Electronic Colloquium on Computational Complexity, Revision 2 of Report No. 45 (2017)

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

NOAH FLEMING, University of Toronto, Canada
DENIS PANKRATOV, University of Toronto, Canada
TONIANN PITASSI, University of Toronto, Canada
ROBERT ROBERE, University of Toronto, Canada

The random $k$-SAT model is one of the most important and well-studied distribution 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 size 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. 2018. Random $\Theta(\log n)$-CNFs are Hard for Cutting Planes. 1, 1 (September 2018), 23 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 one of the classic 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. [27, 38]). 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 [15].

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

Authors' addresses: Noah Fleming, University of Toronto, 10 King's College Road, Toronto, ON, M5S 3G4, Canada, noahfleming@cs.toronto.edu; Denis Pankratov, University of Toronto, 10 King's College Road, Toronto, ON, M5S 3G4, Canada, denisp@cs.toronto.edu; Toniann Pitassi, University of Toronto, 10 King's College Road, Toronto, ON, M5S 3G4, Canada, toni@cs.toronto.edu; Robert Robere, University of Toronto, 10 King's College Road, Toronto, ON, M5S 3G4, Canada, robere@cs.toronto.edu.


Manuscript submitted to ACM



ISSN 1433-8092

area, theorists have proven strong lower bounds for refuting random $k$-SAT formulas in many weak proof systems. For instance, in the *Resolution* proof system — which forms the basis of essentially all modern SAT solvers — a classic result of Chvátal and Szemerédi [11] showed 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, $\mathrm{Res}(k)$, and Sum-of-Squares proof systems [1, 2, 5, 37].

In the present work we focus on Cutting Planes refutations of $\mathcal{F}(m, n, k)$. The Cutting Planes technique was introduced in [19] in the context of linear programming, and was shown [9] 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 [12], 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 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, especially because superpolynomial lower bounds for other formulas have been shown [8, 34]. 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*[1].

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 [23] 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* [8, 28, 29, 34, 36]. 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 form

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

Then, given an assignment $\alpha$ 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

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

interpolation was introduced at this level of generality in a classic work of Krajicék [28] where it was shown, for instance, that lower bounds on *monotone circuit complexity* of $I$ can be used to show *Resolution* proof size lower bounds for the formula $F$ (provided $I$ is "monotone" in a certain technical sense). Pudlák generalized Krajicek's argument, showing that lower bounds on *real* monotone circuit complexity lower bounded Cutting Planes proof size [34].

Of course, the obvious drawback of feasible interpolation as a lower bound technique is that it only applies to "split" formulas of the form $A(\vec{x}, \vec{z}) \wedge B(\vec{y}, \vec{z})$; this is why it can not directly be used to prove lower bounds for random formulas from the distribution $\mathcal{F}(m, n, k)$. Since this is the only known technique for proving strong lower bounds against Cutting Planes proofs, researchers were essentially required to invent new techniques to handle random $k$-SAT instances.

To prove Theorem 1.1 we generalize the method of feasible interpolation so that it can be applied to *any* unsatisfiable CNF formula instead of only "split" formulas. As we have just mentioned, Pudlák [34] showed that from a Cutting Planes refutation of a split formula $F(\vec{x}, \vec{y}, \vec{z})$ one can extract a monotone real circuit for the interpolant $I(\vec{z})$ of roughly the same size. We strengthen this connection by considering the proof system RCC (which is stronger than Cutting Planes, see Section 2) and show that RCC proofs of any formula $F$ *characterize* the size of monotone real circuits computing mCSP-SAT, which is a monotone version of the SAT problem that depends on $F$.

**Theorem 1.2** (Informal). *Let $F$ be any unsatisfiable CNF formula. If there is an* RCC *refutation of $F$ of length $\ell$, 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 $\ell$ then there is an* RCC *refutation of $F$ of length* $\mathrm{poly}(\ell)$.

This equivalence generalizes prior arguments by Krajicék [28] and Bonet, Pitassi, and Raz [8], and crucially relies on a recent technical result by Hrubeš and Pudlák [24] reducing real communication protocols to monotone real circuits. With this equivalence in hand we simply need to prove lower bounds on 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 [7, 21, 25]).

