DRAT Proofs, Propagation Redundancy, and Extended Resolution

Sam Buss[⋆,1] and Neil Thapen[⋆⋆,2]

[1] Dept. of Mathematics, U.C. San Diego, La Jolla, CA, USA, sbuss@ucsd.edu
[2] Institute of Mathematics, Czech Academy of Sciences, Prague, Czech Republic, thapen@math.cas.cz

Abstract. We study the proof complexity of RAT proofs and related systems including BC, SPR, and PR which use blocked clauses and (subset) propagation redundancy. These systems arise in satisfiability (SAT) solving, and allow inferences which preserve satisfiability but not logical implication. We introduce a new inference SR using substitution redundancy. We consider systems both with and without deletion. With new variables allowed, the systems are known to have the same proof theoretic strength as extended resolution. We focus on the systems that do not allow new variables to be introduced.

Our first main result is that the systems DRAT$^-$, DSPR$^-$ and DPR$^-$, which allow deletion but not new variables, are polynomially equivalent. By earlier work of Kiesl, Rebola-Pardo and Heule, they are also equivalent to DBC$^-$. Without deletion and without new variables, we show that SPR$^-$ can polynomially simulate PR$^-$ provided only short clauses are inferred by SPR inferences. Our next main results are that many of the well-known "hard" principles have polynomial size SPR$^-$ refutations (without deletions or new variables). These include the pigeonhole principle, bit pigeonhole principle, parity principle, Tseitin tautologies, and clique-coloring tautologies; SPR$^-$ can also handle or-fication and xor-ification. Our final result is an exponential size lower bound for RAT$^-$ refutations, giving exponential separations between RAT$^-$ and both DRAT$^-$ and SPR$^-$.

1 Introduction

SAT solvers are routinely used for many large-scale instances of satisfiability. It is widely realized that when a solver reports a SAT instance Γ is unsatisfiable, it should also produce a *proof* of unsatisfiability. This is of particular importance as SAT solvers become increasingly complex, and thus more subject to software bugs or even design problems.

The first proof systems proposed for SAT solvers were based on reverse unit propagation (RUP) inferences [8, 27] as this is sufficient to handle resolution and the usual CDCL clause learning schemes. However, RUP inferences only support logical implication, and do not accommodate many "inprocessing" rules. Inprocessing rules may

⋆ This work was initiated on a visit of the first author to the Czech Academy of Sciences in July 2018, supported by ERC advanced grant 339691 (FEALORA). Also supported by Simons Foundation grant 578919.

⋆⋆ Partially supported by GA ČR project 19-05497S and by ERC advanced grant 339691 (FEALORA) and RVO:67985840.

not respect logical implication; instead they only guarantee *equisatisfiability* [15]. In-processing inferences have been formalized with sophisticated inference rules including DRAT (*deletion, reverse asymmetric tautology*), PR (*propagation redundancy*) and SPR (*subset* PR) in a series of papers including [15, 11, 10, 28]. These systems can be used both as proof systems to verify unsatisfiability, and as inference systems to facilitate searching for either a satisfying assignment or a proof of unsatisfiability.

The ability to introduce new variables makes DRAT very powerful, and it can simulate extended resolution [16]. Moreover there are recent results [9, 14, 12, 13] indicating that DRAT and PR are still powerful when restricted to use few new variables, or even no new variables. In particular, [14, 12, 13] showed that the pigeonhole principle clauses have short (polynomial size) refutations in the PR proof system; in fact, these refutations can be found *automatically* by an appropriately configured SAT solver [14]. There are at present no good proof search heuristics for how to introduce new variables with the extension rule. It is possible however that there are good heuristics for searching for proofs that do not use new variables in DRAT and PR and related systems. Thus, these systems could potentially lead to dramatic improvements in the power of SAT solvers.

This paper studies these proof systems viewed as refutation systems, paying particular attention to proof systems that do not allow new variables. The proof systems BC (*blocked clauses*), RAT, SPR, PR, and SR are defined below. (Only SR is new to this paper.) These systems have variants which allow deletion, called DBC, DRAT, DSPR, DPR and DSR. There are also variants of all these systems restricted to not allow new variables: we denote these with a superscript "$-$" as BC$^-$, DBC$^-$, RAT$^-$, DRAT$^-$ etc.

Section 2 studies the relation between these systems and extended resolution. We show that any proof system containing BC$^-$ and closed under restrictions simulates extended resolution. A proof system $\mathcal{P}$ *simulates* a proof system $\mathcal{Q}$ if any $\mathcal{Q}$-proof can be converted, in polynomial time, into a $\mathcal{P}$-proof of the same result. It is known that DBC$^-$ simulates DRAT$^-$ [16], and that DRAT simulates DPR with the use of only one extra variable [9]. Section 3.1 shows DRAT$^-$ simulates DPR$^-$. As a consequence, DBC$^-$ can also simulate DPR$^-$. Section 3.2 gives a method to convert SPR$^-$ refutations into PR$^-$ refutations with a size increase that is exponential in the "discrepancy" of the PR inferences. However, in many cases, the discrepancy will be logarithmic or even smaller.

Section 4 proves new polynomial upper bounds on the size of SPR$^-$ proofs for many of the "hard" tautologies from proof complexity. This includes the pigeonhole principle, the bit pigeonhole principle, the parity principle, the clique-coloring principle, and the Tseitin tautologies. We also show that obfuscation by or-fication and xor-ification does not work against SPR$^-$. Note that SPR$^-$ allows neither deletion nor the the use of new variables. Prior results gave SPR$^-$ proofs for the pigeonhole principle (PHP) [12, 13], and PR$^-$ proofs for the Tseitin tautologies and the 2-1 PHP [9]. These results raise the question of whether SPR$^-$ can simulate, for instance, Frege systems.

Section 5 shows RAT$^-$ cannot simulate either DRAT$^-$ or SPR$^-$, by proving size and width lower bounds for RAT$^-$ proofs of the bit pigeonhole principle (BPHP).

Most of the known inclusions for these systems, including our new results, are summarized in (1)-(3). Allowing new variables (and with or without deletion), we have

$$\text{Res} < \text{BC} \equiv \text{RAT} \equiv \text{SPR} \equiv \text{PR} \equiv \text{SR} \equiv \text{ER}. \tag{1}$$

With deletion and no new variables (except ER may use new variables):

$$\text{Res} < \text{DBC}^- \equiv \text{DRAT}^- \equiv \text{DSPR}^- \equiv \text{DPR}^- \leq \text{DSR}^- \leq \text{ER}. \tag{2}$$

With no deletion and no new variables (except ER may use new variables):

$$\text{Res} < \text{BC}^- \leq \text{RAT}^- < \text{SPR}^- \leq^* \text{PR}^- \leq \text{SR}^- \leq \text{ER}. \tag{3}$$

In these equations, equivalence ($\equiv$) indicates the systems simulate each other. Inequality ($\leq$) indicates only one direction is known for the simulation. Strict inequality ($<$) means that it is known there is no simulation in the other direction. The symbol $\leq^*$ in (3) means PR^- simulates SPR^-, and there is a simulation in the other direction under the additional assumption that the discrepancies of PR inferences are logarithmically bounded.

There are still a number of open questions about the systems with no new variables. Of particular importance is the question of the relative strengths of DPR^-, DSR^- and ER. The system DPR^- is a promising system for effective proof search algorithms, and ER is known to be strong. The results of [9, 12, 13] and the present paper show that DPR^- is also strong. Indeed, Section 4 shows even (the possibly weaker) SPR^- is strong.

Another important question is to understand the strength of deletion for these systems. Deletion is well-known to help the performance of SAT solvers in practice, and for systems such as RAT, it is known that deletion can allow new inferences. Our results in Sections 4 and 5 show that in fact RAT^- does not simulate DRAT^-. This strengthens the case for the importance of deletion.

