Exponential Separations Using Guarded Extension Variables

Emre Yolcu
Carnegie Mellon University, Pittsburgh, PA, USA

Marijn J. H. Heule
Carnegie Mellon University, Pittsburgh, PA, USA

Abstract

We study the complexity of proof systems augmenting resolution with inference rules that allow, given a formula Γ in conjunctive normal form, deriving clauses that are not necessarily logically implied by Γ but whose addition to Γ preserves satisfiability. When the derived clauses are allowed to introduce variables not occurring in Γ, the systems we consider become equivalent to extended resolution. We are concerned with the versions of these systems without new variables. They are called BC^-, RAT^-, SBC^-, and GER^-, denoting respectively blocked clauses, resolution asymmetric tautologies, set-blocked clauses, and generalized extended resolution. Each of these systems formalizes some restricted version of the ability to make assumptions that hold "without loss of generality," which is commonly used informally to simplify or shorten proofs.

Except for SBC^-, these systems are known to be exponentially weaker than extended resolution. They are, however, all equivalent to it under a relaxed notion of simulation that allows the translation of the formula along with the proof when moving between proof systems. By taking advantage of this fact, we construct formulas that separate RAT^- from GER^- and vice versa. With the same strategy, we also separate SBC^- from RAT^-. Additionally, we give polynomial-size SBC^- proofs of the pigeonhole principle, which separates SBC^- from GER^- by a previously known lower bound. These results also separate the three systems from BC^- since they all simulate it. We thus give an almost complete picture of their relative strengths.

2012 ACM Subject Classification Theory of computation → Proof complexity

Keywords and phrases proof complexity, separations, resolution, extended resolution, blocked clauses

Digital Object Identifier 10.4230/LIPIcs.ITCS.2023.101

Related Version *Full Version*: `https://arxiv.org/abs/2211.12456`

Funding This material is based upon work supported by the National Science Foundation under grant CCF-2015445.

Acknowledgements We thank Jakob Nordström for useful discussions and the ITCS reviewers for their comments.

1 Introduction

1.1 Properties of commonly studied proof systems

Most of the commonly studied rule-based propositional proof systems,[1] such as resolution [3, 27], Frege [8], cutting planes [9], polynomial calculus [6], Lovász–Schrijver [23], are in several senses well behaved. For instance, they are monotonic, strongly sound, and strongly closed under restrictions.

- A proof system P is *monotonic* if for all sets Γ and Γ' of formulas such that $\Gamma \subseteq \Gamma'$ and for every formula φ we have

$$\Gamma \vdash_P \varphi \implies \Gamma' \vdash_P \varphi,$$

[1] Throughout this paper, by "proof" we mean a refutation of satisfiability (i.e., a derivation of $\bot$ from a set Γ of formulas, where $\bot$ denotes a contradiction such as the empty clause).

where $\vdash_P$ is the derivability relation for P. Since we often express a proof as a tree, monotonicity naturally holds for most proof systems. It has the consequence that the validity of an inference in the proof relies only on some subset of the previously derived formulas as opposed to the entire set.

- *Strong soundness* is the property of a proof system P that a formula φ can be derived in P from a set Γ of formulas only if φ is logically implied by Γ (i.e., every total assignment satisfying all formulas in Γ also satisfies φ), written as

$$\Gamma \vdash_P \varphi \implies \Gamma \models \varphi. \tag{1}$$

  Soundness is less strict than strong soundness in that it only requires (1) to hold for $\varphi = \bot$.
- For a formula φ and a partial assignment α, let $\varphi|_\alpha$ denote the formula obtained by first replacing every assigned variable x occurring in φ by $\alpha(x)$ and then recursively simplifying all of the subformulas. For a set Γ of formulas, let $\Gamma|_\alpha := \{\psi|_\alpha \mid \psi \in \Gamma\}$. We call $\varphi|_\alpha$ the restriction of φ under α, and similarly for Γ. We say a proof system P is *strongly closed under restrictions* if for every set Γ of formulas, for every formula φ, and for every partial assignment α that does not satisfy φ, we have

$$\Gamma \vdash_P \varphi \implies \Gamma|_\alpha \vdash_P \varphi|_\alpha. \tag{2}$$

  As in the case of soundness, (weak) closure under restrictions only requires (2) to hold for $\varphi = \bot$. Closure under restrictions is often also defined by the more quantitative condition that for every P-proof Π of Γ and every partial assignment α there exist a P-proof Π' of $\Gamma|_\alpha$ of size polynomial in the size of Π.

None of the above properties are necessary for soundness, and proof systems that do not have them can be stronger since such systems are more permissive in terms of the kinds of reasoning they allow. Possibly the most prominent example of such a system in proof complexity is extended Frege [8]. When refuting a set Γ of formulas, extended Frege allows (in addition to the axioms and the rules of the underlying Frege system) proof steps of the form $x \leftrightarrow \varphi$, where x is a variable and φ is an arbitrary formula, with the condition that x not occur in Γ, any of the preceding steps, or φ. Another example is extended resolution [29], which uses a similar rule although in a more restricted form since resolution works only with clauses. With x a "new" variable as before and p, q literals, extended resolution allows introducing $x \leftrightarrow p \wedge q$ via the following clauses, where the overline denotes negation:

$$\overline{x} \vee p \qquad \overline{x} \vee q \qquad x \vee \overline{p} \vee \overline{q} \tag{3}$$

Extended Frege (and similarly extended resolution) has none of the above properties, although the reasons are not particularly interesting:

- Monotonicity fails to hold since we cannot necessarily derive $x \leftrightarrow \varphi$ from $\Gamma' \supseteq \Gamma$ if x already occurs in Γ'.
- Strong soundness fails to hold since Γ may not imply $x \leftrightarrow \varphi$ under assignments α such that $\alpha(x) \neq \alpha(\varphi)$. Nevertheless, extended Frege is sound because Γ and $\Gamma \cup \{x \leftrightarrow \varphi\}$ are equisatisfiable (i.e., Γ is satisfiable if and only if $\Gamma \cup \{x \leftrightarrow \varphi\}$ is satisfiable), seen as follows: if an assignment α satisfies Γ but falsifies $x \leftrightarrow \varphi$, flipping $\alpha(x)$ gives a different assignment α' satisfying both Γ and $x \leftrightarrow \varphi$.
- Strong closure under restrictions fails to hold since otherwise we could choose a partial assignment that only assigns x to *True* and conclude that every formula φ can be derived from Γ, which contradicts the soundness of extended Frege.

In all of the above cases, the counterexamples rely crucially on the fact that x is a new variable. This ability to abbreviate complex formulas by variables significantly increases the difficulty of proving lower bounds for extended Frege and makes it one of the strongest propositional proof systems. (Extended resolution is equivalent to it over refutations of sets of clauses.) From this point on, we use "formula" and "set of clauses" interchangeably.

Proof systems that violate the above properties for more sophisticated reasons (i.e., not simply due to the introduction of new variables) also exist. In this paper we compare the proof complexity of four such systems that augment resolution with inference rules of varying expressiveness. Given a set Γ of clauses, these rules allow deriving clauses that are not necessarily logically implied by Γ but whose addition to Γ preserves satisfiability. We call such clauses redundant. Deciding the redundancy of a clause with respect to a set of clauses is coDP-complete[2] [2], so we consider only the inference rules that rely on polynomial-time verifiable syntactic conditions corresponding to restricted versions of redundancy. These rules may be viewed as capturing the commonly used technique of making assumptions that hold "without loss of generality" when writing informal mathematical proofs. Such assumptions are not logically implied by the hypotheses at hand, but their use is justified by the fact that they can be eliminated at the possible cost of an increase in the size of the proof. The formal rules we study rely on syntactic criteria to justify such assumptions, with weaker criteria allowing the introduction of stronger assumptions. In this way, these rules allow us to directly express various kinds of informal reasoning that are otherwise difficult to formalize.

From the perspective of the broader study of proof complexity, these systems are somewhat unique in that Frege does not simulate even the weakest variant unless Frege and extended Frege are equivalent [4, Corollary 2.5]. It would be interesting to determine whether some variant of these systems, despite having the same limited syntax as resolution and no new variables, simulates a subsystem of Frege stronger than resolution.

1.2 Related work

1.2.1 Proof complexity

The inference rules we study originate from the notion of blocked clauses, developed initially by Kullmann [20, 21] to give improved deterministic algorithms for 3-SAT. We call a clause C blocked with respect to a set Γ of clauses if there exists a literal $p \in C$ such that all possible resolvents of C on p against clauses from Γ are tautological (i.e., contain a literal and its negation). Kullmann [22] showed that blocked clauses are redundant and thus considered an inference rule that, given a set Γ of clauses, allows us to extend Γ with a clause that is blocked with respect to Γ. This rule, along with resolution, gives the proof system called *blocked clauses* (BC). As illustrated below, BC is not monotonic, not strongly sound, and not strongly closed under restrictions.

▶ **Example 1.** The clause $C = \overline{x} \vee \overline{y}$ is blocked with respect to the set $\Gamma = \{x \vee y,\ x \vee \overline{y}\}$.

- Monotonicity fails to hold since we cannot derive C from $\Gamma' = \Gamma \cup \{y\}$ in BC: the set Γ' is satisfiable but $\Gamma' \cup \{C\}$ is unsatisfiable.
- Strong soundness fails to hold since Γ does not imply C under assignments that set both x and y to *True*.

[2] The class $\mathsf{DP} = \{L_1 \cap L_2 \mid L_1 \in \mathsf{NP}, L_2 \in \mathsf{coNP}\}$, which is a superset of both NP and coNP, was defined by Papadimitriou and Yannakakis [25].

- Strong closure under restrictions fails to hold since for an assignment α that sets y to *True* we cannot derive $C|_\alpha = \overline{x}$ from $\Gamma|_\alpha = \{x\}$ in BC: the set $\Gamma|_\alpha$ is satisfiable but $\Gamma|_\alpha \cup \{C|_\alpha\}$ is unsatisfiable.

It is apparent from the definition of a blocked clause that deleting clauses from Γ enlarges the set of clauses that are blocked with respect to Γ. With this observation at hand, Kullmann defined a strengthening of BC called *generalized extended resolution* (GER) that allows the temporary deletion of clauses from Γ. Arbitrary deletion of clauses does not necessarily preserve satisfiability; however, since no subset of a satisfiable Γ is unsatisfiable, it is also possible to further strengthen GER by allowing the arbitrary deletion of a clause as a proof step. The resulting system is called *deletion blocked clauses* (DBC).

Conversely to the above point, the failure of monotonicity becomes particularly important when deletion is not allowed since it implies that the validity of blocked clause additions performed in sequence are order dependent. In particular, not every set of clauses that are all blocked with respect to Γ can be derived from Γ by a sequence of blocked clause additions. For this reason, proving upper bounds for generalizations of BC involves carefully ensuring the validity of sequences of inferences.

Without any additional restrictions, the above systems all simulate extended resolution since the clauses in (3) can be added in sequence as blocked clauses if we are allowed to introduce new variables: starting with a set Γ of clauses not containing the variable x, we can derive

- $\overline{x} \vee p$ followed by $\overline{x} \vee q$ since no occurrence of the literal x precedes either step (so both clauses are vacuously blocked), and then
- $x \vee \overline{p} \vee \overline{q}$ since its resolvents on x against $\overline{x} \vee p$ and $\overline{x} \vee q$ (i.e., the only preceding occurrences of $\overline{x}$) are tautological.

The study of these systems becomes interesting when we disallow new variables. A proof of Γ is *without new variables* if it contains only the variables that already occur in Γ. Throughout this paper, we denote a proof system variant that disallows new variables with the superscript "$-$" (e.g., BC^- is BC without new variables). We denote a variant that allows arbitrary deletion with the prefix D. All of those variants constitute examples of proof systems that share the peculiarities of extended resolution from Section 1.1 without being allowed new variables.

