On the Relative Complexity of Resolution Refinements and Cutting Planes Proof Systems*

Maria Luisa Bonet[†‡] Juan Luis Esteban[§†] Nicola Galesi[¶†] Jan Johannsen[‖]

Abstract

An exponential lower bound for the size of tree-like Cutting Planes refutations of a certain family of CNF formulas with polynomial size resolution refutations is proved. This implies an exponential separation between the tree-like versions and the dag-like versions of resolution and Cutting Planes. In both cases only superpolynomial separations were known [29, 18, 8]. In order to prove these separations, the lower bounds on the depth of monotone circuits of Raz and McKenzie in [25] are extended to monotone real circuits.

An exponential separation is also proved between tree-like resolution and several refinements of resolution: negative resolution and regular resolution. Actually this last separation also provides a separation between tree-like resolution and ordered resolution, thus the corresponding superpolynomial separation of [29] is extended.

Finally, an exponential separation between ordered resolution and unrestricted resolution (also negative resolution) is proved. Only a superpolynomial separation between ordered and unrestricted resolution was previously known [13].

MSC Classification: 03F20, 68Q17, 68T15

1 Introduction

The motivation for research on the proof length of propositional proof systems is double. First, by the work of Cook and Reckhow [10] we know that the claim that *for every propositional proof system there is a class of tautologies that have no polynomial-size proofs* is equivalent to $NP \neq co\text{-}NP$. This connection explains the interest in developing combinatorial techniques to prove lower bounds for proof systems. The second motivation comes from the interest in studying efficiency issues in Automated Theorem Proving. The question is, which proof systems have efficient algorithms to find proofs. Actually, the proof system most widely used for implementations is resolution or refinements of resolution. Our work is relevant to both motivations. On one hand, all the separation results of this paper improve previously known superpolynomial separations to exponential. On the other hand, these exponential separations harden the known results showing inefficiency of several widely used strategies for finding proofs, especially for the resolution system.

*A preliminary version of this paper appeared as [5] and as ECCC TR98-035.

†Departament de Llenguatges i Sistemes Informàtics, Universitat Politècnica de Catalunya, {bonet,esteban,galesi}@lsi.upc.es

‡Partially supported by projects SPRIT 20244 ALCOM-IT, TIC 98-0410-C02-01 and PB98-0937-C04-03.

§Partially supported by project KOALA:DGICYT:PB95-0787.

¶Supported by an European Community grant under the TMR project.

‖Institut für Informatik, Ludwig-Maximilians-Universität München, jjohanns@informatik.uni-muenchen.de. Research of this author done at Department of Mathematics, University of California, San Diego, supported by DFG grant No. Jo 291/1-1.

Haken [16] was the first to prove exponential lower bounds for unrestricted resolution. He showed that the Pigeonhole Principle requires exponential-size resolution refutations. Urquhart [28] found another class of tautologies with the same property. Chvátal and Szemerédi [7] showed that in some sense, almost all classes of tautologies require exponential size resolution proofs (see [2, 3] for simplified proofs of these results). These exponential lower bounds are bad news for Automated Theorem Proving, since they mean that often the time used in finding proofs will be exponentially long in the size of the tautology, just because the shortest proofs are exponentially long in the size of the tautology.

Many strategies for finding resolution proofs are described in the literature, see e.g. Schöning's textbook [27]. One commonly used type of strategy is to reduce the search space by defining restricted versions of resolution that are still complete. Such restricted forms are commonly referred to as *resolution refinements*. One particularly important resolution refinement is tree-like resolution. Its importance stems from the close relationship between the complexity of tree-like resolution proofs and the runtime of a certain class of satisfiability testing algorithms, the so-called *DLL Algorithms* (cf. [24, 1]). We prove an exponential separation between tree-like resolution and unrestricted resolution (Corollary 20), thus showing that finding tree-like resolution proofs is not an efficient strategy for finding resolution proofs. Until now only superpolynomial separations were known [29, 8].

We also consider three more of the most commonly used resolution refinements: negative resolution, regular resolution and ordered resolution. We show an exponential separation between tree-like resolution and each one of the above restrictions (Corollary 20 for negative resolution and Corollary 23 for both regular and ordered resolution).

Goerdt [14, 13, 15] gave several superpolynomial separations between unrestricted resolution and some refinements of resolution, in particular he gave a superpolynomial separation between ordered resolution and unrestricted resolution. In this paper we consider the case of ordered resolution and we improve his separation to exponential. We prove that a certain CNF formula requires exponential size ordered resolution refutations, but can be refuted with a polynomial size negative resolution proof (Corollary 29), thus in particular showing that unrestricted resolution can have an exponential speed-up over ordered resolution.

The Cutting Planes proof system, CP from now on, is a refutation system based on manipulating integer linear inequalities. Exponential lower bounds for the size of CP refutations have already been proven. Impagliazzo et al. [17] proved exponential lower bounds for tree-like CP. Bonet et al. [6] proved a lower bound for the subsystem CP^*, where the coefficients appearing in the inequalities are polynomially bounded in the size of the formula being refuted. This is a very important result because all known CP refutations fulfill this property. Finally, Pudlák [23] and Cook and Haken [9] gave general circuit complexity results from which exponential lower bounds for CP follow. To this day it is still unknown whether CP is more powerful than CP^*, i.e., whether it produces shorter proofs or not.

Since there is an exponential speed-up of CP over resolution, it would be nice to find an efficient algorithm for finding CP proofs and a question to ask is whether trying to find tree-like CP proofs would be an efficient strategy for finding Cutting Planes proofs. Johannsen [18] gave a superpolynomial separation, with a lower bound of the form $\Omega(n^{\log n})$, between tree-like CP and dag-like CP (this was previously known for CP^* from [6]). Here we improve that separation to exponential (Corollary 20). This shows that searching for tree-like proofs is also not a good strategy for finding proofs in CP.

The separation between tree-like and dag-like versions of resolution and CP are obtained using the technique of the interpolation method introduced by Krajíček [21]. Closely related ideas appeared previously in the mentioned works that gave lower bounds for fragments of CP ([17, 6]). The interpolation method applied on CP, translates proofs of certain formulas to monotone real circuits (a generalization of boolean circuits). The translation has two important features. First, it preserves the size, that is, the size of the circuit is similar to the size of the proof from which the circuit is built. Second, if the proof is tree-like, the circuit will be also tree-like, i.e., a formula. So we can prove size lower bounds for tree-like CP proofs by proving size lower bounds for monotone real formulas.

In section 3 we prove that a certain boolean function $\textsc{Gen}_n$ requires exponential size monotone real formulas. This is a consequence of extending the result of Raz and McKenzie [25], proving linear depth lower bounds for monotone boolean circuits, to the case of monotone real circuits. We use these circuit complexity lower bounds to obtain proof complexity lower bound using the interpolation method.

2 Preliminaries and Outline of the Paper

In this section we introduce the notions we use and our main results. We also discuss the structure of the paper and the dependency among our main results.

2.1 Proof Systems

We start by giving a short description of the proof systems studied in this paper. Most proof systems can be used in a tree-like or dag-like fashion. In a tree-like proof any line in the proof can be used only once as a premise. Should the same line be used twice, it must be rederived. A proof system that only produces tree-like proofs is called *tree-like*. Otherwise we will call it *dag-like* or when nothing is said it is understood that the system is dag-like.

2.1.1 Resolution

Resolution is a refutation proof system for CNF formulas, which are represented as sets of *clauses*, i.e., disjunctions of literals. Clauses that contain the same literals are considered equal. The only inference rule is the resolution rule:

$$\frac{C \vee x \qquad D \vee \bar{x}}{C \vee D}\,.$$

That is, from clauses $C \vee x$ and $D \vee \bar{x}$ we get clause $C \vee D$, called the *resolvent*. We say that the variable x is *eliminated* in this resolution step. A resolution refutation of a set Σ of clauses is a derivation of the empty clause from Σ using the resolution rule. Resolution is a sound and complete refutation system, i.e., a set of clauses has a resolution refutation if and only if it is unsatisfiable.

Several refinements of the resolution proof system have been proposed. These refinements reduce the search space by restricting the choice of pairs of clauses to which the resolution rule can be applied. In this paper we consider the following three refinements, all of which are still complete:

1. The *regular* resolution system: Viewing the refutations as graph, in any path from the empty clause to any initial clause, no variable is eliminated twice.

2. The *ordered*[1] resolution system: There exists an ordering of the variables in the formula being refuted, such that if a variable x is eliminated before a variable y on any path from an initial

[1]In Goerdt's paper [13] and in the preliminary version [5] of this paper, this refinement is called *Davis-Putnam resolution*. In the meantime, we have learned that it is better known as *ordered* resolution.

clause to the empty clause, then x is before y in the ordering. As no variable is eliminated twice on any path, ordered resolution is a restriction of regular resolution.

3. The *negative* resolution system: To apply the resolution rule, one of the two clauses must consist of negative literals only.

There is an algorithm (see e.g. Urquhart [29]) that transforms a tree-like resolution proof into a possibly smaller regular tree-like resolution proof, therefore tree-like resolution proofs of minimal size are regular. This means that from the point of view of proof complexity, tree-like resolution and tree-like regular resolution are equivalent.

2.1.2 Cutting Planes

The Cutting Planes proof system, CP for short, is a refutation system for CNF formulas, as resolution is. It works with linear inequalities. The initial clauses are transformed into linear inequalities. A generic clause

$$\bigvee_{i=1}^{k} p_{j_i} \vee \bigvee_{i=1}^{m} \neg p_{l_i}$$

is transformed into a linear inequality

$$\sum_{i=1}^{k} p_{j_i} + \sum_{i=1}^{m} (1 - p_{l_i}) \geq 1.$$

The CP rules are basic algebraic manipulations, additions of two inequalities, multiplication of an inequality by a positive integer and the following division rule:

$$\frac{\sum_{i \in I} a_i x_i \geq k}{\sum_{i \in I} \frac{a_i}{b} x_i \geq \left\lceil \frac{k}{b} \right\rceil},$$

where b is a positive integer that evenly divides all a_i, $i \in I$. A CP refutation of a set E of inequalities is a derivation of $0 \geq 1$ from the inequalities in E and the axioms $x \geq 0$ and $-x \geq -1$ for every variable x, using the CP rules. It can be shown that a set of inequalities has a CP-refutation iff it has no $\{0,1\}$-solution. Any assignment satisfying the original clauses is actually a $\{0,1\}$-solution of the corresponding inequalities, provided that we assign the numerical value 1 to True and the value 0 to False. It is easy to translate, see [11], resolution refutations into CP refutations similar in size to the original resolution refutations. Moreover if the resolution refutation is tree-like, the resulting CP refutation is also tree-like.

2.2 Monotone Real Circuits

An important part of this paper is concerned with monotone real circuits, which were introduced by Pudlák [23]. A *monotone real circuit* is a circuit of fan-in 2 computing with real numbers where every gate computes a nondecreasing real function. We require that monotone real circuits output 0 or 1 on every input of zeroes and ones only, so that they are a generalization of monotone boolean circuits. The depth and size of a monotone real circuit are defined as for boolean circuits. A *formula* is a circuit in which every gate has at most fan-out 1, i.e., a tree-like circuit.