As stated above, Hrubeš and Pudlák have independently proved Theorem 1.1 using nearly identical techniques [23]. 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 [7, 21] 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.

*Related Work.* 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 [10, 13, 18]. In a recent breakthrough this conjecture was resolved for large values of $k$ by appealing to arguments in statistical mechanics [14].

The density parameter $\Delta$ 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 $\tau$ 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 [11] 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 probabilty. In their result, $\kappa$ decays doubly-exponentially as $\Delta$ 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 $\kappa$ to polynomial in $\Delta$ 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 [6], Res($k$) [1], and Sum of Squares [37].

Krajicék [28] introduced feasible interpolation in its modern form, with similar results shown around the same time by Razborov [36] and Bonet, Pitassi, and Raz [8]. Using feasible interpolation techniques, exponential lower bounds were obtained for many proof systems, including Resolution and several variants and generalizations of Cutting Planes. For instance, Krajicék used monotone feasible interpolation to obtain lower bounds on Resolution proofs, as well as proofs in the CC-proof system where lines of the proof are computed by low-depth communication protocols [28]. Bonet, Pitassi and Raz used feasible interpolation techniques to obtain exponential lower bounds on *low-weight* Cutting Planes proofs (where the bit-length of each of the coefficients are bounded by $O(\log n)$) [8]. Pudlák obtained exponential lower bounds on the length of Cutting Planes proofs [34]. Pudlák's result was later improved to hold for *semantic* Cutting Planes proofs by Filmus, Hrubeš and Lauria [16]. Despite the success of feasible interpolation, it has been quite limited since it only applies to "split" formulas in the above sense, and the lower bounds ultimately rely on the strength of underlying monotone circuit lower bounds. In particular, the only family of formulas which are known to be hard for (unrestricted) Cutting Planes are the clique-coclique formulas [8, 34] and the Broken Mosquito Screen formulas [22].

Our equivalence in Theorem 1.2 is inspired by earlier results of Razborov [36] where he gave an interesting characterization of circuit size in terms of certain communication protocols solving Karchmer-Wigderson games (this directly generalizes the seminal Karchmer-Wigderson connection between communication complexity and circuit depth [26]). Razborov's reduction was recently simplified by Sokolov [39] and, inspired by Sokolov's proof, we give a direct equivalence between real monotone circuits and RCC-proofs and also between monotone circuits and CC-proofs (cf. Section 4).

## 2 DEFINITIONS AND PRELIMINARIES

If $x, y \in \{0,1\}^n$ then we write $x \le y$ if $x_i \le y_i$ for all $i$. A function $f : \{0,1\}^n \to \{0,1\}$ is *monotone* if $f(x) \le f(y)$ whenever $x \le y$. 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 $x \le y$, 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, Pudlak [34] introduced *monotone real circuits*. In these circuits each internal gate has two inputs and computes any function $\phi(x, y) : \mathbb{R}^2 \to \mathbb{R}$ 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 the sum 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 $\ell$, the number of lines. If all coefficients appearing in the Cutting Planes proof are bounded by $O(\mathrm{poly}(n))$, 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 inqualities $Ax \geq b$ 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 $f, g, h : \{0, 1\}^n \to \{0, 1\}$ 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 $i = 1, 2, \ldots, m$.*
(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 $\ell$.*

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 [8, 16, 31]. 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 [30].

**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 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 (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 *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 Krajicék in the study of feasible interpolation [28], 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 $\mathrm{CC}_d$-refutation of $F$ with respect to the partition $(X, Y)$ is a semantic refutation $L_1, \ldots, L_\ell$ 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 Krajícek [29].

**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 $\alpha, \beta$ 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 $\mathrm{RCC}_d$-refutation of $F$ is a semantic refutation $L_1, L_2, \ldots, L_\ell$ 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 $\mathrm{RCC}_1$-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-$\ell$ low-weight Cutting Planes refutation of $F$ is a length-$\ell$ $\mathrm{CC}_{O(\log n)}$-refutation of $F$. Similarly, any length-$\ell$ Cutting Planes refutation of $F$ is a length-$\ell$ $\mathrm{RCC}_1$-refutation of $F$.*

Although one only needs to establish the equivalence between $\mathrm{RCC}_1$-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 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 [35] under a different name, and it has re-appeared in recent work on lifting theorems in communication complexity [20, 33].

