DISCRETE APPLIED MATHEMATICS

ELSEVIER Discrete Applied Mathematics 96–97 (1999) 149–176

On a generalization of extended resolution ☆

O. Kullmann *

Johann Wolfgang Goethe-Universität, Fachbereich Mathematik D-60054 Frankfurt, Germany

Received 1 November 1996; revised 1 May 1997; accepted 5 October 1998

Abstract

Motivated by improved SAT algorithms ((O. Kullmann, DIMACS Series, vol. 35, Amer. Math. Soc., Providence, RI, 1997; O. Kullmann, Theoret. Comput. Sci. (1999); O. Kullmann, Inform. Comput., submitted); yielding new worst-case upper bounds) a natural parameterized generalization GER of Extended Resolution (ER) is introduced. ER can simulate polynomially GER, but GER allows special cases for which exponential lower bounds can be proven.

Keywords: Propositional logic; Extended resolution; Blocked clauses; Generalized extended resolution; Lower bounds

1. Introduction

1.1. Extended resolution

G. Tseitin introduced in [21] the *Extension Rule* for the Resolution Calculus:

$$F \to F \cup \{\{\overline{v}, \overline{a}, \overline{b}\}, \{v, a\}, \{v, b\}\}$$

for arbitrary variables a, b and a *new* variable v (new relative to the set F of premises and to a, b). Thereby the clause-set $\{\{\overline{v}, \overline{a}, \overline{b}\}, \{v, a\}, \{v, b\}\}$ is the Conjunctive Normal Form of the formula $v \leftrightarrow (\overline{a} \vee \overline{b})$.

An *Extended Resolution Proof* (for short: *ER proof*) of the empty clause $\bot$ from the clause-set F is an ordinary resolution proof of $\bot$ from F^*, where $F^* \supseteq F$ is obtained by repeated applications of the Extension Rule. The length of an ER proof is the (total) number of (different) clauses in it. We denote by $\mathrm{Comp}_{\mathrm{ER}}(F)$ the minimal length of an ER proof of $\bot$ from F.

Till today no super polynomial lower bound for $\mathrm{Comp}_{\mathrm{ER}}(F)$ is known. For all (concrete) examples of "difficult" formulas we know short ER proofs, because the Extension

☆ This work was supported by DFG-Leibniz-Programm Schn 143/5-1.
* *E-mail address*: kullmann@mi.informatik.uni-frankfurt.de (O. Kullmann)

PII: S0166-218X(99)00037-2

Rule enables one to mimic the (informal) proofs of unsatisfiability for the (concrete) examples. In [4–6,9] it is proved that ER has the same power (up to polynomial transformations) as the most powerful (known) proof systems, Extended Frege Systems or Frege Systems with the substitution rule.

1.2. Blocked clauses

In this note [1] I make some remarks on a (natural) generalization of the concept of ER proofs. This generalization is based on the concept of *Blocked Clauses.* Blocked Clauses are special cases of *redundant clauses*, i.e., they can be satisfiability-equivalently added to or eliminated from the given clause-set. They are defined by the condition that there is a literal l in them such that every candidate for a resolution on this literal, i.e., every clause in the given clause-set containing $\bar{l}$, also contains another complementary literal and hence the resolvent is tautological. Note that for the above Extension Rule all three new clauses are "blocked" for the literal $\overline{v}$ respectively v (in any order of addition), and thus the addition of Blocked Clauses covers the Extension Rule.

The concept of blocked clauses has been developed with the aim to improve worst-case upper bounds for SAT-algorithms. In [14,15] (see also [13]) the addition and elimination of blocked clauses under various circumstances is an important tool for improving the bound for 3-SAT-decision (no clause has more than three literals) to 1.5045^n (n= number of variables) and for improving the bound for SAT-decision to $2^{\ell/9}$ (ℓ= number of literal occurrences).

In all these applications we use blocked clauses *without new variables*. This restriction is important to obtain control on the process of introducing new clauses: If a clause is blocked for a literal which is not new, then this clauses *reflects a certain structure* of the clause-set in consideration, while the Extension Rule *does not depend on the structure* of the set of premises.

A predecessor of these methods (in a more general context) is the concept of Complement Search in [18].

1.3. Generalized extended resolution

Although the addition of blocked clauses already generalizes the above Extension Rule in a symmetrical manner, it is still not fully satisfactory because of the dependence on the order of additions. For example, consider the following $2 \cdot 3 = 6$ clauses coming from two applications of the Extension Rule:

$$C_1 = \{\overline{v}, \overline{a}, \overline{b}\}, \quad C_2 = \{v, a\}, \quad C_3 = \{v, b\},$$
$$C_4 = \{\overline{w}, \overline{v}, \overline{c}\}, \quad C_5 = \{w, v\}, \quad C_6 = \{w, c\}.$$

As mentioned, for $i = 1, \ldots, 6$ the clause C_i is blocked w.r.t. $F \cup \{C_1, \ldots, C_{i-1}\}$, but if we add one of the clauses C_4, C_5, C_6 first, then one of the clauses C_1, C_2, C_3 is thereafter no longer blocked.

[1] A first version appeared in [12].

To overcome this dependence on the order of added clauses, we generalize the addition of blocked clauses in the following natural way:

For a clause-set F we define the *kernel* $K(F)$ as the (uniquely defined) subset of F obtained by repeated eliminations of blocked clauses until no blocked clause is left.

Now a *Blocked Extension* F' of F is any clause-set F' with the property: $K(F \cup F') = K(F)$.

A *Generalized ER proof* (for short: *GER proof*) for F is a resolution proof of $\perp$ from $F \cup F'$, where F' is a blocked extension of F (whether F' is a superset of F, or only contains the new clauses, does not matter).

1.4. Results

The addition of one blocked clause C to F can speed up resolution proofs for F at most by a factor $|C|$ (while, due to "hidden" additions of blocked clauses, an extension by one clause in the GER calculus can already cause an exponential speed up).