Pudlák [23], Cook and Haken [9] and Fu [12] gave lower bounds on the size of monotone real circuits. Rosenbloom [26] showed that they are strictly more powerful than monotone boolean circuits, since every slice function can be computed by a linear-size, logarithmic-depth monotone real circuit, whereas most slice functions require exponential size general boolean circuits. On the other hand, Jukna [19] gives a general lower bound criterion for monotone real circuits, and uses it to show that certain functions in *P/poly* require exponential-size monotone real circuits, hence the computing power of monotone real circuits and general boolean circuits is incomparable.

For a monotone boolean function f, we denote by $d_{\mathbb{R}}(f)$ the minimal depth of a monotone real circuit computing f, and by $s_{\mathbb{R}}(f)$ the minimal size of a monotone real formula computing f.

2.3 Deterministic and Real Communication Complexity

The use of communication complexity as a tool to prove depth lower bounds for monotone circuits was introduced by Karchmer and Wigderson [20]. They gave an $\Omega(\log^2 n)$ lower bound on the depth of monotone circuits computing st-connectivity.

Krajíček [22] introduced a notion of *real communication complexity*, generalizing ordinary communication complexity, that is suitable to prove depth lower bounds for monotone real circuits. This was used by Johannsen [18] to extend the depth lower bound for st-connectivity to monotone real circuits.

Raz and McKenzie [25] proved an $\Omega(n^\epsilon)$ lower bound on the depth of monotone circuits computing a certain function $\textsc{Gen}_n$, which, on the other hand, can be computed by monotone circuits of polynomial size. This gives a strong separation of the depth and size complexity of monotone circuits. We extend this lower bound to monotone real circuits, again using the notion of real communication complexity.

2.3.1 Communication Complexity

Let $R \subseteq X \times Y \times Z$ be a multifunction, i.e., for every pair $(x, y) \in X \times Y$, there is a $z \in Z$ with $(x, y, z) \in R$. We view such a multifunction as a search problem, i.e., given input $(x, y) \in X \times Y$, the goal is to find a $z \in Z$ such that $(x, y, z) \in R$.

A deterministic communication protocol P over $X \times Y \times Z$ specifies the exchange of information bits between two players, I and II, that receive as inputs respectively $x \in X$ and $y \in Y$ and finally agree on a value $P(x, y) \in Z$ such that $(x, y, P(x, y)) \in R$. The *deterministic communication complexity* of R, $CC(R)$, is the number of bits communicated between players I and II in an optimal protocol for R.

2.3.2 Real Communication Complexity

A real communication protocol over $X \times Y \times Z$ is executed by two players I and II who exchange information by simultaneously playing real numbers and then comparing them according to the natural order of $\mathbb{R}$. This generalizes ordinary deterministic communication protocols in the following way: in order to communicate a bit, the sender plays this bit, while the receiver plays a constant between 0 and 1, so that he can determine the value of the bit from the outcome of the comparison.

Formally, such a protocol P is specified by a binary tree, where each internal node v is labeled by two functions $f_v^I : X \to \mathbb{R}$, giving player I's move, and $f_v^{II} : Y \to \mathbb{R}$, giving player II's move, and each leaf is labeled by an element $z \in Z$. On input $(x, y) \in X \times Y$, the players construct a path through the tree according to the following rule:

> At node v, if $f_v^I(x) > f_v^{II}(y)$, then the next node is the left son of v, otherwise the right son of v.

The value $P(x, y)$ computed by P on input (x, y) is the label of the leaf reached by this path.

A real communication protocol P solves a search problem $R \subseteq X \times Y \times Z$ if for every $(x, y) \in X \times Y$, $(x, y, P(x, y)) \in R$ holds. The *real communication complexity* $CC_{\mathbb{R}}(R)$ of a search problem R is the minimal depth of a real communication protocol that solves R.

For a natural number n, let $[n]$ denote the set $\{1, \ldots, n\}$. Let $f : \{0,1\}^n \to \{0,1\}$ be a monotone boolean function, let $X := f^{-1}(1)$ and $Y := f^{-1}(0)$, and let the multifunction $R_f \subseteq X \times Y \times [n]$ be defined by

$$(x, y, i) \in R_f \quad \text{iff} \quad x_i = 1 \text{ and } y_i = 0$$

The Karchmer-Wigderson game for f is defined as follows: Player I receives an input $x \in X$ and Player II an input $y \in Y$. They have to agree on a position $i \in [n]$ such that $(x, y, i) \in R_f$. The Karchmer-Wigderson game for a monotone boolean function f is also denoted by R_f. As happens with monotone boolean functions and communication complexity, there is a relation between the real communication complexity of R_f and the depth of monotone real circuits (and the size of a monotone real formulas) computing f.

Lemma 1 (Krajíček [22]). *Let f be a monotone boolean function. Then*

1. $CC_{\mathbb{R}}(R_f) \leq d_{\mathbb{R}}(f)$;
2. $CC_{\mathbb{R}}(R_f) \leq \log_{3/2} s_{\mathbb{R}}(f)$.

For a proof see [22] or [18]. Notice that by 2 a linear lower bound for the real communication complexity of R_f gives an exponential lower bound for the size of the smallest monotone real formula computing f.

2.4 DART Games and Structured Protocols

Raz and McKenzie [25] introduced a special kind of communication games, called DART games, and a special class of communication protocols, the *structured protocols*, for solving them.

For $m, k \in \mathbb{N}$, $\mathrm{DART}(m, k)$ is the set of communication games specified by a relation $R \subseteq X \times Y \times Z$ such that:

- $X = [m]^k$. I.e., the inputs for Player I are k-tuples of elements $x_i \in [m]$.
- $Y = (\{0,1\}^m)^k$. I.e., the inputs for Player II are k-tuples of binary colorings y_i of $[m]$.
- For all $i = 1, \ldots, k$ let $e_i = y_i(x_i) \in \{0,1\}$ (i.e., the x_i-th bit in the m-bits string y_i). The relation $R \subseteq X \times Y \times Z$ defining the game only depends on $e_1, \ldots, e_k$ and z, i.e., we can describe $R(x, y, z)$ as $R((e_1, \ldots, e_k), z)$.
- $R((e_1, \ldots, e_k), z)$ can be expressed as a DNF-Search-Problem, i.e., there exists a DNF-tautology F_R defined over the variables $e_1, \ldots, e_k$ such that Z is the set of terms of F_R, and $R((e_1, \ldots, e_k), z)$ holds if and only if the term z is satisfied by the assignment $(e_1, \ldots, e_k)$.

A *structured protocol* for a DART game is a communication protocol for solving the search problem R, where player I gets input $x \in X$, player II gets input $y \in Y$, and in each round, player I reveals the value x_i for some i, and II replies with $y_i(x_i)$. The structured communication complexity of $R \in \mathrm{DART}(m, k)$, denoted by $SC(R)$, is the minimal number of rounds in a structured protocol solving R. In [25] it was proved that $CC(R) = SC(R) \cdot \Omega(\log m)$ for $R \in \text{DART}(m, k)$. We generalize this result to real communication complexity, proving

$$CC_{\mathbb{R}}(R) = SC(R) \cdot \Omega(\log m).$$

Observe that at each structured round the two players transmit $\lceil \log m \rceil + 1$ bits. The first player transmits a number in $[m]$ and the second answers with a bit. Since both players know the structure of the protocol for the game, at each round they both know what is the coordinate i of the inputs they are talking about and they have no need to transmit it. So for a DART game R we have $CC_{\mathbb{R}}(R) \leq SC(R) \cdot \Omega(\log m)$.

Proving the opposite inequality, which is one of our main results, is much harder. In Theorem 6 we show that for every relation $R \in \text{DART}(m, k)$, where $m \geq k^{14}$, $CC_{\mathbb{R}}(R) \geq SC(R) \cdot \Omega(\log m)$.

2.5 The Interpolation Method

The separations between tree-like Cutting Planes (respectively resolution) and Cutting Planes (resolution) are among our main results about proof complexity. The lower bound part of the separation is obtained employing the following Theorem which relates the size of Cutting Planes refutations with size of monotone real circuits.

Theorem 2 (Pudlák [23]). *Let $\vec{p}, \vec{q}, \vec{r}$ be disjoint vectors of variables, and let $A(\vec{p}, \vec{q})$ and $B(\vec{p}, \vec{r})$ be sets of inequalities in the indicated variables such that the variables $\vec{p}$ either have only nonnegative coefficients in $A(\vec{p}, \vec{q})$ or have only nonpositive coefficients in $B(\vec{p}, \vec{r})$.*

Suppose there is a CP refutation P of $A(\vec{p}, \vec{q}) \cup B(\vec{p}, \vec{r})$. Then there is a monotone real circuit $C(\vec{p})$ of size $O(|P|)$ such that for any vector $\vec{a} \in \{0,1\}^{|\vec{p}|}$

$$\begin{array}{lcl} C(\vec{a}) = 0 & \rightarrow & A(\vec{a}, \vec{q}) \text{ is unsatisfiable} \\ C(\vec{a}) = 1 & \rightarrow & B(\vec{a}, \vec{r}) \text{ is unsatisfiable} \end{array}$$

Furthermore, if P is tree-like, then $C(\vec{p})$ is a monotone real formula.

The fact that the interpolant $C(\vec{p})$ is in a monotone real formula if the refutation is tree-like is not stated explicitly in [23], but it can be checked easily by analyzing the original proof of theorem 2 in [23],

We use this theorem to get lower bounds for cutting planes refutations from lower bounds for monotone real formulas. Recall that a *minterm* (respectively a *maxterm*) of a boolean function $f : \{0,1\}^n \to \{0,1\}$ is a set of inputs $x \in \{0,1\}^n$ such that $f(x) = 1$ (respectively $f(x) = 0$) and for each $y \in \{0,1\}^n$ obtained from x by changing a bit from 1 to 0 (respectively by changing a bit from 0 to 1) it holds that $f(y) = 0$ (respectively $f(y) = 1$).

For a certain boolean function f we will apply Theorem 2 to a CNF-formula $A(\vec{p}, \vec{q}) \cup B(\vec{p}, \vec{r})$ such that $A(\vec{p}, \vec{q})$ will encode that $\vec{p}$ is a minterm of f and $B(\vec{p}, \vec{r})$ will encode that $\vec{p}$ is maxterm of f. Using the interpolation theorem, from any tree-like CP proof of $A(\vec{p}, \vec{q}) \cup B(\vec{p}, \vec{r})$ we will get an interpolant which is a monotone real formula computing f. Therefore if we prove exponential lower bounds for the size of the tree-like monotone real circuits computing f, we immediately obtain an exponential lower bound for tree-like Cutting Planes refutations for $A(\vec{p}, \vec{q}) \cup B(\vec{p}, \vec{r})$. The same result also holds for tree-like resolution.

To get the separation results we need a monotone boolean function with some nice properties, namely:

1. Exponential lower bounds for monotone real formulas computing the function, and

2. The corresponding $A(\vec{p},\vec{q}) \cup B(\vec{p},\vec{r})$ formula must have polynomial-size resolution (and therefore also CP) refutations.

The chosen monotone boolean function f is the function $\textsc{Gen}_n : \{0,1\}^{n^3} \to \{0,1\}$ considered by Raz and McKenzie [25]. The input bits are called $t_{a,b,c}$ for $a,b,c \in [n]$. The function is defined as follows: $\textsc{Gen}_n(\vec{t}) = 1$ iff $\vdash n$ where for $c \in [n]$, $\vdash c$ (meaning c *is generated*) is defined recursively by

$$\vdash c \quad \text{iff} \quad c = 1 \text{ or there are } a, b \le n \text{ with } \vdash a\ ,\ \vdash b \text{ and } t_{a,b,c} = 1\ .$$

From now on we will write $a, b \vdash c$ for $t_{a,b,c} = 1$.