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 $i$th constraint. The CSP $\mathcal{H}$* accepts *an assignment $\rho \in \{0,1\}^R$ if $TT_i(\rho \upharpoonright \mathrm{vars}(i)) = 1$ 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 $\mathrm{TT}_i$.

**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* $\mathrm{TT}_i : \{0,1\}^{\mathrm{vars}(i)} \to \{0,1\}$. *For any $z \in \{0,1\}^N$,* $\text{mCSP-SAT}_H(z) = 1$ *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 $F$s 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* $\text{mCSP-SAT}_H$*, which is a boolean function on $N$ boolean variables. Define sets of accepting and rejecting instances of* $\text{mCSP-SAT}_H$ *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 $C_i(\alpha, y) = 1$.*

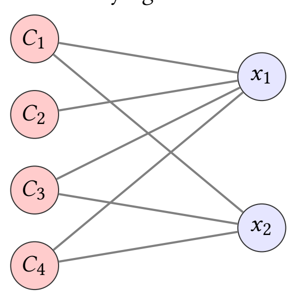
*When the underlying unsatisfiable CNF $F$ is clear from context, we will write* $\mathrm{mCSP\text{-}SAT}_F$ *to mean the partial monotone boolean function corresponding to the above set of accepting and rejecting instances.*

Note that accepting and rejecting inputs to $\mathrm{mCSP\text{-}SAT}_F$ 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 $\mathrm{mCSP\text{-}SAT}_F$ 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 $\mathrm{mCSP\text{-}SAT}_F$ 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 $\mathrm{mCSP\text{-}SAT}_F$ 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 $\mathrm{CC}_d$-proofs and monotone circuits, as well as $\mathrm{RCC}_1$-proofs and monotone *real* circuits. Our argument relating $\mathrm{CC}_d$ and monotone circuits is a direct generalization of main theorem of Bonet, Pitassi, and Raz [8], 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 [36]; Razborov's work was recently simplified by Sokolov [39].

First we give a high-level sketch of the argument. From a $\mathrm{CC}_d$-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 $C^L$ will "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 $\mathrm{CC}_d$-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 $(x, y) \in R$.

**Theorem 4.1.** *Let $F$ be an unsatisfiable CNF formula on $n$ variables and let $X = \{x_1, \ldots, x_{n_1}\}$, $Y = \{y_1, \ldots, y_{n_2}\}$ be any partition of the variables. Let $d$ be a positive integer. If there is a $\mathrm{CC}_d$ refutation of $F$ with respect to the partition $(X, Y)$ of length $\ell$, 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 $O(2^{3d}\ell)$.*

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 $\mathrm{CC}_d$-proof for $F$ with $\ell$ 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 $\ell$, 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 it 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 $C_h^L$ that correctly "separates" $x$ and $y$ for each $(x, y) \in R_L(h)$. By this, we mean that the circuit $C_h^L$ 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 $\alpha, \beta$ 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 $\alpha, \beta$ corresponds to the variable $\mathrm{TT}_i(\alpha)$ of $\mathrm{mCSP\text{-}SAT}_F$, and we define the circuit $C_h^L$ corresponding to line $L = C_i$ and good history $h = 00$ to be the variable $\mathrm{TT}_i(\alpha)$.

