Algorithmica (2017) 79:29–41
DOI 10.1007/s00453-016-0228-6

Strong ETH and Resolution via Games and the Multiplicity of Strategies

Ilario Bonacina[1] · Navid Talebanfard[2]

Received: 10 January 2016 / Accepted: 4 October 2016 / Published online: 19 October 2016

Abstract We consider a proof system intermediate between *regular* Resolution, in which no variable can be resolved more than once along any refutation path, and general Resolution. We call δ-*regular* Resolution such system, in which at most a fraction δ of the variables can be resolved more than once along each refutation path (however, the re-resolved variables along different paths do not need to be the same). We show that when for δ not too large, δ-regular Resolution is consistent with the Strong Exponential Time Hypothesis (SETH). More precisely, for large n and k, we show that there are unsatisfiable k-CNF formulas which require δ-regular Resolution refutations of size $2^{(1-\epsilon_k)n}$, where n is the number of variables and $\epsilon_k = \widetilde{O}(k^{-1/4})$ and $\delta = \widetilde{O}(k^{-1/4})$ is the number of variables that can be resolved multiple times.

Keywords Satisfiability · Resolution · Strong ETH

This work was completed while the 1st author was affiliated to the Computer Science Department of Sapienza University of Rome (Italy). The 1st author was funded by the European Research Council under the European Union's Seventh Framework Programme (FP7/2007–2013) / ERC Grant Agreement No. 279611.

✉ Navid Talebanfard
ntalebanfard@gmail.com

Ilario Bonacina
ilario@kth.se

[1] School of Computer Science and Communication, KTH Royal Institute of Technology, Stockholm, Sweden

[2] Saarland University and the Cluster of Excellence, MMCI, Saarbrücken, Germany

1 Introduction

The SAT problem is one of the most fundamental NP-complete problems. Paturi, Pudlák and Zane [20] proved tight depth-3 circuit lower bounds and from their technique they obtained a k-SAT algorithm which beats exhaustive search. Along similar lines, Santhanam [25] modified a lower bound argument to obtain improved satisfiability algorithms for De Morgan formulas of linear size. Employing stronger lower bound arguments, satisfiability algorithms were given for formulas of larger size in [9] and [10]. In a different direction, Williams [29] showed that even small improvements over exhaustive search for satisfiability on certain circuit classes implies a lower bound against that class. In fact he obtained his seminal $\mathsf{NEXP} \not\subseteq \mathsf{ACC}^0$ result in [30] by giving a non-trivial ACC^0-SAT algorithm.

In this paper we focus on the k-SAT problem. There are several non-trivial algorithms known for this problem, see e.g. [13,19,20,26]. Despite this however, the exact complexity of k-SAT under suitable assumptions remains unknown. Formalizing what this complexity could be, Impagliazzo and Paturi [17] formulated the following two hypotheses. The *Exponential Time Hypothesis* (ETH) which states that the are no sub-exponential time algorithms for the k-SAT problem, for any k. The *Strong Exponential Time Hypothesis* (SETH) which states that the complexity of k-SAT grows as k increases and the running time of the best k-SAT algorithms approach that of exhaustive search. More formally, it says that k-SAT requires running time $2^{(1-\epsilon_k)n}$ where $\epsilon_k \to 0$ as $k \to \infty$.

Both ETH and SETH are stronger than $\mathsf{P} \neq \mathsf{NP}$ and hence we do not expect to be able to verify either of them in any new future. We can however ask whether known algorithms are consistent with these hypotheses. For the PPSZ algorithm [19] strong lower bounds were proved in [11] supporting SETH. But one may ask for such a result that holds for a class of algorithms rather than for a specific one. Proof complexity provides a framework to do this. One can think of the run of a SAT algorithm on an unsatisfiable instance as a proof of unsatisfiability hence, if this proof is structured enough, we can employ tools from proof complexity and obtain lower bounds.

For instance practical SAT-solvers are based on the *Davis-Putnam-Logemann-Loveland* algorithm (DPLL) that is a backtracking method introduced by [14,15] to search for assignments satisfying a CNF formula. It is a well known result that DPLL is equivalent to *tree-like* Resolution, a sub-system of the proof system Resolution where only proofs having a tree structure are allowed. Hence size tree-like Resolution lower bounds transfer to lower bounds for the running time of the DPLL algorithm. In a series of works, [3,18,27] introduced the idea of *Conflict Driven Clause Learning* (CDCL) as a way for DPLL SAT-solvers to cut the search space and avoid duplicated work. This is done by performing a *conflict analysis* when the search for an assignments leads to a contradiction and then *learning* a clause encoding a reason for that failure. By definition Resolution (polynomially) simulates runs of CDCL solvers over unsatisfiable instances, hence lower bounds for Resolution transfer to lower bounds for CDCL solvers. We recall that the converse also holds under certain assumptions on the behaviour of the CDCL solver, see [21] and [2].