We thank the reviewers for suggestions and comments that improved the paper.

1.1 Preliminaries

We use the usual conventions for clauses, variables, literals, truth assignments, etc. *Var* and *Lit* are the sets of all variables and all literals. A set of literals is called *tautological* if it contains a pair of complementary literals p and $\overline{p}$. A *clause* is a non-tautological set of literals; we use $C, D, \ldots$ to denote clauses. The empty clause is denoted $\bot$, and is always false. 0 and 1 denote respectively *False* and *True*; and $\overline{0}$ and $\overline{1}$ are respectively 1 and 0. We use both $C \cup D$ or $C \vee D$ to denote unions of clauses, but usually write $C \vee D$ when the union is a clause. The notation $C = D \dot{\vee} E$ indicates that $C = D \vee E$ is a clause and D and E have no variables in common. If Γ is a set of clauses, $C \vee \Gamma$ is the set $\{C \vee D : D \in \Gamma \text{ and } C \vee D \text{ is a clause}\}$.

A *partial assignment* τ is a mapping from a set of variables to $\{0, 1\}$. It acts on literals by letting $\tau(\overline{p}) = \overline{\tau(p)}$. We sometimes identify a partial assignment τ with the set of unit clauses asserting that τ holds. For C a clause, $\overline{C}$ denotes the partial assignment whose domain is the variables of C and which asserts that C is false. For example, if $C = x \vee \overline{y} \vee z$ then, depending on context, $\overline{C}$ will denote either the set containing the three unit clauses $\overline{x}$ and y and $\overline{z}$, or the partial assignment α with domain $dom(\alpha) = \{x, y, z\}$ such that $\alpha(x) = \alpha(z) = 1$ and $\alpha(y) = 0$.

A *substitution* generalizes the notion of a partial assignment by allowing variables to be mapped also to literals. Formally, a substitution σ is a map from $Var \cup \{0, 1\}$ to $Lit \cup \{0, 1\}$ which is constant on $\{0, 1\}$. Note that a substitution may cause different

literals to become identified. A partial assignment τ can be viewed as a substitution, by defining $\tau(x) = x$ for all variables x outside the domain of τ.

Suppose C is a clause and σ is a substitution. Let $\sigma(C) = \{\sigma(p) : p \in C\}$. We say σ *satisfies* C, written $\sigma \vDash C$, if $1 \in \sigma(C)$ or $\sigma(C)$ is tautological. When $\sigma \nvDash C$, the *restriction* $C_{|\sigma}$ is defined by letting $C_{|\sigma}$ equal $\sigma(C) \setminus \{0\}$. Thus $C_{|\sigma}$ is a clause expressing the meaning of C under σ. For Γ a set of clauses, the restriction of Γ under σ is denoted $\Gamma_{|\sigma}$ and equals $\{ C_{|\sigma} : C \in \Gamma \text{ and } \sigma \nvDash C \}$. The composition of substitutions τ and π is defined by $(\tau \circ \pi)(x) = \tau(\pi(x))$, and in particular $(\tau \circ \pi)(x) = \pi(x)$ if $\pi(x) \in \{0, 1\}$. For partial assignments τ and π, this means that $dom(\tau \circ \pi) = dom(\tau) \cup dom(\pi)$ and that $(\tau \circ \pi)(x)$ equals $\pi(x)$ for $x \in dom(\pi)$ and $\tau(x)$ for $x \in dom(\tau) \setminus dom(\pi)$.

Lemma 1. *For a set of clauses Γ and substitutions τ and π, $\Gamma_{|\tau\circ\pi} = (\Gamma_{|\pi})_{|\tau}$. In particular, $\tau \vDash \Gamma_{|\pi}$ if and only if $\tau \circ \pi \vDash \Gamma$.*

We write $\Gamma \vDash C$, if every total assignment satisfying Γ also satisfies C. Recall that a *unit propagation* refutation of Γ is a resolution refutation in which, in every resolution step, one of the clauses resolved is a unit clause.

Definition 2. *We write $\Gamma \vdash_1 \bot$ to denote that there is a unit propagation refutation of Γ. We define $\Gamma \vdash_1 C$ to mean $\Gamma \cup \overline{C} \vdash_1 \bot$. For a set of clauses Δ, we write $\Gamma \vdash_1 \Delta$ to mean $\Gamma \vdash_1 C$ for every $C \in \Delta$.*

Fact 3. *If $\Gamma \vdash_1 \bot$ and α is any partial assignment or substitution, then $\Gamma_{|\alpha} \vdash_1 \bot$.*

When $\Gamma \vdash_1 C$, then C is said to be derivable from Γ by *reverse unit propagation* (RUP), or is called an *asymmetric tautology* (AT) with respect to Γ [27, 15, 11]. Of course, $\Gamma \vdash_1 C$ implies that $\Gamma \vDash C$. The advantage of working with $\vdash_1$ is that there is a simple polynomial time algorithm to determine whether $\Gamma \vdash_1 C$. We have:

Lemma 4. *If C is derivable from Γ by a single resolution inference, then $\Gamma \vdash_1 C$. Conversely, if $\Gamma \vdash_1 C$, then some $C' \subseteq C$ has a resolution derivation from Γ of length at most $n + 1$, where n is the number of variables appearing in Γ.*

Lemma 5. *Let $C \vee D$ be a clause (so $C \cup D$ is not tautological), and set $\alpha = \overline{C}$. Then*

$$\Gamma_{|\alpha} \vdash_1 D \setminus C \quad \Longleftrightarrow \quad \Gamma_{|\alpha} \vdash_1 D \quad \Longleftrightarrow \quad \Gamma \vdash_1 C \vee D.$$

1.2 RAT and propagation redundancy

We next describe inference rules which can be used to add a clause C to a set of clauses Γ, maintaining satisfiability. In non-strictly increasing order of strength, they are

$$\text{BC} \to \text{RAT} \to \text{SPR} \to \text{PR} \to \text{SR}.$$

The definitions follow [15, 11, 13], except for the new SR ("substitution redundancy"). All of these rules can be viewed as allowing the introduction of clauses that hold "without loss of generality" [22].

Let Γ be a set of clauses and C a clause with a distinguished literal p, so that C has the form $p \dot{\vee} C'$.

Definition 6. *The clause C is a* blocked clause *(*BC*) with respect to p and Γ if, for every clause D of the form $\overline{p} \vee D'$ in Γ, the set $C' \cup D'$ is tautological.*

Definition 7. *A clause C is a* resolution asymmetric tautology *(*RAT*) with respect to p and Γ if, for every clause D of the form $\overline{p} \vee D'$ in Γ, either $C' \cup D'$ is tautological or $\Gamma \vdash_1 p \vee C' \vee D'$.*

We write $p \vee C'$ instead of C to emphasize that we include the literal p (some definitions of RAT omit it). Clearly, being BC implies being RAT.

Two sets Γ and Π of clauses are *equisatisfiable* if they are both satisfiable or both unsatisfiable. A well-known, important property of BC and RAT is:

Theorem 8. ([15]) *If C is* BC *or* RAT *w.r.t. Γ, then Γ and $\Gamma \cup \{C\}$ are equisatisfiable.*

For the rest of this section, let α be the partial assignment $\overline{C}$.

Definition 9. ([13]) *A clause C is* propagation redundant *(*PR*) with respect to Γ if there is a partial assignment τ such that $\tau \vDash C$ and $\Gamma_{|\alpha} \vdash_1 \Gamma_{|\tau}$.*

Theorem 10. ([13]) *If C is* PR *with respect to Γ, then Γ and $\Gamma \cup \{C\}$ are equisatisfiable.*

Of the remaining rules, SPR is a restriction of PR, but is more general than RAT ([13]). The new substitution redundancy rule (SR) generalizes PR, allowing τ to be a substitution rather than a partial assignment. The condition is still polynomial-time checkable.