Kullmann [22] proved that extended resolution simulates GER. He also proved that GER^- is exponentially stronger than resolution and exponentially weaker than extended resolution. In later work, Järvisalo, Heule, and Biere [15] defined a different generalization of BC by essentially replacing "tautological" in the definition of a blocked clause with "implied by Γ through unit propagation." (Unit propagation is an automatizable but incomplete variant of resolution.) The result is still a polynomial-time verifiable redundancy criterion since the only important property of tautologies in the argument for the redundancy of a blocked clause C with respect to Γ is that tautologies are implied by Γ. This generalization is called *resolution asymmetric tautologies* (RAT). Yet another generalization of BC along a different axis is called *set-blocked clauses* (SBC), defined by Kiesl, Seidl, Tompits, and Biere [17]. We call a clause C set-blocked with respect to a set Γ of clauses if there exists some nonempty $L \subseteq C$ such that for all $D \in \Gamma$ with $D \cap \overline{L} \neq \varnothing$ and $D \cap L = \varnothing$ the set $\big(C \setminus L\big) \cup \big(D \setminus \overline{L}\big)$ is tautological. A blocked clause is the special case where L is a singleton, so set-blockedness expands the scope of the literals in C that we consider. Deciding the set-blockedness of a clause with respect to a set of clauses is NP-complete [17]. To ensure that an SBC proof is polynomial-time verifiable, every step in the proof that adds a clause C as set-blocked is expected to indicate the subset $L \subseteq C$ for which C is set-blocked. With that said, to reduce clutter, we leave this requirement out of our definitions and indicate those subsets only informally throughout this paper.

Subsequent works [13, 16] defined further generalizations, showed simulations between some variants, and gave polynomial-size proofs (without new variables) of the pigeonhole principle in a variant called *set-propagation redundancy* (SPR^-) that combines SBC^- and RAT^-. Recently, Buss and Thapen [4] initiated a systematic study of the proof complexity of the many generalizations of BC^-. Among other results, they showed that the bit pigeonhole principle, parity principle, clique-coloring principle, and Tseitin tautologies have polynomial-size SPR^- proofs. They also showed that SPR^- can undo (with polynomial-size derivations) the effects of or-ification, xor-ification, and lifting with index gadgets. In view of these results, SPR^- appears to be surprisingly strong.[3] Buss and Thapen also proved an exponential size lower bound for RAT^-, separating DRAT^- and SPR^- from it. Superpolynomial lower bounds for SPR^- or even SBC^- are currently open.

1.2.2 SAT solving

As the use of SAT solvers in propositional theorem proving increased, it became standard to expect a solver to produce a proof alongside an unsatisfiability claim. Modern SAT solvers are based on conflict-driven clause learning (CDCL) [24] and essentially search for resolution proofs. As a result, the initial proof systems developed to help verify the outputs of CDCL SAT solvers were based on resolution [10, 31]. However, most of the current SAT solvers go beyond CDCL and employ an array of *inprocessing techniques* [15] that transform the formula during the search. These techniques are often not strongly sound, and resolution falls short for expressing them. Järvisalo, Heule, and Biere [15] observed that DRAT simulated all of the common techniques used at the time, and, following the implementation of a practical verifier [32], DRAT became the de facto standard proof system used in SAT solvers. Extended resolution could also be used for verification; however, it is only known to simulate DRAT with polynomial overhead [16, Section 4.5], whereas DRAT simulates extended resolution with no overhead. There are also a few examples of DRAT enabling significant gains over the smallest known extended resolution proofs (see, e.g., [16, Table 1]), which is important for practical purposes.

Another practical motivation for studying these systems is their potential usefulness in proof search due to the surprising strength of the variants without new variables. Recent works have introduced a SAT solving paradigm called *satisfaction-driven clause learning* (SDCL) [14, 12] that can fully automatically discover small proofs of the pigeonhole principle. Its usefulness remains limited, though, since it was observed to improve upon CDCL only on specific classes of formulas. Exploiting the power of these systems might be a promising avenue for the research that aims to improve the performance of practical SAT solvers. To this end, it is important to understand the relative strengths of these systems.

1.3 Results

We prove some results concerning the relative strengths of BC^-, RAT^-, SBC^-, and GER^-, continuing the line of work [22, 13, 16, 4] on the proof complexity of generalizations of BC^-. Figure 1 summarizes the state of the proof complexity landscape surrounding these systems after our results.

[3] As remarked by Buss and Thapen [4, Section 4], the apparent strength of SPR^- stems from the ability to exploit symmetries, which are abundant in the combinatorial principles used for proving lower bounds against the commonly studied proof systems. Other interesting examples of systems that easily prove such combinatorial principles are the variants of Krishnamurthy's *symmetric resolution* [19, 30, 1, 28], obtained by augmenting resolution with rules that explicitly support reasoning about symmetries.

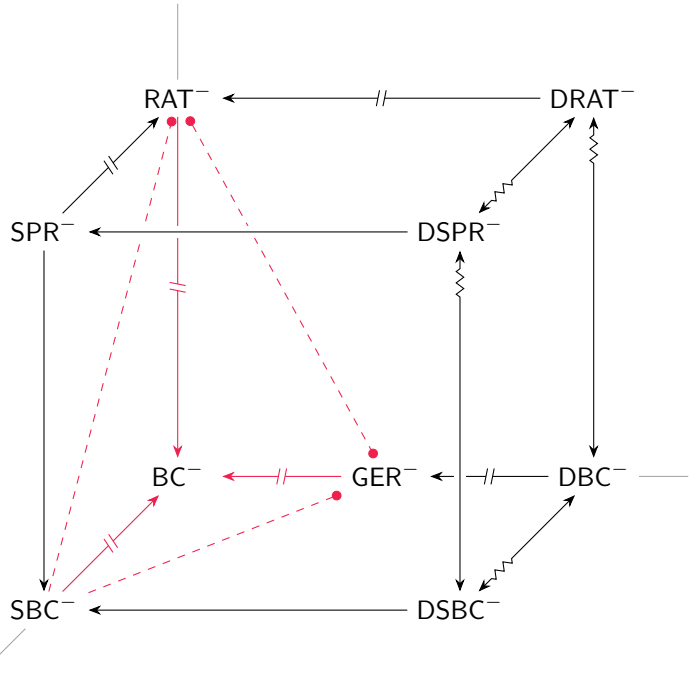

Figure 1 In the above diagram, the proof systems are placed in three-dimensional space with BC^- at the origin. Moving away from the origin along each axis corresponds to a particular way of generalizing a proof system. For systems P and Q, we use $P \longrightarrow Q$ to denote that P simulates Q; (and $P \rightsquigarrow Q$ to indicate an "interesting" simulation, where P is not simply a generalization of Q); $P \text{ - - } \bullet\, Q$ to denote that P is exponentially separated from Q (i.e., there exists an infinite sequence of formulas admitting polynomial-size proofs in P while requiring exponential-size proofs in Q); and $P \dashrightarrow Q$ to denote that P both simulates Q and is exponentially separated from Q. Arrows in red indicate the relationships that are new in this paper. To reduce clutter, some relationships that are implied by transitivity are not displayed (e.g., DBC^- simulates RAT^- and is exponentially separated from it through DRAT^-).

Our first main result is a two-way separation between RAT^- and GER^-.

▶ **Theorem 2.** *There exists an infinite sequence $(\Gamma_n)_{n=1}^{\infty}$ of formulas such that Γ_n admits RAT^- proofs of size $n^{O(1)}$ but requires GER^- proofs of size $2^{\Omega(n)}$. Conversely, there exists an infinite sequence $(\Delta_n)_{n=1}^{\infty}$ of formulas such that Δ_n admits GER^- proofs of size $n^{O(1)}$ but requires RAT^- proofs of size $2^{\Omega(n)}$.*

Since both RAT^- and GER^- are generalizations of BC^-, the above result also separates both systems from BC^-. It was already understood that GER^- is between BC^- and DBC^- in strength, and GER^- is in fact "strictly" between BC^- and DBC^- by Theorem 2.

For both directions of the separation, we follow a strategy that exploits the equivalence of BC^- (without new variables) to extended resolution under *effective simulations* [11, 26], which allow the translation of the formula (in a satisfiability-preserving way) along with the proof when moving between proof systems. In particular, Kullmann [22, Lemma 8.4] and Buss and Thapen [4] observed that it is possible to incorporate new variables into a formula Γ in such a way that BC^-, while still technically only using the variables occurring in the formula, simulates an extended resolution proof of Γ.

▶ **Lemma 3** ([4, Lemma 2.2])**.** *Suppose that a formula Γ has an extended resolution proof of size m and that Γ and the set $X = \{y \vee x_1 \vee \cdots \vee x_m,\ y\}$ of clauses have no variables in common. Then $\Gamma \cup X$ has a BC^- proof of size $O(m)$.*

To separate RAT^- and GER^-, we incorporate new variables into formulas in ways that are useful to only one of the two systems. We achieve this by "guarding" the new variables by clauses instead of providing them as in Lemma 3. Recall that, for a clause to be redundant with respect to a formula according to some syntactic criterion in this paper, every clause in the formula has to satisfy a certain condition. We take advantage of this fact to include the new variables within a strategically chosen set $\mathcal{X}$ of guard clauses alongside an unsatisfiable formula Γ. With a suitable choice of $\mathcal{X}$, we are able to impose enough limitations upon the redundant clauses derivable in a system to ensure that the new variables can essentially be ignored in a proof of $\Gamma \cup \mathcal{X}$.

For each direction of the separation, when proving the upper bound for one of the two systems, we show that it can efficiently work through the guard clauses and use the new variables to simulate the extended resolution proof. When proving the lower bound for the other system, we show essentially that it is closed under restrictions[4] for the specific formulas and partial assignments that we construct. (Neither system is closed under restrictions in general.) In other words, the guard clauses make it impossible for the system to efficiently "access" the new variables, thus preventing it from achieving any speedup. This allows us to use the existing separations of extended resolution from RAT^- and GER^- to separate the two systems without needing to prove lower bounds entirely from scratch. The main difficulty is in coming up with the appropriate ways of incorporating new variables into formulas.

As our next main result, we separate SBC^- from RAT^- (and hence BC^-) with the same strategy. In fact, we reuse the formulas separating GER^- from RAT^- and show that SBC^- can also efficiently work through the guard clauses in them, although in a different manner than GER^-.

▶ **Theorem 4.** *There exists an infinite sequence* $(\Gamma_n)_{n=1}^{\infty}$ *of formulas such that* Γ_n *admits* SBC^- *proofs of size* $n^{O(1)}$ *but requires* RAT^- *proofs of size* $2^{\Omega(n)}$.

Finally, we give polynomial-size SBC^- proofs of the pigeonhole principle, which exponentially separates SBC^- from GER^- by a lower bound due to Kullmann [22, Lemma 9.4].

▶ **Theorem 5.** *There exists an infinite sequence* $(\Gamma_n)_{n=1}^{\infty}$ *of formulas such that* Γ_n *admits* SBC^- *proofs of size* $n^{O(1)}$ *but requires* GER^- *proofs of size* $2^{\Omega(n)}$.

Along the way to our main results, we prove a partial simulation of RAT^- by BC^-. It is partial in the sense that the size of the produced BC^- proof is not always a polynomial in the size of the RAT^- proof (which is impossible due to Theorem 2). It also has the property that, although the produced proof may sometimes be small, the simulation cannot necessarily be carried out in time polynomial in the size of the produced proof. This is because the simulation involves generating satisfying assignments to certain formulas obtained in the process. To our knowledge, all of the "natural" simulations between the commonly studied proof systems are efficient, so the partial simulation of RAT^- by BC^- is an odd example. Another notable aspect of the simulation is that it directly informed the construction of the formulas that we use for separating RAT^- from GER^-. We discuss this further at the end of Section 4. Due to the technical nature of the simulation, we do not state it here in detail.

1.4 Open questions

We leave open the following.

▶ **Question 6.** Is RAT^- exponentially separated from SBC^-?

[4] We use closure under restrictions here in the quantitative sense described in Section 1.

▶ **Question 7.** Is GER^- exponentially separated from SBC^-?

Answering these questions will complete the picture of the relative strengths of the weakest generalizations of BC^- along each axis in Figure 1. However, we do not even have any superpolynomial lower bounds for SBC^-. If a separation of extended resolution from SBC^- is shown, it might be possible to relatively easily separate RAT^- and GER^- from SBC^- by tailoring guarded extension variables that SBC^- cannot access (in the manner of the current paper).