1.1 Previous Work

Exponential lower bounds supporting ETH have long been known for natural proof systems such as Resolution since mid 1980s, see e.g. [28]. These are $2^{\Omega(n)}$ lower bounds for k-CNF formulas on n variables and hence not strong enough to support SETH. Some thirteen years passed until the first lower bounds supporting SETH were shown. Pudlák and Impagliazzo [23] proved such lower bounds for tree-like Resolution via Prover-Delayer games. Another thirteen years later, Beck and Impagliazzo [5] obtained a very strong width lower bound which simplified and improved the result of [23] for tree-like Resolution and they were able to prove lower bounds supporting SETH for *regular* Resolution, a sub-system of Resolution. Beck and Impagliazzo in [5] showed that there are unsatisfiable k-CNF formulas in n variables requiring refutations of size at least $2^{n(1-\epsilon_k)}$ in regular Resolution. Their proof is an adaptation of a probabilistic technique from [4] and, from an high level, it can be seen as a variation of the bottleneck counting of Haken in [16]. In their argument a rule is given which maps assignments to particular clauses of the proof, at which a significant amount of 'work' is done.

Prior to this work, the strongest proof system with lower bounds supporting SETH was regular Resolution [5]. This work proves lower bounds supporting SETH in a subsystem of Resolution stronger than regular Resolution and moreover it gives a different, conceptually simpler, game-theoretic proof of the fact that SETH is consistent with regular Resolution.

1.2 Results

In this work we consider proof systems that are intermediate between regular Resolution and Resolution. The *Resolution* proof system [7,24] is a proof system for refuting unsatisfiable CNF formulas. A Resolution refutation of a CNF formula φ is a sequence of clauses ending with the empty clause such that each clause is either a clause from φ or it is derived from previous clauses in the sequence according to the following inference rule:

$$\frac{C \vee x, \quad D \vee \neg x}{C \vee D},$$

where C and D are clauses and x is a variable. Clearly a Resolution refutation can be annotated with directed edges keeping track of the applications of the inference rule, in particular each clause in the sequence will have either 0 or 2 predecessors according to if it is a clause from φ or an inferred clause. The resulting directed graph is a DAG and the sequence of clauses in the Resolution refutation is a topological ordering of it. Notice that given a resolution Refutation the DAG we can associate is not uniquely determined. Anyway, when defining subsystems of Resolution based on restrictions on the DAG structures allowed for a proof, it is customary to consider a Resolution refutation directly as a DAG with vertices labeled with clauses. This is the case for *tree-like* Resolution where are allowed as valid Resolution refutations only the ones with a tree-like structure.

A *δ-regular Resolution refutation* of a formula φ is a Resolution derivation in which along any path of the refutation DAG at most a fraction δ of the variables of φ are resolved multiple times. Hence a 0-regular Resolution refutation is just a standard regular Resolution refutation, that is a Resolution refutation where no variable is resolved multiple times along any path. A 1-regular Resolution refutation is just one without any constraint. A more formal definition of Resolution and δ-regular Resolution is given in Sect. 2 together with all the other necessary preliminaries.

The main result of this work is the following.

Theorem 1.1 (Main theorem) *For any large n and k, there exists an unsatisfiable k-CNF formula ψ on $n' \geq n$ variables such that any δ-regular Resolution refutation of ψ requires size at least $2^{(1-\epsilon_k)n}$ where both ϵ_k and δ are $\widetilde{O}(k^{-1/4})$.*

We recall that the *width* of a Resolution refutation is the number of literals in the largest clause appearing in the refutation. The way we prove this result is via a *strong* width lower bound, that is a lower bound of the form $(1-\epsilon_k)n$, with $\epsilon_k \to 0$, relative to Resolution refutations of some particular k-CNFs in n variables. Width lower bounds of this form were proved in [5] and improved in the asymptotic in [8] (see Theorem 4.1 for the precise statement we are going to use).

The reader could be tempted to think that once we are given a strong Resolution width of the form above then a size lower bound as in Theorem 1.1 will follow by the standard relation between width and size by Ben-Sasson and Wigderson [6]. In [6] the authors showed that if a formula requires refutations of large width, it also requires refutations with many clauses. More precisely they showed that if a k-CNF formula can only have Resolution refutations of width at least W, then it requires Resolution size at least $2^{(W-k)^2/16n}$, where n is the number of variables. The constant loss in the exponent is the reason why we do not immediately get $2^{(1-\epsilon_k)n}$ size lower bounds from strong width lower bounds. However, the result in [6] holds for any k-CNF without any particular assumption. If the formula is structured in some sense, for instance if it is a *xorification*, we show we can avoid this loss (in a subsystem of Resolution).

The *ℓ-xorification* of a CNF formula φ in n variables is a new CNF formula in $n\ell$ variables obtained substituting each variable x_i in φ with ℓ new variables $y_i^1 \oplus \cdots \oplus y_i^\ell$ and then expanding again as a CNF. We denote the CNF resulting from such operation $\varphi[\oplus^\ell]$. Our main technical result, Theorem 1.2 informally states that if a k-CNF φ requires width w to be refuted in Resolution, then any δ-regular Resolution refutation of $\varphi[\oplus^\ell]$ requires size $2^{(1-\epsilon)n\ell}$, where ϵ is a function of k, ℓ, δ and w. More precisely we prove the following:

Theorem 1.2 *Let φ be an unsatisfiable CNF formula in n variables and w, δ and ℓ be parameters. If the width to refute φ is Resolution is at least w then the size to refute $\varphi[\oplus^\ell]$ in δ-regular Resolution is at least $2^{(1-\epsilon)w\ell}$, where $\epsilon = \frac{1}{\ell}\log(\frac{e^3\ell n}{w}) + \frac{\delta n}{w}\log\frac{e^3\ell}{\delta}$.*

Once we have proved this result then Theorem 1.1 follows just by carefully tuning the parameters.