The concept of GER proofs is a generalization of ER proofs (an extension F^* by ER is also a blocked extension of F), which eliminates the special form of the additional clauses as well as the special ordering of introduction. However, the strength of the calculus is not increased: We show that the ER calculus can polynomially simulate the GER calculus.

GER allows to study the effect of various restrictions on the blocked extension F'. We obtain the following results:

1. 1-clauses in F' containing new variables (relative to F) can be eliminated without affecting the shortest proof length. In minimal unsatisfiable F we can eliminate all 1-clauses from F'.
2. If F' contains only clauses of length less than or equal to 2, then we can eliminate all new variables from F'. Thus new variables play a role only if F' contains at least one clause of length at least 3.
3. "$(i+1,0)$-GER" cannot be simulated polynomially by "$(i,0)$-GER" for $i = 0,1,2$, where "$(i,0)$-GER" stands for the GER calculus with the restrictions:
   3.1. the blocked extensions F' contain only clauses of length less or equal i,
   3.2. F' does not contain new variables,
   (thus "$(0,0)$-GER" is the ordinary resolution calculus).
4. We give an (sub)exponential lower bound for those GER proofs, which only use blocked extensions F' *without new variables* (but with arbitrary clause length). The "hard formulas" here are the Pigeonhole formulas. Hence, although the addition of clauses without new variables can cause an exponential speed up (see result 3 above), in general the introduction of new variables is necessary (ER admits proofs of polynomial size for the Pigeonhole formulas; see [2] or [6]).

Considering the simulation of GER by ER mentioned above, the exponential lower bound in 4 shows that there are some "harmless" applications of the Extension Rule.

Prevailing backtracking algorithms for SAT-decision can be simulated by ordinary resolution and thus the exponential lower bounds carries over. See [10,11] and also [13] for a generalization of the (well-known) simulation by (regular) resolution of SAT decision algorithms using (only) backtracking in its simplest form (i.e., “semantic trees”, see [22]).

However, for algorithms working with blocked clauses (without new variables) the resolution calculus has to be generalized. As a first step by the lower bound for the GER calculus without new variables we obtain a lower bound for a general class of “DPLL-like” algorithms, whose reduction component allows the addition of resolvents and of blocked clauses without new variables (but does not eliminate clauses). See [11].

1.5. Organization of the paper

- Section 2 introduces the notations used in this paper.
- Section 3 gives the background for the Resolution Calculus.
- The notion of Blocked Clauses is introduced in Section 4, and examples for its use for SAT decision are given.
- The maximal speed up achieved by adding one blocked clause is estimated in Section 5.
- Blocked Extensions and the GER calculus are introduced in Section 6.
- Section 7 is devoted to the simulation of GER by ER.
- 1- and 2-clauses in blocked extensions are discussed in Section 8.
- The exponential lower bound for GER without new variables is proven in Section 9.
- Finally some open problems are given in Section 10.

2. Notation

By $\mathscr{VA}$ we denote the set of *variables* and by $\mathscr{LIT} := \{v\colon v \in \mathscr{VA}\} \cup \{\bar{v}\colon v \in \mathscr{VA}\}$ the set of *literals* ($\bar{l}$ is the *complement* of l, $\bar{\bar{l}} = l$).

A *clause* is a finite and complement-free (i.e., non-tautological) set of literals. We denote by $\mathscr{CL}$ the set of all clauses: $\mathscr{CL} := \{C \subseteq \mathscr{LIT}\colon C \text{ finite and } C \cap \overline{C} = \emptyset\}$, where for $L \subseteq \mathscr{LIT}\colon \overline{L} := \{\bar{l}\colon l \in L\}$ (i.e., a set of literals (clauses for example) is complemented elementwise).

A *clause-set* is a finite set of clauses, the set of all clause-sets is $\mathscr{CLS} := \{F \subseteq \mathscr{CL}\colon F \text{ finite}\}$.

A special clause is the *empty clause* $\bot := \emptyset \in \mathscr{CL}$, a special clause-set is the *empty clause-set* $\top := \emptyset \in \mathscr{CLS}$.

By $\mathrm{Var}(l) \in \mathscr{VA}$ for $l \in \mathscr{LIT}$ we denote the variable of l ($l = \mathrm{Var}(l)$ or $\bar{l} = \mathrm{Var}(l)$), and we use $\mathrm{Var}(C) := \{\mathrm{Var}(l)\colon l \in C\}$, $\mathrm{Var}(F) := \bigcup_{C \in F} \mathrm{Var}(C)$. Furthermore, $\mathrm{Lit}(F) := \bigcup_{C \in F} C$.

A literal l is *pure* for $F \in \mathscr{CLS}$ iff $\bar{l} \notin \mathrm{Lit}(F)$.

A *partial assignment* φ is a complement-preserving mapping $\varphi : L \to \{0,1\}$ for $L \subseteq \mathscr{LIT}$ closed under complement: $\overline{L} = L$. We define $\mathrm{Var}(\varphi) := \mathrm{Var}(L)$.

$\varphi(C)=1$ holds for a clause C iff $\exists l \in C$: $\varphi(l)=1$, and $\varphi(C)=0$ holds iff $\forall l \in C : \varphi(l)= 0$; otherwise (i.e., no literal of C is mapped to 1 by φ, and φ is not defined for at least one literal of C) the term $\varphi(C)$ is undefined.

$\varphi(F) = 1$ for a clause-set F iff $\forall C \in F$: $\varphi(C) = 1$; $\varphi(F) = 0$ iff $\exists C \in F$: $\varphi(C) = 0$; otherwise, $\varphi(F)$ is undefined.