Definition 11. ([13]) *A clause C is* subset propagation redundant *(*SPR*) w.r.t. Γ if there is a partial assignment τ with $dom(\tau) = dom(\alpha)$ such that $\tau \vDash C$ and $\Gamma_{|\alpha} \vdash_1 \Gamma_{|\tau}$.*

Definition 12. *A clause C is* substitution redundant *(*SR*) with respect to Γ if there is a substitution τ such that $\tau \vDash C$ and $\Gamma_{|\alpha} \vdash_1 \Gamma_{|\tau}$.*

Theorem 13. *If C is* SPR *or* SR *w.r.t. Γ, then Γ and $\Gamma \cup \{C\}$ are equisatisfiable.*

Proof. This follows by Theorem 10 or, in the case of SR, by an identical proof. □

We next give a technical lemma giving a kind of normal form for propagation redundancy. It implies that if C is PR with respect to Γ, then without loss of generality $dom(\tau)$ includes $dom(\alpha)$.

Lemma 14. *If C is* PR *w.r.t. Γ, witnessed by partial assignment τ, then $\Gamma_{|\alpha} \vdash_1 \Gamma_{|\alpha \circ \tau}$.*

Proof. Let $\pi = \alpha \circ \tau$. Suppose $E \in \Gamma$ is such that $\pi \nvDash E$. We must show that $\Gamma_{|\alpha} \vdash_1 E_{|\pi}$. We can decompose E as $E_1 \vee E_2 \vee E_3$ where E_1 contains the literals in $dom(\tau)$, E_2 the literals in $dom(\alpha) \setminus dom(\tau)$ and E_3 the remaining literals. Then $E_{|\tau} = E_2 \vee E_3$ and by the PR assumption $\Gamma_{|\alpha} \vdash_1 E_{|\tau}$, so there is a derivation $\Gamma_{|\alpha} \cup \overline{E_2} \cup \overline{E_3} \vdash_1 \bot$. But neither $\Gamma_{|\alpha}$ nor $\overline{E_3}$ contain any variables from $dom(\alpha)$, so the literals in $\overline{E_2}$ are not used in this derivation. Hence $\Gamma_{|\alpha} \cup \overline{E_3} \vdash_1 \bot$, which completes the proof since $E_3 = E_{|\pi}$. □

1.3 Proof systems

This section introduces proof systems based on the BC, RAT, SPR, PR and SR inferences. Some of the systems also allow the use of the deletion rule: these systems are denoted DBC, DRAT, etc. All the proof systems are *refutation systems*. They start with a set of clauses Γ, and successively derive sets Γ_i of clauses, first $\Gamma_0 = \Gamma$, then $\Gamma_1, \Gamma_2, \ldots, \Gamma_m$ until reaching a set Γ_m containing the empty clause. It will always be the case that if Γ_i is satisfiable, then Γ_{i+1} is satisfiable. Since the empty clause $\bot$ is in Γ_m, this last set is not satisfiable. This suffices to show that Γ is not satisfiable.

Definition 15. *A* BC*,* RAT*,* SPR*,* PR*, or* SR *proof (or refutation) of* Γ *is a sequence* $\Gamma_0, \ldots, \Gamma_m$ *such that* $\Gamma_0 = \Gamma$*,* $\bot \in \Gamma_m$ *and each* $\Gamma_{i+1} = \Gamma_i \cup \{C\}$*, where either*

- $\Gamma_i \vdash_1 C$ *(that is, "*C *is* RUP *with respect to* Γ_i*"), or*
- C *is* BC*,* RAT*,* SPR*,* PR*, or* SR *(respectively) with respect to* Γ_i*.*

For BC *or* RAT *steps, the proof must specify some* p*, and for* SPR*,* PR *or* SR*, it must specify some* τ*.*

There is no constraint on the variables that appear in clauses C introduced in BC, RAT etc. steps. They are free to include new variables that did not occur in $\Gamma_0, \ldots, \Gamma_i$.

Definition 16. *A* DBC*,* DRAT*,* DSPR*,* DPR*, or* DSR *proof allows the same rules of inference (respectively) as Definition 15, plus the* deletion *inference rule:*

- $\Gamma_{i+1} = \Gamma_i \setminus \{C\}$ *for some* $C \in \Gamma_i$*.*

Since RUP inferences simulate resolution, these systems all simulate resolution. By Theorems 8, 10 and 13, they are sound. Since the inferences are defined using $\vdash_1$ they are polynomial time verifiable, as the description of τ is included with every SPR, PR or SR inference. Hence they are all proof systems in the sense of Cook-Reckhow [6, 7].

The deletion rule allows *any* clause to be deleted, even initial clauses. So it can happen that Γ_i is unsatisfiable but Γ_{i+1} is satisfiable. This is okay for us since we focus on refuting sets of unsatisfiable clauses. Surprisingly, deletion is important because the property of being BC, RAT etc. involves a universal quantification over the current set of clauses Γ_i. Thus deletion can make the systems more powerful, by making more inferences possible. For this, see Corollary 47. (Also, an early paper on this by Kullmann [19] exploited deletions to generalize the power of BC inferences.)

All the systems defined so far are equivalent to extended resolution (ER), because of their ability to freely introduce new variables. The main topic of the paper is the systems in the next definition, which lack this ability.

Definition 17. *A* BC *refutation of* Γ without new variables*, or, for short, a* BC^- *refutation of* Γ*, is a* BC *refutation of* Γ *in which only variables from* Γ *appear. The systems* RAT^-*,* SPR^-*,* PR^-*,* SR^- *and* DBC^-*,* DRAT^-*,* DSPR^-*,* DPR^-*,* DSR^- *are defined similarly.*

2 Relations with extended resolution

2.1 With new variables

It is well-known that RAT, and even BC, can simulate extended resolution (ER) if new variables are allowed. To see this, consider an extended resolution inference which uses the extension rule to introduce a new variable x to stand for the conjunction $p \wedge q$ of two literals. This means that the three extension clauses

$$x \vee \overline{p} \vee \overline{q} \qquad \overline{x} \vee p \qquad \overline{x} \vee q \tag{4}$$

are introduced. We can instead add these clauses using the BC rule. Let Γ be the original set of clauses, and let $\Gamma_1, \Gamma_2, \Gamma_3$ be Γ with the three clauses above successively added. Then $x \vee \overline{p} \vee \overline{q}$ is BC with respect to Γ and x because no clause in Γ contains $\overline{x}$. The clause $\overline{x} \vee p$ is BC with respect to Γ_1 and $\overline{x}$ because the only clause in Γ_1 containing x is $x \vee \overline{p} \vee \overline{q}$, and resolving this with $\overline{x} \vee p$ gives a tautological conclusion. The clause $\overline{x} \vee q$ is BC with respect to Γ_2 and $\overline{x}$ in a similar way. Thus BC, and hence all the other systems which allow new variables, simulate ER. The converse holds as well:

Theorem 18. *The system* ER *simulates* DSR*, and hence every other system above.*

For space reasons, we omit the proof here, but it is known already by [16, 9] that ER simulates DPR. A similar proof works for DSR.

2.2 Without new variables

In the systems without the ability to freely add new variables, we can still imitate extended resolution by adding dummy variables to the formula we want to refute.