The way we prove Theorem 1.2 relies on two known games characterizations: the *Pudlák game* characterizing Resolution size [22] and the *Atserias-Dalmau game* characterizing Resolution width [1]. In Sect. 3 we give a precise account for both games in a common setting and terminology, see Definition 3.1. We conclude this introductory part giving some intuition behind those games and the proof of Theorem 1.2.

In the *Pudlák game*, informally, we have two players, Prover and Delayer , that play on some formula φ. Prover has the objective of showing that the formula φ is unsatisfiable by querying variables. Delayer on the other hand wants to play as long as possible before the formula is falsified while answering to the queries Prover asks her. The size of Resolution proofs of φ is then characterized as the minimal number of *records*, i.e. partial assignments, Prover has to consider in a winning strategy [22]. If we force the Prover in each game to re-query a fraction of at most δ variables from φ then the minimal number of records such Prover has to consider in a winning strategy characterize the size in δ-regular Resolution. This is the content of Theorem 3.3 which we leave without proof since it is a trivial observation over the result in [22].

Hence to prove a Resolution (or a δ-regular Resolution) size lower bound we show that, in order to win, Prover must keep a large number of records and we can do that by producing a lot of sufficiently different strategies for Delayer . Prover must win against each of them, hence in his winning strategy he must have a lot of distinct records, since the strategies of Delayer are sufficiently different. In the literature this is done essentially by making Prover play against a Delayer that plays accordingly to a random strategy [12,22]. Then the size lower bound, that is a lower bound on the number of records that Prover must have in a winning strategy, is obtained by probabilistic arguments. This may very likely lead to some loss in the constants that we need to avoid to prove a SETH lower bound for Resolution size so we choose another way: we play the Pudlák game over a xorified formula $\varphi[\oplus^{\ell}]$.

The construction of multiple strategies for Delayer relies on the characterization of Resolution width as a game [1] where again we have a Prover and a Delayer , the goal of the prover is to falsify the formula φ but he can use assignments with a bounded number of variables. At a very high level, a winning strategy for Delayer in the width game on φ gives rise to a multitude of strategies for Delayer on the Pudlák game on $\varphi[\oplus^{\ell}]$. The new strategies act differently from each other on $\varphi[\oplus^{\ell}]$, but in a sense they all act the same as the original strategy for Delayer in the width game on the original formula φ. The size lower bound then follows by a counting argument exploiting the combinatorial properties of the xorified formula in such a way that the number of Delayer strategies, for the Pudlák game played on $\varphi[\oplus^{\ell}]$, does indeed hugely amplify.

Notice that Theorem 1.2 does not depend on the particular formula we choose to apply it but, to get the result in Theorem 1.1, we need to apply it to a particular formula φ for which we have strong width lower bounds, such formulas are provided by [5] and, with better asymptotic, by [8].

1.3 Outline of Paper

In the next Section we give some preliminaries and notations about Resolution and δ-regular Resolution. Section 3 contains the common framework for *Pudlák games* [22] characterizing *size* in Resolution and the Atserias and Dalmau games [1] characterizing *width* in Resolution. Section 4 contains the core results of this work (Theorem 1.1 and Theorem 1.2).

1.4 Open Problems

In this work we prove that there exist unsatisfiable k-CNF formulas in n variables that require δ-regular Resolution refutations of size at least $2^{(1-\epsilon)n}$, where $k = \widetilde{O}(\epsilon^{-4})$ and where $\delta = \widetilde{O}(\epsilon^{-4})$. Hence a natural question is whether it is possible to improve the dependency of δ and k on ϵ.

More generally, we have some proof systems stronger than δ-regular Resolution, such as Resolution itself, Polynomial Calculus + Resolution, RES(k), Cutting Planes, for which we know that there are some unsatisfiable CNFs in n variables which require refutations of size $2^{\Omega(n)}$. Are those proof systems consistent with SETH?

2 Preliminaries

A *literal* is either a variable x or its negation $\neg x$. A *clause* C is a disjunction of literals and by its *width* we mean the number of literals appearing in C and we denote this by $|C|$. A *Conjunctive Normal Form* (CNF) formula is a conjunction of a set of clauses.

Given a boolean function f on a set of variables X, a *partial assignment* is a function $\rho : X \to \{0, 1, *\}$. We call *domain* of ρ, $\text{dom}(\rho)$ the set $\rho^{-1}(\{0, 1\})$. The restriction of f to ρ denoted by $f|_\rho$ is a function on $\rho^{-1}(*)$ obtained from f by fixing the value of all variables in $\rho^{-1}(0) \cup \rho^{-1}(1)$ according to ρ. We write $\rho \subseteq \sigma$ if for all $x \in X$, $\rho(x) \neq *$ implies $\sigma(x) = \rho(x)$. For a partial assignment ρ for which $\rho(x) = *$, by $\rho \cup \{(x, b)\}$ we denote a partial assignment ρ' such that for all $y \neq x$, $\rho'(y) = \rho(y)$ and $\rho'(x) = b$. Given a (partial) assignment ρ and a subset $B \subseteq X$, $\rho|_B$ is a partial assignment defined only on the variables in B such that for all $x \in B$, $\rho|_B(x) = \rho(x)$.

