monotone real gate with inputs f_{u_1} and f_{u_2}. First of all, the value of f_v is determined by values of f_{u_1} and f_{u_2}, and secondly, increasing values of f_{u_1} and/or f_{u_2} increases the feasible region of xs over which the maximum is taken in the definition of f_v.

Thus, it is left to show that $f_v(z)$ satisfies (4). We shall prove this by induction. The base case is given by (3). Inductive assumption (IA): suppose that we proved (4) for children u_1, u_2 of v. Consider an arbitrary $x \in \{0,1\}^{|X|}$. By IA, we have $f_{u_1}(\mathcal{U}(x)) \geq A_{u_1}(x)$ and $f_{u_2}(\mathcal{U}(x)) \geq A_{u_2}(x)$. Thus, the region over which the max is taken in the definition of $f_v(\mathcal{U}(x))$ is nonempty and contains x. It follows that $f_v(\mathcal{U}(x)) \geq A_v(x)$. Now, consider an arbitrary $y \in \{0,1\}^{|Y|}$. Assume for contradiction that $f_v(\mathcal{V}(y)) > B_v(y)$. Since $B_v(y) \geq 0$, we have $f_v(\mathcal{V}(y)) = A_v(x)$ for some $x \in \{0,1\}^{|X|}$. Thus we have $A_v(x) > B_v(y)$, and by soundness of the refutation it follows that either $A_{u_1}(x) > B_{u_1}(y)$ or $A_{u_2}(x) > B_{u_2}(y)$. Assume without loss of generality that $A_{u_1}(x) > B_{u_1}(y)$. By definition of $f_v(\mathcal{V}(y))$ we have $f_{u_1}(\mathcal{V}(y)) \geq A_{u_1}(x) > B_{u_1}(y)$. This contradicts the IA. □

The above lemma proves the first part of Theorem 4.3. The following lemma proves the second part of the theorem.

Lemma A.3. *With the setting as in the previous lemma, a monotone real circuit separating the inputs of* mCSP-SAT$_F$ *implies a* RCC$_1$ *refutation of* F *of the same size.*

Proof. The RCC$_1$ refutation that we shall construct will have the exact same topology as the given monotone real circuit. Turn each input variable $\mathrm{TT}_i(\alpha)$ of the circuit into the corresponding clause C_i in the refutation. Turn each gate v in the circuit into the line in the refutation computed by the following RCC$_1$ protocol. On input x, Alice privately runs the circuit on $\mathcal{U}(x)$ and sends the value A_v computed by the circuit at gate v to the referee. On input y, Bob acts analogously — he simulates the circuit privately on input $\mathcal{V}(y)$ and sends the value B_v computed by the circuit at gate v to the referee. The referee outputs 0 if and only if $A_v > B_v$. Since at the top gate the circuit is identically 1 on $\mathcal{U}(x)$ and 0 on $\mathcal{V}(y)$, the referee always outputs 0 at the last line in the refutation. Thus, the only thing left to see is that the refutation is sound. Let u_1 and u_2 be the children of v, then $A_v = f(A_{u_1}, A_{u_2})$ and $B_v = f(B_{u_1}, B_{u_2})$ for some monotone function f. Thus, if $A_v > B_v$ then either $A_{u_1} > B_{u_1}$ or $A_{u_2} > B_{u_2}$. □

Appendix B

In this appendix we give a formal proof of Lemma 5.9.

Lemma B.1 (Restatement of Lemma 5.9)**.** *Let C be a k-clause over the Y variables, sampled as in the statement of Lemma 5.8. Then*

$$\Pr[C \textit{ is empty}] = 1/2^k.$$

Proof. Observe that $|X|$ is a binomial random variable consisting of n trials with probability $p = 1/2$ of success, and so $\Pr[|X| = t] = \binom{n}{t}2^{-n}$. Then

$$\begin{aligned}\Pr[C \text{ is empty}] &= \sum_{t=0}^{n} \Pr[|X| = t] \Pr[C \text{ is empty} | |X| = t] \\ &= \sum_{t=0}^{n} \frac{\binom{n}{t}}{2^n} \cdot \frac{\binom{t}{k}}{\binom{n}{k}} \\ &= \frac{1}{2^n \binom{n}{k}} \sum_{t=k}^{n} \binom{n}{t}\binom{t}{k}\end{aligned}$$

where the change in indices follows since if $t < k$ then C can never be contained in X. This sum counts the number of ways to first choose a t-subset A of $[n]$, and then choose a k-subset B of A. Equivalently, we can first choose the k-subset B of $[n]$, and then generate A by extending B to a t-subset. Thus

$$\begin{aligned}\frac{1}{2^n \binom{n}{k}} \sum_{t=k}^{n} \binom{n}{t}\binom{t}{k} &= \frac{1}{2^n \binom{n}{k}} \sum_{t=k}^{n} \binom{n}{k}\binom{n-k}{t-k} \\ &= \frac{1}{2^n} \sum_{t=k}^{n} \binom{n-k}{t-k} \\ &= \frac{2^{n-k}}{2^n} = \frac{1}{2^k}.\end{aligned}$$

□

Appendix C

In this appendix we prove Claim 5.12. The notation is the same as in Lemma 5.11.