For $C \in \mathcal{CL}$ with $\varphi(C) \neq 1$ we denote by

$$\varphi * C := C \setminus \{l \in C: \varphi(l) = 0\} \in \mathcal{CL}$$

the result of substituting truth-values via φ in C ("$\varphi(l) = 0$" implies that φ is defined on l), and for $F \in \mathcal{CLS}$:

$$\varphi * F := \{\varphi * C: C \in F \wedge \varphi(C) \neq 1\} \in \mathcal{CLS}.$$

By $\langle l_1 \rightarrow \epsilon_1, \dots, l_n \rightarrow \epsilon_n \rangle$ we denote the partial assignment φ with $\mathrm{Var}(\varphi)=\mathrm{Var}(\{l_1, \dots, l_n\})$ and $\varphi(l_i) = \epsilon_i \in \{0, 1\}$ for $i \in \{1, \dots, n\}$.

A *substitution* σ is a complement-preserving mapping σ: $\mathcal{LIT} \rightarrow \mathcal{LIT}$. For $C \in \mathcal{CL}$ we define $\sigma(C) := \{\sigma(l)$: $l \in C\}$. Note that $\sigma(C) \notin \mathcal{CL}$ is possible since clauses must be complement-free. And for $F \in \mathcal{CLS}$ we define

$$\sigma(F) := \{\sigma(C): C \in F \wedge \sigma(C) \in \mathcal{CL}\} \in \mathcal{CLS}.$$

A *renaming* is a bijective substitution. We write σ: $L_1 \rightarrow L_2$ for substitutions σ and subsets $L_1, L_2 \subseteq \mathcal{LIT}$, iff $\sigma(L_1) \subseteq \sigma(L_2)$ holds and σ is the identity on $\mathcal{VA} \setminus \mathrm{Var}(L_1)$.

3. The resolution calculus

Definition 3.1. A clause $R \in \mathcal{CL}$ *is the Resolvent of clauses* $C_1, C_2 \in \mathcal{CL}$ $(C_1, C_2 \vdash R)$ iff there is $l \in C_1$ with $\bar{l} \in C_2$ such that $R = (C_1 \setminus \{l\}) \cup (C_2 \setminus \{\bar{l}\})$. (Note that, because R is a clause which are defined to be complement-free, the resolution literal l is uniquely determined: $C_1 \cap \overline{C_2} = \{l\}$.)

A *Resolution Proof* $\mathcal{P}$ *of* $F_2 \in \mathcal{CLS}$ *from* $F_1 \in \mathcal{CLS}$ is a sequence $\mathcal{P}=(C_1, \dots, C_n)$ of clauses ($C_i \in \mathcal{CL}$, $n \geqslant 0$), such that the following holds:

1. For each $C \in F_2$ there is $i \in \{1, \dots, n\}$ with $C_i \subseteq C$.
2. For $i \in \{1, \dots, n\}$ we have $C_i \in F_1$, or there are $j, k \in \{1, \dots, i-1\}$ with $C_j, C_k \vdash C_i$.

The *length* of $\mathcal{P}$ is n. A *Resolution Proof for* $F \in \mathcal{CLS}$ is a resolution proof of $\{\bot\}$ from F. For $F, F_1, F_2 \in \mathcal{CLS}$ and $C \in \mathcal{CL}$ we define [2]

$$\mathrm{Comp_R}(F_1, F_2) := \inf\{n: \exists \mathcal{P} \text{ resolution proof of } F_2 \text{ from } F_1 \text{ with length } n\},$$
$$\mathrm{Comp_R}(F, C) := \mathrm{Comp_R}(F, \{C\}),$$
$$\mathrm{Comp_R}(F) := \mathrm{Comp_R}(F, \bot).$$

[2] $\inf \emptyset = +\infty$.

Remark

1.
   1.1. $\mathrm{Comp}_\mathrm{R}(F_1, F_2) = 0 \Leftrightarrow F_2 = \top$,
   1.2. $\mathrm{Comp}_\mathrm{R}(F) = \infty \Leftrightarrow F \in \mathcal{SAT}$.
2. Resolution proofs have the structure of a forest.
3. Resolution proofs $(C_1, \dots, C_n)$ of $\bot$ can be restricted w.l.o.g. to proofs containing less than n^2 literal occurrences ($\sum_{i=1}^n |C_i| < n^2$), because the length of a clause decreases at most by one by a resolution step, and thus clauses C_i with $|C_i| > n - i$ can be eliminated.

Lemma 3.1. *For $F, F_1, F_1', F_2, F_2' \in \mathcal{CLS}$ and a partial assignment φ we have*

1. *If the two conditions*
   1.1. $\forall C \in F_1 \exists C' \in F_1'\colon C' \subseteq C$,
   1.2. $\forall C' \in F_2' \exists C \in F_2\colon C \subseteq C'$
   hold, then: $\mathrm{Comp}_\mathrm{R}(F_1', F_2') \leqslant \mathrm{Comp}_\mathrm{R}(F_1, F_2)$.
2. $\mathrm{Comp}_\mathrm{R}(\varphi * F_1, \varphi * F_2) \leqslant \mathrm{Comp}_\mathrm{R}(F_1, F_2)$.
3. *If* $\forall C \in F_2\colon \varphi(C) \neq 1$, *and furthermore one of the following two conditions hold*:
   3.1. $\varphi * F_1 \subseteq F_1$, *or*
   3.2. $\forall C \in F_2\colon \mathrm{Var}(\varphi) \subseteq \mathrm{Var}(C)$
   then $\mathrm{Comp}_\mathrm{R}(\varphi * F_1, \varphi * F_2) = \mathrm{Comp}_\mathrm{R}(F_1, F_2)$.

Proof. 1. For a resolution proof $(C_1, \dots, C_n)$ of F_2 from F_1 one constructs inductively (in a straight forward manner) a resolution proof $(C_1', \dots, C_n')$ of F_2' from F_1' with $C_i' \subseteq C_i$ for $i = 1, \dots, n$.