Resolution [7,24] is a proof system for refuting unsatisfiable CNF formulas. The only inference rule in Resolution is given as follows

$$\frac{C \vee x, \quad D \vee \neg x}{C \vee D},$$

where C and D are clauses and x is a variable. We say that x is *resolved* and $C \vee D$ is called the *resolvant* of $C \vee x$ and $D \vee \neg x$. A *Resolution derivation* of a clause D from a CNF φ is a sequence $\Pi = \langle C_1, \ldots, C_\tau \rangle$ of clauses such that $C_\tau = D$ and each C_i is either an *axiom*, that is a clause from φ, or it is derived by applying the Resolution rule on some clause C_j and $C_{j'}$ such that $j, j' < i$. We will denote this by $\Pi : \varphi \vdash D$. When defining subsystems of Resolution we consider hardcoded in the sequence of clauses Π also a function providing from which previous clauses a clause in Π is inferred or if it is a clause from φ. Having at hand such function then a Resolution derivation Π is given a structure of a DAG and hence we can talk of *paths* in the derivation intending paths in the DAG associated to the derivation. If φ is an unsatisfiable formula, a *Resolution refutation* of φ is a derivation of $\bot$, the empty clause, from φ. Resolution is *sound* and *complete*, that is we can derive $\bot$ from a CNF formula if and only if it is unsatisfiable.

A *δ-regular Resolution derivation* of a clause D from a formula φ in n variables is a Resolution derivation in which along any derivation path at most a fraction of δ variables are resolved multiple times. Hence a 0-regular Resolution refutation is just a standard regular refutation and a 1-regular Resolution refutation is one without any constraint.

The *size* of a Resolution derivation is the number of clauses appearing in it. We denote the minimum size of a derivation of D from φ by $\mathsf{size}(\varphi \vdash D)$. We also denote the minimum size of a δ-regular derivation of D from φ by $\mathsf{size}_\delta(\varphi \vdash D)$. Similarly we define the *width* of a derivation to be the width of the largest clause appearing in it. We denote the minimum width of a derivation of D from φ by $\mathsf{width}(\varphi \vdash D)$.

3 A Game View of Resolution

In this section we present a common framework, Definition 3.1, for the games described by Atserias and Dalmau [1] and Pudlák [22] and then we recall the characterizations of width and size in Resolution.

Definition 3.1 ($\mathsf{Game}(\varphi, \mathcal{R}\ )$). Given an unsatisfiable CNF formula φ in n variables and a set of partial assignments $\mathcal{R}$ containing the empty assignment, we define a game, $\mathsf{Game}(\varphi, \mathcal{R}\ )$, between two players Prover (he) and Delayer (she).

At each step i of the game a partial assignment $\alpha_i \in \mathcal{R}$ is maintained (α_0 is the empty partial assignment), then at step $i+1$ the following moves take place:

1. Prover picks some variable $x \notin \mathrm{dom}(\alpha_i)$.
2. Delayer then has to answer $x = b$ for some bit $b \in \{0, 1\}$.
3. Prover set $\alpha_{i+1} \in \mathcal{R}$ such that $\alpha_{i+1} \subseteq \alpha_i \cup \{(x, b)\}$.

If at any point in the game α_i falsifies φ then Prover wins; otherwise we say that Delayer wins. As customary, we say that Prover has a *winning strategy* for the game if for any strategy of Delayer , he can play so that he wins the game. Otherwise we say that Delayer has a *winning strategy*.

If in each run of the game Prover can query at most a fraction of δ variables, we call the corresponding game $\mathsf{Game}_\delta(\varphi, \mathcal{R}\ )$.

For a suitable choice of $\mathcal{R}$ the $\mathsf{Game}(\varphi, \mathcal{R}\ )$ is exactly the one used by Atserias and Dalmau [1] to characterise the minimal width of Resolution refutations of φ. In particular in [1] the following result is shown (rephrased here with the notations we just set up).

Theorem 3.2 (Atserias and Dalmau [1]). *Let φ be an unsatisfiable CNF formula and let $\mathcal{R}$ be the set of all possible partial assignments with a domain of size strictly less than w. The following are equivalent*

1. Prover *has a winning strategy for* $\mathsf{Game}(\varphi, \mathcal{R}\ )$ *;*
2. $\mathsf{width}(\varphi \vdash \bot) < w$.

Due to this equivalence, for this particular choice of $\mathcal{R}$, we will denote $\mathsf{Game}(\varphi, \mathcal{R}\ )$ *by* $\mathsf{width\text{-}Game}(\varphi, w)$.

The next result is essentially due to Pudlák [22]: he shows that we can also characterize the minimal size of Resolution refutations of φ in terms of these games. From a Resolution refutation Π we can construct a winning strategy for Prover with a set $\mathcal{R}$

of the same size of Π and vice versa. Moreover a play of the $\mathsf{Game}_\delta(\varphi, \mathcal{R})$ corresponds to a path in Π and, if Π is δ-regular, in each run the set of variables Prover is going to query many times is at most a δ fraction of the total number of variables.