For $m \geq 1$, define X^m to be the set consisting of only the two clauses

$$y \vee x_1 \vee \cdots \vee x_m \qquad \text{and} \qquad y.$$

Lemma 19. *Suppose* Γ *has an ER refutation* Π *of size* m*, and that* Γ *and* X^m *have no variables in common. Then* $\Gamma \cup X^m$ *has a* BC^-*-refutation* Π^* *of size* $O(m)$*, which can furthermore be constructed from* Π *in polynomial time.*

Proof. We describe how to change Π into Π^*. We first rename all extension variables to use names from $\{x_1, \ldots, x_m\}$ and replace all resolution steps with $\vdash_1$ inferences. Now consider an extension rule in Π which introduces the three extension clauses (4) expressing $x_i \leftrightarrow (p \wedge q)$, where we may assume that p and q are either variables of Γ or from $\{x_1, \ldots, x_{i-1}\}$. We simulate this by introducing successively the three clauses

$$x_i \vee \overline{p} \vee \overline{q} \qquad \overline{x}_i \vee p \vee \overline{y} \qquad \overline{x}_i \vee q \vee \overline{y}$$

using the BC rule. The first clause, $x_i \vee \overline{p} \vee \overline{q}$, is BC with respect to x_i, because $\overline{x}_i$ has not appeared yet. The second clause is BC with respect to $\overline{x_i}$, because x_i appears only in two earlier clauses, namely $y \vee x_1 \vee \cdots \vee x_m$, which contains y, and $x_i \vee \overline{p} \vee \overline{q}$, which contains $\overline{p}$. In both cases the resolvent with $\overline{x}_i \vee p \vee \overline{y}$ is tautological. The third clause is similar. The unit clause y is in X^m, so we can then derive the remaining two needed extension clauses $\overline{x}_i \vee p$ and $\overline{x}_i \vee q$ by two $\vdash_1$ inferences. □

As the next corollary shows, this lemma can be used to construct examples of usually-hard formulas which have short proofs in BC^-. (We will give less artificial examples of short SPR^- proofs in Section 4.) Let $m(n)$ be the polynomial size upper bound on ER refutations of the pigeonhole principle PHP_n which follows from [7].

Corollary 20. *The set of clauses* $\mathrm{PHP}_n \cup X^{m(n)}$ *has polynomial size proofs in* BC^-*, but requires exponential size proofs in constant depth Frege.*

Proof. The upper bound is by Lemma 19. For the lower bound, let Π be a refutation in depth-d Frege. Then we can restrict Π by setting $y = 1$ to obtain a depth-d refutation of PHP_n which, by [18, 20], must have exponential size. □

The same argument can give a more general result. A propositional proof system $\mathcal{P}$ is *closed under restrictions* if given any $\mathcal{P}$-refutation of Γ and any partial assignment ρ, we can construct a $\mathcal{P}$-refutation of $\Gamma_{|\rho}$ in polynomial time.

Theorem 21. *Let* $\mathcal{P}$ *be any propositional proof system which is closed under restrictions. If* $\mathcal{P}$ *simulates* BC^-*, then* $\mathcal{P}$ *simulates ER.*

Proof. Suppose Γ has a refutation Π in ER of length m. Take a copy of X^m in disjoint variables from Γ. By Lemma 19 we can construct a BC^--refutation of $\Gamma \cup X^m$. Since $\mathcal{P}$ simulates BC^-, we can then construct a $\mathcal{P}$-refutation of $\Gamma \cup X^m$. Let ρ be the restriction which just sets $y = 1$, so that $(\Gamma \cup X^m)_{|\rho} = \Gamma$. Since $\mathcal{P}$ is closed under restrictions, we can construct a $\mathcal{P}$-refutation of Γ. All constructions are polynomial time. □

Corollary 22. *If, as is expected, the Frege proof system is strictly weaker than* ER*, then Frege does not simulate* BC^-.

3 Simulations

3.1 DRAT^- simulates DPR^-

The following relations were known between DBC^-, DRAT^- and DPR^-.

Theorem 23. ([16]) DBC^- *simulates* DRAT^-.

Theorem 24. ([9]) *Suppose* Γ *has a* DPR *refutation* Π*. Then it has a* DRAT *refutation constructible in polynomial time from* Π*, using at most one variable not used in* Π*.*

We will show, in Theorem 30 below, that DRAT^- simulates DPR^-. Thus the systems DBC^-, DRAT^-, DSPR^- and DPR^- are all equivalent. Our proof relies on the main step in the proof of Theorem 24:

Lemma 25. ([9]) *Suppose* C *is* PR *w.r.t.* Γ*. Then there is a polynomial size* DRAT *derivation of* $\Gamma \cup \{C\}$ *from* Γ*, using at most one variable not appearing in* Γ *or* C*.*

Definition 26. *Let* Γ *be a set of clauses and* x *any variable. Then* $\Gamma^{(x)}$ *consists of every clause in* Γ *which does not mention* x*, together with every clause of the form* $E \vee F$ *where both* $x \vee E$ *and* $\overline{x} \vee F$ *are in* Γ*.*

In other words, $\Gamma^{(x)}$ is formed from Γ by doing all possible resolutions with respect to x and then deleting all clauses containing either x or $\overline{x}$. (This is exactly like the first step of the Davis-Putnam procedure.)

Lemma 27. *There is a polynomial size* DRAT *derivation of* Γ *from* $\Gamma^{(x)}$*, using only variables from* Γ.

Proof. We first derive every clause of the form $E \dot{\vee} x$ in Γ, by RAT on x. As $\overline{x}$ has not appeared yet the RAT condition is satisfied. Then we derive each clause of the form $F \dot{\vee} \overline{x}$ in Γ, by RAT on $\overline{x}$. The only possible resolutions are with clauses of the form $E \dot{\vee} x$ which we have just introduced, but in this case either $E \cup F$ is tautological or $E \vee F$ is in $\Gamma^{(x)}$ so $\Gamma^{(x)} \vdash_1 \overline{x} \vee F \vee E$. Finally we delete all clauses not in Γ. □

The next two lemmas show that, under suitable conditions, if we can derive C from Γ in DPR$^-$, then we can derive it from $\Gamma^{(x)}$. We will use a kind of normal form for PR inferences. Say that a clause C is PR$_0$ with respect to Γ if there is a partial assignment τ such that $\tau \vDash C$, all variables in C are in $dom(\tau)$, and

$$C \vee \Gamma_{|\tau} \subseteq \Gamma. \tag{5}$$

The PR$_0$ inference rule lets us derive $\Gamma \cup \{C\}$ from Γ when (5) holds. It is not hard to see that (5) implies $\Gamma_{|\alpha} \vdash_1 \Gamma_{|\tau}$, where $\alpha = \overline{C}$, so this is a special case of the PR rule.

Lemma 28. *Any* PR *inference can be replaced with a* PR$_0$ *inference together with polynomially many* $\vdash_1$ *and deletion steps, using no new variables.*

Proof. Suppose $\Gamma_{|\alpha} \vdash_1 \Gamma_{|\tau}$, where $\alpha = \overline{C}$ and $\tau \vDash C$. By Lemma 14 we may assume $dom(\alpha) \subseteq dom(\tau)$ so $dom(\tau)$ contains all variables in C. Let $\Delta = C \vee \Gamma_{|\tau}$ and $\Gamma^* = \Gamma \cup \Delta$. Note that $\Delta_{|\tau}$ is empty, as τ satisfies C. This implies that $C \vee \Gamma^*_{|\tau} = C \vee \Gamma_{|\tau} \subseteq \Gamma^*$, so C is PR$_0$ w.r.t. Γ^*. Furthermore the condition $\Gamma_{|\alpha} \vdash_1 \Gamma_{|\tau}$ implies that every clause in Δ is derivable from Γ by a $\vdash_1$ step, by Lemma 5. Thus we can derive Γ^* from Γ by $\vdash_1$ steps, then introduce C by the PR$_0$ rule, and recover $\Gamma \cup \{C\}$ by deleting everything else. □

Lemma 29. *Suppose* C *is* PR$_0$ *with respect to* Γ*, witnessed by* τ *with* $x \notin dom(\tau)$*. Then* C *is* PR$_0$ *with respect to* $\Gamma^{(x)}$.