As an aside, the formulas that we use for the separations in this paper are arguably "artificial" in that they do not encode any combinatorial principles. Separations with "natural" formulas give more intuitive insight into the relative capabilities of the proof systems being considered, so it is desirable to reprove Theorems 2 and 4 using formulas that encode some combinatorial principles. On the other hand, those artificial formulas enable relatively simple and modular proofs of the separations. An interesting open question is whether our strategy of separating two proof systems P and Q, both of which effectively simulate a strong system R, through syntactic manipulations of formulas that separate R from P and Q is more generally applicable.

Another desirable goal is to establish tighter connections between the more commonly studied proof systems and the generalizations of BC^-. As mentioned earlier, Frege does not simulate BC^- unless it also simulates extended Frege. Additionally, it is already known that bounded-depth Frege does not simulate BC^- [4, Corollary 2.3]. We naturally wonder about the converse direction.

▶ **Question 8.** Is there a subsystem of Frege stronger than resolution that DBC^- simulates?

2 Preliminaries

We denote the set of positive integers by $\mathbb{N}^+$. For $n \in \mathbb{N}^+$, we let $[n] \coloneqq \{1, \ldots, n\}$. For a sequence $S = (x_1, \ldots, x_n)$, its length is n and we denote it by $|S|$. We use $\langle x \rangle$ to compactly denote an infinite sequence $(x_n)_{n=1}^{\infty}$.

2.1 Propositional logic

For notation we mostly follow Buss and Thapen [4].

We use 0 and 1 to denote *False* and *True*, respectively. A *literal* is a propositional variable or its negation. A set of literals is *tautological* if it contains a pair of complementary literals x and $\overline{x}$. A *clause* is the disjunction of a nontautological set of literals. We denote by $\mathbf{V}$, $\mathbf{L}$, and $\mathbf{C}$ respectively the sets of all variables, all literals, and all clauses. A *conjunctive normal form formula* (CNF) is a conjunction of clauses. We identify clauses with sets of literals and CNFs with sets of clauses. In the rest of this section we use C, D to denote clauses and Γ, Δ to denote CNFs.

When we know $C \cup D$ to be nontautological, we write it as $C \vee D$. We write $C \dot{\vee} D$ to indicate a *disjoint disjunction*, where C and D have no variables in common. We sometimes write $\Gamma \cup \{C\}$ as $\Gamma \wedge C$.

We denote by $\mathrm{var}(\Gamma)$ the set of all the variables occurring in Γ. We say C *subsumes* D, denoted $C \sqsupseteq D$, if either D is tautological or $C \subseteq D$. For CNFs, we say Γ subsumes Δ, denoted $\Gamma \sqsupseteq \Delta$, if for all $D \in \Delta$ there exists some $C \in \Gamma$ such that $C \sqsupseteq D$. We take the disjunction of a clause and a CNF as $C \vee \Delta \coloneqq \{C \vee D \mid D \in \Delta \text{ and } C \cup D \text{ is nontautological}\}$. We say Γ and Δ are *equisatisfiable*, denoted $\Gamma \equiv_{\mathrm{sat}} \Delta$, if they are either both satisfiable or both unsatisfiable. With respect to Γ, a clause C is *redundant* if $\Gamma \setminus \{C\} \equiv_{\mathrm{sat}} \Gamma \equiv_{\mathrm{sat}} \Gamma \cup \{C\}$.

A *partial assignment* α is a partial function $\alpha\colon \mathbf{V} \rightharpoonup \{0,1\}$, which also acts on literals by letting $\alpha(\overline{x}) \coloneqq \overline{\alpha(x)}$. We identify α with the set $\{p \in \mathbf{L} \mid \alpha(p) = 1\}$, consisting of all the literals it satisfies. For a set L of literals, we let $\overline{L} \coloneqq \{\overline{x} \mid x \in L\}$. In particular, we use $\overline{C}$ to denote the smallest partial assignment that falsifies all the literals in C. We say α *satisfies* C, denoted $\alpha \models C$, if there exists some $p \in C$ such that $\alpha(p) = 1$. We say α satisfies Γ if for all $C \in \Gamma$ we have $\alpha \models C$. For C that α does not satisfy, the *restriction* of C under α is $C|_\alpha \coloneqq C \setminus \{p \in C \mid \alpha(p) = 0\}$. Extending the above to CNFs, the restriction of Γ under α is $\Gamma|_\alpha \coloneqq \{C|_\alpha \mid C \in \Gamma \text{ and } \alpha \not\models C\}$.

2.2 Proof complexity

We recall the definition of a proof system (in the sense of Cook and Reckhow [8]) and the basic notions of proof complexity, which can also be found in the recent textbook by Krajíček [18, Chapter 1].

In the rest of this section we think of Γ as a formula and Π as a proof, each encoded by a string over some finite alphabet.

▶ **Definition 9.** *A* proof system *is a polynomial-time computable binary relation P such that the following hold.*

- Soundness: *For all Γ and Π, if $P(\Gamma, \Pi)$ holds then Γ is unsatisfiable.*
- Completeness: *For all unsatisfiable Γ, there exists some Π such that $P(\Gamma, \Pi)$ holds.*

*We call any Π satisfying $P(\Gamma, \Pi)$ a P-*proof *of Γ.*

Proof complexity is concerned with the sizes (or lengths) of proofs. For a proof system P and a formula Γ, we define $\mathrm{size}_P(\Gamma) \coloneqq \min\{|\Pi| \mid \Pi \text{ is a } P\text{-proof of } \Gamma\}$ if Γ is unsatisfiable and $\mathrm{size}_P(\Gamma) \coloneqq \infty$ otherwise.

▶ **Definition 10.** *A proof system P* simulates *Q if for all unsatisfiable Γ we have $\mathrm{size}_P(\Gamma) = \mathrm{size}_Q(\Gamma)^{O(1)}$. Additionally, P* polynomially simulates *Q if there exists a polynomial-time algorithm for converting a Q-proof of Γ into a P-proof of Γ.*

▶ **Definition 11.** *Proof systems P and Q are* equivalent *if they simulate each other. Additionally, P and Q are* polynomially equivalent *if they polynomially simulate each other.*

We say P is *exponentially separated* from Q if there exists some sequence $\langle\Gamma\rangle$ of formulas such that $\mathrm{size}_P(\Gamma_n) = n^{O(1)}$ while $\mathrm{size}_Q(\Gamma_n) = 2^{\Omega(n)}$. We call such $\langle\Gamma\rangle$ *easy* for P and *hard* for Q.

2.3 Resolution

▶ **Definition 12.** *The* resolution rule *is*

$$\frac{A \mathbin{\dot\vee} x \qquad B \mathbin{\dot\vee} \overline{x}}{A \vee B},$$

where A, B are clauses and x is a variable. We call $A \vee B$ the resolvent *of $A \vee x$ and $B \vee \overline{x}$ on x.*

▶ **Definition 13.** *The* weakening rule *is*

$$\frac{A}{A \vee B},$$

where A and B are clauses. We call $A \vee B$ a weakening *of A.*

We define a resolution proof in a slightly different form than usual: as a sequence of CNFs instead of a sequence of clauses.

▶ **Definition 14.** *A* resolution proof *of a CNF Γ is a sequence $\Pi = (\Gamma_1, \dots, \Gamma_N)$ of CNFs such that $\Gamma_1 = \Gamma$, $\bot \in \Gamma_N$, and, for all $i \in [N-1]$, we have $\Gamma_{i+1} = \Gamma_i \cup \{C\}$, where*
- *C is a resolvent of two clauses $D, E \in \Gamma_i$ or*
- *C is a weakening of some clause $D \in \Gamma_i$.*

The size of Π is N.

We write Res to denote the resolution proof system. A well known fact is that resolution proofs are preserved under restrictions: if $(\Gamma_1, \Gamma_2, \dots, \Gamma_N)$ is a resolution proof of Γ, then, for every partial assignment α, the sequence $(\Gamma_1|_\alpha, \Gamma_2|_\alpha, \dots, \Gamma_N|_\alpha)$ contains a resolution proof of $\Gamma|_\alpha$. This implies in particular the following.

▶ **Lemma 15.** *For every CNF Γ and every partial assignment α, we have* $\mathrm{size}_{\mathsf{Res}}(\Gamma|_\alpha) \leq \mathrm{size}_{\mathsf{Res}}(\Gamma)$.

We next define a weakened version of resolution that comes up often in the study of decision algorithms for satisfiability.

▶ **Definition 16.** *A* unit propagation proof *is a resolution proof where each use of the resolution rule is of the form*

$$\frac{A \dot\vee x \qquad \overline{x}}{A}.$$

Unit propagation is not complete. With Γ, Δ CNFs and $L = \{p_1, \dots, p_k\}$ a set of literals, we define $\Gamma \wedge \overline{L} \coloneqq \Gamma \wedge \overline{p_1} \wedge \dots \wedge \overline{p_k}$ and write
- $\Gamma \vdash_1 \bot$ to denote that there exists a unit propagation proof of Γ,
- $\Gamma \vdash_1 L$ to denote $\left(\Gamma \wedge \overline{L}\right) \vdash_1 \bot$,
- $\Gamma \vdash_1 \Delta$ to denote that for all $D \in \Delta$ we have $\Gamma \vdash_1 D$.

Note that $\Gamma \vdash_1 \Delta$ implies $\Gamma \models \Delta$. Moreover, whether $\Gamma \vdash_1 \Delta$ holds can be decided in polynomial time. This makes it useful as a component in defining inference rules.

As in the case of resolution, unit propagation proofs are preserved under restrictions.

▶ **Lemma 17.** *For every CNF Γ, every set L of literals, and every partial assignment α such that $\alpha \not\models L$, if $\Gamma \vdash_1 L$, then $\Gamma|_\alpha \vdash_1 L|_\alpha$.*

Proof. The proof is available in the full version. ◀

From this point on, we discuss some strengthenings of the resolution proof system.

▶ **Definition 18.** *Let Γ be a CNF and p, q be arbitrary literals. Consider a* new *variable x (i.e., not occurring in any one of Γ, p, q). We call $\{\overline{x} \vee p,\ \overline{x} \vee q,\ x \vee \overline{p} \vee \overline{q}\}$ a set of* extension clauses *for Γ. In this context, we refer to x as the* extension variable.

▶ **Definition 19.** *A CNF Λ is an* extension *for a CNF Γ if there exists a sequence $(\lambda_1, \dots, \lambda_t)$ such that $\Lambda = \bigcup_{i=1}^{t} \lambda_i$, and, for all $i \in [t]$, we have that λ_i is a set of extension clauses for $\Gamma \cup \bigcup_{j=1}^{i-1} \lambda_j$.*

▶ **Definition 20.** *An* extended resolution proof *of a CNF Γ is a pair (Λ, Π), where Λ is an extension for Γ and Π is a resolution proof of $\Gamma \cup \Lambda$. The size of (Λ, Π) is defined to be $|\Lambda| + |\Pi|$.*

We write ER to denote the extended resolution proof system.

3 Inference rules

We recall the redundancy criteria that lead to the inference rules we use to augment resolution proofs. The definitions are adapted from previous works [22, 15, 17, 13, 4].

▶ **Definition 21.** *A clause $C = p \dot\vee C'$ is a* blocked clause *(BC) for p with respect to a CNF Γ if, for every clause D of the form $\overline{p} \dot\vee D'$ in Γ, the set $C' \cup D'$ is tautological.*

A strict generalization of the notion of a blocked clause is a resolution asymmetric tautology, defined as follows.