Theorem 3.3 *Let φ be an unsatisfiable CNF and let δ be any real in the interval* $[0, 1]$. *The following are equivalent*

1. *there exists a set of partial assignments $\mathcal{R}$ such that $|\mathcal{R}| \leq s$ for which* Prover *has a winning strategy for* $\mathsf{Game}_\delta(\varphi, \mathcal{R})$ *;*
2. $\mathsf{size}_\delta(\varphi \vdash \bot) \leq s$.

Notice that this Theorem states an equivalence but in what follows we will only use the fact that (2) implies (1).

4 Games and Xorifications

Given a CNF formula φ on the variables $x_1, \ldots, x_n$, we define the ℓ-*xorification* of φ as follows: it is a formula on the new variables y_i^j, where $1 \leq i \leq n$ and $1 \leq j \leq \ell$ and it is obtained by replacing each x_i with $y_i^1 \oplus \cdots \oplus y_i^\ell$ and expanding the formula as a CNF formula. We denote the obtained CNF formula by $\varphi[\oplus^\ell]$ and note that if φ is a k-CNF in n variables, then $\varphi[\oplus^\ell]$ is a $k\ell$-CNF in $n\ell$ variables. Due to this notation we will refer to the variables of φ as the x-*variables* and to the variables of $\varphi[\oplus^\ell]$ as the y-*variables*. Moreover we say that all the y-variables $y_i^1, \ldots, y_i^\ell$ form a *block* of variables corresponding to the x-variable x_i. We say that a partial assignment over the y-variables *fixes* a value for a x-variable x_i if it assigns all the y-variables in the block corresponding to x_i.

Restated Theorem 1.2 *Let φ be an unsatisfiable CNF formula in n variables and w, δ and ℓ be parameters. If the width to refute φ is Resolution is at least w then the size to refute $\varphi[\oplus^\ell]$ in δ-regular Resolution is at least $2^{(1-\epsilon)w\ell}$, where $\epsilon = \frac{1}{\ell}\log(\frac{e^3\ell n}{w}) + \frac{\delta n}{w}\log\frac{e^3\ell}{\delta}$.*

Proof For each partial assignment α over the y-variables there is naturally associated a partial assignment α' over the x-variables, defined as follows

$$\alpha'(x_i) = \begin{cases} \alpha(y_i^1) \oplus \cdots \oplus \alpha_r(y_i^\ell) & \text{if } \forall j = 1, \ldots, \ell,\ y_i^j \in \mathrm{dom}(\alpha), \\ * & \text{otherwise.} \end{cases}$$

By Theorem 3.3, it is enough to show that if Prover wins $\mathsf{Game}_\delta(\varphi[\oplus^\ell], \mathcal{R})$ then

$$|\mathcal{R}| \geq 2^{w(\ell - \log(\frac{e^3\ell n}{w}) - \frac{\delta\ell n}{w}\log\frac{e^3\ell}{\delta})}.$$

So suppose Prover wins $\mathsf{Game}_\delta(\varphi[\oplus^\ell], \mathcal{R})$ for some set of partial assignments $\mathcal{R}$. Since $\mathsf{width}(\varphi \vdash \bot) \geq w$, by Theorem 3.2, there is a winning strategy σ for Delayer in the game $\mathsf{width\text{-}Game}(\varphi, w)$.

For each total assignment β on the y-variables, consider a strategy σ_β for Delayer in the game $\mathsf{Game}_\delta(\varphi[\oplus^\ell], \mathcal{R}\ )$ as follows. Let α_r be the partial assignment on y-variables at stage r of the game $\mathsf{Game}_\delta(\varphi[\oplus^\ell], \mathcal{R}\ )$ and y^i_j the variable queried by Prover at stage $r+1$. Then the strategy σ_β for Delayer goes as follows:

1. if there exists $j' \neq j$ such that $y_i^{j'} \notin \mathrm{dom}(\alpha_r)$, set y_i^j to $\beta(y^i_j)$;
2. otherwise, if for all $j' \neq j$, $y_i^{j'} \in \mathrm{dom}(\alpha_r)$, then look at the value $b \in \{0, 1\}$ the strategy σ sets the variable x_i when given the partial assignment α'_r. Then set y_i^j to $q \in \{0, 1\}$ such that

$$q \oplus \bigoplus_{j' \neq j} \alpha_r(y_i^j) = b.$$

This can be done since $x_i \equiv y_i^1 \oplus \cdots \oplus y_i^\ell$ and the value of x_i can be set freely to 0 or 1 appropriately even after all but one of $y_i^1, \ldots, y_i^\ell$ have been set. Moreover, by induction on r, it is easy to see that the strategy σ must provide an answer when challenged by Prover by any variable x_i not on the record α_r, hence σ_β is well-defined.

Moreover, it is easy to see that for each total assignment β over the y-variables, σ_β is a winning strategy for Delayer in the game $\mathsf{width\text{-}Game}(\varphi[\oplus^\ell], w\ell)$. Since we are assuming that Prover has a winning strategy for $\mathsf{Game}_\delta(\varphi[\oplus^\ell], \mathcal{R}\ )$, in particular, this means that for any β he wins against the Delayer 's strategy σ_β. This means that for each total assignment β over the y-variables, $\mathcal{R}$ must contain some partial assignment, denoted by ρ_β, with domain of size at least $w\ell$ and such that at least w blocks of y-variables are completely fixed by ρ_β. Without loss of generality we assume that each ρ_β fixes exactly w blocks of y-variables, that is if ρ_β is setting more y-variables we simply ignore some of the variables and only consider w blocks. Our goal is to show that we have 'many distinct' such partial assignments ρ_β.