Proof. The PR$_0$ condition implies the variable x does not occur in C. We are given that $C \vee \Gamma_{|\tau} \subseteq \Gamma$ and want to show $C \vee (\Gamma^{(x)})_{|\tau} \subseteq \Gamma^{(x)}$. Let $D \in \Gamma^{(x)}$ with $\tau \not\vDash D$. First suppose D is in Γ and x does not occur in D. Then $C \vee D_{|\tau} \in \Gamma$ by assumption, so $C \vee D_{|\tau} \in \Gamma^{(x)}$. Otherwise, $D = E \vee F$ where both $E \dot{\vee} x$ and $F \dot{\vee} \overline{x}$ are in Γ. Then by assumption both $C \vee E_{|\tau} \vee x$ and $C \vee F_{|\tau} \vee \overline{x}$ are in Γ. Hence $C \vee D_{|\tau} = C \vee E_{|\tau} \vee F_{|\tau} \in \Gamma^{(x)}$. □

Theorem 30. DRAT$^-$ *simulates* DPR$^-$.

Proof. We are given a DPR$^-$ refutation of some set Δ, using only the variables in Δ. By Lemma 28 we may assume without loss of generality that the refutation uses only $\vdash_1$, deletion and PR$_0$ steps. Consider a PR$_0$ inference in this refutation, which derives $\Gamma \cup \{C\}$ from a set of clauses Γ, witnessed by a partial assignment τ. We want to derive $\Gamma \cup \{C\}$ from Γ in DRAT using only variables in Δ.

Suppose τ is a total assignment to all variables in Γ. The set Γ is necessarily unsatisfiable, or it could not occur as a line in a refutation. Therefore $\Gamma_{|\tau}$ is simply $\bot$, so the PR_0 condition tells us that $C \in \Gamma$ and we do not need to do anything.

Otherwise, there is some variable x which occurs in Γ but is outside the domain of τ, and thus in particular does not occur in C. We first use $\vdash_1$ and deletion steps to replace Γ with $\Gamma^{(x)}$. By Lemma 29, C is PR_0, and thus PR, with respect to $\Gamma^{(x)}$. By Lemma 25 there is a short DRAT derivation of $\Gamma^{(x)} \cup \{C\}$ from $\Gamma^{(x)}$, using one new variable which does not occur in $\Gamma^{(x)}$ or C. We choose x for this variable. Finally, observing that here $\Gamma^{(x)} \cup \{C\} = (\Gamma \cup \{C\})^{(x)}$, we recover $\Gamma \cup \{C\}$ using Lemma 27. □

3.2 Towards a simulation of PR^- by SPR^-

Our next result shows how to replace a PR inference with SPR inferences, without additional variables. It is not a polynomial simulation of PR^- by SPR^- however, as it depends exponentially on the "discrepancy" as defined next. Recall that C is PR w.r.t. Γ if $\Gamma_{|\alpha} \vdash_1 \Gamma_{|\tau}$, where $\alpha = \overline{C}$ and τ is a partial assignment satisfying C. We will keep this notation throughout this section. C is SPR w.r.t. Γ if additionally $dom(\tau) = dom(\alpha)$.

Definition 31. *The* discrepancy *of a* PR *inference is* $|dom(\tau) \setminus dom(\alpha)|$. *That is, it is the number of variables which are assigned by* τ *but not by* α.

Theorem 32. *Suppose that* Γ *has a* PR *refutation* Π *of size* S *in which every* PR *inference has discrepancy bounded by* δ. *Then* Γ *has a* SPR *refutation of size* $O(2^\delta S)$ *which does not use any variables not present in* Π.

If the discrepancy is logarithmically bounded, Theorem 32 gives polynomial size SPR refutations automatically. We need a couple of lemmas before proving the theorem.

Lemma 33. *Suppose* $\Gamma_{|\alpha} \vdash_1 \Gamma_{|\tau}$ *and* α^+ *is a partial assignment extending* α, *such that* $dom(\alpha^+) \subseteq dom(\tau)$. *Then* $\Gamma_{|\alpha^+} \vdash_1 \Gamma_{|\tau}$

Proof. Suppose $E \in \Gamma_{|\tau}$. Then E contains no variables from α^+ and by assumption there is a refutation $\Gamma_{|\alpha}, \overline{E} \vdash_1 \bot$. Thus $\Gamma_{|\alpha^+}, \overline{E} \vdash_1 \bot$, by Fact 3. □

Definition 34. *A clause* C subsumes *a clause* D *if* $C \subseteq D$. *A set* Γ *of clauses* subsumes *a set* Γ' *if each clause of* Γ' *is subsumed by some clause of* Γ.

Lemma 35. *Suppose* $\Gamma \subseteq \Gamma'$ *and* Γ *subsumes* Γ'. *Suppose* α *and* τ *are substitutions and* $\Gamma_{|\alpha} \vdash_1 \Gamma_{|\tau}$ *holds. Then* $\Gamma'_{|\alpha} \vdash_1 \Gamma'_{|\tau}$. *Consequently, if* C *can be inferred from* Γ *by an* SPR, PR *or* SR *rule, then* C *can also be inferred from* Γ' *by the same rule.*

Proof. Suppose $D \in \Gamma'$ and $\tau \not\models D$. We must show $\Gamma'_{|\alpha} \vdash_1 D_{|\tau}$. Let $E \in \Gamma$ with $E \subseteq D$. Then $\tau \not\models E$, so by assumption $\Gamma_{|\alpha} \vdash_1 E_{|\tau}$. Also $E_{|\tau} \subseteq D_{|\tau}$, so $\Gamma_{|\alpha} \vdash_1 D_{|\tau}$. It follows that $\Gamma'_{|\alpha} \vdash_1 D_{|\tau}$. □

Proof (of Theorem 32). Our main task is to show that a PR inference with discrepancy at most δ can be simulated by multiple SPR inferences, while bounding the increase in proof size in terms of δ. Suppose C is derivable from Γ by a PR inference. That is,

$\Gamma_{|\alpha} \vdash_1 \Gamma_{|\tau}$ where $\alpha = \overline{C}$ and $\tau \vDash C$; by Lemma 14 we may assume that $dom(\tau) \supseteq dom(\alpha)$. List the variables in $dom(\tau) \setminus dom(\alpha)$ as $p_1, \ldots, p_s$, where $s \leq \delta$.

Enumerate as $D_1, \ldots, D_{2^s}$ all clauses containing exactly the variables $p_1, \ldots, p_s$ with some pattern of negations. Let $\sigma_i = \overline{C \vee D_i}$, so that $\sigma_i \supseteq \alpha$ and $dom(\sigma_i) = dom(\tau)$. By Lemma 33, $\Gamma_{|\sigma_i} \vdash_1 \Gamma_{|\tau}$. Since $\tau \vDash C \vee D_j$ for every j, in fact $\Gamma_{|\sigma_i} \vdash_1 (\Gamma \cup \{C \vee D_1, \ldots, C \vee D_{i-1}\})_{|\tau}$. Thus we may introduce all clauses $C \vee D_1, \ldots, C \vee D_{2^s}$ one after another by SPR inferences. We can then use $2^s - 1$ resolution steps to derive C.

The result is a set $\Gamma' \supseteq \Gamma$ which contains C plus extra clauses subsumed by C. By Lemma 35 these extra clauses do not affect the validity of later PR inferences. □

4 Upper bounds for some hard tautologies

This section proves that SPR$^-$ — without new variables — can give polynomial size refutations for essentially all the usual "hard" propositional principles. Heule, Kiesl and Biere [13, 12] showed that the tautologies based on the pigeonhole principle (PHP) and the 2-1 pigeonhole principle have polynomial size SPR$^-$ proofs, and Heule and Biere discuss polynomial size PR$^-$ proofs of the Tseitin tautologies in [9]. The SPR$^-$ proof of the PHP tautologies can be viewed as a version of the original extended resolution proof of PHP given by Cook and Reckhow [7]. Here we describe polynomial size SPR$^-$ proofs for several well-known principles. We also show that orification and xorification can be handled in SPR$^-$. This is surprising since the proofs contain only clauses in the original literals, and it is well-known that such clauses are limited in what they can express.