To get the exponential separation the task to be done is:

1. Prove exponential lower bounds for the size of monotone real formulas computing $\textsc{Gen}_n$;

2. Find CNF-formulas $A(\vec{p},\vec{q})$ and $B(\vec{p},\vec{r})$ expressing respectively a minterm and a maxterm of $\textsc{Gen}_n$.

3. Show polynomial-size resolution (and CP) refutations for $A(\vec{p},\vec{q}) \cup B(\vec{p},\vec{r})$

In section 3 we will show, among other things, that $CC_{\mathbb{R}}(R_{\textsc{Gen}_n}) \ge \Omega(n^{\epsilon})$, for some $\epsilon > 0$. From this, it follows by part 2 of Lemma 1 that $s_{\mathbb{R}}(\textsc{Gen}_n) \ge 2^{\Omega(n^{\epsilon})}$, thus task 1 is achieved. Tasks 2 and 3 will be developed in section 4.

3 Lower bounds for Real Communication Complexity

In this section we prove a $\Omega(n^{\epsilon})$ lower bound for the real communication complexity of the Karchmer-Wigderson game associated to $\textsc{Gen}_n$, denoted by $R_{\textsc{Gen}_n}$.

Theorem 3. *For some $\epsilon > 0$ and sufficiently large n*

$$CC_{\mathbb{R}}(R_{\textsc{Gen}_n}) \ge \Omega(n^{\epsilon}).$$

To prove theorem 3 we define a DART game $\textsc{PyrGen}(m,d)$ in section 3.1 related to the $\textsc{Gen}_n$ function. This game is used with parameters $m = d^{28}$ and $n = \binom{d+1}{2}m + 2$, so that $d \approx n^{1/30}$. Then we will prove the following results from which Theorem 3 directly follows.

$$\begin{aligned} SC(\textsc{PyrGen}(m,d)) &\ge d && \text{(Lemma 4)}\\ CC_{\mathbb{R}}(\textsc{PyrGen}(m,d)) &\ge SC(\textsc{PyrGen}(m,d))\,\Omega(\log m) && \text{(Theorem 6)}\\ CC_{\mathbb{R}}(R_{\textsc{Gen}_n}) &\ge CC_{\mathbb{R}}(\textsc{PyrGen}(m,d)) && \text{(Lemma 5)} \end{aligned}$$

Lemma 4 is proved in [25], therefore we omit its proof. Theorem 6 is proved in section 3.2 for any DART game R. Lemma 5 is proved in section 3.1. In section 3.3 we deduce some lower bounds for monotone real circuits from these results.

3.1 The Pyramidal Generation Game

For $d \in \mathbb{N}$, let $Pyr_d := \{(i,j) : 1 \le j \le i \le d\}$. Following [25], a communication game in $DART(m, \binom{d+1}{2})$ called $\textsc{PyrGen}(m,d)$ is defined as follows: We regard the indices as elements of Pyr_d, so that the inputs for the two players I and II in the $\textsc{PyrGen}(m,d)$ game are respectively sequences of elements $x_{i,j} \in [m]$ and $y_{i,j} \in \{0,1\}^m$ with $(i,j) \in Pyr_d$, and we picture these as laid out in a pyramidal form with $(1,1)$ at the top and (d,j), $1 \le j \le d$ and the bottom. The goal of the game is to find either an element colored 0 at the top of the pyramid, or an element colored 1 at the bottom of the pyramid, or an element colored 1 with the two elements below it colored 0. That is we have to find indices (i,j) such that one of the following holds:

1. $i = j = 1$ and $y_{1,1}(x_{1,1}) = 0$, or
2. $y_{i,j}(x_{i,j}) = 1$ and $y_{i+1,j}(x_{i+1,j}) = 0$ and $y_{i+1,j+1}(x_{i+1,j+1}) = 0$, or
3. $i = d$ and $y_{d,j}(x_{d,j}) = 1$.

Observe that, setting $e_{i,j} = y_{i,j}(x_{i,j})$ for $1 \le j \le i \le d$, this search problem can be defined as a DNF search problem given by the following DNF tautology:

$$\bar{e}_{1,1} \vee \bigvee_{1 \le j \le i \le d-1} (e_{i,j} \wedge \bar{e}_{i+1,j} \wedge \bar{e}_{i+1,j+1}) \vee \bigvee_{1 \le j \le d} e_{d,j}$$

Therefore, $\textsc{PyrGen}(m,d)$ is a game in $\mathrm{DART}(m, \binom{d+1}{2})$.

A lower bound on the structured communication complexity of $\textsc{PyrGen}(m,d)$ was proved in [25]:

Lemma 4 (Raz/McKenzie [25]). *$SC(\textsc{PyrGen}(m,d)) \ge d$.*

The following reduction shows that the real communication complexity of the game $\textsc{PyrGen}(m,d)$ is bounded by the real communication complexity of the Karchmer-Wigderson game for $\textsc{Gen}_n$ (denoted by $R_{\textsc{Gen}_n}$) for a suitable n. The proof is taken from [25]. It is included because it can help to understand other parts of this paper.

Lemma 5. *Let $d, m \in \mathbb{N}$ and let $n := m \cdot \binom{d+1}{2} + 2$. Then*

$$CC_{\mathbb{R}}(\textsc{PyrGen}(m,d)) \le CC_{\mathbb{R}}(R_{\textsc{Gen}_n}).$$

Proof. We prove that any protocol solving the Karchmer-Wigderson game for $\textsc{Gen}_n$ can be used to solve the $\textsc{PyrGen}(m,d)$ game. Recall that $\textsc{PyrGen}(m,d)$ is a $DART(m, \binom{d+1}{2})$ game, so the two players I and II receive inputs respectively of the form $(x_{1,1}, \ldots, x_{d,d})$ where $x_{i,j} \in [m]$ for all $(i,j) \in Pyr_d$ and $(y_{1,1}, \ldots, y_{d,d})$ where $y_{i,j} \in \{0,1\}^m$ for all $(i,j) \in Pyr_d$.

From their respective inputs for the $\textsc{PyrGen}(m,d)$ game, Player I and II compute respectively a minterm $t^x_{a,b,c}$ and a maxterm $t^y_{a,b,c}$, for $\textsc{Gen}_n$ and then they play the Karchmer-Wigderson game applying the protocol P.

As in [25] we consider fixed the element 1 as a bottom generator and the element n as the element we want to generate. We interpret the remaining $n-2 = \binom{d+1}{2}m$ elements between 2 and $n-1$ as triples (i,j,k), where $(i,j) \in Pyr_d$ and $k \in [m]$.

Now player I computes from his input $(x_{1,1}, \ldots, x_{d,d})$ an input $t^x_{a,b,c}$ for $\textsc{Gen}_n$ such that $\textsc{Gen}_n(t^x_{a,b,c}) = 1$ by setting the following (recall that $a, b \vdash c$ means $t_{a,b,c} = 1$):

$$\begin{aligned} &1, 1 \vdash g_{d,j} && \text{for } 1 \le j \le d \\ &g_{1,1}, g_{1,1} \vdash n && \\ &g_{i+1,j}, g_{i+1,j+1} \vdash g_{i,j} && \text{for } (i,j) \in Pyr_{d-1} \end{aligned}$$

where $g_{i,j} := (i, j, x_{i,j}) \in \{2, \ldots, n-1\}$ and all the other bits $t^x_{a,b,c} = 0$. This completely determines $t^x_{a,b,c}$, and obviously $\textsc{Gen}_n(t^x_{a,b,c}) = 1$ since we have forced a generation of n (in a pyramidal form).

Likewise Player II computes from his input $(y_{1,1}, \dots, y_{d,d})$ a coloring col of the elements from $[n]$ by setting $col(1) = 0$, $col(n) = 1$ and $col((i,j,k)) = y_{i,j}(k)$ (the k-th bit of $y_{(i,j)}$). From this coloring, he computes an input $t^y_{a,b,c}$ by setting $t^y_{a,b,c} = 1$ iff it is not the case that $col(c) = 1$ and $col(a) = col(b) = 0$. Obviously $\text{GEN}_n(t^y_{a,b,c}) = 0$.

Running the protocol for the Karchmer-Wigderson game for GEN_n now yields a triple (a,b,c) such that $t^x_{a,b,c} = 1$ and $t^y_{a,b,c} = 0$. By definition of t^y, this means that $col(a) = col(b) = 0$ and $col(c) = 1$, and by definition of t^x one of the following cases must hold:

- $a = b = 1$ and $c = g_{d,j}$ for some $j \le d$. By definition of col, $y_{d,j}(x_{d,j}) = 1$.
- $c = n$ and $a = b = g_{1,1}$. In this case, $y_{1,1}(x_{1,1}) = 0$.
- $a = g_{i+1,j}$, $b = g_{i+1,j+1}$ and $c = g_{i,j}$. Then we have $y_{i,j}(x_{i,j}) = 1$, and $y_{i+1,j}(x_{i+1,j}) = y_{i+1,j+1}(x_{i+1,j+1}) = 0$.

In either case, the players have solved $\text{PYRGEN}(m,d)$ without any additional communication. $\square$

3.2 Relation between Structured Complexity and Real Communication Complexity

We prove here the following general Theorem for DART games.

Theorem 6. *Let $m, k \in \mathbb{N}$. For every relation $R \in \text{DART}(m,k)$, where $m \ge k^{14}$,*

$$CC_{\mathbb{R}}(R) \ge SC(R) \cdot \Omega(\log m) .$$

We first need some combinatorial notions from [25] and some lemmas. Let $A \subseteq [m]^k$ and $1 \le j \le k$. For $x \in [m]^{k-1}$, let $\deg_j(x, A)$ be the number of $\xi \in [m]$ such that $(x_1, \dots, x_{j-1}, \xi, x_j, \dots, x_{k-1}) \in A$. Then we define

$$\begin{aligned} A[j] &:= \left\{ x \in [m]^{k-1} \ : \ \deg_j(x, A) > 0 \right\} \\ AVDEG_j(A) &:= \frac{|A|}{|A[j]|} \\ MINDEG_j(A) &:= \min_{x \in A[j]} \deg_j(x, A) \\ Thickness(A) &:= \min_{1 \le j \le k} MINDEG_j(A) . \end{aligned}$$

The following lemmas about these notions were proved in [25]:

Lemma 7 ([25]). *For every $A' \subseteq A$ and $1 \le j \le k$,*

$$AVDEG_j(A') \ge \frac{|A'|}{|A|} AVDEG_j(A) \tag{1}$$

$$Thickness(A[j]) \ge Thickness(A) \tag{2}$$

Lemma 8 ([25]). *Let $0 < \delta < 1$ be given. If for every $1 \le j \le k$, $AVDEG_j(A) \ge \delta m$, then for every $\alpha > 0$ there is $A' \subseteq A$ with $|A'| \ge (1 - \alpha)|A|$ and*

$$Thickness(A') \ge \frac{\alpha \delta m}{k} .$$

In particular, setting $\alpha = \frac{1}{2}$ and $\delta = 4m^{-\frac{1}{14}}$, we get

Corollary 9. *If $m \geq k^{14}$ and for every $1 \leq j \leq k$, $AVDEG_j(A) \geq 4m^{\frac{13}{14}}$, then there is $A' \subseteq A$ with $|A'| \geq \frac{1}{2}|A|$ and $Thickness(A) \geq m^{\frac{11}{14}}$.*

For a relation $R \in \mathrm{DART}(m,k)$, $A \subseteq X$ and $B \subseteq Y$, let $CC_{\mathbb{R}}(R,A,B)$ be the real communication complexity of R restricted to $A \times B$.