2. By part 1 we have

$$\mathrm{Comp}_\mathrm{R}(F_1, F_2) \geqslant \mathrm{Comp}_\mathrm{R}((\varphi * F_1) \cup \{C \in F_1\colon \varphi(C) = 1\}, F_2).$$

Let $\mathcal{P} = (C_1, \dots, C_n)$ be a proof of F_2 from $(\varphi * F_1) \cup \{C \in F_1\colon \varphi(C) = 1\}$, and define $G := \{C_i\colon C_i \in F_1 \setminus (\varphi * F)\}$.

Prove inductively that for $C_j \in G^*$ we have $\varphi(C_j) = 1$, where G^* is the set of all successors in G in the underlying forest structure of $\mathcal{P}$.

Thus after elimination of G^* from $\mathcal{P}$ we still have a resolution proof of $F_2 \setminus \{C \in F_2\colon \varphi(C) = 1\}$ from $\varphi * F_1$, which in fact must be a proof of $\varphi * F_2$ from $\varphi * F_1$.

3. Because of part 2 we only have to prove

$$\mathrm{Comp}_\mathrm{R}(\varphi * F_1, \varphi * F_2) \geqslant \mathrm{Comp}_\mathrm{R}(F_1, F_2).$$

3.1. Here $(*)$ follows from part 1.
3.2. Consider a proof $(C_1, \dots, C_n)$ of $\varphi * F_2$ from $\varphi * F_1$. Replace all axioms $\varphi * C$ for $C \in F_1$ (with $\varphi(C) \neq 1$) by C and obtain a proof of F_1 from F_2. □

Definition 3.2. For a clause C we define the corresponding partial assignment φ_C by

$$\varphi_C := \langle l \to 0 \colon l \in C \rangle.$$

Corollary 3.2. *For $F \in \mathscr{CLS}$ and $C \in \mathscr{CL}$ we have*

$$\mathrm{Comp}_{\mathrm{R}}(F, C) = \mathrm{Comp}_{\mathrm{R}}(\varphi_C * F).$$

Lemma 3.3 (Generalized splitting lemma). *For clause-sets $F, T \in \mathscr{CLS}$ such that T is minimally unsatisfiable (i.e., $T \notin \mathscr{SAT}$, and $\forall C \in T \colon T \setminus \{C\} \in \mathscr{SAT}$) we have*

$$\begin{aligned} \mathrm{Comp}_{\mathrm{R}}(F) &\leqslant \mathrm{Comp}_{\mathrm{R}}(F, T) + \mathrm{Comp}_{\mathrm{R}}(T) - |T| \\ &\leqslant \sum_{C \in T} \mathrm{Comp}_{\mathrm{R}}(\varphi_C * F) + \mathrm{Comp}_{\mathrm{R}}(T) - |T|. \end{aligned}$$

Proof. For the first inequality note that in a proof for T all clauses of T must occur (because of the minimality condition), and thus when combining the proofs of T from F with the proof for T we can subtract $|T|$ clauses.

The second inequality follows by $\mathrm{Comp}_{\mathrm{R}}(F, T) \leqslant \sum_{C \in T} \mathrm{Comp}_{\mathrm{R}}(F, C)$ and Corollary 3.2. □

Lemma 3.4. *For $F \in \mathscr{CLS}$ and a substitution σ we have* $\mathrm{Comp}_{\mathrm{R}}(\sigma(F)) \leqslant \mathrm{Comp}_{\mathrm{R}}(F)$.

Proof. Consider a resolution proof $(C_1, \ldots, C_n)$ for F. Replace axioms $C_i \in F$ by $\sigma(C_i)$ and obtain a "pre"-resolution proof for $\{\sigma(C) \colon C \in F\}$, containing possibly tautological clauses, which can be eliminated by the following observations:

- if the resolvent is non-tautological, then at least one of its parent clauses is also non-tautological;
- a non-tautological resolvent R of a non-tautological clause C_1 with a tautological clause C_2 properly contains C_1: $C_1 \subset R$. □

3.1. Lower bounds

For a natural number $n \geqslant 2$ consider variables $v_{i,j}$ with $i \in \{1, \ldots, n\}$ and $j \in \{1, \ldots, n-1\}$. By PHP we denote the *Pigeonhole Principle*:

$$\begin{aligned} &\mathrm{PHP} := \mathrm{PHP}_n^p \cup \mathrm{PHP}_n^n, \\ &\mathrm{PHP}_n^p := \{\{v_{i,j}\}_{j \in \{1,\ldots,n-1\}}\}_{i \in \{1,\ldots,n\}}, \\ &\mathrm{PHP}_n^n := \{\{\overline{v_{i_1,j}}, \overline{v_{i_2,j}}\}\}_{\substack{i_1, i_2 \in \{1,\ldots,n\}, i_1 < i_2 \\ j \in \{1,\ldots,n-1\}}}. \end{aligned}$$

Variable $v_{i,j}$ has the meaning "pigeon i in hole j". The "positive" part PHP_n^p states that every pigeon i is in some hole j. And the "negative" part PHP_n^p states that no hole contains two pigeons. Thus, because there are more pigeons than holes, we have $\mathrm{PHP} \notin \mathscr{SAT}$. The size of PHP is: $|\mathrm{Var}(\mathrm{PHP})| = \mathrm{O}(n^2)$, and $c(\mathrm{PHP}), \ell(\mathrm{PHP}) = \mathrm{O}(n^3)$

where $c(F) := |F|$ is the number of clauses, and $\ell(F) := \sum_{C \in F} |C|$ is the number of literal occurrences.

Theorem 1 (Haken [8]). *For all $n \geqslant 1676$ using $c := 2^{1/20} = 1.03526..$ we have*

$$\mathrm{Comp}_{\mathrm{R}}(\mathrm{PHP}) \geqslant c^n.$$

The value for c is taken from [1]. Later on we will strengthen this result and then we give an outline of the proof.