Now suppose that $L$ is derived from $L_1$ and $L_2$, and inductively we have circuits $C_{h'}^{L_1}, C_{h''}^{L_2}$ 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 $C_h^L$. It will use all of the circuits that were built for $L_1$ and $L_2$ ($\{C_{h'}^{L_1}, C_{h''}^{L_2}\}$ for all good $h'$ and $h''$) and an additional $2^d$ gates. To build $C_h^L$ 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 $\{C_{h'}^{L_1}, C_{h''}^{L_2}\}$ 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

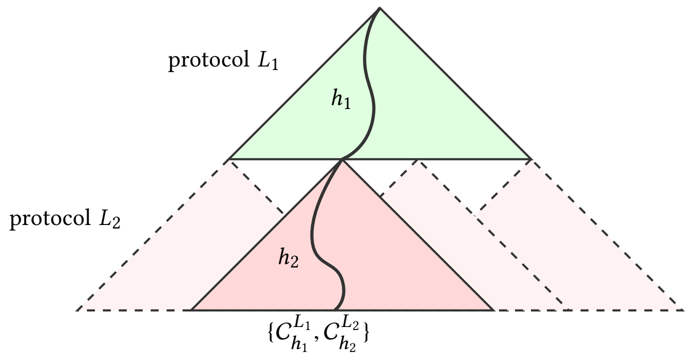



Fig. 1. Protocol tree $T$.

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[2] 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 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 $(x, y) \in R_L(h)$.

*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 $\alpha = x \upharpoonright \mathrm{vars}(C_i)$. In our construction the circuit corresponding to $C^L_h$ is labelled by the variable $\mathrm{TT}_i(\alpha)$, and it is easy to check that $\mathcal{U}(x)$ sets $\mathrm{TT}_i(\alpha)$ to true, and $\mathcal{V}(y)$ sets $\mathrm{TT}_i(\alpha)$ to false.

If $L$ is not an axiom, then we will prove the lemma 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

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

(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 $h'$ and assume that Alice owns $v$. 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 AND 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 $\mathrm{CC}_{O(\log n)}$-proofs to be of independent interest.

**Theorem 4.2.** *If there is a monotone circuit separating the inputs of* mCSP-SAT$_F$ *of size $\ell$, then there is a* $\mathrm{CC}_2$*-refutation of $F$ of length $\ell$ with respect to this variable partition.*

Proof. We show that from a small monotone circuit $F$ for mCSP-SAT$_F$ that separates $\mathcal{U}(\{0,1\}^{|X|})$ and $\mathcal{V}(\{0,1\}^{|Y|})$, we can construct a small $\mathrm{CC}_2$-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.

At a leaf $\ell$ 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 $R_\ell(x, y)$ to be 0 if and only if $x$ is consistent with $\alpha$ 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 $x \upharpoonright \mathrm{vars}(j) = \alpha$ and $\mathcal{R}(\alpha, y, j)$ holds.

Now suppose that $g$ is an OR gate of $C$, with inputs $g_1, g_2$. 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 AND 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 OR 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 $\mathrm{RCC}_1$ proofs and monotone real circuits. The proof (which is in the Appendix) crucially uses a recent technical result regarding real monotone circuits due to Pavel Hrubeš and Pavel Pudlák [24].

**Theorem 4.3** (cf. Theorem 1.2)**.** *Let $F$ be an unsatisfiable CNF formula on $n$ variables and let $(X, Y)$ be any partition of the variables. If there is a $\mathrm{RCC}_1$ refutation of $F$ with respect to the partition $(X, Y)$ of length $\ell$, then there is a monotone real circuit separating the accepting and rejecting instances $\mathcal{U}(\{0,1\}^{|X|}), \mathcal{V}(\{0,1\}^{|Y|})$ of $\text{mCSP-SAT}_F$ of size $\ell$. Conversely, a monotone real circuit separating the inputs of $\text{mCSP-SAT}_F$ implies a $\mathrm{RCC}_1$ refutation of $F$ of the same size.*

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 $S$ 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 $\text{mCSP-SAT}_F$. 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 $\mathrm{RCC}_1$-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 $F^1 = C_1^1 \wedge C_2^1 \wedge \dots \wedge C_m^1$ from $\mathcal{F}(m, n, k)$ on the $X$ variables, and then $F^2 = C_1^2 \wedge C_2^2 \wedge \dots \wedge C_m^2$ from $\mathcal{F}(m, n, k)$ on the $Y$ variables independently. Then output*

$$F = (C_1^1 \vee C_1^2) \wedge (C_2^1 \vee C_2^2) \wedge \dots \wedge (C_m^1 \vee C_m^2).$$

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 $i$th 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-proofs, which is obtained by using Theorem 4.3 and applying the well-known method of symmetric approximations [7, 21] 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 [25]. 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 $f : \{0,1\}^N \to \{0,1\}$ 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 $m = cn^2 2^k$ 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 $k = 4\log n, m = n^6$. 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 [40]. The lemma is stated in general terms for re-use in the next section.