Definition ((α,β,ℓ)-game). *Let $m \in \mathbb{N}$, $m \geq k^{14}$. Let $A \subseteq X$ and $B \subseteq Y$. A triple (R,A,B) is called an (α,β,ℓ)-game if the following conditions hold:*

1. *$R \in \mathrm{DART}(m,k)$,*
2. *$SC(R) \geq \ell$,*
3. *$|A| \geq 2^{-\alpha}|X|$ and $|B| \geq 2^{-\beta}|Y|$,*
4. *$Thickness(A) \geq m^{\frac{11}{14}}$.*

The following lemma and its proof are slightly different from the corresponding lemma in [25], because we use the strong notion of real communication complexity where [25] use ordinary communication complexity. The modification we apply is analogous to that introduced by Johannsen [18] to improve the result of Karchmer and Wigderson [20] to the case of real communication complexity. This modification will affect the proof of the first point of the next lemma. We include a proof of the second part for completeness.

Lemma 10. *For every $\alpha, \ell \geq 0$ and $0 \leq \beta \leq m^{\frac{1}{7}}$, $m \geq 1000^{14}$ and every (α,β,ℓ)-game (R,A,B),*

1. *if for every $1 \leq j \leq k$, $AVDEG_j(A) \geq 8m^{\frac{13}{14}}$, then there is an $(\alpha+2,\beta+1,\ell)$-game (R',A',B') with*
$$CC_{\mathbb{R}}(R',A',B') \leq CC_{\mathbb{R}}(R,A,B) - 1 \ .$$
2. *if $\ell \geq 1$ and for some $1 \leq j \leq k$, $AVDEG_j(A) < 8m^{\frac{13}{14}}$, then there is an $(\alpha + 3 - \frac{\log m}{14}, \beta + 1, \ell - 1)$-game (R',A',B') with*
$$CC_{\mathbb{R}}(R',A',B') \leq CC_{\mathbb{R}}(R,A,B) \ .$$

Proof of Lemma 10 (Part 1). Let (R,A,B) be a (α,β,ℓ)-game. First we show that $CC_{\mathbb{R}}(R,A,B) \neq 0$. Assume by contradiction that $CC_{\mathbb{R}}(R,A,B) = 0$. Then the players have no need to transmit information to solve R. This means that the answer to the game is implicit in the domain $A \times B$ and therefore by requirement (4) of DART games there is a term in the DNF tautology F_R defining R that is satisfied for every $(x,y) \in A \times B$. Therefore there is at least a coordinate j, $1 \leq j \leq k$ such that $y_j(x_j)$ is constant (i.e., is always 0 or always 1) . If γ denotes the number of possible different values of x_j in elements of A, then this implies that $|B| \leq 2^{mk-\gamma}$. On the other hand, $|B| \geq 2^{mk-\beta}$, hence it follows that $\beta \geq \gamma$, which is a contradiction since $\beta \leq m^{\frac{1}{7}}$, whereas $AVDEG_j(A) \geq 8m^{\frac{13}{14}}$ implies $\gamma \geq 8m^{\frac{13}{14}}$.

Let an optimal real communication protocol solving R restricted to $A \times B$ be given. For $a \in A$ and $b \in B$, let ρ_a and σ_b be the real numbers played by I and II in the first round on input a and b, respectively. W.l.o.g. we can assume that these are $|A| + |B|$ pairwise distinct real numbers.

Now consider a $\{0,1\}$-matrix of size $|A| \times |B|$ with columns indexed by the ρ_a and rows indexed by the σ_b, both in increasing order, and where the entry in position (ρ_a, σ_b) is 1 if $\rho_a > \sigma_b$ and 0 if $\rho_a \leq \sigma_b$. Thus this entry determines the outcome of the first round, when these numbers are played. It is now obvious that either the upper right quadrant or the lower left quadrant must form a monochromatic rectangle.

Hence there are $A^\circ \subseteq A$ and $B' \subseteq B$ with $|A^\circ| \geq \frac{1}{2}|A|$ and $|B'| \geq \frac{1}{2}|B|$ such that R restricted to $A^\circ \times B'$ can be solved by a protocol with one round fewer than the original protocol. This means that $CC_{\mathbb{R}}(R, A^\circ, B') \leq CC_{\mathbb{R}}(R, A, B) - 1$. By Equation (1) of Lemma 7, $AVDEG_j(A^\circ) \geq 4m^{\frac{13}{14}}$ for every $1 \leq j \leq k$, hence by Corollary 9 there is $A' \subseteq A^\circ$ with $|A'| \geq \frac{1}{2}|A^\circ| \geq \frac{1}{4}|A|$ and $Thickness(A') \geq m^{\frac{11}{14}}$. Thus (R, A', B') is an $(\alpha+2, \beta+1, \ell)$-game, moreover, since $A' \subseteq A^\circ$, we have that $CC_R(R, A', B') \leq CC_R(R, A^\circ, B')$, from which the Lemma follows.

(Part 2) We proceed like in the proof of the corresponding lemma of [25], with the numbers slightly adjusted. Assume without loss of generality that k is the coordinate for which $AVDEG_k(A) < 8m^{\frac{13}{14}}$. Let R_0 and R_1 be the restrictions of R in which the k-th coordinate $e_k = y_k(x_k)$ is fixed to 0 and 1, respectively. Obviously, R_0 and R_1 are $DART(m, k-1)$ relations, and therefore at least one of $SC(R_0)$ and $SC(R_1)$ is at least $\ell - 1$. Assume without loss of generality that $SC(R_0) \geq \ell - 1$. We will prove that there are two sets $A' \subseteq [m]^{k-1}$ and $B' \subseteq (\{0,1\}^m)^{k-1}$ such that the following properties hold:

$$|A'| \geq \frac{m^{k-1}}{2^{\alpha+3-\frac{\log m}{14}}} \tag{3}$$

$$|B'| \geq \frac{2^{m(k-1)}}{2^{\beta+1}} \tag{4}$$

$$Thickness(A') \geq m^{\frac{11}{14}} \tag{5}$$

$$CC_{\mathbb{R}}(R_0, A', B') \leq CC_{\mathbb{R}}(R, A, B) \tag{6}$$

This means that there is a $(\alpha+3-\frac{\log m}{14}, \beta+1, \ell-1)$-game (R_0, A', B') such that $CC_{\mathbb{R}}(R_0, A', B') \leq CC_{\mathbb{R}}(R, A, B)$ and this proves part 2 of Lemma 10.

Given any set $U \subset [m]$ consider the sets $A_U \subseteq [m]^{k-1}$ and $B_U \subseteq (\{0,1\}^m)^{k-1}$ associated to the set U by the following definition of [25]:

- $(x_1, \ldots x_{k-1}) \in A_U$ iff there is an $u \in U$ such that $(x_1, \ldots x_{k-1}, u) \in A$;
- $(y_1, \ldots y_{k-1}) \in B_U$ iff there is a $w \in \{0,1\}^m$ such that $w(u) = 0$ for all $u \in U$ and $(y_1, \ldots y_{k-1}, w) \in B$.

The following two claims can be proved exactly as the corresponding Claims of [25] and we omit their proof.

Claim 11. *For a random set U of size $m^{\frac{5}{14}}$, with $m \geq 1000^{14}$, we have that*

$$Prob_U\big[A_U = A[k]\big] \geq \frac{3}{4}.$$

Claim 12. *For a random set U of size $m^{\frac{5}{14}}$, with $m \geq 1000^{14}$ we have that*

$$Prob_U\big[|B_U| \geq \frac{|B|}{2^{m+1}}\big] \geq \frac{3}{4}.$$

Moreover it is immediate to see that the same reduction used in Claim 6.3 of [25] also works for the case of real communication complexity. Therefore we get:

Claim 13. *For every set $U \subset [m]$*

$$CC_{\mathbb{R}}(R_0, A_U, B_U) \le CC_{\mathbb{R}}(R, A, B) \ .$$

Take a random set U which with probability greater than $\frac{1}{2}$, satisfies both the properties of Claim 11 and Claim 12, and define $A' := A_U$ and $B' := B_U$. This means that with probability at least $\frac{1}{2}$ both $A' = A[k]$ and $|B'| \ge \frac{|B|}{2^{m+1}}$ hold.

Recall that $\frac{|A|}{|A'|} = \frac{|A|}{|A[k]|} = AVDEG_k(A)$ and that, by hypothesis on Part 2 of the lemma $|\,AVDEG_k(A)| \le 8m^{\frac{13}{14}}$. Therefore we have that

$$|A'| \ge \frac{|A|}{8m^{\frac{13}{14}}} \ge \frac{m^k}{2^{\alpha} 8m^{\frac{13}{14}}} = \frac{m^{k-1}}{2^{\alpha+3-\frac{\log m}{14}}} \ .$$

This proves (3). For (4) observe that by Claim 12 we have

$$|B'| \ge \frac{|B|}{2^{m+1}} \ge \frac{2^{mk}}{2^{\beta} 2^{m+1}} = \frac{2^{m(k-1)}}{2^{\beta+1}} \ .$$

The property (5) follows directly from Lemma 7 (2), and finally (6) follows from Claim 13. ☐

3.2.1 Proof of Theorem 6

Proof. Let $k \in \mathbb{N}$, $k \ge 1000$. We prove that for any $\alpha, \beta, \ell, m \ge 0$, with $\beta \le m^{1/7}$, $\ell \ge 1$, and $m \ge k^{14}$, every (α, β, ℓ)-game (R, A, B) is such that

$$CC_{\mathbb{R}}(R, A, B) \ge \ell \cdot \Big(\frac{\log m}{42} - \frac{4}{3}\Big) - \frac{\alpha + \beta}{3} \ . \tag{7}$$

Observe that by Definition of (α, β, ℓ)-game, when $\alpha = \beta = 0$ we have that $A = X$ and $B = Y$. Therefore $CC_{\mathbb{R}}(R, A, B) = CC_{\mathbb{R}}(R)$. Moreover the right side of Equation 7 reduces to $\ell \cdot \Omega(\log m)$. Since by the same Definition $\ell \le SC(R)$, for $\alpha = \beta = 0$ we get the claim of the theorem:

$$CC_{\mathbb{R}}(R) \ge SC(R) \cdot \Omega(\log m)$$

To prove Equation 7, we proceed by induction on $\ell \ge 1$ and $\beta \le m^{1/7}$. In the base case $\ell < 1$ (that is $\ell = 0$) and $\beta > m^{\frac{1}{7}}$, the inequality (7) is trivial, since the right hand side gets negative for large m. In the inductive step consider (R, A, B) be an (α, β, ℓ)-game, and assume that (7) holds for all (α', β', ℓ')-games with $\ell' \le \ell$ and $\beta' > \beta$. For sake of contradiction, suppose that $CC_{\mathbb{R}}(R, A, B) < \ell \cdot \Big(\frac{\log m}{42} - \frac{4}{3}\Big) - \frac{\alpha+\beta}{3}$. Then either for every $1 \le j \le k$, $AVDEG_j(A) \ge 8m^{\frac{13}{14}}$, and Lemma 10 gives an $(\alpha + 2, \beta + 1, \ell)$-game (R', A', B') with

$$\begin{aligned} CC_{\mathbb{R}}(R', A', B') &\le CC_{\mathbb{R}}(R, A, B) - 1 \\ &< \ell \cdot \Big(\frac{\log m}{42} - \frac{4}{3}\Big) - \frac{(\alpha + 2) + (\beta + 1)}{3} \ , \end{aligned}$$

or for some $1 \le j \le k$, $AVDEG_j(A) < 8m^{\frac{13}{14}}$, then Lemma 10 gives an $(\alpha + 3 - \frac{\log m}{14}, \beta + 1, \ell - 1)$-game (R', A', B') with

$$\begin{aligned} CC_{\mathbb{R}}(R', A', B') &< \ell \cdot \Big(\frac{\log m}{42} - \frac{4}{3}\Big) - \frac{\alpha + \beta}{3} \\ &= (\ell - 1) \cdot \Big(\frac{\log m}{42} - \frac{4}{3}\Big) - \frac{(\alpha + 3 - \frac{\log m}{14}) + (\beta + 1)}{3} \ , \end{aligned}$$

both contradicting the assumption. ☐

3.3 Consequences for Monotone Real Circuits

As a first corollary to Theorem 6, we observe that for DART games, real communication protocols are no more powerful than deterministic communication protocols.