Claim C.1 (Restatement of Claim 5.12). *For any fixed $(X, Y) \in \mathcal{P}$, the number of X-heavy and Y-heavy clauses are each upper bounded by $(3/2)m_U$ and lower bounded by $(1/2)m_L$, except with probability at most $\exp(-\Omega(m_L))$.*

Proof. For each clause C_i let T_i be the random variable indicating whether this clause is X-heavy. Clearly the probability of a clause being X-heavy is maximized when $|X|$ is as large as possible. Since we are considering $|X| \in [n/2 - n^{2/3}, n/2 + n^{2/3}]$, it suffices to bound the probability of a clause being X-heavy for $|X| = n/2 + n^{2/3}$. Let $n' = n^{2/3}$ for convenience. We can bound the probability of a clause being X-heavy given (X, Y) as follows:

$$\begin{aligned}\Pr[T_i = 1 | (X, Y)] &= \sum_{\ell=0}^{\epsilon k} \frac{\binom{|Y|}{\ell}\binom{|X|}{k-\ell}}{\binom{n}{k}} \\ &\le \sum_{\ell=0}^{\epsilon k} \binom{n/2 - n'}{\ell}\binom{n/2 + n'}{k-\ell} \frac{1}{\binom{n}{k}} \\ &= \sum_{\ell=0}^{\epsilon k} \frac{\binom{k}{\ell}}{2^k} \cdot \frac{(n/2 - n')!}{(n/2 - n' - \ell)!} \cdot \frac{(n/2 + n')!}{(n/2 + n' - k + \ell)!} \cdot \frac{2^k (n-k)!}{n!}.\end{aligned}$$

The expression $\sum_{\ell=0}^{\epsilon k} \frac{\binom{k}{\ell}}{2^k}$ is what we had before in the analysis of part (2). Thus, if we can bound the term $\frac{(n/2-n')!}{(n/2-n'-\ell)!} \cdot \frac{(n/2+n')!}{(n/2+n'-k+\ell)!} \cdot \frac{2^k(n-k)!}{n!}$ by a constant, we are done. It is maximized when $\ell = 0$, therefore it suffices to bound $\frac{(n/2+n')!}{(n/2+n'-k)!} \cdot \frac{2^k(n-k)!}{n!}$. For that we use the fact that there exist constants c_0 and c_1 such that[5] $c_0 n^{n+1/2}e^{-n} \leq n! \leq c_1 n^{n+1/2}e^{-n}$. Let $c = c_1^2/c_0^2$, then

$$\begin{aligned}
\frac{(n/2+n')!}{(n/2+n'-k)!} \cdot \frac{2^k(n-k)!}{n!} &\leq c \cdot \frac{(n/2+n')^{n/2+n'+1/2}e^{-n/2-n'}}{(n/2+n'-k)^{n/2+n'-k+1/2}e^{-n/2-n'+k}} \cdot \frac{2^k(n-k)^{n-k+1/2}e^{-n+k}}{n^{n+1/2}e^{-n}} \\
&= c \cdot \left(\frac{n/2+n'}{n/2+n'-k}\right)^{n/2+n'-k+1/2} \cdot \frac{2^k(n-k)^{n-k+1/2}(n/2+n')^k}{n^{n+1/2}} \\
&= c\left(1+\frac{k}{z_U}\right)^{z_U+1/2}\left(1-\frac{k}{n}\right)^{n-k+1/2}\left(1+\frac{2n'}{n}\right)^k \\
&\leq c \cdot \exp(k + k/(2z_U))\exp(-k + k^2/n - k/(2n))\exp((2n'k)/n) \\
&\leq c \cdot \exp(k/(2z_U) + k^2/n + (2n'k)/n) \leq c \cdot \exp(3) = O(1),
\end{aligned}$$

where $z_U := n/2 + n' - k$. Therefore, $\Pr[T_i = 1|(X,Y)] \leq c_U \sum_{\ell=0}^{\varepsilon k}\binom{k}{\ell}2^{-k}$ for some constant $c_U > 0$.