It is open whether extended resolution, or the Frege proof system, can be simulated by PR$^-$ or DPR$^-$, or more generally by DSR$^-$. The examples below show that any separation of these systems must involve a new technique.

For space reasons, we omit the proofs and the descriptions of the clauses for Theorems 38, 41, 42 and 43. They can be found in the full version of the paper. We include the proof of Theorem 39 for the bit pigeonhole principle as an example, as it is relatively easy, and is used in Section 5.

Definition 36. *A* Γ-symmetry *is an invertible substitution* π *such that* $\Gamma_{|\pi} = \Gamma$.

If π is a Γ-symmetry and $\alpha = \overline{C}$ is a partial assignment, then by Lemma 1 we have $\Gamma_{|\alpha} = (\Gamma_{|\pi})_{|\alpha} = \Gamma_{|\alpha\circ\pi}$. Hence, if $\alpha\circ\pi \vDash C$, we can infer C from Γ by an SR inference with $\tau = \alpha\circ\pi$. If furthermore π is the identity outside $dom(\alpha)$, then $\alpha\circ\pi$ behaves as a partial assignment and $dom(\alpha\circ\pi) = dom(\alpha)$, so this becomes an SPR inference.

Below we write $\overline{\alpha}$ for the clause expressing that the partial assignment α does not hold (so $C = \overline{\alpha}$ if and only if $\alpha = \overline{C}$).

Lemma 37. *Suppose* $(\alpha_0, \tau_0), \ldots, (\alpha_m, \tau_m)$ *is a sequence of pairs of partial assignments such that for each i,*

1. $\Gamma_{|\alpha_i} = \Gamma_{|\tau_i}$
2. α_i *and* τ_i *are contradictory and have the same domain*
3. *for all* $j < i$, *either* α_j *and* τ_i *are disjoint or they are contradictory.*

Then we can derive $\Gamma \cup \{\overline{\alpha_i} : i = 0, \ldots, m\}$ *from* Γ *by a sequence of* SPR *inferences.*

Proof. We write C_i for $\overline{\alpha_i}$. By item 2, $\tau_i \vDash C_i$. Thus it is enough to show that for each i,

$$\big(\Gamma \cup \{C_0, \dots, C_{i-1}\}\big)_{|\alpha_i} \supseteq \big(\Gamma \cup \{C_0, \dots, C_{i-1}\}\big)_{|\tau_i}.$$

We have $\Gamma_{|\alpha_i} = \Gamma_{|\tau_i}$. For $j < i$, either α_j and τ_i are disjoint, and so $(C_j)_{|\alpha_i} = (C_j)_{|\tau_i} = C_j$, or they are contradictory and so $\tau_i \vDash C_j$ and C_j vanishes from the right hand side. □

4.1 Pigeonhole principles

Let $n \geq 1$. The *pigeonhole principle* PHP_n asserts that $n+1$ pigeons can be mapped to n holes with no collisions.

Theorem 38 ([13]). PHP_n *has polynomial size* SPR^- *refutations.*

Let $n = 2^k$. The *bit pigeonhole principle* contradiction, BPHP_n, asserts that each of $n+1$ pigeons can be assigned a distinct k-bit binary string. For each pigeon x, $0 \leq x < n+1$, it has variables $p_1^x, \dots, p_k^x$ for the bits of the string assigned to x. We think of strings $y \in \{0,1\}^k$ as holes. When convenient we will identify holes with numbers $y < n$. We write $(x \to y)$ for the conjunction $\bigwedge_i (p_i^x = y_i)$ asserting that pigeon x goes to hole y. We write $(x \not\to y)$ for its negation $\bigvee_i (p_i^x \neq y_i)$. The axioms of BPHP_n are then

$$(x \not\to y) \vee (x' \not\to y) \quad \text{for all holes } y \text{ and all distinct pigeons } x, x'.$$

The set $\{(x \not\to y) : y < n\}$ consists of the 2^k clauses containing the variables $p_1^x, \dots, p_k^x$ with all patterns of negations. We can derive $\bot$ from them in $2^k - 1$ resolution steps.

Theorem 39. *The* BPHP_n *clauses have polynomial size* SPR^- *refutations.*

The theorem is proved below. It is essentially the same as the proof of PHP in [13] (or Theorem 38 above). For each $m < n-1$ and each pair $x, y > m$, we define a clause

$$C_{m,x,y} := (m \not\to y) \vee (x \not\to m).$$

Let Γ be the set of all such clauses $C_{m,x,y}$. We will show these clauses can be introduced by SPR inferences, but first we show they suffice to derive BPHP_n.

Lemma 40. $\mathrm{BPHP}_n \cup \Gamma$ *has a polynomial size resolution refutation.*

Proof. Using induction on $m = 0, 1, 2, \dots, n-1$ we derive all clauses $\{(x \not\to m) : x > m\}$. So suppose $m < n$ and $x > m$. For each $y > m$, we have the clause $(m \not\to y) \vee (x \not\to m)$, as this is $C_{m,x,y}$. We also have the clause $(m \not\to m) \vee (x \not\to m)$, as this is an axiom of BPHP_n. Finally, for each $m' < m$, we have $(m \not\to m')$ by the inductive hypothesis (or, in the base case $m = 0$, there are no such clauses). Resolving all these together gives $(x \not\to m)$.

This derives all clauses in $\{(n \not\to m) : m < n\}$. Resolving these yields $\bot$. □

Thus it is enough to show that we can introduce all clauses in Γ using SPR inferences. We use Lemma 37. For $m < n-1$ and each pair $x, y > m$, define partial assignments

$$\alpha_{m,x,y} := (m \to y) \wedge (x \to m) \qquad \text{and} \qquad \tau_{m,x,y} := (m \to m) \wedge (x \to y)$$

so that $C_{m,x,y} = \overline{\alpha_{m,x,y}}$ and $\tau_{m,x,y} = \alpha_{m,x,y} \circ \pi$ where π swaps all variables for pigeon m and x. Hence $(\mathrm{BPHP}_n)_{|\alpha_{m,x,y}} = (\mathrm{BPHP}_n)_{|\tau_{m,x,y}}$ as required.

For the other conditions for Lemma 37, first observe that assignments $\alpha_{m,x,y}$ and $\tau_{m,x',y'}$ are always inconsistent, since they map m to different places. Now suppose that $m' < m$ and $\alpha_{m,x,y}$ and $\tau_{m',x',y'}$ are not disjoint. Then they must have some pigeon in common, so either $m' = x$ or $x' = x$. In both cases $\tau_{m',x',y'}$ contradicts $(x \to m)$, in the first case because it maps x to m' and in the second because it maps x to y' with $y' > m'$.

4.2 Other tautologies

The *parity principle* states that there is no (undirected) graph on an odd number of vertices in which each vertex has degree exactly one (see [1, 2]). For n odd, let PAR_n be a set of clauses expressing (a violation of) the parity principle on n vertices.

Theorem 41. *The* PAR_n *clauses have polynomial size* SPR^- *refutations.*

The *clique-coloring principle* $\mathrm{CC}_{n,m}$ states, informally, that a graph with n vertices cannot have both a clique of size m and a coloring of size $m - 1$ (see [17, 21]).