**Lemma 5.5.** *Let $n$ be any sufficiently large positive integer. Let $k, m$ be positive integers and sample $F \sim \mathcal{F}(m,n,k)$. Let $s \leq n/ek^2$ be a positive integer. For any subset $S \subseteq F$ of clauses let* vars$(S)$ *denote the subset of variables appearing in clauses $S$. If*

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

*for some $0 < \delta < 1$, then every set $S \subseteq F$ of size $s$ satisfies $|\text{vars}(S)| \geq ks/2$ 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 $|\text{vars}(S)| < ks/2$ 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 $m = n^2 2^k$. 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 $x \restriction \mathrm{vars}(i) = \alpha$; 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. However, it is easy to see that with high probability every $x$ variable 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 $(\alpha, i)$ 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 $\alpha$ 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 $i$th 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 $\alpha$ 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 $\alpha$ and $\widehat{\alpha}$ agree on column $i$ is at most $1-\frac{k}{2^k n}$. Since columns are sampled independently, the probability that $\alpha$ 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 $\alpha$ 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 $(2s)A_1(1,U) = (2s)2^{n-k} = (2s)2^n/n^2 \leq 2^{n-1}$. 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.*

We recall some basic facts from probability theory which will be used in our main lemma.

**Lemma 5.7** (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 $\Omega$. 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 $\Pr(E) \leq q(1-q)^{|\Gamma(E)|}$, then the probability that none of the events $E_i$ occur is at least $\Pr(\overline{E_1} \wedge \overline{E_2} \wedge \cdots \wedge \overline{E_n}) \geq (1-q)^n$.*

**Fact 5.8** (Entropy bound on binomial tail (Lemma 6.19 in [17])). *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.*

**Fact 5.9** (Multiplicative Chernoff Bound (Theorems 4.4 and 4.5 in [32])). *Suppose $Z_1, \ldots, Z_n$ are independent random variables taking values in $\{0, 1\}$. Let $Z$ denote their sum and let $\mu = \mathbb{E}(Z)$ denote the sum's expected value. Then for any $0 < \delta \leq 1$ we have*

$$\Pr(Z \geq (1+\delta)\mu) \leq e^{-\delta^2\mu/3} \text{ and } \Pr(Z \leq (1-\delta)\mu) \leq e^{-\delta^2\mu/3}.$$

We now prove the main lemma of this section, which shows that for $F \sim \mathcal{F}(m, n, k)$ a good partition of the variables exists with high probability. There is a delicate balance of parameters. In particular, there is tension between the distinct profiles lemma (Lemma 5.6), which requires $m$ to be large, and the Lovasz Local Lemma (Lemma 5.7) which requires $m$ to be small. This is further complicated because we would like to retain all but a constant fraction of assignments in part (3). Because of this we need to set our parameters with precision.

**Lemma 5.10.** *Let $\varepsilon = 1/50$, and let $n$ be a sufficiently large positive integer. Let $k = 240 \log n$, and let $m = n2^{(1+1/16)k}$. 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 $(3/2)m2^{-k/2}$.*
(3) *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. We prove the existence of such a partition by the probabilistic method. For each variable, flip a fair coin and place it in $X$ if the coin is heads and in $Y$ otherwise.