▶ **Definition 22.** *A clause $C = p \dot\vee C'$ is a* resolution asymmetric tautology *(RAT) for p with respect to a CNF Γ if, for every clause D of the form $\overline{p} \dot\vee D'$ in Γ, we have $\Gamma \vdash_1 C' \cup D'$.*

Another strict generalization of a blocked clause is a set-blocked clause.[5]

▶ **Definition 23.** *A clause C is a* set-blocked clause *(SBC) for a nonempty $L \subseteq C$ with respect to a CNF Γ if, for every clause $D \in \Gamma$ with $D \cap \overline{L} \neq \varnothing$ and $D \cap L = \varnothing$, the set $\big(C \setminus L\big) \cup \big(D \setminus \overline{L}\big)$ is tautological.*

We say C is a BC with respect to Γ if there exists a literal $p \in C$ for which C is a BC with respect to Γ, and similarly for RAT and SBC. Note that the above definitions do not prohibit BCs, RATs, or SBCs with respect to Γ from containing variables not occurring in Γ.

It was shown by Kullmann [22], Järvisalo, Heule, and Biere [15], and Kiesl, Seidl, Tompits, and Biere [17] that BCs, RATs, and SBCs are redundant, which makes it possible to use them to define proof systems.

▶ **Theorem 24.** *If a clause C is a BC, RAT, or SBC with respect to a CNF Γ, then $\Gamma \setminus \{C\} \equiv_{\text{sat}} \Gamma \equiv_{\text{sat}} \Gamma \cup \{C\}$.*

▶ **Definition 25.** *A* blocked clauses proof *of a CNF Γ is a sequence $\Pi = (\Gamma_1, \dots, \Gamma_N)$ of CNFs such that $\Gamma_1 = \Gamma$, $\bot \in \Gamma_N$, and, for all $i \in [N-1]$, we have $\Gamma_{i+1} = \Gamma_i \cup \{C\}$, where either*

- *C is a resolvent of two clauses $D, E \in \Gamma_i$,*
- *C is a weakening of some clause $D \in \Gamma_i$, or*
- *C is a blocked clause with respect to Γ_i.*

The size of Π is N.

We write BC to denote the blocked clauses proof system. Replacing "blocked clause" by "resolution asymmetric tautology" in the above definition gives the resolution asymmetric tautologies proof system, which we denote by RAT. Replacing it by "set-blocked clause" gives the set-blocked clauses proof system, which we denote by SBC.

RAT and SBC are two generalizations of BC, and we now define another, designed to overcome the dependence of the validity of BC inferences on the order of clause additions (see [22, Section 1.3]).

For a CNF Γ and a set V of variables, we let

$$\mathbf{B}_V(\Gamma) \coloneqq \{C \in \mathbf{C} \mid C \text{ is a BC for a literal of some } x \in V \text{ with respect to } \Gamma\}.$$

We also let

[5] We define a set-blocked clause in a slightly different, although equivalent, way compared with the original [17, Definition 4.1].

- $\mathbf{B}(\Gamma) \coloneqq \mathbf{B}_{\mathbf{V}}(\Gamma)$,
- $\mathbf{B}^{\text{in}}(\Gamma) \coloneqq \mathbf{B}(\Gamma) \cap \Gamma$,
- $\mathbf{B}^{-}_{V}(\Gamma) \coloneqq \{C \in \mathbf{B}_{V}(\Gamma) \mid \text{var}(C) \subseteq \text{var}(\Gamma)\}$,
- $\mathbf{B}^{-}(\Gamma) \coloneqq \mathbf{B}^{-}_{\text{var}(\Gamma)}(\Gamma)$.

Before proceeding, we observe the below result, which follows immediately from the definition of a blocked clause.

▶ **Lemma 26.** *For all CNFs Γ and Δ such that $\Gamma \subseteq \Delta$, we have $\mathbf{B}(\Gamma) \supseteq \mathbf{B}(\Delta)$.*

Thus, we may assume without loss of generality that all of the blocked clause additions in a BC proof are performed before any resolution steps. (A similar assumption does not necessarily hold for RAT proofs.)

▶ **Definition 27.** *A sequence $(C_1, \ldots, C_m)$ of some clauses from a CNF Γ is a* maximal blocked sequence *for Γ if for all $i \in [m]$ the clause C_i is blocked with respect to $\Gamma \setminus \bigcup_{j=1}^{i-1}\{C_j\}$ and $\mathbf{B}^{\text{in}}(\Gamma \setminus \bigcup_{i=1}^{m}\{C_i\})$ is empty.*

For a CNF Γ, a maximal blocked sequence is unique up to the ordering of its clauses [22, Lemma 6.1], which makes the following notion well defined.

▶ **Definition 28.** *Let $(C_1, \ldots, C_m)$ be a maximal blocked sequence for a CNF Γ. The* kernel *of Γ is $\text{ker}(\Gamma) \coloneqq \Gamma \setminus \bigcup_{i=1}^{m}\{C_i\}$.*

▶ **Definition 29.** *A CNF Λ is a* blocked extension *for a CNF Γ if $\text{ker}(\Gamma \cup \Lambda) = \text{ker}(\Gamma)$.*

▶ **Definition 30.** *A* generalized extended resolution proof *of a CNF Γ is a pair (Λ, Π), where Λ is a blocked extension for Γ and Π is a resolution proof of $\Gamma \cup \Lambda$. The size of (Λ, Π) is defined to be $|\Lambda| + |\Pi|$.*

We write GER to denote the generalized extended resolution proof system. The relationship between GER and BC is made clear by the following characterization of blocked extensions.

▶ **Lemma 31** ([22, Lemma 6.5])**.** *A CNF Λ is a blocked extension for a CNF Γ if and only if there exists a CNF $\Gamma' \subseteq \Gamma$ and an ordering $(C_1, \ldots, C_m)$ of all the clauses in $\Lambda \cup (\Gamma \setminus \Gamma')$ such that for all $i \in [m]$ the clause C_i is blocked with respect to $\Gamma' \cup \bigcup_{j=1}^{i-1}\{C_j\}$.*

This result gives a view of GER as a version of BC that allows the *temporary* deletion of clauses from the initial formula (i.e., clauses can be deleted as long as they are added back later).

In this paper, we study the variants of BC, RAT, SBC, and GER that disallow the use of new variables. We say that a proof of a CNF Γ is *without new variables* if all the variables occurring in the proof are in $\text{var}(\Gamma)$. In the case of GER, this constraint applies to the blocked extension. We use BC^{-}, RAT^{-}, SBC^{-}, and GER^{-} to denote the variants without new variables.

3.1 Useful facts

We conclude this section with a few standalone results that we will refer back to later. We defer their proofs to the full version.

▶ **Lemma 32.** *For every CNF Γ such that $\text{ker}(\Gamma) = \Gamma$, we have $\text{size}_{\text{GER}^-}(\Gamma) = \text{size}_{\text{BC}^-}(\Gamma)$.*

▶ **Lemma 33.** *For every CNF Γ, we have $\text{size}_{\text{BC}^-}(\Gamma) \geq \text{size}_{\text{Res}}(\Gamma \cup \mathbf{B}^{-}(\Gamma))$.*

▶ **Definition 34.** *The* projection *of a CNF* Γ *onto a literal* p *is the CNF* $\mathrm{proj}_p(\Gamma) \coloneqq \{C \setminus \{p\} \mid C \in \Gamma \text{ and } p \in C\}$.

This definition plays a role in both our (partial) simulation of RAT^- by BC^- and our GER^- lower bounds. In particular, we use the following fact, which was already observed by Kullmann [22, Section 4].

▶ **Lemma 35.** *A clause* $C = p \dot\vee C'$ *is a BC for* p *with respect to a CNF* Γ *if and only if the partial assignment* $\overline{C'}$ *satisfies* $\mathrm{proj}_{\overline{p}}(\Gamma)$.

The next result is essentially due to Chang [5, Theorem 1]. Although its original form is slightly weaker, the exact statement below can be obtained by a modification of Chang's proof. We provide its proof in the full version.

▶ **Lemma 36.** *For every CNF* Γ *and every clause* C *such that* $\Gamma \vdash_1 C$*, there exists a resolution derivation* $(\Gamma_1, \dots, \Gamma_N)$ *with* $N \leq |\mathrm{var}(\Gamma)| + 1$ *such that* $\Gamma_1 = \Gamma$*,* $C \in \Gamma_N$*, and* $\Gamma \cup \{C\} \sqsupseteq \Gamma_N$.

The following gives a simple condition under which we regain monotonicity.

▶ **Lemma 37** ([4, Lemma 1.20])**.** *Let* Γ *and* Δ *be CNFs such that* $\Gamma \subseteq \Delta$ *and* $\Gamma \sqsupseteq \Delta$*. If a clause is a BC, RAT, or SBC with respect to* Γ*, then it is a BC, RAT, or SBC with respect to* Δ.

4 Partial simulation of RAT^- by BC^-

We will show how to convert a RAT addition into a sequence of BC additions and resolution steps. Assume that all the BCs and RATs in this section are without new variables.

▶ **Definition 38.** *The* nonblocking CNF *of a clause* C *for a literal* $p \in C$ *with respect to a CNF* Γ *is* $\mathrm{NB}_p^\Gamma(C) \coloneqq \left\{D \setminus C \;\middle|\; D \in \mathrm{proj}_{\overline{p}}(\Gamma) \text{ and } (C \setminus \{p\}) \cup D \text{ is nontautological}\right\}$.

As a consequence of the above definition, we have $\mathrm{var}\left(\mathrm{NB}_p^\Gamma(C)\right) \cap \mathrm{var}(C) = \varnothing$.

We say an assignment α *minimally satisfies* a CNF Γ, denoted $\alpha \models_{\min} \Gamma$, if α satisfies Γ while no proper subset $\alpha' \subsetneq \alpha$ satisfies Γ. We let $\mu(\Gamma) \coloneqq \left\{E \in \mathbf{C} \;\middle|\; \overline{E} \models_{\min} \Gamma\right\}$. Since two different minimally satisfying assignments cannot contain one another, no clause $E \in \mu(\Gamma)$ is contained in a different clause $E' \in \mu(\Gamma)$.

▶ **Example 39.** Let $\Gamma = \{x,\ y \vee \overline{z}\}$. This CNF has two minimally satisfying assignments: $\{x, y\}$ and $\{x, \overline{z}\}$. We thus have $\mu(\Gamma) = \{\overline{x} \vee \overline{y},\ \overline{x} \vee z\}$.

Noting that $\Gamma \cup \mu(\Gamma)$ is unsatisfiable for every CNF Γ, we let $s(\Gamma) \coloneqq |\mu(\Gamma)| + \mathrm{size}_{\mathsf{Res}}(\Gamma \cup \mu(\Gamma))$. When Γ is unsatisfiable, we simply have $s(\Gamma) = \mathrm{size}_{\mathsf{Res}}(\Gamma)$.

▶ **Theorem 40.** *Let* $C = p \dot\vee C'$ *be a RAT for* p *with respect to a CNF* Γ*. There exists a* BC^- *derivation* $(\Gamma_1, \dots, \Gamma_N)$ *such that* $\Gamma_1 = \Gamma$*,* $C \in \Gamma_N$*, and* $\Gamma \cup \{C\} \sqsupseteq \Gamma_N$*, where, letting* $\Sigma = \mathrm{NB}_p^\Gamma(C)$ *and letting* $n = |\mathrm{var}(\Gamma)|$*, we have* $N \leq |\Sigma|(n+1) + s(\Sigma)$.

Proof. Since C is a RAT for p, for all $D \in \mathrm{NB}_p^\Gamma(C)$ we have $\Gamma \vdash_1 C' \dot\vee D$, which implies in particular that $\Gamma \vdash_1 C \dot\vee D$. Then, using Lemma 36, for all $D \in \mathrm{NB}_p^\Gamma(C)$ we derive $C \dot\vee D$ from Γ in resolution using at most $n+1$ steps. More formally, we derive $\Gamma' \cup \left(C \dot\vee \mathrm{NB}_p^\Gamma(C)\right)$ from Γ, where Γ' is the set of intermediate clauses, guaranteed by Lemma 36 to satisfy $\{C\} \sqsupseteq \Gamma'$.

We proceed differently depending on the satisfiability of $\mathrm{NB}_p^\Gamma(C)$.