Theorem 42. *The* $\mathrm{CC}_{n,m}$ *clauses have polynomial size* SPR^- *refutations.*

The *Tseitin tautologies* $\mathrm{TS}_{G,\gamma}$ are hard examples for many proof systems (see [24, 25]). Let G be an undirected graph with n vertices, with each vertex i labelled with a charge $\gamma(i) \in \{0, 1\}$ such that the total charge on G is odd. For each edge e of G there is a variable x_e. Then $\mathrm{TS}_{G,\gamma}$ is the CNF consisting of clauses expressing that, for each vertex i, the parity of the values x_e over the edges e touching i is equal to the charge $\gamma(i)$. For a vertex i of degree d, this requires 2^{d-1} clauses, using one clause to rule out each assignment to the edges touching i with the wrong parity. If G has constant degree then this has size polynomial in n. It is well-known to be unsatisfiable.

Theorem 43. *The* $\mathrm{TS}_{G,\gamma}$ *clauses have polynomial size* SPR^- *refutations.*

Orification and xorification have also been used to make hard instances of propositional tautologies. (See [4, 3, 26].) Nonetheless, SPR-inferences can be used to "undo" the effects of orification and xorification, without using any new variables. (This is argued in the full version of the paper.) As a consequence, these techniques are not likely to be helpful in establishing lower bounds for the size of PR refutations.

The principles considered above exhaust most of the known "hard" tautologies that have been shown to require exponential size, constant depth Frege proofs. It is open whether SPR^- or SR^- simulates Frege; and by the above results, any separation of SPR^- and Frege systems will likely require new techniques.

Paul Beame [private comm., 2018] suggested the graph PHP principles (see [5]) may separate systems such as SPR^-, or even SR^-, from Frege systems. However, it is plausible that the graph PHP principles also have short SPR^- proofs. Namely, SPR inferences can infer a lot of clauses from the graph PHP clauses. If an instance of graph PHP has every pigeon with outdegree ≥ 2, then there must be an alternating cycle of pigeons $i_1, \ldots i_{\ell+1}$ and holes $j_1, \ldots j_\ell$ such that $i_\ell = i_1$, the edges (i_s, j_s) and (i_{s+1}, j_s)

are all in the graph, and $\ell = O(\log n)$. An SPR inference can be used to learn the clause $\overline{x_{i_1,j_1}} \vee \overline{x_{i_2,j_2}} \vee \cdots \vee \overline{x_{i_\ell,j_\ell}}$, by using the fact that a satisfying assignment that falsifies this clause can be replaced by the assignment that maps instead each pigeon i_{s+1} to hole j_s.

This allows SPR inferences to infer many clauses from the graph PHP clauses. However, it remains open whether a polynomial size SPR$^-$ refutation exists.

5 Lower bounds

This section gives an exponential separation between DRAT$^-$ and RAT$^-$, by showing that the bit pigeonhole principle BPHP$_n$ requires exponential size refutations in RAT$^-$. This lower bound still holds if we allow some deletions, as long as no initial clause of BPHP$_n$ is deleted. On the other hand, with unrestricted deletions, Theorems 23, 30 and 39 imply that BPHP$_n$ has polynomial size refutations in DRAT$^-$ and even DBC$^-$, and Theorem 39 shows that it has polynomial size SPR$^-$ refutations.

We define the *pigeon-width* of a clause or assignment to be the number of distinct pigeons that it mentions. Our lower bound proof uses a conventional strategy: we first show a width lower bound (on pigeon-width), and then use a random restriction argument to show that a proof of subexponential size can be made into one of small pigeon-width. However, RAT$^-$ refutation size may not behave well under restrictions (see Section 2.2). So, rather than using restrictions directly to reduce width, we will define a partial random matching ρ of pigeons to holes and show that if BPHP$_n$ has a RAT$^-$ refutation of small size, then BPHP$_n \cup \rho$ has one of small pigeon-width.

We will sometimes identify resolution refutations of Γ with winning strategies for the Prover in the Prover-Adversary game on Γ, in which the Adversary claims to know a satisfying assignment and the Prover tries to force her into a contradiction by querying variables; the Prover can also forget variables to save memory and simplify his strategy.

Lemma 44. *Let β be a partial assignment corresponding to a partial matching of m pigeons to holes. Then* BPHP$_n \cup \beta$ *requires pigeon-width $n+1-m$ to refute in resolution.*

Proof. A refutation of pigeon-width less than $n + 1 - m$ would give a Prover-strategy in which the Prover never has information about more than $n - m$ pigeons; namely, traverse Π upwards from $\bot$ to an initial clause, remembering only the values of variables mentioned in the current clause. This strategy is easy for the Adversary to defeat, as BPHP$_n \cup \beta$ is essentially the pigeonhole principle with $n - m$ holes. □

Theorem 45. *Let ρ be a partial matching of size at most $n/4$. Let Π be a* DRAT *refutation of* BPHP$_n \cup \rho$ *in which no new variables are introduced and no clause of* BPHP$_n$ *is ever deleted. Then some clause in Π has pigeon-width more than $n/3$.*

Proof. Suppose for a contradiction there is a such a refutation Π in pigeon-width $n/3$. We consider each RAT inference in Π in turn, and show that it can be eliminated and replaced with standard resolution reasoning, without increasing the pigeon-width.

Inductively suppose Γ is a set of clauses derivable from BPHP$_n \cup \rho$ in pigeon-width $n/3$, using only resolution and weakening. Suppose a clause C in Π of the form $p \dot\vee C'$ is RAT w.r.t. Γ and p. Let $\alpha = \overline{C}$, so $\alpha(p) = 0$ and α mentions at most $n/3$ pigeons. We consider three cases.

Case 1: the assignment α is inconsistent with ρ. This means that ρ satisfies a literal which appears in C, so C can be derived from ρ by a single weakening step.

Case 2: the assignment $\alpha \cup \rho$ can be extended to a partial matching β of the pigeons it mentions. We will show that this cannot happen. Let x be the pigeon associated with the literal p. Let $y = \beta(x)$ and let y' be the hole β would map x to if the bit p were flipped to 1. If $y' = \beta(x')$ for some pigeon x' in the domain of β, let $\beta' = \beta$. Otherwise let $\beta' = \beta \cup \{(x', y')\}$ for some pigeon x' outside the domain of β.

Let H be the hole axiom $(x \nrightarrow y') \vee (x' \nrightarrow y')$ in Γ. The clause $(x \nrightarrow y')$ contains the literal $\overline{p}$, since $(x \rightarrow y')$ contains p. So $H = \overline{p} \vee H'$ for some clause H'. By the RAT condition, either $C' \cup H'$ is a tautology or $\Gamma \vdash_1 C \vee H'$. Either way, $\Gamma \cup \overline{C} \cup \overline{H'} \vdash_1 \bot$. Since $\beta' \supseteq \alpha$, β' falsifies C. It also falsifies H', since it satisfies $(x \rightarrow y') \wedge (x' \rightarrow y')$ except at p. It follows that $\Gamma \cup \beta' \vdash_1 \bot$. By assumption, Γ is derivable from $\mathrm{BPHP}_n \cup \rho$ in pigeon-width $n/3$, and $\beta' \supseteq \rho$. As unit propagation does not increase pigeon-width, this implies that $\mathrm{BPHP}_n \cup \beta'$ is refutable in resolution in pigeon-width $n/3$, by first deriving Γ and then using unit propagation. This contradicts Lemma 44 as β' is a matching of at most $n/3 + n/4 + 1$ pigeons.

Case 3: the assignment $\alpha \cup \rho$ cannot be extended to a partial matching of the pigeons it mentions. Consider a position in the Prover-Adversary game on $\mathrm{BPHP}_n \cup \rho$ in which the Prover knows α. The Prover can ask all remaining bits of the pigeons mentioned in α, and since there is no suitable partial matching this forces the Adversary to reveal a collision and lose the game. This strategy has pigeon-width $n/3$; it follows that C is derivable from $\mathrm{BPHP}_n \cup \rho$ in resolution in this pigeon-width. □

Theorem 46. *Let* Π *be a* DRAT^- *refutation of* BPHP_n *in which no clause of* BPHP_n *is ever deleted. Then* Π *has size at least* $2^{n/80}$.