4. Blocked Clauses and their use for SAT-decision

Definition 4.1. A clause $C \in \mathscr{CL}$ is called *blocked for $l \in \mathscr{LIT}$ with respect to $F \in \mathscr{CLS}$* iff all (envisaged) resolvents of C with $C' \in F$ for $\bar{l} \in C'$ are tautological, i.e., iff

$$l \in C \wedge \forall C' \in F[\bar{l} \in C' \Rightarrow \exists \overline{x} \in C'[\overline{x} \neq \bar{l} \wedge x \in C]]$$

holds. C is called *blocked w.r.t. F* iff there is a literal l such that C is blocked for l w.r.t. F.

For brevity we use for $F \in \mathscr{CLS}$ and $l \in \mathscr{LIT}$:

$$B_l(F) := \{C \in \mathscr{CL} \colon C \text{ blocked for } l \text{ w.r.t. } F\},$$

$$B(F) := \{C \in \mathscr{CL} \colon C \text{ blocked w.r.t. } F\} = \bigcup_{l \in \mathscr{LIT}} B_l(F)$$

$$B^{\mathrm{in}}(F) := B(F) \cap F.$$

Remark (for $F_1, F_2,\ F \in \mathscr{CLS}$, $C_1, C_2, C \in \mathscr{CL}$ and $l \in \mathscr{LIT}$)

1. As a first example consider
$$F := \{\{a,b\},\{\overline{a},b\},\{a,\overline{b}\}\} \Rightarrow B^{\mathrm{in}}(F) = \{\{\overline{a},b\},\{a,\overline{b}\}\}.$$
2. Smaller clause-sets have more blocked clauses: $F_1 \subseteq F_2 \Rightarrow B(F_2) \subseteq B(F_1)$.
3. Superclauses of blocked clauses are also blocked: $C_1 \subseteq C_2 \wedge C_1 \in B(F) \Rightarrow C_2 \in B(F)$.
4. Clauses with pure literals are blocked: $\overline{C} \nsubseteq \mathrm{Lit}(F) \Rightarrow C \in B(F)$ (especially clauses with new variables are blocked).
5. C is blocked w.r.t. $F \Leftrightarrow C$ is blocked w.r.t. $F \setminus \{C\} \Leftrightarrow C$ is blocked w.r.t. $F \cup \{C\}$.
6. $C \in B_l(F) \Leftrightarrow l \in C \wedge \varphi_{C \setminus \{l\}}(\{C \in F : \bar{l} \in C\}) = 1$.
7. If the clauses $C_1, \ldots, C_r$ are the only occurrences of l in F, then every clause C, containing $\bar{l}$ and from every C_i another complementary literal ($C \cap \overline{C_i \setminus \{l\}} \neq \emptyset$), is blocked for $\bar{l}$ w.r.t. F. For example:
   - 7.1. If $\{l, x_1, \ldots, x_m\}$ is the only occurrence of l in F, then the clauses $\{\bar{l}, \overline{x_1}\}, \ldots, \{\bar{l}, \overline{x_m}\}$ are blocked for $\bar{l}$ w.r.t. F.
   - 7.2. And if $\{l, x_1, \ldots, x_m\}$ and $\{l, y_1, \ldots, y_n\}$ are the only occurrences of l in F, then the clauses $\{\bar{l}, \overline{x_i}, \overline{y_j}\}$ ($i \in \{1, \ldots, m\}$, $j \in \{1, \ldots, n\}$) are blocked for $\bar{l}$ w.r.t. F.

(If $x_i = y_j$, then $\{\bar{l}, \overline{x_i}, \overline{y_j}\}$ in fact is a 2-clause, and if $x_i = \overline{y_j}$, then it is no clause at all (in our notation).)

8. Generalizing the example from 1 the following holds:[3]

$$\exists C \in F \forall C' \in F[\mathrm{Var}(C) = \mathrm{Var}(C')] \Rightarrow (B^{\mathrm{in}}(F) = \emptyset \Leftrightarrow |F| = 2^{|\mathrm{Var}(F)|}).$$

Lemma 4.1. *For $F \in \mathscr{CLS}$ and $C \in B(F)$ we have $F \setminus \{C\} \overset{\mathrm{sat}}{\equiv} F \overset{\mathrm{sat}}{\equiv} F \cup \{C\}$, where $\overset{\mathrm{sat}}{\equiv}$ denotes equivalence w.r.t. satisfiability ($F_1 \overset{\mathrm{sat}}{\equiv} F_2$ iff both F_1, F_2 are satisfiable or both F_1, F_2 are unsatisfiable).*

Proof. It is enough to show $F \overset{\mathrm{sat}}{\equiv} F \setminus \{C\}$ for $C \in F$. The direction $F \in \mathscr{SAT} \Rightarrow F \setminus \{C\} \in \mathscr{SAT}$ is obvious.

Consider a satisfying assignment φ for F: $\varphi(F \setminus \{C\}) = 1$. W.l.o.g.: $\mathrm{Var}(\varphi) = \mathrm{Var}(F)$. If $\varphi(C) = 1$ then immediately $\varphi(F) = 1$ also.

Otherwise, let C be blocked for l w.r.t. F. Thus we have $\varphi(l) = 0$. Define φ' by flipping the value of l: $\varphi'(l) := 1$ ($\varphi'(\bar{l}) := 0$) and $\varphi'(x) := \varphi(x)$ else.

Now $\varphi'(F) = 1$ holds, since on the one hand we have

$$\varphi'(l) = 1 \Rightarrow \varphi'(C) = 1$$

and on the other hand we have for $C' \in F \setminus \{C\}$:

- if $\bar{l} \notin C'$, then $\varphi'(C') = 1$ because of $\varphi(C') = 1$ and $\varphi'(l) = 1$;
- if $\bar{l} \in C'$ then there is another $a \in C \setminus \{l\}$ with $\bar{a} \in C'$ (because of the blocking condition) and by $\varphi'(a) = \varphi(a) = 0$ we have $\varphi'(C') = 1$ as well.