Corollary 14. *Let $m, k \in \mathbb{N}$. For $R \in \mathrm{DART}(m,k)$ with $m \geq k^{14}$,*

$$CC_{\mathbb{R}}(R) = \Theta(CC(R)) .$$

Proof. $CC(R) \geq CC_{\mathbb{R}}(R) \geq SC(R) \cdot \Omega(\log m) \geq \Omega(CC(R))$. □

From Theorem 3 we obtain consequences for monotone real circuits analogous to those obtained in [25] for monotone boolean circuits. An immediate consequences of Theorem 3 and Lemma 1 is the following

Theorem 15. *Any tree-like monotone real circuit computing the boolean function $\textsc{Gen}_n$ must have size $2^{\Omega(n^{\epsilon})}$, for some $\epsilon > 0$.*

Definition (Pyramidal Generation). *Let $\vec{t}$ be an input to $\textsc{Gen}_n$. We say that n is generated in a depth d pyramidal fashion by $\vec{t}$ if there is a mapping $m : Pyr_d \to [n]$ such that the following hold (recall that $a, b \vdash c$ means $t_{a,b,c} = 1$):*

$$\begin{array}{ll} 1, 1 \vdash m(d, j) & \text{for every } j \leq d \\ m(i+1, j), m(i+1, j+1) \vdash m(i, j) & \text{for every } (i,j) \in Pyr_{d-1} \\ m(1,1), m(1,1) \vdash n & \end{array}$$

We can obtain an analogous of Theorem 15 also for the simpler case in which the generation is restricted to be only in a pyramidal form.

Corollary 16. *Every monotone real formula that outputs 1 on every input to $\textsc{Gen}_n$ for which n is generated in a depth d pyramidal fashion, and outputs 0 on all inputs where $\textsc{Gen}_n$ is 0, has to be of size $\Omega(2^{n^{\epsilon}})$, for some $\epsilon > 0$.*

Proof. To simplify, let $Pyrgen_n$ be any monotone boolean function that outputs 1 on every input to $\textsc{Gen}_n$ for which n is generated in a depth d pyramidal fashion, and outputs 0 on all inputs where $\textsc{Gen}_n$ is 0. Note that there are many such functions, since the output is not specified in the case where n can be generated, but not in a depth d pyramidal fashion. Observe that in Lemma 5, Player I builds from his input an input for $\textsc{Gen}_n$ which enforces a depth d pyramidal generation. So the proof of Lemma 6 also shows that $CC_{\mathbb{R}}(\textsc{PyrGen}(m,d)) \leq CC_{\mathbb{R}}(R_{Pyrgen_n})$. Lemma 4 and Theorem 6 then imply that $CC_{\mathbb{R}}(R_{Pyrgen_n}) \geq \Omega(n^{\epsilon})$, for some $\epsilon > 0$. Finally Lemma 1 gives the statement of the corollary. □

The other consequences drawn from Theorem 6 and Lemma 4 in [25] apply to monotone real circuits as well, e.g. we just state without proof the following result:

Theorem 17. *There are constants $0 < \epsilon, \gamma < 1$ such that for every function $d(n) \leq n^{\epsilon}$, there is a family of monotone functions $f_n : \{0,1\}^n \to \{0,1\}$ that can be computed by monotone boolean circuits of size $n^{O(1)}$ and depth $d(n)$, but cannot be computed by monotone real circuits of depth less than $\gamma \cdot d(n)$.*

The method also gives a simpler proof of the lower bounds in [18], in the same way as [25] simplifies the lower bound of [20].

4 Separation between Tree-like and Dag-like Versions of Resolution and Cutting Planes

We will define an unsatisfiable CNF-formula $Gen(\vec{p},\vec{q}) \wedge Col(\vec{p},\vec{r})$ that fulfills the assumptions of Theorem 2, so any CP refutation of it can be transformed into a monotone real circuit, and any tree-like CP refutation into a monotone real formula. This circuit (or formula) is similar in size to the original CP refutation. We will show that it computes a boolean function related to GEN_n: It outputs 1 if n is generated in a pyramidal way, so the exponential size lower bound in corollary 16 imply an exponential size lower bound for tree-like CP refutations of $Gen(\vec{p},\vec{q}) \wedge Col(\vec{p},\vec{r})$. Besides we give a polynomial-size resolution refutation of $Gen(\vec{p},\vec{q}) \wedge Col(\vec{p},\vec{r})$. As CP refutations are shorter than resolution refutations, we get the separation between tree-like CP and CP, in fact also a separation of tree-like resolution from resolution.

Let n and d be natural numbers whose values are to be fixed. Recall that the set Pyr_d is $\{ (i,j) : 1 \le j \le i \le d \}$. The vector $\vec{p}$, that is, the variables $p_{a,b,c}$ for $a,b,c \in [n]$, represent the input to GEN_n.

The set of clauses $Gen(\vec{p},\vec{q})$ is designed to be satisfiable if in the input $\vec{p}$, n is generated in a depth d pyramidal fashion. To this end, the variables $q_{i,j,a}$ for $(i,j) \in Pyr_d$ and $a \in [n]$ encode a mapping $m : Pyr_d \to [n]$ as in the definition of pyramidal generation in section 3.3, where $q_{i,j,a}$ is intended to express that $m(i,j) = a$.

On the other hand, the set of clauses $Col(\vec{p},\vec{r})$ is designed to be satisfiable if for the input $\vec{p}$, $\text{GEN}_n(\vec{p}) = 0$. To achieve this, the variables r_a for $a \in [n]$ encode a coloring of the elements of $[n]$ such that element 1 is colored 0, element n is colored 1, and the elements colored 0 are closed under generation, i.e., if a and b are colored 0 and $a,b \vdash c$, then c is also colored 0.

The set $Gen(\vec{p},\vec{q})$ is given by (8) - (11), and $Col(\vec{p},\vec{r})$ by (12) - (14).

$$\bigvee_{1 \le a \le n} q_{i,j,a} \qquad \text{for } (i,j) \in Pyr_d \tag{8}$$

$$\bar{q}_{d,j,a} \vee p_{1,1,a} \qquad \text{for } 1 \le j \le d \text{ and } a \in [n] \tag{9}$$

$$\bar{q}_{1,1,a} \vee p_{a,a,n} \qquad \text{for } a \in [n] \tag{10}$$

$$\bar{q}_{i+1,j,a} \vee \bar{q}_{i+1,j+1,b} \vee \bar{q}_{i,j,c} \vee p_{a,b,c} \qquad \text{for } (i,j) \in Pyr_{d-1} \text{ and } a,b,c \in [n] \tag{11}$$

$$\bar{p}_{1,1,a} \vee \bar{r}_a \qquad \text{for } a \in [n] \tag{12}$$

$$\bar{p}_{a,a,n} \vee r_a \qquad \text{for } a \in [n] \tag{13}$$

$$r_a \vee r_b \vee \bar{p}_{a,b,c} \vee \bar{r}_c \qquad \text{for } a,b,c \in [n] \tag{14}$$

Obviously, $Gen(\vec{p},\vec{q}) \wedge Col(\vec{p},\vec{r})$is unsatisfiable. Observe that the variables $\vec{p}$ occur only positively in $Gen(\vec{p},\vec{q})$ and only negatively in $Col(\vec{p},\vec{r})$, thus Theorem 2 yields an interpolating monotone real formula $C(\vec{p})$.

Now if for a assignment $\vec{t}$ to the variables $\vec{p}$, n is generated in a depth d pyramidal fashion, then $Gen(\vec{t},\vec{q})$ is satisfiable by setting the values of the variables $q_{i,j,a}$ according to the mapping m. Therefore $Col(\vec{t},\vec{r})$ must be unsatisfiable, and thus $C(\vec{t}) = 1$.

If on the other hand $\text{GEN}_n(\vec{t}) = 0$, then $Col(\vec{t},\vec{r})$ can be satisfied by assigning the color 0 to precisely those elements that can be generated in $\vec{t}$. Therefore $Gen(\vec{t},\vec{q})$ must be unsatisfiable, and so $C(\vec{t}) = 0$.

Thus $C(\vec{p})$ is a monotone real formula satisfying the assumptions of Corollary 16, therefore it has to be of size $2^{\Omega(n^\epsilon)}$. Note that Theorem 2 gives no information about the behavior of $C(\vec{t})$ in the case where $Gen(\vec{t},\vec{q})$ and $Col(\vec{t},\vec{r})$ are both unsatisfiable, thus we need Corollary 16 in precisely the general form in which it is stated. From the size bounds in Theorem 2 we now obtain:

Theorem 18. *Every tree-like CP refutation of the clauses $Gen(\vec{p},\vec{q}) \cup Col(\vec{p},\vec{r})$ has to be of size $2^{\Omega(n^{\epsilon})}$, for some $\epsilon > 0$.*

On the other hand, there are polynomial size dag-like resolution refutations of these clauses.

Theorem 19. *There are (dag-like) resolution refutations of size $n^{O(1)}$ of the clauses $Gen(\vec{p},\vec{q}) \cup Col(\vec{p},\vec{r})$.*

Proof. First we resolve clauses (9) and (12) to get

$$\bar{q}_{d,j,c} \vee \bar{r}_c \tag{15}$$

for $1 \le j \le d$ and $1 \le c \le n$.

Now we want to derive $\bar{q}_{i,j,c} \vee \bar{r}_c$ for every $(i,j) \in Pyr_d$ and $1 \le c \le n$, by induction on i downward from d to 1. The induction base is just (15).

Now by induction we have

$$\bar{q}_{i+1,j,a} \vee \bar{r}_a \quad \text{and} \quad \bar{q}_{i+1,j+1,b} \vee \bar{r}_b \ ,$$

we resolve them against (14) to get $\bar{q}_{i+1,j,a} \vee \bar{q}_{i+1,j+1,b} \vee \bar{p}_{a,b,c} \vee \bar{r}_c$ for $1 \le a,b,c \le n$ and then resolve them against (11) and get

$$\bar{q}_{i+1,j,a} \vee \bar{q}_{i+1,j+1,b} \vee \bar{q}_{i,j,c} \vee \bar{r}_c$$

for every $1 \le a, b \le n$. All of these are then resolved against two instances of (8), and we get the desired $\bar{q}_{i,j,c} \vee \bar{r}_c$ for every $1 \le c \le n$.