Proof. Construct a random restriction ρ by selecting each pigeon independently with probability $1/5$ and then randomly matching them with distinct holes. Let $m = n/4$. Let C be a clause mentioning at least m distinct pigeons $x_1, \ldots, x_m$ and choose literals $p_1, \ldots, p_m$ in C such that p_i belongs to pigeon x_i. The probability p_i is satisfied by ρ is $1/10$. These events are not quite independent for different p_i, as the holes used by other pigeons are blocked for pigeon x_i. But since $m = n/4$, fewer than half of the holes that would satisfy p_i are blocked. That is, the probability that p_i is satisfied by ρ, on the worst-case condition that all other literals p_j are not satisfied by ρ, is at least $1/20$. Therefore the probability that C is not satisfied by ρ is at most $(1 - 1/20)^m < e^{-m/20} = e^{-n/80}$.

Now suppose Π contains no more than $2^{n/80}$ clauses. By the union bound, there is some restriction ρ which satisfies all clauses in Π of pigeon-width at least $n/4$, and by the Chernoff bound we may assume that ρ sets no more than $n/4$ pigeons.

We now observe inductively that for each clause C in Π, some subclause of C is derivable from $\mathrm{BPHP}_n \cup \rho$ in resolution in pigeon-width $n/3$, ultimately contradicting Lemma 44. If C has pigeon-width more than $n/3$, this follows because C is subsumed by ρ. Otherwise, if C is derived by a RAT inference, we repeat the proof of Theorem 45; in case 2 we additionally use the observation that if $\Gamma \vdash_1 C \vee H'$ and Γ' subsumes Γ, then $\Gamma' \vdash_1 C \vee H'$. □

Corollary 47. RAT^- *does not simulate* DRAT^-. RAT^- *does not simulate* SPR^-.

References

1. Ajtai, M.: Parity and the pigeonhole principle. In: Feasible Mathematics: A Mathematical Sciences Institute Workshop held in Ithaca, New York, June 1989. pp. 1–24. Birkhäuser (1990)
2. Beame, P., Impagliazzo, R., Krajíček, J., Pitassi, T., Pudlák, P.: Lower bounds on Hilbert's Nullstellensatz and propositional proofs. Proceedings of the London Mathematical Society **73**(3), 1–26 (1996)
3. Ben-Sasson, E.: Size space tradeoffs for resolution. SIAM Journal on Computing **38**(6), 2511–2525 (2009)
4. Ben-Sasson, E., Impagliazzo, R., Wigderson, A.: Near optimal separation of tree-like and general resolution. Combinatorica **24**(4), 585–603 (2004)
5. Ben-Sasson, E., Wigderson, A.: Short proofs are narrow — resolution made simple. Journal of the ACM **48**, 149–169 (2001)
6. Cook, S.A., Reckhow, R.A.: On the lengths of proofs in the propositional calculus, preliminary version. In: Proceedings of the Sixth Annual ACM Symposium on the Theory of Computing. pp. 135–148 (1974)
7. Cook, S.A., Reckhow, R.A.: The relative efficiency of propositional proof systems. Journal of Symbolic Logic **44**, 36–50 (1979)
8. Goldberg, E.I., Novikov, Y.: Verification of proofs of unsatisfiability for CNF formulas. In: Design, Automation and Test in Europe Conference (DATE). pp. 10886–10891. IEEE Computer Society (2003)
9. Heule, M.J.H., Biere, A.: What a difference a variable makes. In: Tools and Algorithms for the Construction and Analysis of Systems - 24th International Conference (TACAS). pp. 75–92. Lecture Notes in Computer Science 10806, Springer Verlag (2018)
10. Heule, M.J.H., Hunt Jr., W.A., Wetzler, N.: Trimming while checking clausal proofs. In: Formal Methods in Computer-Aided Design (FMCAD). pp. 181–188. IEEE (2013)
11. Heule, M.J.H., Hunt Jr., W.A., Wetzler, N.: Verifying refutations with extended resolution. In: Automated Deduction - 24th International Conference (CADE). pp. 345–359. Lecture Notes in Computer Science 7898, Springer Verlag (2013)
12. Heule, M.J.H., Kiesl, B., Biere, A.: Short proofs without new variables. In: Automated Deduction - 26th International Conference (CADE). pp. 130–147. Lecture Notes in Computer Science 10395, Springer Verlag (2017)
13. Heule, M.J.H., Kiesl, B., Biere, A.: Strong extension-free proof systems. Journal of Automated Reasoning pp. 1–22 (2019). https://doi.org/10.1007/s10817-019-09516-0, extended version of [12]
14. Heule, M.J.H., Kiesl, B., Seidl, M., Biere, A.: PRuning through satisfaction. In: Hardware and Software: Verification and Testing - 13th International Haifa Verification Conference (HVC). pp. 179–194. Lecture Notes in Computer Science 10629, Springer Verlag (2017)
15. Järvisalo, M., Heule, M.J.H., Biere, A.: Inprocessing rules. In: Automated Reasoning - 6th International Joint Conference (IJCAR). pp. 355–270. Lecture Notes in Computer Science 7364, Springer Verlag (2012)
16. Kiesl, B., Rebola-Pardo, A., Heule, M.J.H.: Extended resolution simulates DRAT. In: Automated Reasoning - 6th International Joint Conference (IJCAR). pp. 516–531. Lecture Notes in Computer Science 10900, Springer Verlag (2018)
17. Krajíček, J.: Interpolation theorems, lower bounds for proof systems, and independence results for bounded arithmetic. Journal of Symbolic Logic **62**, 457–486 (1997)
18. Krajíček, J., Pudlák, P., Woods, A.: Exponential lower bound to the size of bounded depth Frege proofs of the pigeonhole principle. Random Structures and Algorithms **7**, 15–39 (1995)
19. Kullmann, O.: On a generalizaton of extended resolution. Discrete Applied Mathematics **96-97**, 149–176 (1999)

20. Pitassi, T., Beame, P., Impagliazzo, R.: Exponential lower bounds for the pigeonhole principle. Computational Complexity **3**, 97–140 (1993)
21. Pudlák, P.: Lower bounds for resolution and cutting planes proofs and monotone computations. Journal of Symbolic Logic **62**, 981–998 (1997)
22. Rebola-Pardo, A., Suda, M.: A theory of satisfiability-preserving proofs in SAT solving. In: Proc., 22nd Intl. Conf. on Logic for Programming, Artificial Intelligence and Reasoning (LPAR'22). pp. 583–603. EPiC Series in Computing 57, EasyChair (2018)
23. Siekmann, J., Wrightson, G.: Automation of Reasoning, vol. 1&2. Springer-Verlag, Berlin (1983)
24. Tsejtin, G.S.: On the complexity of derivation in propositional logic. Studies in Constructive Mathematics and Mathematical Logic **2**, 115–125 (1968), reprinted in: [23, vol 2], pp. 466-483.
25. Urquhart, A.: Hard examples for resolution. J. ACM **34**, 209–219 (1987)
26. Urquhart, A.: A near-optimal separation of regular and general resolution. SIAM Journal on Computing **40**(1), 107–121 (2011)
27. Van Gelder, A.: Verifying RUP proofs of propositional unsatisfiability. In: 10th International Symposium on Artificial Intelligence and Mathematics (ISAIM) (2008), http://isaim2008.unl.edu/index.php?page=proceedings
28. Wetzler, N., Heule, M.J.H., Hunt Jr., W.A.: DRAT-trim: Efficient checking and trimming using expressive clausal proofs. In: Theory and Applications of Satisfiability Testing - 17th International Conference (SAT). pp. 422–429. Lecture Notes in Computer Science 8561, Springer Verlag (2014)