Let $B \subseteq [n]$ denote a generic set of size w and consider for each possible such B the set S_B of the total assignments βs such that ρ_β is fixing all the $y_i^1, \ldots, y_i^\ell$ corresponding to some i in B. There are $2^{n\ell}$ possible total assignments β and $\binom{n}{w}$ possible sets B, hence by the pigeonhole principle, there is a set $B^* \subseteq [n]$ of size w such that

$$|S_{B^*}| \geq \frac{2^{n\ell}}{\binom{n}{w}}. \tag{1}$$

Let S'_{B^*} be the set of partial assignments $\beta|_{B^*}$ where $\beta \in S_{B^*}$. We clearly have that

$$|S_{B^*}| \leq |S'_{B^*}| \cdot 2^{n\ell - \ell|B^*|} = |S'_{B^*}| \cdot 2^{n\ell - w\ell}.$$

By Eq. (1), we get

$$|S'_{B^*}| \geq \frac{2^{w\ell}}{\binom{n}{w}}. \tag{2}$$

We have now that both S'_{B^*} and $\{\rho_\beta \in \mathcal{R}\ : \beta \in S_{B^*}\}$ consist of assignments with domain the y-variables y_i^j such that $i \in B^*$ and $1 \leq j \leq \ell$. We show that $|\{\rho_\beta \in \mathcal{R}\ :$

$\beta \in S_{B^*}\}|$ cannot be too small compared to $|S'_{B^*}|$, this will be, intuitively, due to the fact that the βs we start with are very different.

Let Z^β be the set of variables that Prover re-queried when playing against σ_β and for any $i = 1, \dots, n$ let $Z_i^\beta = Z^\beta \cap \{y_i^1, \dots, y_i^\ell\}$. By hypothesis, Prover is allowed to re-query in each game at most a δ fraction of variables, hence $|Z^\beta| \le \delta \ell n$.

When Delayer follows the strategy σ_β and fixes all y-variables in a block corresponding to x_i, the assignment produced ρ_β is within Hamming distance $|Z_i^\beta| + 1$ from β in that block. This means that for each $\beta \in S_{B^*}$ and for each i, $\rho_\beta|_{\{y_1^i,\dots,y_\ell^i\}}$ has Hamming distance at most $|Z_i^\beta| + 1$ from some partial assignment in S'_{B^*} restricted to $\{y_1^i, \dots, y_\ell^i\}$. Let $\mathcal{Z}$ be the set of all possible sets Z that are subsets of the y-variables of size $\delta\ell n$ and such that there exists $\beta \in S_{B^*}$ with $Z^\beta \subseteq Z$. For any $i = 1, \dots, n$ let $Z_i = Z \cap \{y_i^1, \dots, y_i^\ell\}$. Then, by counting the variables where ρ_β and an assignment in S'_{B^*} could differ, we have that

$$|S'_{B^*}| \le |\{\rho_\beta \in \mathcal{R} \ : \beta \in S_{B^*}\}| \cdot \sum_{Z\in\mathcal{Z}} \prod_{i\in B^*} 2^{|Z_i|+1} \binom{\ell}{|Z_i| + 1}. \tag{3}$$

Hence we have the following chain of inequalities

$$|S'_{B^*}| \overset{eq.(3)}{\le} |\{\rho_\beta \in \mathcal{R} \ : \beta \in S_{B^*}\}| \cdot \sum_{Z\in\mathcal{Z}} \prod_{i\in B^*} 2^{|Z_i|+1} \binom{\ell}{|Z_i| + 1} \tag{4}$$

$$\le |\{\rho_\beta \in \mathcal{R} \ : \beta \in S_{B^*}\}| \cdot \sum_{Z\in\mathcal{Z}} \prod_{i\in B^*} \left(\frac{e^2\ell}{|Z_i| + 1}\right)^{|Z_i|+1} \tag{5}$$

$$\le |\{\rho_\beta \in \mathcal{R} \ : \beta \in S_{B^*}\}| \cdot \sum_{Z\in\mathcal{Z}} \left(\frac{\sum_{i\in B^*} e^2\ell}{\sum_{i\in B^*}(|Z_i| + 1)}\right)^{\sum_{i\in B^*}(|Z_i|+1)} \tag{6}$$

$$\le |\{\rho_\beta \in \mathcal{R} \ : \beta \in S_{B^*}\}| \cdot \binom{\ell n}{\delta \ell n} \cdot \left(\frac{\sum_{i\in B^*} e^2\ell}{w}\right)^{\delta\ell n + w} \tag{7}$$

$$= |\{\rho_\beta \in \mathcal{R} \ : \beta \in S_{B^*}\}| \cdot \binom{\ell n}{\delta \ell n} \cdot \left(e^2\ell\right)^{\delta\ell n + w} \tag{8}$$

The inequality (6) follows from the weighted AM-GM inequality [1] and the inequality (7) follows from the fact that $w \le \sum_{i\in B^*}(|Z_i| + 1) \le \delta\ell n + w$. Putting all together we have that