Case 1 ($\mathrm{NB}_p^\Gamma(C)$ *is unsatisfiable.*) There exists a resolution proof $\Pi = (\Delta_1, \ldots, \Delta_m)$, where $\Delta_1 = \mathrm{NB}_p^\Gamma(C)$ and $\bot \in \Delta_m$. Suppose Π is a minimum-size proof, so it does not use weakening. Consider the sequence $\Pi' = (C \vee \Delta_1, \ldots, C \vee \Delta_m)$. This sequence is a valid resolution derivation of C from $C \dot\vee \mathrm{NB}_p^\Gamma(C)$, seen as follows:

- By the definition of $\mathrm{NB}_p^\Gamma(C)$, it has no variables in common with C. Since we assumed that Π does not use weakening, no subsequent CNF in Π has any variables in common with C either.
  Let $i \in [m-1]$. The sequence Π is a resolution proof, so we have $\Delta_{i+1} = \Delta_i \cup \{E\}$, where E is a resolvent of some $F, G \in \Delta_i$. Since Δ_i has no variables in common with C, it is not possible to resolve F and G on a variable of C. Then the clause $C \dot\vee E$ is a resolvent of $C \dot\vee F$ and $C \dot\vee G$, which are in $C \vee \Delta_i$. This proves by induction that Π' is a valid resolution derivation.
- Finally, since $\bot \in \Delta_m$, we have $C \in (C \vee \Delta_m)$.

Thus, resolution can derive C from Γ in at most $\left|\mathrm{NB}_p^\Gamma(C)\right|(n+1) + \mathrm{size}_{\mathsf{Res}}\left(\mathrm{NB}_p^\Gamma(C)\right)$ steps.

Case 2 ($\mathrm{NB}_p^\Gamma(C)$ *is satisfiable.*) Let $\Psi = \Gamma \cup \Gamma' \cup \left(C \dot\vee \mathrm{NB}_p^\Gamma(C)\right)$ (i.e., the current CNF). Since $C \sqsupseteq (\Psi \setminus \Gamma)$, the literal $\overline{p}$ does not occur in $\Psi \setminus \Gamma$. Then we have $\mathrm{proj}_{\overline{p}}(\Psi) = \mathrm{proj}_{\overline{p}}(\Gamma)$, and, consequently, $\mathrm{NB}_p^\Psi(C) = \mathrm{NB}_p^\Gamma(C)$.

Let E be a clause such that $\mathrm{var}(E) \cap \mathrm{var}(C) = \varnothing$. By Lemma 35, the clause $p \dot\vee C' \dot\vee E$ is a BC for p with respect to Ψ if and only if the partial assignment $\overline{C' \dot\vee E}$ satisfies $\mathrm{proj}_{\overline{p}}(\Psi)$. By the definition of a nonblocking CNF, $\overline{C'}$ already satisfies $\mathrm{proj}_{\overline{p}}(\Psi) \setminus \mathrm{NB}_p^\Psi(C)$, so $p \dot\vee C' \dot\vee E$ is a BC for p with respect to Ψ if and only if $\overline{E}$ satisfies $\mathrm{NB}_p^\Psi(C)$.

Let us write μ for $\mu\left(\mathrm{NB}_p^\Psi(C)\right)$. All clauses in $C \dot\vee \mu$ are blocked for p with respect to Ψ, and the addition of each such clause of the form $C \dot\vee E$ rules out, for $C \dot\vee \mathrm{NB}_p^\Psi(C)$, every partial assignment α containing $\overline{C \cup E}$. Then we have $\left(C \dot\vee \mathrm{NB}_p^\Psi(C)\right) \cup (C \dot\vee \mu) \models C$, where every partial assignment containing $\overline{C}$ falsifies the left-hand side (i.e., $\mathrm{NB}_p^\Psi(C) \cup \mu$ is unsatisfiable). Also, no clause $E \in \mu$ contains $\overline{p}$ and no clause $E \in \mu$ is a subset of a different clause $E' \in \mu$. This implies in particular that, for every subset $\mu' \subseteq \mu$ and for every clause $E \in \mu \setminus \mu'$, if $\overline{E} \models \mathrm{NB}_p^\Psi(C)$, then $\overline{E} \models \mathrm{NB}_p^\Psi(C) \cup \mu'$. We thus derive $C \dot\vee \mu$ from Ψ by a sequence of blocked clause additions.

As in the previous case, attaching C to a resolution proof of $\mathrm{NB}_p^\Psi(C) \cup \mu$ gives a resolution derivation of C, so we derive C from $\left(C \dot\vee \mathrm{NB}_p^\Psi(C)\right) \cup (C \dot\vee \mu)$ in resolution.

In the end, BC^- can derive C from Γ in at most $\left|\mathrm{NB}_p^\Gamma(C)\right|(n+1) + |\mu| + \mathrm{size}_{\mathsf{Res}}\left(\mathrm{NB}_p^\Gamma(C) \cup \mu\right)$ steps. ◀

Given a RAT^- proof, we can apply the above theorem to recursively replace the earliest RAT addition in the proof by a BC^- derivation. The intermediate clauses in the derivation replacing the addition of a RAT C are all subsumed by C, which ensures by Lemma 37 that the validity of later RAT additions are preserved. We can thus translate an entire RAT^- proof to a BC^- proof.

In the above simulation, when $\mathrm{NB}_p^\Gamma(C)$ is unsatisfiable, we do not use any blocked clause additions. By Lemma 35, no blocked clause for p exists, and since the RAT addition by itself only gives useful information about the clauses containing $\overline{p}$, a simulation where we add a clause that is blocked for a different literal needs to be more sophisticated. In particular, such a simulation is unlikely to be *local* in the sense of the output consisting of a sequence of derivations that each simulate a single step in the input. (Most simulations in proof complexity are local.) Assuming that the above simulation is the best possible, if every RAT

addition in a RAT^- proof Π has an unsatisfiable nonblocking CNF, then BC^- essentially falls back to refuting the nonblocking CNFs for locally simulating Π. This observation hints at the transformation in (4) for separating RAT^- from GER^-.

5 Incomparability of RAT^- and GER^-

We now show that RAT^- is exponentially separated from GER^- and vice versa, which also exponentially separates both systems from BC^-. For both directions, we follow a strategy similar at a high level to the one that Kullmann [22, Lemma 8.4] used to prove an exponential separation of BC^- from resolution.

Let P and Q be proof systems (without new variables) that simulate BC^-. To separate P from Q, we take a sequence $\langle\Gamma\rangle$ of CNFs separating ER from Q and we incorporate extension variables into the formulas in a way that allows P to simulate the ER proof while preventing Q from achieving any speedup. This strategy is made possible by the fact that BC^- effectively simulates ER. See also the discussion by Buss and Thapen [4, Section 2.2].

From this point on, given a CNF Γ, we use (Λ^*, Π^*) to denote a minimum-size ER proof of Γ, where Λ^* is the union of a sequence of $t(\Gamma) \coloneqq |\Lambda^*|/3$ sets of extension clauses such that the ith set λ_i is of the form $\{\overline{x_i} \vee p_i,\ \overline{x_i} \vee q_i,\ x_i \vee \overline{p_i} \vee \overline{q_i}\}$. Thus, we implicitly reserve $\{x_1, \dots, x_{t(\Gamma)}\}$ as the set of extension variables used in Λ^*. We assume without loss of generality that the variables of p_i and q_i are in $\mathrm{var}(\Gamma) \cup \{x_1, \dots, x_{i-1}\}$ for all $i \in [t(\Gamma)]$.

5.1 Exponential separation of RAT^- from GER^-

Let Γ be a CNF and (Λ^*, Π^*) be a minimum-size ER proof of Γ as described above. Consider the transformation

$$\mathcal{G}(\Gamma) \coloneqq \Gamma \cup \bigcup_{i=1}^{t(\Gamma)} \big[(x_i \vee \Gamma) \cup (\overline{x_i} \vee \Gamma)\big], \tag{4}$$

where $x_1, \dots, x_{t(\Gamma)}$ are the extension variables used in Λ^*. When Γ is unsatisfiable, each extension variable above is "locked" behind the projection Γ, which RAT^- can overcome but BC^- cannot.

▶ **Lemma 41.** *For every CNF Γ, we have* $\mathrm{size}_{\mathsf{RAT}^-}(\mathcal{G}(\Gamma)) \le \mathrm{size}_{\mathsf{ER}}(\Gamma)$.

Proof. We will show that the minimum-size ER proof (Λ^*, Π^*) of Γ directly gives a RAT^- proof of $\mathcal{G}(\Gamma)$ of the same size.

We write t for $t(\Gamma)$. Let $(\lambda_1, \dots, \lambda_t)$ be the sequence of t sets of extension clauses that make up Λ^*. Consider an arbitrary $i \in [t]$, and suppose that we have derived $\bigcup_{j=1}^{i-1} \lambda_j$ from $\mathcal{G}(\Gamma)$ by a sequence of RAT additions, so the current CNF is $\Delta = \mathcal{G}(\Gamma) \cup \bigcup_{j=1}^{i-1} \lambda_j$. We will introduce the clauses in $\lambda_i = \{\overline{x_i} \vee p_i,\ \overline{x_i} \vee q_i,\ x_i \vee \overline{p_i} \vee \overline{q_i}\}$ by a sequence of RAT additions. Note that, since λ_i is a set of extension clauses for $\Gamma \cup \bigcup_{j=1}^{i-1} \lambda_j$, so far the variable x_i occurs only in $\mathcal{G}(\Gamma) \setminus \Gamma$.

1. The clause $\overline{x_i} \vee p_i$ is a RAT for $\overline{x_i}$ with respect to Δ because all earlier occurrences of x_i are clauses of the form $x_i \vee D$, where $D \in \Gamma$. We thus require $\Delta \vdash_1 \{p_i\} \cup D$ for all $D \in \Gamma$. This is indeed the case since we actually have $D \in \Delta$ by the construction of $\mathcal{G}(\Gamma)$, which implies $\Delta \vdash_1 \{p_i\} \cup D$.
2. The clause $\overline{x_i} \vee q_i$ is similarly a RAT for $\overline{x_i}$ with respect to $\Delta \cup \{\overline{x_i} \vee p_i\}$.
3. The clause $x_i \vee \overline{p_i} \vee \overline{q_i}$ is similarly a RAT for x_i with respect to Δ. Moreover, it is a BC for x_i with respect to $\{\overline{x_i} \vee p_i,\ \overline{x_i} \vee q_i\}$ since $\{p_i, \overline{p_i}, \overline{q_i}\}$ and $\{q_i, \overline{p_i}, \overline{q_i}\}$ are both tautological. As a result, $x_i \vee \overline{p_i} \vee \overline{q_i}$ is a RAT with respect to $\Delta \cup \{\overline{x_i} \vee p_i,\ \overline{x_i} \vee q_i\}$.

It follows by induction that we can derive Λ^* from $\mathcal{G}(\Gamma)$ in RAT^-. Since Π^* is a resolution proof of $\Gamma \cup \Lambda^*$, and since $\mathcal{G}(\Gamma)$ contains Γ, we also have a resolution proof of $\mathcal{G}(\Gamma) \cup \Lambda^*$. Thus, we have a RAT^- proof of $\mathcal{G}(\Gamma)$ of size $|\Lambda^*| + |\Pi^*| = \mathrm{size}_{\mathsf{ER}}(\Gamma)$. ◀

As a consequence of the above, if a sequence $\langle\Gamma\rangle$ of CNFs is easy for ER, then, independent of whether $\langle\Gamma\rangle$ is easy or hard for RAT^-, the sequence $\mathcal{G}(\langle\Gamma\rangle) \coloneqq (\mathcal{G}(\Gamma_1), \mathcal{G}(\Gamma_2), \dots)$ is easy for RAT^-. In contrast, the following result implies that the extension variables added by $\mathcal{G}$ are of no use to BC^-.

▶ **Lemma 42.** *For every CNF Γ, we have* $\mathrm{size}_{\mathsf{BC}^-}(\mathcal{G}(\Gamma)) \geq \mathrm{size}_{\mathsf{Res}}(\Gamma \cup \mathbf{B}^-(\Gamma))$.

Proof. When Γ is satisfiable, the inequality holds trivially, so suppose that Γ is unsatisfiable.