Finally, we have in particular $\bar{q}_{1,1,a} \vee \bar{r}_a$ for every $1 \le c \le n$. We resolve them with (13) and get $\bar{q}_{1,1,a} \vee \bar{p}_{a,a,n}$ for every $1 \le a \le n$. These are resolved with (10) to get $\bar{q}_{1,1,a}$ for every $1 \le a \le n$. Finally, this clause is resolved with another instance of (10) (the one with $i = j = 1$) to get the empty clause. ☐

It is easy to check that the above refutation is a negative resolution refutation. The following corollary is an easy consequence of the above theorems and known simulation results.

Corollary 20. *The clauses $Gen(\vec{p},\vec{q}) \cup Col(\vec{p},\vec{r})$ exponentially separate tree-like resolution from dag-like resolution, if fact it separates tree-like resolution from dag-like negative resolution. They also separate tree-like Cutting Planes from dag-like Cutting Planes.*

The resolution refutation of $Gen(\vec{p},\vec{q}) \cup Col(\vec{p},\vec{r})$ that appears in the proof of Theorem 19 is not regular. We do not know whether $Gen(\vec{p},\vec{q}) \cup Col(\vec{p},\vec{r})$ has polynomial size regular resolution refutations. To obtain a separation between tree-like resolution and regular resolution we will modify the clauses $Col(\vec{p},\vec{r})$.

4.1 Separation of Tree-like CP from Regular Resolution

The clauses $Col(\vec{p},\vec{r})$ are modified (and the modification called $RCol(\vec{p},\vec{r})$), so that $Gen(\vec{p},\vec{q}) \cup RCol(\vec{p},\vec{r})$ allow small regular resolutions, but in such a way that the lower bound proof still applies. We replace the variables r_a by $r_{a,i,D}$ for $a \in [n]$, $1 \le i \le d$ and $D \in \{L, R\}$, giving the coloring of element a, with auxiliary indices i being a row in the pyramid and D distinguishing whether an element is used as a left or right predecessor in the generation process.

The set $RCol(\vec{p},\vec{r})$ is defined as follows:

$$\bar{p}_{1,1,a} \vee \bar{r}_{a,d,D} \qquad \text{for } a \in [n] \text{ and } D \in \{L,R\} \tag{16}$$

$$\bar{p}_{a,a,n} \vee r_{a,1,D} \qquad \text{for } a \in [n] \text{ and } D \in \{L,R\} \tag{17}$$

$$r_{a,i+1,L} \vee r_{b,i+1,R} \vee \bar{p}_{a,b,c} \vee \bar{r}_{c,i,D} \qquad \text{for } i < d,\ a,b,c \in [n] \text{ and } D \in \{L,R\} \tag{18}$$

$$\bar{r}_{a,i,D} \vee r_{a,i,\bar{D}} \qquad \text{for } 1 \le i \le d \text{ and } D \in \{L,R\} \tag{19}$$

$$\bar{r}_{a,i,D} \vee r_{a,j,D} \qquad \text{for } 1 \le i,j \le d \text{ and } D \in \{L,R\} \tag{20}$$

Due to the clauses (19) and (20), the variables $r_{a,i,D}$ are equivalent for all values of the auxiliary indices i, D. Hence a satisfying assignment for $RCol(\vec{p},\vec{r})$ still codes a coloring of $[n]$ such that elements a with $1,1 \vdash a$ are colored 0, the elements b with $b,b \vdash n$ are colored 1, and the 0-colored elements are closed under generation. Hence if $RCol(\vec{t},\vec{r})$ is satisfiable, then $\textsc{Gen}(\vec{t}) = 0$.

Hence any interpolant for the clauses $Gen(\vec{p},\vec{q}) \cup RCol(\vec{p},\vec{r})$ satisfies the assumptions of Corollary 16, and we can conclude

Theorem 21. *Tree-like CP refutations of the clauses $Gen(\vec{p},\vec{q}) \cup RCol(\vec{p},\vec{r})$ have to be of size $2^{\Omega(n^\epsilon)}$.*

On the other hand, we have the following upper bound on (dag-like) regular resolution refutations of these clauses:

Theorem 22. *There are (dag-like) regular resolution refutations of the clauses $Gen(\vec{p},\vec{q}) \cup RCol(\vec{p},\vec{r})$ of size $n^{O(1)}$.*

Proof. First we resolve clauses (9) and (16) to get

$$\bar{q}_{d,j,a} \vee \bar{r}_{a,d,D} \tag{21}$$

for $1 \le j \le d$, $1 \le a \le n$ and $D \in \{L,R\}$. Next we resolve (10) and (17) to get

$$\bar{q}_{1,1,a} \vee r_{a,1,D} \tag{22}$$

for $1 \le a \le n$ and $D \in \{L,R\}$. Finally, from (11) and (18) we obtain

$$\bar{q}_{i+1,j,a} \vee \bar{q}_{i+1,j+1,b} \vee \bar{q}_{i,j,c} \vee r_{a,i+1,L} \vee r_{b,i+1,R} \vee \bar{r}_{c,i,D} \tag{23}$$

for $1 \le j \le i < d$, $1 \le a,b,c \le n$ and $D \in \{L,R\}$.

Now we want to derive $\bar{q}_{i,j,a} \vee \bar{r}_{a,i,D}$ for every $(i,j) \in Pyr_d$, $1 \le a \le n$ and $D \in \{L,R\}$, by induction on i downward from d to 1. The induction base is just (21).

For the inductive step, resolve (23) against the clauses

$$\bar{q}_{i+1,j,a} \vee \bar{r}_{a,i+1,L} \quad \text{and} \quad \bar{q}_{i+1,j+1,b} \vee \bar{r}_{b,i+1,R}\ ,$$

which we have by induction, to give

$$\bar{q}_{i+1,j,a} \vee \bar{q}_{i+1,j+1,b} \vee \bar{q}_{i,j,c} \vee \bar{r}_{c,i,D}$$

for every $1 \le a,b \le n$. All of these are then resolved against two instances of (8), and we get the desired $\bar{q}_{i,j,c} \vee \bar{r}_{c,i,D}$.

Finally, we have in particular $\bar{q}_{1,1,a} \vee \bar{r}_{a,1,L}$, which we resolve against (22) to get $\bar{q}_{1,1,a}$ for every $a \le n$. From these and an instance of(8) we get the empty clause. $\square$

Note that the refutation given in the proof of Theorem 22 is actually a ordered refutation: It respects the following elimination order

$$\begin{array}{l} p_{1,1,1} \cdots p_{n,n,n} \\ r_{1,d,L} \;\; r_{1,d,R} \;\; \cdots \;\; r_{n,d,L} \;\; r_{n,d,R} \\ q_{1,d,1} \cdots q_{1,d,n} \cdots q_{d,d,1} \cdots q_{d,d,n} \\ r_{1,d-1,L} \cdots r_{n,d-1,R} \;\; q_{1,d-1,1} \cdots q_{d-1,d-1,n} \\ \vdots \\ r_{1,1,L} \;\; r_{1,1,R} \;\; q_{1,1,1} \cdots q_{1,1,n} \,. \end{array}$$

Corollary 23. *The clauses $Gen(\vec{p},\vec{q}) \cup RCol(\vec{p},\vec{r})$ exponentially separate tree-like resolution from ordered resolution, therefore they also separate exponentially tree-like resolution from regular resolution.*

5 Lower Bound for Ordered Resolution

Goerdt [13] showed that ordered resolution is strictly weaker than unrestricted resolution, by giving a superpolynomial lower bound (of the order $\Omega(n^{\log\log n})$) for ordered resolutions of a certain family of clauses, which on the other hand has polynomial size unrestricted resolution refutations. In this section we improve this separation to an exponential one, in fact, we give an exponential separation of ordered resolution from negative resolution.

To simplify the exposition, we apply the method of [13] to a set of clauses $SP_{n,m}$ expressing a combinatorial principle that we call the *String-of-Pearls* principle: From a bag of m pearls, which are colored red and blue, n pearls are chosen and placed on a string. The string-of-pearls principle $SP_{n,m}$ says that, if the first pearl is red and the last one is blue, then there must be a blue pearl next to a red pearl somewhere on the string.

$SP_{n,m}$ is given by an unsatisfiable set of clauses in variables $p_{i,j}$ and q_j for $i \in [n]$ and $j \in [m]$, where $p_{i,j}$ is intended to say that pearl j is at position i on the string, and q_j means that pearl j is colored blue. The clauses forming $SP_{n,m}$ are:

$$\bigvee_{j=1}^{m} p_{i,j} \qquad i \in [n] \tag{24}$$

$$\bar{p}_{i,j} \vee \bar{p}_{i,j'} \qquad i \in [n],\ j,j' \in [m],\ j \neq j' \tag{25}$$

$$\bar{p}_{i,j} \vee \bar{p}_{i',j} \qquad i,i' \in [n],\ j \in [m],\ i \neq i' \tag{26}$$

These first three sets of clauses express that there is a unique pearl at each position.

$$\bar{p}_{1,j'} \vee \bar{q}_{j'} \qquad j' \in [m] \tag{27}$$

$$\bar{p}_{n,j} \vee q_j \qquad j \in [m] \tag{28}$$

$$\bar{p}_{i,j} \vee \bar{p}_{(i+1),j'} \vee q_j \vee \bar{q}_{j'} \qquad 1 \leq i < n,\ j,j' \in [m],\ j \neq j' \tag{29}$$

These last three sets of clauses express that the first pearl is red, the last one is blue, and that a pearl sitting next to a red pearl is also colored red. The clauses $SP_{n,m}$ are a modified and simplified version of the clauses related to the *st*-connectivity problem that were introduced by Clote and Setzer [8].

Proposition 24. *The clauses $SP_{n,m}$ have negative resolution refutations of size $O(nm^2)$.*

Proof. For every $i \in [n]$, we will derive the clauses $\bar{p}_{i,j} \vee \bar{q}_j$ for $j \in [m]$ from $SP_{n,m}$ by a negative resolution derivation. For $i = 1$, these are the clauses (27) from $SP_{n,m}$. Inductively, assume we have derived $\bar{p}_{i,j'} \vee \bar{q}_{j'}$ for $j' \in [m]$, and we want to derive $\bar{p}_{(i+1),j} \vee \bar{q}_j$ from these.

Consider the clauses (29) of the form $\bar{p}_{i,j'} \vee \bar{p}_{(i+1),j} \vee q_{j'} \vee \bar{q}_j$ for $j' \in [m]$. Using the inductive assumption, we derive from these the clauses $\bar{p}_{i,j'} \vee \bar{p}_{(i+1),j} \vee \bar{q}_j$ for $j' \in [m]$. Note that these are negative clauses.

By a derivation of length m, we obtain $\bar{p}_{(i+1),j} \vee \bar{q}_j$ from these and the clause $\bigvee_{j' \in [m]} p_{i,j'}$ from $SP_{n,m}$. The whole derivation is of length $O(m)$, and we need m of them, giving a total length of $O(m^2)$ for the induction step.