[1] The *weighted Arithmetic Mean - Geometric Mean inequality* says that given non-negative numbers $a_1, \dots, a_n$ and non-negative weights $w_1, \dots, w_n$ then

$$\prod_i a_i^{w_i} \le \left(\frac{\sum_i w_i a_i}{w}\right)^w,$$

where $w = \sum_i w_i$. We applied this inequality with $a_i = \frac{e^2\ell}{|Z_i|+1}$ and $w_i = |Z_i| + 1$.

$$
\begin{aligned}
|\mathcal{R}\,| \overset{(\dagger\dagger)}{\geq} |\{\rho_\beta \in \mathcal{R}\ : \beta \in S_{B^*}\}| \geq \frac{|S'_{B^*}|}{\binom{n\ell}{\delta\ell n}\left(e^2\ell\right)^{\delta\ell n+w}} &\overset{\text{(eq. 2)}}{\geq} \frac{2^{w\ell}}{\binom{n}{w}\binom{\ell n}{\delta\ell n}\left(e^2\ell\right)^{\delta\ell n+w}} \\
&\geq \frac{2^{w\ell}}{(\frac{en}{w})^w\left(\frac{e}{\delta}\right)^{\delta\ell n}\left(e^2\ell\right)^{\delta\ell n+w}} \\
&= 2^{w(\ell-\log(\frac{e^3\ell n}{w})-\frac{\delta\ell n}{w}\log\frac{e^3\ell}{\delta})},
\end{aligned}
$$

where the inequality (††) follows by the definition of ρ_β. □

The next step now is to obtain formulas which require very large Resolution width. Such a construction is given by Beck and Impagliazzo in [5] and improved in [8].

Theorem 4.1 ([8]) *For any large n and k, there exist an unsatisfiable k-CNF formula φ on n variables and some $\zeta_k = \widetilde{O}(k^{-1/3})$ such that*

$$\mathsf{width}(\varphi \vdash \bot) \geq (1-\zeta_k)n.$$

Now, we have set all the preliminary results to prove Theorem 1.1 and in particular, our SETH lower bound for Resolution will follow from the existence of a CNF formula requiring very high Resolution width (Theorem 4.1) and the previous theorem about xorifications (Theorem 1.2).

Restated Theorem 1.1 (Main theorem) *For any large n and k, there exists an unsatisfiable k-CNF formula ψ on $n' \geq n$ variables such that any δ-regular Resolution refutation of ψ requires size at least $2^{(1-\epsilon_k)n}$ where both ϵ_k and δ are $\widetilde{O}(k^{-1/4})$.*

Proof Let φ be the k-CNF formula given by Theorem 4.1, in particular $\mathsf{width}(\varphi \vdash \bot) \geq (1-\zeta_k)n$ where $\zeta_k = \widetilde{O}(k^{-1/3})$. Then $\varphi[\oplus^\ell]$ is a k'-CNF formula on $n\ell$ variables where $k' = k\ell$. By the choice of $\ell = \widetilde{\Theta}(k^{1/3})$, $\delta = \widetilde{O}(k^{-1/3})$ and by Theorem 1.2, it follows that

$$
\begin{aligned}
\mathsf{size}_\delta(\varphi[\oplus^\ell] \vdash \bot) &\geq 2^{(1-\zeta_k)n(\ell-\log(\frac{e^3\ell n}{w})-\frac{\delta\ell n}{w}\log\frac{e^3\ell}{\delta})} \\
&\overset{(\dagger)}{=} 2^{(1-\zeta_k)n(\ell-O(\log k)-\ell\widetilde{O}(k^{-1/3}))} = 2^{(1-\widetilde{O}(k^{-1/3}))n\ell} \\
&= 2^{(1-\epsilon_{k'})n\ell}.
\end{aligned}
$$

In particular the equality (†) follows from the choice of $\ell = \widetilde{\Theta}(k^{1/3})$ and $\delta = \widetilde{O}(k^{-1/3})$. To obtain the asymptotic behaviour of $\epsilon_{k'}$ with respect to k', just observe that $k' = k\ell = \widetilde{\Theta}(k^{4/3})$ and $\epsilon_{k'} = \widetilde{O}(k^{-1/3})$, hence $\epsilon_{k'} = \widetilde{O}(k'^{-1/4})$. Similarly we get the asymptotic behaviour of δ as a function of k'. So the formula ψ in the statement is the constructed formula $\varphi[\oplus^\ell]$. □

Acknowledgements We would like to thank Nicola Galesi for discussions on the topic. We would also like to thank Jakob Nordström and Massimo Lauria for discussions on Resolution size and strong width lower bounds.