**(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 5.9.

**(2)** For convenience, let $m = n2^{(1+\tau)k}$ where we set $\tau = 1/16$. For each clause $C_i$ in $F$ let $Z_i$ be the random variable indicating whether this clause is $X$-heavy. Using both inequalities in Fact 5.8 we have that

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

and

$$\Pr(Z_i = 1) = \sum_{j=0}^{\varepsilon k} \binom{k}{j} 2^{-k} \geq 2^{-k} \frac{2^{H(\varepsilon)k}}{\sqrt{8k\varepsilon(1-\varepsilon)}} > \frac{2^{-(0.85)k}}{\sqrt{k}}$$

since $0.14 < H(\varepsilon) < 0.15$ and $\sqrt{8\varepsilon(1-\varepsilon)} < 1$ for our choice of $\varepsilon$. Let $Z = \sum_{i=1}^{m} Z_i$; then these two bounds and linearity of expectation imply $m2^{-(0.85)k}/\sqrt{k} \leq \mathbb{E}[Z] \leq m2^{-k}2^{H(\varepsilon)k} = n2^{(\tau+H(\varepsilon))k}$. Denote $m_0 := n2^{(\tau+H(\varepsilon))k}$ and observe that $m_0 < m2^{-k/2}$. By the Chernoff bound (see Fact 5.9) we have

$$\Pr(Z > 3m_0/2) \leq \Pr(Z > 3\mathbb{E}[Z]/2) \leq \exp(-\mathbb{E}[Z]/12) \leq \exp(-m2^{-(0.85)k}/(12\sqrt{k})).$$

Since $m = n2^{(1+\tau)k}$ and $k = 240 \log n$ this occurs with high probability. An identical calculation applies to the $Y$-heavy clauses. It follows by a union bound that the partition satisfies both of the above properties with high probability.

**(3)** Assuming that our partition $(X, Y)$ satisfies properties (1) and (2), we show that the third property is also satisfied with high probability. We first bound the number of times a variable appears in a heavy clause in $F$ with the goal of applying the Lovász Local Lemma. Arbitrarily fix $z$ to be any of the $n$ variables occurring as possible inputs to $F$. Because the partition $(X, Y)$ satisfies (2), the number of $X$-heavy and $Y$-heavy clauses are both bounded by $3m_0/2$. Let $Z_i$ be the indicator random variable which is 1 iff the variable $z$ occurs in the $i$th heavy clause and let $Z = \sum_i Z_i$. Since $F \sim \mathcal{F}(m, n, k)$ we have $\Pr(Z_i = 1) = k/n$ and so $\mathbb{E}[Z] = 3km_0/2n$. Applying the Chernoff bound we get

$$\Pr(Z > 3km_0/n) = \Pr(Z > 2\mathbb{E}[Z]) < \exp(-3km_0/12n). \tag{1}$$

Taking a union bound over the $n$ variables, we conclude that each variable occurs in at most $3km_0/n$ $X$-heavy and $Y$-heavy clauses with high probability.

Now, consider selecting a random assignment to the $X$ variables. Let $E_i$ be the event that the $i$th $X$-heavy clause is not satisfied by the random assignment, and observe that $\Pr(E_i) \leq 2^{-(1-\varepsilon)k}$ since the clause is $X$-heavy. By property (2), the number of events $E_i$ is at most $3m_0/2$. As well, for any event $E_i$, by Equation 1, the number of events that share any $X$ variable with $E_i$ is at most $3m_0k^2/n$. Set $q = 2^{-\delta k}$ for $\delta = 1/15 + H(\varepsilon)$. Then for each $E_i$ we must show

$$q(1-q)^{|\Gamma(E_i)|} \geq qe^{-6qm_0k^2/n} \geq 2^{-(1-\epsilon)k}, \tag{2}$$

or equivalently

$$(1-\varepsilon)k \geq 6\log(e)\frac{qm_0k^2}{n} - \log q.$$

Since $m_0 = n2^{(\tau+H(\varepsilon))k}$ and $q = 2^{-\delta k}$ we have

$$6\log(e)\frac{qm_0k^2}{n} - \log q = 6\log(e)k^2 2^{(\tau+H(\varepsilon)-\delta)k} + \delta k.$$

By our setting $\tau = 1/16$ and $\delta = 1/15 + H(\varepsilon)$ we have $\tau + H(\varepsilon) - \delta < 0$, and thus

$$\delta k + 6\log(e)k^2 2^{(\tau+H(\varepsilon)-\delta)k} \leq \delta k + 6\log e \leq (1-\varepsilon)k$$

for sufficiently large $n$ (note $k \to \infty$ when $n \to \infty$), and so (2) holds.