Applying Lemma 33 to $\mathcal{G}(\Gamma)$, we have

$$\mathrm{size}_{\mathsf{BC}^-}(\mathcal{G}(\Gamma)) \geq \mathrm{size}_{\mathsf{Res}}\big(\mathcal{G}(\Gamma) \cup \mathbf{B}^-(\mathcal{G}(\Gamma))\big). \tag{5}$$

We claim that no clause in $\mathbf{B}^-(\mathcal{G}(\Gamma))$ is blocked for a literal of any of the variables in $X = \{x_1, \dots, x_{t(\Gamma)}\}$. To see this, consider a clause C of the form $x \mathbin{\dot\vee} C'$, where $x \in X$. If C is blocked for x with respect to $\mathcal{G}(\Gamma)$, then $\overline{C'}$ is a satisfying assignment to $\mathrm{proj}_{\overline{x}}(\mathcal{G}(\Gamma)) = \Gamma$ by Lemma 35. Since Γ is unsatisfiable, no such assignment exists. Therefore, C cannot be blocked for x, which leaves us with

$$\mathbf{B}^-(\mathcal{G}(\Gamma)) = \mathbf{B}^-_{\mathrm{var}(\Gamma)}(\mathcal{G}(\Gamma)). \tag{6}$$

Furthermore, since $\Gamma \subseteq \mathcal{G}(\Gamma)$, every clause in $\mathbf{B}^-_{\mathrm{var}(\Gamma)}(\mathcal{G}(\Gamma))$ has to be blocked in particular with respect to Γ. This requires $\mathbf{B}^-_{\mathrm{var}(\Gamma)}(\mathcal{G}(\Gamma))$ to consist of clauses of the form $C \mathbin{\dot\vee} D$, where $C \in \mathbf{B}^-(\Gamma)$ and $\mathrm{var}(D) \subseteq X$ (with D possibly empty). In light of this, consider a partial assignment α such that

$$\alpha(z) = \begin{cases} 1 & \text{if } z \in X \\ \text{undefined} & \text{otherwise.} \end{cases}$$

It is straightforward to see that $\mathcal{G}(\Gamma)|_\alpha = \Gamma$. Additionally, for every clause $C \mathbin{\dot\vee} D$ (of the above form) in $\mathbf{B}^-_{\mathrm{var}(\Gamma)}(\mathcal{G}(\Gamma))$, the restriction $(C \mathbin{\dot\vee} D)|_\alpha$ is either 1 or C. We thus have

$$\Big(\mathcal{G}(\Gamma) \cup \mathbf{B}^-_{\mathrm{var}(\Gamma)}(\mathcal{G}(\Gamma))\Big)\Big|_\alpha = \Gamma \cup \mathbf{B}^-(\Gamma). \tag{7}$$

Putting (5), (6), and (7) together, we finally obtain

$$\begin{aligned} \mathrm{size}_{\mathsf{BC}^-}(\mathcal{G}(\Gamma)) &\geq \mathrm{size}_{\mathsf{Res}}\big(\mathcal{G}(\Gamma) \cup \mathbf{B}^-(\mathcal{G}(\Gamma))\big) \\ &= \mathrm{size}_{\mathsf{Res}}\Big(\mathcal{G}(\Gamma) \cup \mathbf{B}^-_{\mathrm{var}(\Gamma)}(\mathcal{G}(\Gamma))\Big) \\ &\geq \mathrm{size}_{\mathsf{Res}}\Big(\Big(\mathcal{G}(\Gamma) \cup \mathbf{B}^-_{\mathrm{var}(\Gamma)}(\mathcal{G}(\Gamma))\Big)\Big|_\alpha\Big) && \text{(Lemma 15)} \\ &= \mathrm{size}_{\mathsf{Res}}\big(\Gamma \cup \mathbf{B}^-(\Gamma)\big), \end{aligned}$$

which is the desired inequality. ◀

For certain CNFs, the above result carries over to GER^-.

▶ **Lemma 43.** *For every CNF Γ such that* $\ker(\Gamma) = \Gamma$*, we have* $\mathrm{size}_{\mathsf{GER}^-}(\mathcal{G}(\Gamma)) \geq \mathrm{size}_{\mathsf{Res}}(\Gamma \cup \mathbf{B}^-(\Gamma))$.

Proof. The proof is available in the full version. ◀

To prove the separation, we invoke the above results with Γ as the *pigeonhole principle*, which states that every "pigeon" $i \in [n+1]$ is mapped to some "hole" $k \in [n]$ and that no two distinct pigeons $i, j \in [n+1]$ are mapped to the same hole. It is defined for $n \in \mathbb{N}^+$ as

$$\mathrm{PHP}_n \coloneqq \bigcup_{i \in [n+1]} \{p_{i,1} \vee \cdots \vee p_{i,n}\} \cup \bigcup_{\substack{i,j \in [n+1],\ i \neq j \\ k \in [n]}} \{\overline{p_{i,k}} \vee \overline{p_{j,k}}\},$$

where we call the first set of clauses the *pigeon axioms* and the second set the *hole axioms*.

▶ **Theorem 44.** RAT^- *is exponentially separated from* GER^-.

Proof. Cook [7] constructed polynomial-size ER proofs of PHP_n, which implies by Lemma 41 that $\mathrm{size}_{\mathsf{RAT}^-}(\mathcal{G}(\mathrm{PHP}_n)) = n^{O(1)}$. Kullmann [22, Theorem 2] proved that $\mathrm{size}_{\mathsf{Res}}(\mathrm{PHP}_n \cup \mathbf{B}^-(\mathrm{PHP}_n)) = 2^{\Omega(n)}$. Noting that $\ker(\mathrm{PHP}_n) = \mathrm{PHP}_n$ for all $n \in \mathbb{N}^+$, we apply Lemma 43 to obtain $\mathrm{size}_{\mathsf{GER}^-}(\mathcal{G}(\mathrm{PHP}_n)) = 2^{\Omega(n)}$. Thus, $\mathcal{G}(\langle \mathrm{PHP} \rangle)$ exponentially separates RAT^- from GER^-. ◀

5.2 Exponential separation of GER^- from RAT^-

We proceed in a similar way to the previous section. Let Γ be a CNF and (Λ^*, Π^*) be a minimum-size ER proof of Γ. Take a set $\{y_1, \ldots, y_{t(\Gamma)}\} \subseteq \mathbf{V} \setminus \mathrm{var}(\Gamma \cup \Lambda^*)$ of $t(\Gamma)$ distinct variables. Consider the transformation

$$\mathcal{H}(\Gamma) \coloneqq \Gamma \cup \bigcup_{i=1}^{t(\Gamma)} \{\overline{x_i} \vee y_i,\ x_i \vee \overline{y_i}\}, \tag{8}$$

where $x_1, \ldots, x_{t(\Gamma)}$ are the extension variables used in Λ^*. As before, only one of the two systems can make any use of the extension variables incorporated into the formula. This time, the temporary deletion available to GER^- makes the difference.

▶ **Lemma 45.** *For every CNF* Γ*, we have* $\mathrm{size}_{\mathsf{GER}^-}(\mathcal{H}(\Gamma)) \leq \mathrm{size}_{\mathsf{ER}}(\Gamma)$.

Proof. Let (Λ^*, Π^*) be the minimum-size ER proof of Γ. We will show that the clauses in $\Lambda^* \cup (\mathcal{H}(\Gamma) \setminus \Gamma)$ can be derived from Γ in some sequence by blocked clause additions, which implies by Lemma 31 that Λ^* is a blocked extension for $\mathcal{H}(\Gamma)$.

Recall that extension clauses can be derived in sequence by blocked clause additions. Then, since Λ^* is an extension for Γ, we derive Λ^* by such a sequence. Next, from $\Gamma \cup \Lambda^*$, we derive the clauses in $\mathcal{H}(\Gamma) \setminus \Gamma$. Let us write t for $t(\Gamma)$. Consider the sequence $(\overline{x_1} \vee y_1,\ \ldots,\ \overline{x_t} \vee y_t,\ x_1 \vee \overline{y_1},\ \ldots,\ x_t \vee \overline{y_t})$. For each $i \in [t]$, the ith clause $\overline{x_i} \vee y_i$ in the first half of the sequence is blocked for y_i since $\overline{y_i}$ does not occur in any of the earlier clauses. Similarly, the ith clause $x_i \vee \overline{y_i}$ in the second half of the sequence is blocked for $\overline{y_i}$ since the only earlier occurrence of y_i is the clause $\overline{x_i} \vee y_i$ and $\{x_i, \overline{x_i}\}$ is tautological. As a result, Λ^* is a blocked extension for $\mathcal{H}(\Gamma)$.

Since Π^* is a resolution proof of $\Gamma \cup \Lambda^*$, and since $\mathcal{H}(\Gamma)$ contains Γ, we also have a resolution proof of $\mathcal{H}(\Gamma) \cup \Lambda^*$. Thus, we have a GER^- proof of $\mathcal{H}(\Gamma)$ of size $|\Lambda^*| + |\Pi^*| = \mathrm{size}_{\mathsf{ER}}(\Gamma)$. ◀

▶ **Lemma 46.** *Let* Γ *be a CNF, and let* $n = |\mathrm{var}(\Gamma)|$*. We have* $\mathrm{size}_{\mathsf{RAT}^-}(\mathcal{H}(\Gamma)) \geq \frac{\mathrm{size}_{\mathsf{RAT}^-}(\Gamma)}{n+1}$.

Proof. We write t for $t(\Gamma)$. Let $V = \mathrm{var}(\mathcal{H}(\Gamma) \setminus \Gamma)$ (i.e., the set of variables added by $\mathcal{H}$), and let α be a partial assignment such that

$$\alpha(z) = \begin{cases} 1 & \text{if } z \in V \\ \text{undefined} & \text{otherwise.} \end{cases}$$

We claim that, for all $N \in \mathbb{N}^+$, given a RAT^- derivation $\Pi = (\Delta_1, \ldots, \Delta_N)$ with $\Delta_1 = \mathcal{H}(\Gamma)$, there exists a RAT^- derivation $\Pi' = (\Psi_1, \ldots, \Psi_{N'})$ with $N' \leq N \cdot (|\mathrm{var}(\Gamma)| + 1)$ such that

- $\Delta_1|_\alpha = \Gamma = \Psi_1$,
- $\Delta_N|_\alpha \subseteq \Psi_{N'}$, and
- $\Delta_N|_\alpha \sqsupseteq \Psi_{N'}$.

The second condition above implies in particular that if $\bot \in \Delta_N$, then $\bot \in \Psi_{N'}$. This in turn implies the desired statement, since it means that if $\mathcal{H}(\Gamma)$ has a RAT^- proof of size N, then Γ has a RAT^- proof of size at most $N \cdot (|\mathrm{var}(\Gamma)| + 1)$.

We proceed by induction. For a derivation $\Pi = (\Delta_1)$ of size 1 with $\Delta_1 = \mathcal{H}(\Gamma)$, the derivation $\Pi' = (\Delta_1|_\alpha)$ satisfies the conditions above. Let $\Pi = (\Delta_1, \ldots, \Delta_m)$ be a RAT^- derivation with $\Delta_1 = \mathcal{H}(\Gamma)$. Suppose that $\Pi' = (\Psi_1, \ldots, \Psi_{m'})$ is a RAT^- derivation with $m' \leq m \cdot (|\mathrm{var}(\Gamma)| + 1)$ satisfying the above conditions. Let C be a clause that is derived from Δ_m either by resolution, weakening, or RAT addition. We will show that there exists a RAT^- derivation from $\Psi_{m'}|_\alpha$ of a CNF that contains and is subsumed by $(\Delta_m \cup \{C\})|_\alpha$. For simplicity, we will establish this for $\Delta_m|_\alpha$ instead of $\Psi_{m'}|_\alpha$, with the understanding that the containment and the subsumption conditions above imply by Lemma 37 that the same derivation can be made also from $\Psi_{m'}|_\alpha$. There exists a trivial derivation when $\alpha \models C$ since it implies that $(\Delta_m \cup \{C\})|_\alpha = \Delta_m|_\alpha$, so suppose that $\alpha \not\models C$. As a consequence, α does not satisfy any subset of C.