References

1. Atserias, A., Dalmau, V.: A combinatorial characterization of resolution width. J. Comput. Syst. Sci. **74**, 323–334 (2008)
2. Atserias, A., Fichte, J.K., Thurley, M.: Clause-learning algorithms with many restarts and bounded-width resolution. J. Artif. Intell. Res. (JAIR) **40**, 353–373 (2011)
3. Bayardo Jr., R.J.B., Schrag, R.: Using CSP look-back techniques to solve real-world SAT instances. In: Kuipers, B., Webber, B.L. (eds.) Proceedings of the Fourteenth National Conference on Artificial Intelligence and Ninth Innovative Applications of Artificial Intelligence Conference, AAAI 97, IAAI 97, pp. 203–208, AAAI Press/The MIT Press, Providence 27–31 July 1997
4. Beame, P., Beck, C., Impagliazzo, R.: Time-space tradeoffs in resolution: superpolynomial lower bounds for superlinear space. In: Karloff, H.J., Pitassi, T. (eds.) Proceedings of the 44th Symposium on Theory of Computing Conference, STOC 2012, pp. 213–232, ACM, New York, 19–22 May 2012
5. Beck, C., Impagliazzo, R.: Strong ETH holds for regular resolution. In: Proceedings of the Forty-fifth Annual ACM Symposium on Theory of Computing, STOC '13, pp. 487–494, ACM (2013)
6. Ben-Sasson, E., Wigderson, A.: Short proofs are narrow - resolution made simple. J. ACM **48**, 149–169 (2001)
7. Blake, A.: Canonical Expressions in Boolean Algebra, PhD thesis. University of Chicago (1937)
8. Bonacina, I., Talebanfard, N.: Improving resolution width lower bounds for k-CNFs with applications to the strong exponential time hypothesis. Inf. Process. Lett. **116**, 120–124 (2015)
9. Chen, R., Kabanets, V., Kolokolova, A., Shaltiel, R., Zuckerman, D.: Mining circuit lower bound proofs for meta-algorithms. In: IEEE 29th Conference on Computational Complexity, CCC, pp. 262–273 (2014)
10. Chen, R., Kabanets, V., Saurabh, N.: An improved deterministic #SAT algorithm for small de morgan formulas. In: Mathematical Foundations of Computer Science 2014 - 39th International Symposium, MFCS, pp. 165–176 (2014)
11. Chen, S., Scheder, D., Talebanfard, N., Tang, B.: Exponential lower bounds for the PPSZ k-SAT algorithm. In: SODA, pp. 1253–1263 (2013)
12. Dantchev, S.S.: Relativisation provides natural separations for resolution-based proof systems. In: Proceedings of Computer Science - Theory and Applications, First International Computer Science Symposium in Russia, pp. 147–158, CSR 2006, St. Petersburg, 8–12 June 2006
13. Dantsin, E., Goerdt, A., Hirsch, E.A., Kannan, R., Kleinberg, J.M., Papadimitriou, C.H., Raghavan, P., Schöning, U.: A deterministic $(2–2/(k+1))^n$ algorithm for k-SAT based on local search. Theor. Comput. Sci. **289**, 69–83 (2002)
14. Davis, M., Logemann, G., Loveland, D.W.: A machine program for theorem-proving. Commun. ACM **5**, 394–397 (1962)
15. Davis, M., Putnam, H.: A computing procedure for quantification theory. J. ACM **7**, 201–215 (1960)
16. Haken, A.: The intractability of resolution. Theor. Comput. Sci. **39**, 297–308 (1985)
17. Impagliazzo, R., Paturi, R.: On the complexity of k-SAT. J. Comput. Syst. Sci. **62**, 367–375 (2001)
18. Moskewicz, M.W., Madigan, C.F., Zhao, Y., Zhang, L., Malik, S.: Chaff: engineering an efficient SAT solver. In: Proceedings of the 38th Design Automation Conference, DAC 2001, pp. 530–535, ACM, Las Vegas, 18–22 June 2001
19. Paturi, R., Pudlák, P., Saks, M.E., Zane, F.: An improved exponential-time algorithm for k-SAT. J. ACM **52**, 337–364 (2005)
20. Paturi, R., Pudlák, P., Zane, F.: Satisfiability coding lemma. In: 38th Annual Symposium on Foundations of Computer Science, FOCS, pp. 566–574 (1997)
21. Pipatsrisawat, K., Darwiche, A.: On the power of clause-learning SAT solvers as resolution engines. Artif. Intell. **175**, 512–525 (2011)
22. Pudlák, P.: Proofs as games. Am. Math. Mon. **107**, 541–550 (2000)
23. Pudlák, P., Impagliazzo, R.: A lower bound for DLL algorithms for k-SAT (preliminary version). In: Proceedings of the Eleventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA '00, pp. 128–136 (2000)
24. Robinson, J.A.: A machine-oriented logic based on the resolution principle. J. ACM **12**, 23–41 (1965)
25. Santhanam, R.: Fighting perebor: new and improved algorithms for formula and QBF satisfiability. In: 51th Annual IEEE Symposium on Foundations of Computer Science. FOCS 2010, 183–192 (2010)
26. Schöning, U.: A probabilistic algorithm for k-SAT and constraint satisfaction problems. In: 40th Annual Symposium on Foundations of Computer Science, FOCS, pp. 410–414 (1999)

27. Silva, J.P.M., Sakallah, K.A.: GRASP: a search algorithm for propositional satisfiability. IEEE Trans. Comput. **48**, 506–521 (1999)
28. Urquhart, A.: Hard examples for resolution. J. ACM **34**, 209–219 (1987)
29. Williams, R.: Improving exhaustive search implies superpolynomial lower bounds. In: Proceedings of the 42nd ACM Symposium on Theory of Computing, STOC, pp. 231–240 (2010)
30. Williams, R.: Non-uniform ACC circuit lower bounds. In: Proceedings of the 26th Annual IEEE Conference on Computational Complexity, CCC, pp. 115–125 (2011)