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 $\tau, \delta$, and $k$,

$$qm_0 = 2^{-\delta k} n 2^{(\tau+H(\varepsilon))k} = n2^{-(\delta-(H(\varepsilon)+\tau)k} = n2^{-(1/15-1/16)240\log n} = 1.$$

Applying the Lovász Local Lemma (Lemma 5.7) we get that the probability that an assignment satisfies all $X$-heavy clauses is at least

$$(1-q)^{3m_0/2} \geq e^{-3qm_0} = e^{-3}.$$

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 in more or less the same way that we proceeded 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 $A$ and $B$ 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 5.11.** *There exists a constant $c > 0$ such that the following holds. Let $n \geq c$ 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 5.10 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 property (3) of Lemma 5.10. 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 $m - 3m2^{-k/2} \geq m/2$ 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$_F$. As in the previous section, we shall apply Theorem 5.2 to $U$ 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 $\rho$ 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 $C_\rho$ separates $U$ and $V$. Because $F'$ is balanced, we can then perform the same argument with for $C_\rho$ with respect to $\mathcal{U}(\mathcal{A})$ and $\mathcal{U}(\mathcal{B})$ as we did for balanced random CNFs in the previous section. A lower bound on the size of $C_\rho$ then implies a lower bound on the size of the unrestricted circuit $C$.

We define the restriction $\rho$ 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 5.12.** *The circuit $C_\rho$ obtained from applying the restriction $\rho$ 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 $\rho$ to an input to mCSP-SAT$_F$. Thus, $C_\rho$ 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 $z$ is defined by setting $TT_i(\alpha) = 1$ if and only if $x \restriction_{vars(i)} = \alpha$. Let $\alpha'$ 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 $x \restriction_{vars(i)} = \alpha'$, 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 $C_\rho$ 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 $\rho$ (if there is no such $\alpha$). The original circuit $C$ outputs 0 on $z$ therefore, by monotonicity, it also outputs outputs 0 on $z' \circ \rho$. This completes the proof of the claim.

The rest of the proof proceeds identically to the proof of Theorem 5.3. We will apply Theorem 5.2 to $U$ and $V$, and counting with respect to the balanced clauses. We begin by calculating $|U|$; to do this, we show that with high probability $\mathcal{U}$ is 1-1 on $\mathcal{A}$ when looking only at the truth tables of balanced clauses. That is, each $z' \in U$ maps 1-1 to a $z \in \mathcal{U}(\mathcal{A})$. This implies that $|U| = |\mathcal{A}|$. By definition, in an accepting input $\mathcal{U}(x)$ we set $TT_i(\alpha) = 1$ 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 is some variable that doesn't appear in any clause. It is easy to see that with high probability the $m/2$ balanced clauses contain all variables occurring in the formula. This implies that $\mathcal{U}$ is 1-1 when restricted to $\mathcal{A}$ and looking only at truth tables of balanced clauses. Therefore $|U| = |\mathcal{A}| = 2^{|X| - 3\log(e))}$.

Similarly, we show that with high probability $\mathcal{V}$ is 1-1 on $\mathcal{B}$ when looking only at the truth tables of balanced clauses. That is, each $z' \in \mathcal{V}(\mathcal{B})$ maps 1-1 to a $z \in \mathcal{V}(\mathcal{B})$. Letting $\mathcal{S} = \mathcal{V}(\mathcal{B})$, we can apply Lemma 5.6 with respect to the the $m/2$ balanced clauses. Note that the $m/2$ balanced clauses $F'$ are obtained from $F$ by discarding heavy clauses; heavy-ness depends only on the partition $(X, Y)$ and so $F'$ is distributed as $\mathcal{F}(m/2, n, k)$. Since