Another possibility for a proof is to use satisfiability equivalence of F and $\mathrm{DP}_l(F)$, where $\mathrm{DP}_l(F)$ denotes the result of substituting all clauses of F containing l or $\bar{l}$ by their non-tautological resolvents (on l): If C is blocked for l w.r.t. F, then addition of C to F has no effect on DP_l.

Or one uses completeness of (non-tautological) SL-resolution for any start clause from a minimal unsatisfiable clause-set and for any selected resolution literal from that clause. □

To become familiar with the concept of blocked clauses, and for later use, we determine the blocked clauses without new variables for PHP:

[3] **Proof.** For $G \in \mathscr{CLS}$ we define the "Resolution Graph" $R(G)$ as the undirected graph (without parallel edges and loops) whose vertices are the clauses of G, and an edge joins clauses C and C' iff $|C \cap \overline{C'}| = 1$ holds (i.e., C and C' have *exactly* one clashing literal).

Now consider $R(F)$ for the special F here, where all clauses contain the same variables. $R(F)$ is a sub-graph of $R(F^*)$ where F^* is that clause-set containing all 2^n clauses C with $\mathrm{Var}(C) = \mathrm{Var}(F)$ ($n = |\mathrm{Var}(F)|$).

If for $0 < |F| < 2^n$ no clause of F would be blocked w.r.t. F, then F would be a connected component of $R(F^*)$, since for every clause of F^* and every literal in it there is *exactly* one resolution partner (in F^*). But if G is unsatisfiable (for any G), then at least one connected component of $R(G)$ is also unsatisfiable. (The proof for that is not completely trivial since in the def. of $R(G)$ not all complementary literal pairs correspond to edges.)

Now we obtain a contradiction since F^* is minimally unsatisfiable.

Lemma 4.2. *For* $C \in \mathscr{CL}$ *and* $i \in \{1,\dots,n\}$, $j \in \{1,\dots,n-1\}$ *the following holds*:

1. $C \in B_{v_{i,j}}(\text{PHP}) \Leftrightarrow \{v_{1,j},\dots,v_{n,j}\} \subseteq C$.
2. $C \in B_{\overline{v_{i,j}}}(\text{PHP}) \Leftrightarrow \exists k \in \{1,\dots,n-1\}\backslash\{j\} : \{\overline{v_{i,j}}, \overline{v_{i,k}}\} \subseteq C$.

4.1. Blocked Clauses for SAT-decision

Two new features are invented for the improved 3-SAT-algorithm in [14] (see also [13]): "Generalized Autarkness" and "Blocked Clauses" [4]. The use of blocked clauses (*without new variables*) in [14] for 3-SAT-decision can be summarized as follows: [5]

1. Testing of a *not-blocked* 2-*clause* $\{a,b\}$, i.e., branching via $(\langle a \to 0, b \to 1\rangle, \langle a \to 1\rangle)$, has a greater impact on the formula than testing a blocked 2-clause. The existence of a not blocked 2-clause is established by a combination of autarkness with the blocking concept (called "Br-Autarkness" in [14]).
2. The main idea in [14] is to consider not only the decrease in the number of variables in the course of the algorithm, but also to consider the alteration in the number of 2-clauses: An *increase in the number of* 2-*clauses* can shorten the computation. Now in the situation of point 7.2 in Remark (given after Definition 4.1) one can add blocked 3-clauses, which become new 2-clauses after applying $\langle l \to 1\rangle$.
3. And also the situation of point 7.1 in the Remarks (given after Definition 4.1) is applied, but with a different effect: Here, by applying $\langle l \to 1\rangle$, more variables vanish because of 1-clause eliminations.
4. Blocked 3-clauses are eliminated to establish some "normal form". (Blocked 2-clauses are not eliminated since they count for the analysis.)

The application of point 7 of the Remark to SAT-decision is already prescribed in [18], called "Complement Search" there (in a more general setting).

The elimination of blocked clauses and the (implicit) addition of blocked 2-clauses is also part of the "$\mathscr{L}$"-algorithm from [15] with the improved bound $2^{1/10 \cdot \ell}$ for SAT-decision.

5. An upper bound for the speed up achieved by adding one blocked clause

Lemma 5.1. *For* $F \in \mathscr{CLS}$, *a clause* $C \in \mathscr{CL}$, *a literal* l *and a partial assignment* φ *fulfilling* $\varphi(C) \neq 1$ *and* $\text{Var}(l) \notin \text{Var}(\varphi)$ *we have*

$$C \in B_l(F) \;\;\Rightarrow\;\; \varphi * C \in B_l(\varphi * F).$$

(*The opposite direction is true under the additional assumption* $\text{Var}(\varphi) \subseteq \text{Var}(C)$.)

[4] The "Autarkness Principle" has been introduced in [17] to ensure the existence of a 2-clause in the course of the (recursive) decision procedure, and also [16] used a similar scheme. "Generalized Autarkness" is a branching scheme which gives an alternative branching in the case of an *arbitrary* number of new 2-clauses. See [14].

[5] Ref. [19] applies our concept of blocked clauses in a manner similar to 2 and 3 below.

Now we are able to prove an upper bound for the speed up achieved by adding one blocked clause.

Lemma 5.2. *For $F \in \mathscr{CLS}$ and $C \in B(F)$ we have*

$$\mathrm{Comp}_{\mathrm{R}}(F) \leqslant |C| \cdot \mathrm{Comp}_{\mathrm{R}}(F \cup \{C\}) + |C| - 1.$$

Proof. (By induction on $|C|$)

$|C| = 1$: Thus $C = \{l\}$, where l is pure for $F \Rightarrow \mathrm{Comp}_{\mathrm{R}}(F) = \mathrm{Comp}_{\mathrm{R}}(F \cup \{C\})$.

$|C| > 1$: Let $C \in B_l(F)$ and choose $x \in C \backslash \{l\}$.