We end up with a derivation of the clauses $\bar{p}_{n,j} \vee \bar{q}_j$ for $j \in [m]$ of length $O(nm^2)$. In another m steps we resolve these with the initial clauses (28), obtaining the singleton clauses $\bar{p}_{n,j}$ for $j \in [m]$. Finally we derive a contradiction from these and the clauses $\bigvee_{j \in [m]} p_{n,j}$. $\square$

The above refutation of $SP_{n,m}$ is not ordered, since it is not even regular: the variables q_j for every pearl j are eliminated at every stage of the induction. Nevertheless, we are unable to show that there are no short ordered refutations of $SP_{n,m}$. In order to obtain a lower bound for ordered resolution refutations, we shall modify the clauses $SP_{n,m}$. The lower bound is then proved by a bottleneck counting argument similar to that used in [13], which is based on the original argument of Haken [16]. Note that the clauses (24) - (26) are similar to the clauses expressing the Pigeonhole Principle, which makes the bottleneck counting technique applicable in our situation.

We call the pearls numbered 1 through $\frac{n}{4}$ (which we assume to be integer, for simplicity) the *special* pearls. The positions 1 to $\frac{n}{2}$ on the string are called the *left half*, and the positions $\frac{n}{2}+1$ to n the *right half* of the string.

For each special pearl j placed on the string, an associated position $\hat{\imath} = \hat{\imath}(j)$ is defined, depending on where on the string j is placed. If j is placed in the left half, then $\hat{\imath}$ is in the right half, say $\hat{\imath} = \frac{n}{2} + 2j - 1$ for definiteness, and if j is placed in the right half, then $\hat{\imath}$ is in the left half, say $\hat{\imath} = 2j$.

The set $SP'_{n,m}$ is obtained from $SP_{n,m}$ by adding additional literals to those clauses that restrict the coloring of the special pearls placed on the string. First, the clauses (27) and (29) for $1 \le i < \frac{n}{2}$, where $j' \le \frac{n}{4}$ is special, are replaced by m clauses each, namely

$$\bar{p}_{\hat{\imath},\ell} \vee \bar{p}_{1,j'} \vee \bar{q}_{j'} \tag{30}$$

$$\bar{p}_{\hat{\imath},\ell} \vee \bar{p}_{i,j} \vee \bar{p}_{(i+1),j'} \vee q_j \vee \bar{q}_{j'} \tag{31}$$

for every $\ell \in [m]$, where $\hat{\imath} := \frac{n}{2} + 2j' - 1$, since j' is placed in the left half. Similarly, the clauses (28) and (29) for $\frac{n}{2} < i < n$ and special $j \le \frac{n}{4}$ are replaced by

$$\bar{p}_{\hat{\imath},\ell} \vee \bar{p}_{n,j} \vee q_j \tag{32}$$

$$\bar{p}_{\hat{\imath},\ell} \vee \bar{p}_{i,j} \vee \bar{p}_{(i+1),j'} \vee q_j \vee \bar{q}_{j'} \tag{33}$$

for every $\ell \in [m]$, where now $\hat{\imath} := 2j$, since j is placed in the right half. All other clauses remain unchanged. The modified clauses $SP'_{n,m}$ do not have an intuitive combinatorial interpretation different from the meaning of the original clauses $SP_{n,m}$. The added literals only serve to make the clauses hard for ordered refutations. The idea is that, for the clauses (30)-(33) to be used as one would use the original (27)-(29) in the natural short, inductive proof above, the additional literals $\bar{p}_{\hat{\imath},\ell}$ have to be removed first. The positions $\hat{\imath}$ are chosen in such a way that this cannot be done in a manner consistent with a global ordering of the variables.

Theorem 25. *The clauses $SP'_{n,m}$ have negative resolution refutations of size $O(nm^2)$.*

Proof. We modify the refutation of $SP_{n,m}$ given above for the modified clauses $SP'_{n,m}$. First, note that the original clauses (27) can be obtained from (30) by a negative derivation of length m.

Next, we modify those places in the inductive step where the clauses (29) are used that have been modified. First, we resolve the modified clauses (31) resp. (33) with the inductive assumption, yielding the negative clauses

$$\bar{p}_{\hat{\imath},\ell} \vee \bar{p}_{i,j} \vee \bar{p}_{(i+1),j'} \vee \bar{q}_{j'}$$

for $\ell \in [m]$. These are then resolved with the clause $\bigvee_{j=1}^{m} p_{\hat{\imath},j}$, after which we can continue as in the original refutation.

In the places where the clauses (28) are used in the original refutation, we first resolve (32) with the clauses $\bar{p}_{n,j} \vee \bar{q}_j$, yielding $\bar{p}_{\hat{\imath},\ell} \vee \bar{p}_{n,j}$, which can be resolved with $\bigvee_{j=1}^{m} p_{\hat{\imath},j}$ to get the singleton clauses $\bar{p}_{n,j}$ as in the original refutation. □

In particular, there are polynomial size unrestricted resolution refutations of the clauses $SP'_{n,m}$. The next theorem gives a lower bound for ordered resolution refutations of these clauses.

Theorem 26. *For sufficiently large n and $m \geq \frac{9}{8}n$, every ordered resolution refutation of the clauses $SP'_{n,m}$ contains at least $2^{\frac{n}{8}(\log n-5)}$ clauses.*

Proof. For sake of simplicity, let n be divisible by 8, say $n = 8k$. Let $N := nm + m$ be the number of variables, and let an ordering $x_1, x_2, \ldots, x_N$ of the variables be given, i.e., each x_ν is one of the variables $p_{i,j}$ or q_j. Let R be a ordered resolution refutation of $SP'_{n,m}$ respecting this elimination ordering, i.e., on every path through R the variables are eliminated in the prescribed order. We shall show that R contains at least $k!$ different clauses, which is at least $2^{\frac{n}{8}(\log n-5)}$ for large n.

For a position $i \in [n]$ and $\nu \leq N$, let $S(i,\nu)$ be the set of special pearls $j \leq 2k = \frac{n}{4}$ such that $p_{i,j}$ is among the first ν eliminated variables, i.e.,

$$S(i,\nu) := \{\, j \leq 2k \,:\, p_{i,j} \in \{x_1, \ldots, x_\nu\}\,\} \; .$$

Let ν_0 be the smallest index such that $|S(i,\nu_0)| = k$ for some position i, and call this position i_0. It follows that for all $i \neq i_0$, $|S(i,\nu_0)| < k$. In other words, i_0 is the first position for which k of the variables $p_{i_0,j}$ with $j \leq 2k$ special are eliminated.

Let the elements of $S(i_0,\nu_0)$ be denoted by $j_1, \ldots, j_k$, enumerated in increasing order for definiteness. For each $1 \leq \mu \leq k$, let i_μ be the position $\hat{\imath}(j_\mu)$ associated to j_μ when j_μ is placed on the string at position i_0, i.e.,

$$i_\mu := \begin{cases} \frac{n}{2} + 2j_\mu - 1 & \text{if } i_0 \leq \frac{n}{2} \\ 2j_\mu & \text{if } i_0 > \frac{n}{2} \end{cases} \; .$$

Further we define for the set $R_\mu := [2k] \setminus S(i_\mu,\nu_0)$, i.e., R_μ is the set of special pearls j with the property that, on every path in the refutation, the variable $p_{i_\mu,j}$ is eliminated only after all the variables p_{i_0,j_κ} for $1 \leq \kappa \leq k$ have been eliminated. Note that by the definition of ν_0, $|S(i_\mu,\nu_0)| < k$ and therefore $|R_\mu| \geq k$ for all $1 \leq \mu \leq k$.

Definition. *A* critical assignment *is an assignment that satisfies all the clauses of $SP'_{n,m}$ except for exactly one of the clauses (24). From a critical assignment α, we define the following data:*

- *The unique position $i_\alpha \in [n]$ such that no pearl is placed at position i_α by α, i.e., $\alpha(p_{i_\alpha,j}) = 0$ for every $j \in [m]$. We call i_α the* gap *of α.*
- *A 1-1 mapping $m_\alpha : [n] \setminus \{i_\alpha\} \to [m]$, where for every $i \neq i_\alpha$, $m_\alpha(i)$ is the pearl placed at position i by α, i.e., the unique $j \in [m]$ such that $\alpha(p_{i,j}) = 1$.*

For every $j \in [m]$, we refer to the value $\alpha(q_j)$ as the color *of j, where we identify the value* 0 *with red and* 1 *with blue.*

A critical assignment α is called 0-critical*, if the gap is $i_\alpha = i_0$ and $m_\alpha(i_\mu) \in R_\mu$ for each $1 \leq \mu \leq k$, and moreover*

- *if i_0 is in the left half, then $j_1, \dots, j_k$ are colored blue (i.e., $\alpha(q_{j_1}) = \dots = \alpha(q_{j_k}) = 1$,)*
- *if i_0 is in the right half, then $j_1, \dots, j_k$ are colored red (i.e., $\alpha(q_{j_1}) = \dots = \alpha(q_{j_k}) = 0$.)*

Note that the positions $i_0, i_1, \dots, i_k$ and the pearls $j_1, \dots, j_k$, and thus the notion of 0-critical assignment, only depend on the elimination order and not on the refutation R.

As in other bottleneck counting arguments, the lower bound will now be proved in two steps: First, we show that there are many 0-critical assignments. Second, we will map each 0-critical assignment α to a certain clause C_α in R, and then show that not too many different assignments α can be mapped to the same clause C_α, thus there must be many of the clauses C_α.

The first goal, showing there are many 0-critical assignments, is attained with the following claim:

Claim 27. *For every choice of pairwise distinct pearls $b_1, \dots, b_k$ with $b_\mu \in R_\mu$ for $1 \leq \mu \leq k$, there is a 0-critical assignment α with $m_\alpha(i_\mu) = b_\mu$ for $1 \leq \mu \leq k$. In particular, there are at least $k!$ 0-critical assignments that disagree on the values $m_\alpha(i_\mu)$ for $1 \leq \mu \leq k$.*

Proof of Claim 27. For those positions i such that $m_\alpha(i)$ is not defined yet, i.e., $i \notin \{i_0, i_1, \dots, i_k\}$, assign pearls $m_\alpha(i) \in [m] \setminus \{j_1, \dots, j_k\}$ arbitrarily but consistently, i.e., choose an arbitrary 1-1 mapping from $[n] \setminus \{i_0, i_1, \dots, i_k\}$ to $[m] \setminus \{b_1, \dots b_k, j_1, \dots, j_k\}$. This is always possible, since by assumption $m \geq 9k$.

Finally, color those pearls that are assigned to positions to the left of the gap red, and those that are assigned to positions to the right of the gap blue, i.e., set $\alpha(q_{m_\alpha(i)}) = 0$ for $i < i_0$ and $\alpha(q_{m_\alpha(i)}) = 1$ for $i > i_0$. The pearls $j_1, \dots, j_k$ are colored according to the requirement in the definition of a 0-critical assignment.