Lower bounding the probability of a clause being X-heavy can be done analogously. The probability of a clause being X-heavy is minimized when $|X|$ is as small as possible. Therefore,

$$\begin{aligned}
\Pr[T_i = 1|(X,Y)] &\geq \sum_{\ell=0}^{\varepsilon k}\binom{n/2+n'}{\ell}\binom{n/2-n'}{k-\ell}\frac{1}{\binom{n}{k}} \\
&= \sum_{\ell=0}^{\varepsilon k}\frac{\binom{k}{\ell}}{2^k}\frac{(n/2+n')!}{(n/2+n'-\ell)!} \cdot \frac{(n/2-n)!}{(n/2-n'-k+\ell)!} \cdot \frac{2^k(n-k)!}{n!}
\end{aligned}$$

The term $\frac{(n/2+n')!}{(n/2+n'-\ell)!} \cdot \frac{(n/2-n)!}{(n/2-n'-k+\ell)!} \cdot \frac{2^k(n-k)!}{n!}$ is minimized whenever $\ell = 0$, therefore it suffices to bound $\frac{(n/2-n')!}{(n/2-n'-k)!} \cdot \frac{2^k(n-k)!}{n!}$ from below by a constant. Using the same bound on $n!$ as above,

$$\begin{aligned}
\frac{(n/2-n')!}{(n/2-n'-k)!} \cdot \frac{2^k(n-k)!}{n!} &\geq c^{-1} \cdot \left(\frac{n/2-n'}{n/2-n'-k}\right)^{n/2-n'-k+1/2} \cdot \frac{2^k(n-k)^{n-k+1/2}(n/2-n')^k}{n^{n+1/2}} \\
&= c^{-1} \cdot \left(1+\frac{k}{z_L}\right)^{z_L+1/2}\left(1-\frac{k}{n}\right)^{n-k+1/2}\left(1-\frac{2n'}{n}\right)^k \\
&\geq \frac{c^{-1}}{2^3}\exp\big((k/z_L)(z_L+1/2)\big)\exp\big((-k/n)(n-k+1/2)\big)\exp\big(-2n'k/nk\big) \\
&= \frac{c^{-1}}{2^3}\exp\big(k/(2z_L) + k^2/n - k/(2n) - 2n'/n\big) = \Omega(1),
\end{aligned}$$

where $z_L := n/2 - n' - k$. The third line follows from the fact that k/z_L, k/n, and $2n'k/n$ are all less than $1/2$ for sufficiently large n, and therefore we can use the inequality $(1+x) >= e^x/2$ when $|x| < 1/2$. Therefore, $\Pr[T_i = 1|(X,Y)] \geq \sum_{\ell=0}^{\varepsilon k} c_L\binom{k}{\ell}2^{-k}$ for some constant $0 < c_L < 1$.

[5] More specifically, one can take $c_0 = \sqrt{2\pi}$ and $c_1 = e$.

The remainder of the proof is similar to the proof of part **(2)** of Lemma 5.11. Using both of the inequalities in Fact 5.10, we have

$$\Pr[T_i = 1|(X,Y)] \leq c_U \sum_{\ell=1}^{\varepsilon k} \binom{k}{\ell} 2^{-k} \leq c_U \cdot 2^{H(\varepsilon)k-k}$$
$$\Pr[T_i = 1|(X,Y)] \geq c_L \sum_{\ell=1}^{\varepsilon k} \binom{k}{\ell} 2^{-k} \geq c_L \cdot 2^{-k} \frac{2^{H(\varepsilon)k}}{\sqrt{8k\varepsilon(1-\varepsilon)}} \geq \frac{2^{-(0.86)k}}{\sqrt{k}},$$

since $0.14 < H(\varepsilon) < 0.15$ and $\sqrt{8\varepsilon(1-\varepsilon)} < 1$ for our choice of ε. Let $T := \sum_{i=1}^{m} T_i$. Then by linearity of expectation,

$$c_L \cdot m2^{-(0.86)k}/\sqrt{k} \leq \mathbb{E}[T] \leq c_U \cdot m2^{H(\varepsilon)k-k} = c_U \cdot n2^{(\tau+H(\varepsilon))k},$$

where $\tau = 1/16$. Let $m_L := m2^{-(0.86)k}/\sqrt{k}$ and $m_U := n2^{(\tau+H(\varepsilon))k}$ as before, and define $\delta_U := 3/2c_U - 1$. By the Chernoff bound, we have

$$\begin{aligned}\Pr[T > 3m_U/2|(X,Y)] &\leq \Pr[T > 3\mathbb{E}[T]/(2c_U)|(X,Y)] \\ &= \Pr[T > (1+\delta_U)\mathbb{E}[T]|(X,Y)] \\ &\leq \exp(-\delta_U^2\mathbb{E}[T]/3) \leq \exp(-\delta_U^2 m_L/3) = \exp(-\Omega(m_L)),\end{aligned}$$

where the final equality holds because δ_U is a constant. Similarly, for $\delta_L := 1 - 2/c_L$,

$$\begin{aligned}\Pr[T < m_L/2|(X,Y)] &\leq \Pr[T < (1-\delta_L)\mathbb{E}[T]|(X,Y)] \\ &\leq \exp(-\delta_L^2\mathbb{E}[T]/3) \leq \exp(-\delta_L^2 m_L/3) = \exp(-\Omega(m_L))\end{aligned}$$

Thus we have that $m_L/2 < T < 3m_U/2$ with probability at least $1 - 2\exp(-\Omega(m_L))$. Exactly the same conclusion holds via the same calculations for the Y-variables as well.

□