$$km/8n2^k = 3n^{15}\log(n) > \log|\mathcal{S}|$$

for sufficiently large $n$ this lemma implies that $\mathcal{V}$ is 1-1 on $\mathcal{B}$ when restricted to the truth tables of the balanced clauses with high probability. Therefore $|V| = |\mathcal{B}| = 2^{|Y| - 3\log(e)}$. 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 5.13.** *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 $|\mathrm{vars}(S)| \geq k_0 s/2$ 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 $\alpha$ 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 5.13. 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 5.13 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 5.13 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 5.14** (Theorem 1.1)**.** *Let $F$ be distributed as above. There exists $\varepsilon > 0$ such that with high probability any* $\mathrm{RCC}_1$*-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 $k$. Another standard formula which is believed to be hard for Cutting Planes are the Tseitin tautologies (conjectured, for instance, in [25]). However, $\mathrm{CC}_2$ proofs admits linear-length refutations of the Tseitin graph principles on any underlying graph — simply consider the lines as mod 2 linear equations and add the constraints, using the fact that each variable occurs in exactly two clauses. Therefore our techniques cannot be directly applied to obtain lower bounds for the Tseitin graph principles.

## 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] 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

[6] 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

[7] 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

[8] 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

[9] 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

[10] 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

[11] 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

[12] 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

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

[14] 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

[15] 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

[16] Yuval Filmus, Pavel Hrubes, 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

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

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

[19] 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

[20] 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.

[21] 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

[22] 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

[23] Pavel Hrubes 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

[24] Pavel Hrubes 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

[25] 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

[26] 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

[27] 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

[28] Jan Krajícek. 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

[29] Jan Krajícek. 1998. Interpolation by a Game. *Math. Log. Q.* 44 (1998), 450–458. https://doi.org/10.1002/malq.19980440403
[30] Eyal Kushilevitz and Noam Nisan. 1997. *Communication complexity.* Cambridge University Press.
[31] 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
[32] Michael Mitzenmacher and Eli Upfal. 2005. *Probability and computing - randomized algorithms and probabilistic analysis.* Cambridge University Press.
[33] Igor C. Oliveira. 2015. *Unconditional lower bounds in complexity theory.* Ph.D. Dissertation. Columbia University.
[34] 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
[35] Ran Raz and Pierre McKenzie. 1999. Separation of the Monotone NC Hierarchy. *Combinatorica* 19, 3 (1999), 403–435. https://doi.org/10.1007/s004930050062
[36] Alexander Razborov. 1995. Unprovability of Lower Bounds on Circuit Size in Certain Fragments of Bounded Arithmetic. *Izvestiya Mathematics* 59, 1 (1995), 205–227.
[37] 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
[38] 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
[39] 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
[40] Salil P. Vadhan. 2012. Pseudorandomness. *Foundations and Trends in Theoretical Computer Science* 7, 1-3 (2012), 1–336. https://doi.org/10.1561/0400000010

## 7 APPENDIX

In this appendix, we prove Theorem 4.3. Theorem 4.3 follows from the following lemmas. The first lemma shows how to translate an $\mathrm{RCC}_1$ refutation for $F$ into a real monotone circuit for mCSP-SAT$_F$; the proof is a modification of a recent technical result of Hrubeš and Pudlák relating real communication protocols with real monotone circuits [24]. The second lemma shows a converse, and is a simple direct argument analogous to Theorem 4.2.

**Lemma 7.1** (cf. Theorem 5 in [24]). *Let $F$ be an unsatisfiable CNF formula on $n$ variables and let $X = \{x_1, \dots, x_{n_1}\}$, $Y = \{y_1, \dots, y_{n_2}\}$ be any partition of the variables. If there is a $\mathrm{RCC}_1$ refutation of $F$ with respect to the partition $(X, Y)$ of length $\ell$, then there is a real monotone circuit separating the accepting and rejecting instances $\mathcal{U}(\{0,1\}^{|X|}), \mathcal{V}(\{0,1\}^{|Y|})$ of* mCSP-SAT$_F$ *with $\ell$ gates.*

Proof. Fix an $\mathrm{RCC}_1$-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 $\alpha_i$ 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

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

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 $\mathrm{RCC}_1$ 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 $x$s 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 7.2.** *With the setting as in the previous lemma, a monotone real circuit separating the inputs of* mCSP-SAT$_F$ *implies a* $\mathrm{RCC}_1$ *refutation of F of the same size.*

Proof. The $\mathrm{RCC}_1$ 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 $\mathrm{RCC}_1$ 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 $A_{u_2} > B_{u_2}$. □



ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il