This coloring of the pearls is well-defined even if some of the pearls $b_1, \dots b_k$ are among the $j_1, \dots, j_k$, because the positions $i_1, \dots, i_k$ and i_0 are in opposing halves of the string: if i_0 is in the left half, then every i_μ is in the right half, and in particular, $i_\mu > i_0$. Similarly, if i_0 is in the right half, then $i_\mu < i_0$, so in both cases, the pearls $j_1, \dots, j_k$ get the same color as $b_1, \dots, b_k$. The remaining pearls can be colored arbitrarily. ☐

Now we map 0-critical assignments to certain clauses in R. For a 0-critical assignment α, let C_α be the first clause in R such that α does not satisfy C_α, and

$$\{\, j \,:\, p_{i_0,j} \text{ occurs in } C_\alpha \,\} = [m] \setminus \{j_1, \dots, j_k\}\ .$$

This clause exists because α determines a path through R from the clause $\bigvee_{j \in [m]} p_{i_0,j}$ to the empty clause, such that α does not satisfy any clause on this path. The variables $p_{i_0,j}$ with $j \leq 2k$ are eliminated along that path, and $p_{i_0,j_1}, \dots p_{i_0,j_k}$ are the first among them in the elimination order.

Claim 28. *Let α be a 0-critical assignment. For every $1 \leq \mu \leq k$, the literal $\bar{p}_{i_\mu,\ell_\mu}$, where $\ell_\mu := m_\alpha(i_\mu)$, occurs in C_α.*

Proof of Claim 28. Let α' be the assignment defined by $\alpha'(p_{i_0,j_\mu}) := 1$ and $\alpha'(x) := \alpha(x)$ for all other variables x. As p_{i_0,j_μ} does not occur in C_α, α' does not satisfy C_α either.

There is exactly one clause in $SP'_{n,m}$ that is not satisfied by α', depending on where the gap i_0 is, this clause is

$$\begin{array}{rll}
i_0 = 1 \ : & \bar{p}_{i_\mu,\ell_\mu} \vee \bar{p}_{1,j_\mu} \vee \bar{q}_{j_\mu} & \\
1 < i_0 \leq \dfrac{n}{2} \ : & \bar{p}_{i_\mu,\ell_\mu} \vee \bar{p}_{i_0-1,h} \vee \bar{p}_{i_0,j_\mu} \vee q_h \vee \bar{q}_{j_\mu} & \text{where } h = m_\alpha(i_0-1) \\
\dfrac{n}{2} < i_0 < n \ : & \bar{p}_{i_\mu,\ell_\mu} \vee \bar{p}_{i_0,j_\mu} \vee \bar{p}_{i_0+1,h} \vee q_{j_\mu} \vee \bar{q}_h & \text{where } h = m_\alpha(i_0+1) \\
i_0 = n \ : & \bar{p}_{i_\mu,\ell_\mu} \vee \bar{p}_{n,j_\mu} \vee q_{j_\mu} &
\end{array}$$

The requirement for the coloring of the j_μ in the definition of a 0-critical assignment entails that these clauses are not satisfied by α', and that all other clauses are satisfied by α'.

In any case, the literal $\bar{p}_{i_\mu,\ell_\mu}$ occurs in this clause, and there is a path through R leading from the clause in question to C_α, such that α' does not satisfy any clause on that path. The variable that is eliminated in the last inference on that path must be one of the p_{i_0,j_κ} for $1 \leq \kappa \leq k$, by the definition of C_α. Since $\ell_\mu \in R_\mu$, the variable p_{i_μ,ℓ_μ} appears after p_{i_0,j_κ} in the elimination order, by the definition of R_μ . Therefore p_{i_μ,ℓ_μ} cannot have been eliminated on that path, so $\bar{p}_{i_\mu,\ell_\mu}$ still occurs in C_α. □

Finally we are ready to finish the proof of the theorem. Let α, β be two 0-critical assignments such that $\ell_\mu := m_\alpha(i_\mu) \neq m_\beta(i_\mu)$ for some $1 \leq \mu \leq k$, so that $\beta(p_{i_\mu,\ell_\mu}) = 0$. By Claim 28, the literal $\bar{p}_{i_\mu,\ell_\mu}$ occurs in C_α, therefore β satisfies C_α, and hence $C_\beta \neq C_\alpha$.

By Claim 27, there are at least $k!$ 0-critical assignments α that disagree on at least one of the values $m_\alpha(i_\mu)$. Thus R contains at least $k!$ distinct clauses of the form C_α. □

The following corollary is a direct consequence of Theorems 26 and 25.

Corollary 29. *The clauses $SP'_{n,m}$ for $m \geq \frac{9}{8}n$ exponentially separate ordered resolution from unrestricted resolution and negative resolution.*

A modification similar to the one that transforms $SP_{n,m}$ into $SP'_{n,m}$ can also be applied to the clauses $Gen(\vec{p}, \vec{q})$, yielding a set $DPGen(\vec{p}, \vec{q})$. Then for the clauses $DPGen(\vec{p}, \vec{q}) \cup Col(\vec{p}, \vec{r})$, an exponential lower bound for ordered resolutions can be proved by the method of Theorem 26 (this was presented in the conference version [5] of this paper). Also the negative resolution proofs of Theorem 19 can be modified for these clauses. Thus the clauses $DPGen(\vec{p}, \vec{q}) \cup Col(\vec{p}, \vec{r})$ exponentially separate ordered from negative resolution as well.

6 Open Problems

We would like to conclude by stating some open problems related to the topics of this paper.

1. For boolean circuits (monotone as well as general), circuit depth and formula size are essentially the same complexity measure, as they are exponentially related by the well-known Brent-Spira theorem. Is there an analogous theorem for monotone real circuits, i.e., is

$d_{\mathbb{R}}(f) = \Theta(\log s_{\mathbb{R}}(f))$ for every monotone function f? This would be implied by the converse to Lemma 1, i.e., $d_{\mathbb{R}}(f) \leq CC_{\mathbb{R}}(R_f)$. Does this hold for every monotone function f?

2. The separation between tree-like and dag-like resolution was recently improved to a strongly exponential one, with a lower bound of the form $2^{n/\log n}$ ([3, 4, 24]). Can we prove the same strong separation between tree-like and dag-like CP?

3. A solution for the previous problem would follow from a strongly exponential separation of monotone real formula size from monotone circuit size. Such a strong separation is not even known for monotone *boolean* circuits.

4. Can the superpolynomial separations of regular and negative resolution from unrestricted resolution [14, 15] be improved to exponential as well? And is there an exponential speed-up of regular over ordered resolution?

Acknowledgments

We would like to thank Ran Raz for reading a previous version of this work and discovering an error, Andreas Goerdt for sending us copies of his papers, Sam Buss for helpful discussions and finally Peter Clote for suggesting us to work on resolution separations.

References

[1] P. Beame, R. Karp, T. Pitassi, and M. Saks. The efficiency of resolution and Davis-Putnam procedures. Submitted for publication, 1999.

[2] P. Beame and T. Pitassi. Simplified and improved resolution lower bounds. In *Proc. 37th IEEE Symposium on Foundations of Computer Science*, 1996.

[3] E. Ben-Sasson and A. Wigderson. Short proofs are narrow — resolution made simple. In *Proc. 31st ACM Symposium on Theory of Computing*, 1999.

[4] E. Ben-Sasson, R. Impagliazzo and A. Wigderson. Near-optimal separation of tree-like and general resolution. Manuscript. Available as ECCC TR 00-005 at the page `http://www.eccc.uni-trier.de/eccc-local/Lists/TR-2000.html`

[5] M. L. Bonet, J. L. Esteban, N. Galesi, and J. Johannsen. Exponential separations between restricted resolution and cutting planes proof systems. In *Proc. 39th Symposium on Foundations of Computer Science*, pages 638–647, 1998.

[6] M. L. Bonet, T. Pitassi, and R. Raz. Lower bounds for cutting planes proofs with small coefficients. *Journal of Symbolic Logic*, 62:708–728, 1997. Preliminary version in *Proc. 27th ACM Symposium on Theory of Computing*, 1995.

[7] V. Chvátal and E. Szemerédi. Many hard examples for resolution. *Journal of the ACM*, 35:759–768, 1988.

[8] P. Clote and A. Setzer. On *PHP*, *st*-connectivity and odd charged graphs. In P. Beame and S. R. Buss, editors, *Proof Complexity and Feasible Arithmetics*, pages 93–117. AMS DIMACS Series Vol. 39, 1998.

[9] S. Cook and A. Haken. An exponential lower bound for the size of monotone real circuits. *J. Comp. System Sciences* 58:326–335, 1999.

[10] S. A. Cook and R. A. Reckhow. The relative efficiency of propositional proof systems. *Journal of Symbolic Logic*, 44:36–50, 1979.

[11] W. Cook, C. Coullard, and G. Turán. On the complexity of cutting plane proofs. *Discrete Applied Mathematics*, 18:25–38, 1987.

[12] X. Fu. Lower bounds on sizes of cutting planes proofs for modular coloring principles. In P. Beame and S. R. Buss, editors, *Proof Complexity and Feasible Arithmetics*, pages 135–148. AMS DIMACS Series Vol. 39, 1998.

[13] A. Goerdt. Davis-Putnam resolution versus unrestricted resolution. *Annals of Mathematics and Artificial Intelligence*, 6:169–184, 1992.

[14] A. Goerdt. Unrestricted resolution versus N-resolution. *Theoretical Computer Science*, 93:159–167, 1992.

[15] A. Goerdt. Regular resolution versus unrestricted resolution. *SIAM Journal of Computing*, 22:661–683, 1993.

[16] A. Haken. The intractability of resolution. *Theoretical Computer Science*, 39:297–308, 1985.

[17] R. Impagliazzo, T. Pitassi, and A. Urquhart. Upper and lower bounds for tree-like cutting planes proofs. In *Proc. 9th Symposium on Logic in Computer Science*, pages 220–228, 1994.

[18] J. Johannsen. Lower bounds for monotone real circuit depth and formula size and tree-like cutting planes. *Information Processing Letters*, 67:37–41, 1998.

[19] S. Jukna. Combinatorics of monotone computations. *Combinatorica*. 19(1) pp. 65-85, 1999. Preliminary version available as ECCC TR98-041, 1998.

[20] M. Karchmer and A. Wigderson. Monotone circuits for connectivity require super-logarithmic depth. *SIAM Journal on Discrete Mathematics*, 3:255–265, 1990. Preliminary version in *Proc. 20th ACM Symposium on Theory of Computing*, 1988.

[21] J. Krajíček. Interpolation theorems, lower bounds for proof systems and independence results for bounded arithmetic. *Journal of Symbolic Logic*, 62:457–486, 1997.

[22] J. Krajíček. Interpolation by a game. *Mathematical Logic Quarterly*, 44:450–458, 1998.

[23] P. Pudlák. Lower bounds for resolution and cutting plane proofs and monotone computations. *Journal of Symbolic Logic*, 62:981–998, 1997.

[24] P. Pudlák and R. Impagliazzo. A Lower Bound for DLL algorithms for k-SAT. To appear in SODA 2000.

[25] R. Raz and P. McKenzie. Separation of the monotone NC hierarchy. *Combinatorica* 19(3) pp. 403-435, 1999. Preliminary version in *Proc. 38th Symposium on Foundations of Computer Science*, pp. 234–243, 1997.

[26] A. Rosenbloom. Monotone real circuits are more powerful than monotone boolean circuits. *Information Processing Letters*, 61:161–164, 1997.

[27] U. Schöning. *Logic for Computer Scientists*. Birkhäuser, 1989.

[28] A. Urquhart. Hard examples for resolution. *Journal of the ACM*, 34:209–219, 1987.

[29] A. Urquhart. The complexity of propositional proofs. *Bulletin of Symbolic Logic*, 1:425–467, 1995.