From this point on, we write Δ instead of Δ_m to reduce clutter.

Case 1 (C *is a resolvent of* $D, E \in \Delta$ *on* v.) Without loss of generality, suppose that $v \in D$ and $\overline{v} \in E$.

- **Case 1.1** ($v \in V$.) We have $E|_\alpha = (E \setminus \{\overline{v}\})|_\alpha$. Since $(E \setminus \{\overline{v}\}) \subseteq C$, we have $E|_\alpha \in \Delta|_\alpha$ and $E|_\alpha \subseteq C|_\alpha$, so $C|_\alpha$ is derived from $\Delta|_\alpha$ by weakening.
- **Case 1.2** ($v \notin V$.) Since $(D \setminus \{v\}) \subseteq C$ and $(E \setminus \{\overline{v}\}) \subseteq C$, and since α does not set v, we have $D|_\alpha \in \Delta|_\alpha$ and $E|_\alpha \in \Delta|_\alpha$. Moreover, $C|_\alpha$ is a resolvent of $D|_\alpha \in \Delta|_\alpha$ and $E|_\alpha \in \Delta|_\alpha$, so $C|_\alpha$ is derived from $\Delta|_\alpha$ by resolution.

Case 2 (C *is a weakening of a clause* $D \in \Delta$.) Since $D \subseteq C$, we have $D|_\alpha \in \Delta|_\alpha$ and $D|_\alpha \subseteq C|_\alpha$, so $C|_\alpha$ is derived from $\Delta|_\alpha$ by weakening.

Case 3 (C *is a RAT for* $p \in C$ *with respect to* Δ.) Since we assumed $\alpha \not\models C$, there are two possibilities: either $\overline{p} \in V$ or $\mathrm{var}(p) \notin V$.

- **Case 3.1** ($\overline{p} \in V$.) Either $p = \overline{x_i}$ or $p = \overline{y_i}$ for some $i \in [t]$. Suppose $p = \overline{x_i}$. (The case for $p = \overline{y_i}$ is symmetric.) Since C is a RAT for $\overline{x_i}$ with respect to Δ, and since $x_i \vee \overline{y_i} \in \Delta$, we have $\Delta \vdash_1 (C \setminus \{\overline{x_i}\}) \cup \{\overline{y_i}\}$. By Lemma 17, we also have $\Delta|_\alpha \vdash_1 \big((C \setminus \{\overline{x_i}\}) \cup \{\overline{y_i}\}\big)\big|_\alpha$, which simplifies to $\Delta|_\alpha \vdash_1 C|_\alpha$. By Lemma 36, there exists a resolution derivation of $C|_\alpha$ from $\Delta|_\alpha$ of size $|\mathrm{var}(\Delta|_\alpha)| + 1 \leq |\mathrm{var}(\Gamma)| + 1$. Moreover, the final CNF in this derivation is subsumed by $\Delta|_\alpha \cup C|_\alpha = (\Delta \cup \{C\})|_\alpha$ as desired.
- **Case 3.2** ($\mathrm{var}(p) \notin V$.) The clause C is of the form $C' \dot\vee p$ with $\mathrm{var}(p) \notin V$. Since C is a RAT for p with respect to Δ, for every clause $D \in \Delta$ of the form $D' \dot\vee \overline{p}$, we have $\Delta \vdash_1 C' \cup D'$. We will show that $C|_\alpha$ is a RAT with respect to $\Delta|_\alpha$. Every clause $D \in \Delta$ such that $\alpha \models D$ simply disappears from $\Delta|_\alpha$, so such clauses are irrelevant when determining whether $C|_\alpha$ is a RAT with respect to $\Delta|_\alpha$. On the other hand, for $D \in \Delta$ of the form $D' \dot\vee \overline{p}$ such that $\alpha \not\models D$, we have $\Delta|_\alpha \vdash_1 (C' \cup D')|_\alpha$ by Lemma 17. Thus, $C|_\alpha$ is a RAT for p with respect to $\Delta|_\alpha$. ◀

Let $n = 2^k$ for $k \in \mathbb{N}^+$. For a propositional variable x, let us write $x \neq 0$ and $x \neq 1$ to denote the literals x and $\overline{x}$, respectively. To prove the separation, we invoke the above results with Γ as the *bit pigeonhole principle*, which states that for all $i, j \in [n+1]$ such that $i \neq j$ the binary strings $p_1^i \dots p_k^i$ and $p_1^j \dots p_k^j$ are different. It is defined for n as

$$\mathrm{BPHP}_n \coloneqq \bigcup_{\substack{i,j\in[n+1],\ i\neq j \\ (h_1,\dots,h_k)\in\{0,1\}^k}} \left(\bigvee_{\ell=1}^{k} p_\ell^i \neq h_\ell \vee \bigvee_{\ell=1}^{k} p_\ell^j \neq h_\ell \right).$$

▶ **Theorem 47.** GER^- *is exponentially separated from* RAT^-.

Proof. Buss and Thapen [4, Theorem 4.4] gave polynomial-size proofs of BPHP_n in SPR^-, which ER simulates. By Lemma 45, we have $\mathrm{size}_{\mathsf{GER}^-}(\mathcal{H}(\mathrm{BPHP}_n)) = n^{O(1)}$. They [4, Theorem 5.4] also proved that $\mathrm{size}_{\mathsf{RAT}^-}(\mathrm{BPHP}_n) = 2^{\Omega(n)}$. Applying Lemma 46 gives $\mathrm{size}_{\mathsf{RAT}^-}(\mathcal{H}(\mathrm{BPHP}_n)) = 2^{\Omega(n)}$. Thus, $\mathcal{H}(\langle \mathrm{BPHP} \rangle)$ exponentially separates GER^- from RAT^-. ◀

Theorem 2 follows directly from Theorems 44 and 47.

6 Exponential separation of SBC^- from RAT^-

Recall the transformation in (8), which we used to construct formulas separating GER^- from RAT^-. We had

$$\mathcal{H}(\Gamma) = \Gamma \cup \bigcup_{i=1}^{t(\Gamma)} \{\overline{x_i} \vee y_i,\ x_i \vee \overline{y_i}\},$$

where $x_1, \dots, x_{t(\Gamma)}$ are the extension variables used in a minimum-size ER proof (Λ^*, Π^*) of Γ. We prove below that SBC^- can use those variables and simulate the ER proof of Γ, so deletion is not the only way to overcome the obstacles in $\mathcal{H}(\Gamma)$ that prevent RAT^- from gaining any speedup.

▶ **Lemma 48.** *For every CNF* Γ*, we have* $\mathrm{size}_{\mathsf{SBC}^-}(\mathcal{H}(\Gamma)) \leq 2 \cdot \mathrm{size}_{\mathsf{ER}}(\Gamma)$.

Proof. We write t for $t(\Gamma)$. Let $(\lambda_1, \dots, \lambda_t)$ be the sequence of t sets of extension clauses that make up Λ^*. For each $i \in [t]$, we will first derive the clauses in $\lambda_i' \coloneqq \{\overline{x_i} \vee \overline{y_i} \vee p_i,\ \overline{x_i} \vee \overline{y_i} \vee q_i,\ x_i \vee y_i \vee \overline{p_i} \vee \overline{q_i}\}$ by a sequence of SBC additions. Consider an arbitrary $i \in [t]$, and suppose that we have derived $\bigcup_{j=1}^{i-1} \lambda_j'$ from $\mathcal{H}(\Gamma)$ by a sequence of SBC additions, so the current CNF is $\Delta = \mathcal{H}(\Gamma) \cup \bigcup_{j=1}^{i-1} \lambda_j'$.

1. The clause $E_i^1 \coloneqq \overline{x_i} \vee \overline{y_i} \vee p_i$ is an SBC for $L = \{\overline{x_i}, \overline{y_i}\}$ with respect to Δ because $\overline{x_i} \vee y_i$ and $x_i \vee \overline{y_i}$ are the only clauses in Δ that intersect with $\overline{L}$, and both of these clauses also intersect with L (i.e., there is nothing to check).
2. The clause $E_i^2 \coloneqq \overline{x_i} \vee \overline{y_i} \vee q_i$ is similarly an SBC for $L = \{\overline{x_i}, \overline{y_i}\}$ with respect to Δ. Furthermore, we have $E_i^1 \cap \overline{L} = \varnothing$, so E_i^2 is an SBC with respect to $\Delta \cup \{E_i^1\}$.
3. The clause $E_i^3 \coloneqq x_i \vee y_i \vee \overline{p_i} \vee \overline{q_i}$ is similarly an SBC for $M = \{x_i, y_i\}$ with respect to Δ. It is also an SBC for M with respect to $\{E_i^1, E_i^2\}$ since $(E_i^3 \setminus M) \cup (E_i^1 \setminus \overline{M}) = \{\overline{p_i}, \overline{q_i}, p_i\}$ and $(E_i^3 \setminus M) \cup (E_i^2 \setminus \overline{M}) = \{\overline{p_i}, \overline{q_i}, q_i\}$ are both tautological. As a result, E_i^3 is an SBC with respect to $\Delta \cup \{E_i^1, E_i^2\}$.

It follows by induction that we can derive $\bigcup_{i=1}^{t} \lambda_i'$ from $\mathcal{H}(\Gamma)$ in SBC^-. For each $i \in [t]$, resolving E_i^1 and E_i^2 against $\overline{x_i} \vee y_i$ and resolving E_i^3 against $x_i \vee \overline{y_i}$ gives λ_i, thus we can derive Λ^* from $\mathcal{H}(\Gamma)$ in SBC^-. Since Π^* is a resolution proof of $\Gamma \cup \Lambda^*$, and since $\mathcal{H}(\Gamma)$ contains Γ, we also have a resolution proof of $\mathcal{H}(\Gamma) \cup \Lambda^*$. In the end, we have an SBC^- proof of $\mathcal{H}(\Gamma)$ of size at most $2|\Lambda^*| + |\Pi^*| \leq 2 \cdot \mathrm{size}_{\mathsf{ER}}(\Gamma)$. ◀

It is now straightforward to deduce Theorem 4 by invoking Lemmas 46 and 48 with Γ as the bit pigeonhole principle (in the manner of the proof of Theorem 47).

7 Exponential separation of SBC^- from GER^-

We now give polynomial-size SBC^- proofs of PHP_n. More specifically, we observe that the SPR^- proofs of PHP_n constructed by Buss and Thapen [4, Theorem 4.3] are in fact valid SBC^- proofs, so the proof below closely follows theirs. (No knowledge of SPR^- is required to follow this section.)

▶ **Lemma 49.** $\mathrm{size}_{\mathsf{SBC}^-}(\mathrm{PHP}_n) = n^{O(1)}$.

Proof. We essentially formalize in SBC^- a short inductive proof of PHP_n, which assumes without loss of generality that the smallest pigeon is mapped to the smallest hole, derives from this assumption a renamed instance of PHP_{n-1}, and inductively repeats these steps until deriving a trivial contradiction.

Recall that PHP_n consists of the clauses $P_i \coloneqq p_{i,1} \vee \cdots \vee p_{i,n}$ and $H_{i,j,k} \coloneqq \overline{p_{i,k}} \vee \overline{p_{j,k}}$ for $i, j \in [n+1]$ and $k \in [n]$ with $i \neq j$.

For $i \in [n-1]$, $j \in [n+1]$, and $k \in [n]$ such that $j, k > i$, let

$$C_{i,j,k} \coloneqq \overline{p_{i,k}} \vee \overline{p_{j,i}} \vee \left(\bigvee_{\substack{\ell \in [n+1] \\ \ell \neq i}} p_{\ell,k} \right) \vee \left(\bigvee_{\substack{\ell \in [n+1] \\ \ell \neq j}} p_{\ell,i} \right).$$

Also, for $i \in [n-1]$, let $\Lambda_i \coloneqq \{C_{i,j,k} \mid j \in [n+1],\ k \in [n],\ \text{and } j, k > i\}$. We will first show that, for all $i \in [n-1]$, from $\Psi_{i-1} \coloneqq \mathrm{PHP}_n \cup \bigcup_{\ell=1}^{i-1} \Lambda_\ell$ we can derive the clauses in Λ_i in any order by a sequence of set-blocked clause additions, which implies that we can obtain Ψ_{n-1} from PHP_n in SBC^-. Afterwards, we will give a polynomial-size resolution derivation of the empty clause from Ψ_{n-1}, concluding the proof.

For every clause $C_{i,j,k} \in \Lambda_i$ and every subset $\Lambda'_i \subseteq \Lambda_i \setminus \{C_{i,j,k}\}$, we claim that $C_{i,j,k}$ is an SBC for $L = \{\overline{p_{i,k}}, \overline{p_{j,i}}, p_{i,i}, p_{j,k}\} \subseteq C_{i,j,k}$ with respect to $\Psi_{i-1} \cup \Lambda'_i$. This requires us to show that for all $D \in \Psi_{i-1} \cup \Lambda'_i$ with $D \cap \overline{L} \neq \varnothing$ and $D \cap L = \varnothing$ the set $\big(C_{i,j,k} \setminus L\big) \cup \big(D \setminus \overline{L}\big)$ is tautological. There are three cases.

Case 1 ($D \in \mathrm{PHP}_n$.) If D is a clause in PHP_n such that $D \cap \overline{L} \neq \varnothing$ and $D \cap L = \varnothing$, then either $D = H_{i,i',i}$ for some $i' \in [n+1]$ such that $i' \neq j$ or $D = H_{j,j',k}$ for some $j' \in [n+1]$ such that $j' \neq i$. If $D = H_{i,i',i}$, then we have $p_{i',i} \in C_{i,j,k} \setminus L$ and $\overline{p_{i',i}} \in D \setminus \overline{L}$, so the union of the two sets is tautological. The argument for the case of $D = H_{j,j',k}$ is similar.

Case 2 ($D \in \bigcup_{\ell=1}^{i-1} \Lambda_\ell$.) Let $D = C_{i',j',k'}$ be an arbitrary clause in $\bigcup_{\ell=1}^{i-1} \Lambda_\ell$, where $i' < i$ and $j', k' > i'$. A simple inspection shows that we have $D \cap \overline{L} \neq \varnothing$ if and only if $k' = i$ or $k' = k$. If $k' = i$, then $p_{i,i} \in D$. Noting that $i' \neq j$, if $k' = k$, then $p_{j,k} \in D$. Either way, $D \cap L = \varnothing$ fails to hold, so there exists no $D \in \bigcup_{\ell=1}^{i-1} \Lambda_\ell$ such that $D \cap \overline{L} \neq \varnothing$ and $D \cap L = \varnothing$.

Case 3 ($D \in \Lambda'_i$.) Let $D = C_{i,j',k'}$ be an arbitrary clause in Λ'_i, where $j', k' > i$. It is straightforward to see that we have $p_{i,i} \in D$ unless $j' = i$. Then, since $j' > i$, there exists no $D \in \Lambda'_i$ such that $D \cap L = \varnothing$.

Thus, we obtain Ψ_{n-1} from PHP_n in SBC^-, and now we construct a resolution proof of Ψ_{n-1}. First, for each (i, j, k) such that $C_{i,j,k}$ is defined, we resolve $C_{i,j,k}$ against the axioms for the holes k and i to derive $\overline{p_{i,k}} \vee \overline{p_{j,i}}$. Then, for each $i \in [n]$ and $j \in [n+1]$ such that $j > i$, we derive the clause $\overline{p_{j,i}}$ by induction on i as follows: Fix i and j such that $j > i$. Let

$$\Delta_{i,j} \coloneqq \{\overline{p_{i,k}} \mid k \in [n] \text{ and } k < i\} \cup \{\overline{p_{i,i}} \vee \overline{p_{j,i}}\} \cup \{\overline{p_{i,k}} \vee \overline{p_{j,i}} \mid k \in [n] \text{ and } k > i\},$$

where we have the clauses in the first set from the induction hypothesis, the second set from PHP_n, and the third set from resolving $C_{i,j,k}$ against the hole axioms in the previous step. Resolving each clause in $\Delta_{i,j}$ against the pigeon axiom P_i thus gives $\overline{p_{j,i}}$. Finally, we resolve for each $i \in [n]$ the clause $\overline{p_{n+1,i}}$ against P_{n+1} to derive the empty clause. ◀

Kullmann [22, Lemma 9.4] showed that $\mathrm{size}_{\mathsf{GER}^-}(\mathrm{PHP}_n) = 2^{\Omega(n)}$, so Theorem 5 follows by Lemma 49.

References

1. Noriko H. Arai and Alasdair Urquhart. Local symmetries in propositional logic. In *Proceedings of the 9th International Conference on Automated Reasoning with Analytic Tableaux and Related Methods (TABLEAUX)*, number 1847 in Lecture Notes in Computer Science, pages 40–51. Springer, 2000. doi:10.1007/10722086_3.
2. Lee A. Barnett, David Cerna, and Armin Biere. Covered clauses are not propagation redundant. In *Proceedings of the 10th International Joint Conference on Automated Reasoning (IJCAR)*, number 12166 in Lecture Notes in Computer Science, pages 32–47. Springer, 2020. doi:10.1007/978-3-030-51074-9_3.
3. Archie Blake. *Canonical Expressions in Boolean Algebra*. PhD thesis, The University of Chicago, 1937.
4. Sam Buss and Neil Thapen. DRAT and propagation redundancy proofs without new variables. *Logical Methods in Computer Science*, 17(2:12), 2021. doi:10.23638/LMCS-17(2:12)2021.
5. Chin-Liang Chang. The unit proof and the input proof in theorem proving. *Journal of the ACM*, 17(4):698–707, 1970. doi:10.1145/321607.321618.
6. Matthew Clegg, Jeffery Edmonds, and Russell Impagliazzo. Using the Groebner basis algorithm to find proofs of unsatisfiability. In *Proceedings of the 28th Symposium on Theory of Computing (STOC)*, pages 174–183. Association for Computing Machinery, 1996. doi:10.1145/237814.237860.
7. Stephen A. Cook. A short proof of the pigeon hole principle using extended resolution. *ACM SIGACT News*, 8(4):28–32, 1976. doi:10.1145/1008335.1008338.
8. Stephen A. Cook and Robert A. Reckhow. The relative efficiency of propositional proof systems. *The Journal of Symbolic Logic*, 44(1):36–50, 1979. doi:10.2307/2273702.
9. William Cook, Collette R. Coullard, and György Turán. On the complexity of cutting-plane proofs. *Discrete Applied Mathematics*, 18(1):25–38, 1987. doi:10.1016/0166-218X(87)90039-4.
10. Evgueni Goldberg and Yakov Novikov. Verification of proofs of unsatisfiability for CNF formulas. In *Proceedings of the Design, Automation and Test in Europe Conference (DATE)*, pages 886–891. IEEE Computer Society, 2003. doi:10.1109/DATE.2003.1253718.
11. Alexander Hertel, Philipp Hertel, and Alasdair Urquhart. Formalizing dangerous SAT encodings. In *Proceedings of the 10th International Conference on Theory and Applications of Satisfiability Testing (SAT)*, number 4501 in Lecture Notes in Computer Science, pages 159–172. Springer, 2007. doi:10.1007/978-3-540-72788-0_18.
12. Marijn J. H. Heule, Benjamin Kiesl, and Armin Biere. Encoding redundancy for satisfaction-driven clause learning. In *Proceedings of the 25th International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS)*, number 11427 in Lecture Notes in Computer Science, pages 41–58. Springer, 2019. doi:10.1007/978-3-030-17462-0_3.
13. Marijn J. H. Heule, Benjamin Kiesl, and Armin Biere. Strong extension-free proof systems. *Journal of Automated Reasoning*, 64(3):533–554, 2020. doi:10.1007/s10817-019-09516-0.
14. Marijn J. H. Heule, Benjamin Kiesl, Martina Seidl, and Armin Biere. PRuning through satisfaction. In *Proceedings of the 13th Haifa Verification Conference (HVC)*, number 10629 in Lecture Notes in Computer Science, pages 179–194. Springer, 2017. doi:10.1007/978-3-319-70389-3_12.

15 Matti Järvisalo, Marijn J. H. Heule, and Armin Biere. Inprocessing rules. In *Proceedings of the 6th International Joint Conference on Automated Reasoning (IJCAR)*, number 7364 in Lecture Notes in Computer Science, pages 355–370. Springer, 2012. doi:10.1007/978-3-642-31365-3_28.

16 Benjamin Kiesl, Adrián Rebola-Pardo, Marijn J. H. Heule, and Armin Biere. Simulating strong practical proof systems with extended resolution. *Journal of Automated Reasoning*, 64(7):1247–1267, 2020. doi:10.1007/s10817-020-09554-z.

17 Benjamin Kiesl, Martina Seidl, Hans Tompits, and Armin Biere. Local redundancy in SAT: Generalizations of blocked clauses. *Logical Methods in Computer Science*, 14(4:3), 2018. doi:10.23638/LMCS-14(4:3)2018.

18 Jan Krajíček. *Proof Complexity*. Cambridge University Press, 2019. doi:10.1017/9781108242066.

19 Balakrishnan Krishnamurthy. Short proofs for tricky formulas. *Acta Informatica*, 22(3):253–275, 1985. doi:10.1007/BF00265682.

20 Oliver Kullmann. Worst-case analysis, 3-SAT decision and lower bounds: Approaches for improved SAT algorithms. In *Satisfiability Problem: Theory and Applications*, number 35 in DIMACS Series in Discrete Mathematics and Theoretical Computer Science, pages 261–313. American Mathematical Society, 1997. doi:10.1090/dimacs/035.

21 Oliver Kullmann. New methods for 3-SAT decision and worst-case analysis. *Theoretical Computer Science*, 223(1–2):1–72, 1999. doi:10.1016/S0304-3975(98)00017-6.

22 Oliver Kullmann. On a generalization of extended resolution. *Discrete Applied Mathematics*, 96–97:149–176, 1999. doi:10.1016/S0166-218X(99)00037-2.

23 László Lovász and Alexander Schrijver. Cones of matrices and set-functions and 0–1 optimization. *SIAM Journal on Optimization*, 1(2):166–190, 1991. doi:10.1137/0801013.

24 João P. Marques-Silva and Karem A. Sakallah. GRASP: A search algorithm for propositional satisfiability. *IEEE Transactions on Computers*, 48(5):506–521, 1999. doi:10.1109/12.769433.

25 Christos H. Papadimitriou and Mihalis Yannakakis. The complexity of facets (and some facets of complexity). *Journal of Computer and System Sciences*, 28(2):244–259, 1984. doi:10.1016/0022-0000(84)90068-0.

26 Toniann Pitassi and Rahul Santhanam. Effectively polynomial simulations. In *Proceedings of the 1st Innovations in Computer Science (ICS)*, pages 370–382. Tsinghua University Press, 2010.

27 John A. Robinson. A machine-oriented logic based on the resolution principle. *Journal of the ACM*, 12(1):23–41, 1965. doi:10.1145/321250.321253.

28 Stefan Szeider. The complexity of resolution with generalized symmetry rules. *Theory of Computing Systems*, 38(2):171–188, 2005. doi:10.1007/s00224-004-1192-0.

29 Grigori S. Tseitin. On the complexity of derivation in propositional calculus. *Zapiski Nauchnykh Seminarov LOMI*, 8:234–259, 1968.

30 Alasdair Urquhart. The symmetry rule in propositional logic. *Discrete Applied Mathematics*, 96–97:177–193, 1999. doi:10.1016/S0166-218X(99)00039-6.

31 Allen Van Gelder. Verifying RUP proofs of propositional unsatisfiability. In *Proceedings of the 10th International Symposium on Artificial Intelligence and Mathematics (ISAIM)*, 2008.

32 Nathan Wetzler, Marijn J. H. Heule, and Warren A. Hunt, Jr. DRAT-trim: Efficient checking and trimming using expressive clausal proofs. In *Proceedings of the 17th International Conference on Theory and Applications of Satisfiability Testing (SAT)*, number 8561 in Lecture Notes in Computer Science, pages 422–429. Springer, 2014. doi:10.1007/978-3-319-09284-3_31.