$\langle x \to 0\rangle * C = C \backslash \{x\} \in B_l(\langle x \to 0\rangle * F)$ by the previous lemma. Thus by the induction hypothesis (and Lemma 3.1, part 2):

$$\begin{aligned}\mathrm{Comp}_{\mathrm{R}}(\langle x \to 0\rangle * F) &\leqslant (|C| - 1) \cdot \mathrm{Comp}_{\mathrm{R}}((\langle x \to 0\rangle * F) \cup (\langle x \to 0\rangle * \{C\})) \\ &\qquad + |C| - 2 \\ &= (|C| - 1) \cdot \mathrm{Comp}_{\mathrm{R}}(\langle x \to 0\rangle * (F \cup \{C\})) + |C| - 2 \\ &\leqslant (|C| - 1) \cdot \mathrm{Comp}_{\mathrm{R}}(F \cup \{C\}) + |C| - 2.\end{aligned}$$

On the other hand we have

$$\mathrm{Comp}_{\mathrm{R}}(\langle x \to 1\rangle * F) = \mathrm{Comp}_{\mathrm{R}}(\langle x \to 1\rangle * (F \cup \{C\})) \leqslant \mathrm{Comp}_{\mathrm{R}}(F \cup \{C\}).$$

Together by Lemma 3.3

$$\begin{aligned}\mathrm{Comp}_{\mathrm{R}}(F) &\leqslant \mathrm{Comp}_{\mathrm{R}}(\langle x \to 0\rangle * F) + \mathrm{Comp}_{\mathrm{R}}(\langle x \to 1\rangle * F) + 3 - 2 \\ &\leqslant |C| \cdot \mathrm{Comp}_{\mathrm{R}}(F \cup \{C\}) + |C| - 1. \qquad \square\end{aligned}$$

6. Blocked extensions and generalized extended resolution

Definition 6.1. A sequence $C_1, \ldots, C_n$ $(n \geqslant 0)$ of clauses is called a *blocking sequence for* the clause-set $F \in \mathscr{CLS}$ iff

$$\forall i \in \{1, \ldots, n\}[C_i \in B^{\mathrm{in}}(F \backslash \{C_1, \ldots, C_{i-1}\})]$$

holds. Literals $l_1, \ldots, l_n$ are called *blocking literals for* $C_1, \ldots, C_n$ iff all C_i are blocked for l_i w.r.t. $F \backslash \{C_1, \ldots, C_{i-1}\}$.

A blocking sequence $C_1, \ldots, C_n$ is called *maximal* iff $B^{\mathrm{in}}(F \backslash \{C_1, \ldots, C_n\}) = \emptyset$.

Note that a blocked clause w.r.t. F may or may not be in F, while a blocking sequence for F is always contained in F.

Lemma 6.1. *Two maximal blocking sequences for $F \in \mathscr{CLS}$ differ only by a permutation of the clauses.*

Proof. Consider two maximal blocking sequences $C_1, \ldots, C_n$ and $D_1, \ldots, D_m$ for F. We prove by induction that for $i \in \{1, \ldots, n\}$ we have

$$C_i \in \{D_1, \ldots, D_m\}$$

and thus by symmetry the assertion follows. Suppose $\{C_1,\dots,C_{i-1}\}\subseteq\{D_1,\dots,D_m\}$. Assume $C_i\notin\{D_1,\dots,D_m\}$.

Then $C_i\in F\setminus\{D_1,\dots,D_m\}$. Furthermore we have $F\setminus\{D_1,\dots,D_m\}\subseteq F\setminus\{C_1,\dots,C_{i-1}\}$, and thus by Remark 2 after Definition 4.1 also $D_1,\dots,D_m,C_i$ would be a blocking sequence for F contradicting the maximality of $D_1,\dots,D_m$. □

Definition 6.2. By the previous lemma we are justified to define the *kernel* $K(F)$ for $F\in\mathcal{CLS}$:

$$K(F):=F\setminus\{C_1,\dots,C_n\}\quad\text{where } C_1,\dots,C_n \text{ is a maximal blocking sequence for } F.$$

K is a "kernel operator": $F_1\subseteq F_2\Rightarrow K(F_1)\subseteq K(F_2)$, $K(K(F))=K(F)$, $K(F)\subseteq F$.

Lemma 6.2. *For $F\in\mathcal{CLS}$ we have $K(F)\overset{\text{sat}}{\equiv}F$.*

Definition 6.3. $F'\in\mathcal{CLS}$ is called a *Blocked Extension* for $F\in\mathcal{CLS}$ iff $K(F\cup F')=K(F)$ holds.

Lemma 6.3. *F' blocked extension for $F\Rightarrow F\overset{\text{sat}}{\equiv}F\cup F'$.*

Proof. $F\overset{\text{sat}}{\equiv}K(F)=K(F\cup F')\overset{\text{sat}}{\equiv}F\cup F'$. □

Remark

1. The property "F' is a blocked extension for F" is polynomially decidable.
2. If C is blocked w.r.t. F, then $\{C\}$ is a blocked extension for F. If $K(F)=F$ holds, then also the opposite direction is true (for example: F minimal unsatisfiable $\Rightarrow K(F)=F$).
3. $F_1\subseteq F_2$, $F_1'\subseteq F_2'$, F_2' blocked extension for $F_2\Rightarrow F_1'$ blocked extension for F_1.
4. Consider $F\in\mathcal{CLS}$, an arbitrary propositional formula A (over $\mathcal{VA}$), and a variable $v\in\mathcal{VA}\setminus(\mathrm{Var}(F)\cup\mathrm{Var}(A))$. Suppose E is a CNF of $v\leftrightarrow A$ (i.e., the formula $\bigwedge_{C\in E}\bigvee_{l\in C} l$ is (logically) equivalent to $v\leftrightarrow A$). Then E is a blocked extension for F: Every clause $C\in E$ is blocked for v or $\overline{v}$ w.r.t. E (otherwise a clause C' with $v\notin\mathrm{Var}(C')$ would follow from $v\leftrightarrow A$).
5. Hence the Extension Rule (and its obvious generalization) is covered by the use of blocked extensions.
6. Unlike extensions by the Extension Rule, blocked extensions are not conservative extensions since for example $\{\{\overline{a},b\}\}$ is a blocked extension for $\{\{a,b\}\}$ while $\{\{\overline{a},b\}\}$ does not contain new variables and does not follow from $\{\{a,b\}\}$.
7. If F' is a blocked extension for F with $F'\cap F=\emptyset$, then $F'\in\mathcal{SAT}$ holds, because F' is also a blocked extension for $\top\in\mathcal{SAT}$. More generally it holds for any blocked extension F' for $F\in\mathcal{CLS}$:

$$\forall S\subseteq F[F\setminus S\in\mathcal{SAT}\Rightarrow(F\cup F')\setminus S\in\mathcal{SAT}].$$

The next lemma provides an alternative characterization of blocked extensions by iterated addition of single clauses.

Lemma 6.4. *A clause-set $F' \in \mathscr{CLS}$ is a blocked extension for $F \in \mathscr{CLS}$ iff there exists an order $F' = \{C_1, \dots, C_n\}$ such that*

$$\forall i \in \{1, \dots, n\}[\{C_i\} \text{ is a blocked extension for } F \cup \{C_1, \dots, C_{i-1}\}] \qquad (\dagger)$$

holds, and also iff for all orders $F' = \{C_1, \dots, C_n\}$ condition $(\dagger)$ holds. Thus the notion of "Blocked Extension" overcomes the special order inherent to the Extension Rule.

Proof. (a) Consider a blocked extension F' for F and an (arbitrary) order $F' = \{C_1, \dots, C_n\}$. Then $(\dagger)$ is a simple conclusion from Remark 2 of Section 4.

(b) Consider $F' \in \mathscr{CLS}$, $F' = \{C_1, \dots, C_n\}$ such that $(*)$ holds. W.l.o.g. $F' \cap F = \emptyset$.

Suppose that $D_1, \dots, D_m$ is a maximal blocking sequence for $F \cup \{C_1, \dots, C_n\}$. We have to show $\{C_1, \dots, C_n\} \subseteq \{D_1, \dots, D_m\}$.

Assume that there is a maximal index $i \in \{1, \dots, n\}$ with $C_i \notin \{D_1, \dots, D_m\}$. We know that $\{C_i\}$ is a blocked extension for $F \cup \{C_1, \dots, C_{i-1}\}$. Now, since $\{C_{i+1}, \dots, C_n\} \subseteq \{D_1, \dots, D_m\}$ holds, by Lemma 6.1 we conclude $C_i \in \{D_1, \dots, D_m\}$ contradicting our assumption. □

Lemma 6.5. *Call clause-set $F' \in \mathscr{CLS}$ a "simple blocked extension" for $F \in \mathscr{CLS}$ iff there is an order $F' \backslash F = \{C_1, \dots, C_n\}$ such that*

$$\forall i \in \{1, \dots, n\}[C_i \text{ is blocked w.r.t. } F \cup \{C_1, \dots, C_{i-1}\}]$$

holds. Now for F, $F' \in \mathscr{CLS}$ the following assertions are equivalent:

1. *F' is a blocked extension for F.*
2. *There exists $F_0 \subseteq F$ such that $F' \cup (F \backslash F_0)$ is a simple blocked extension for F_0.*
3. *$F' \cup (F \backslash K(F))$ is a simple blocked extension for $K(F)$.*

Proof. W.l.o.g. $F \cap F' = \emptyset$.

(i) $\Rightarrow$ (iii): Reverse the order of a maximal blocking sequence for $F \cup F'$ and obtain a sequence required in the definition of "simple blocked extensions".

(iii) $\Rightarrow$ (ii): $F_0 := K(F)$.

(ii) $\Rightarrow$ (i): Under the assumption (ii) we have $K(F \cup F') \subseteq F_0$ and thus F' is a blocked extension for F. □

Definition 6.4. A *Generalized Extended Resolution Proof* $\mathscr{P}$ (for short: *GER proof*) for $F \in \mathscr{CLS}$ is a pair $\mathscr{P} = (F', (C_1, \dots, C_n))$, such that F' is a blocked extension for F, and $(C_1, \dots, C_n)$ is a resolution proof for $F \cup F'$. The *length* of $\mathscr{P}$ is n. For $F \in \mathscr{CLS}$ we define:

$$\mathrm{Comp}_{\mathrm{GER}}(F) := \inf\{n \in \mathbb{N}\colon \exists \mathscr{P} \text{ GER proof for } F \text{ of length } n\}.$$

A GER proof $\mathscr{P} = (F', (C_1, \dots, C_n))$ is, more specifically, an (a, b)-*Resolution Proof* for functions $a, b : \mathscr{CLS} \to \mathbb{N}_0 \cup \{+\infty\}$, iff F' *is an* (a, b)-*extension for* F, which means that the length of the new clauses is bounded by a: $\forall C \in F'[|C| \leqslant a(F)]$, and the number of new variables is bounded by b: $|\mathrm{Var}(F') \setminus \mathrm{Var}(F)| \leqslant b(F)$.

$$\mathrm{Comp}_{(a,b)}(F) := \inf\{n \in \mathbb{N}\colon \exists \mathscr{P}\ (a,b)\text{-resolution proof for } F \text{ of length } n\}.$$

Remark

1.
   1.1. $\mathrm{Comp}_{(0,0)} = \mathrm{Comp}_{(0,\infty)} = \mathrm{Comp}_{\mathrm{R}}$;
   1.2. $\mathrm{Comp}_{(3,\infty)} \leqslant \mathrm{Comp}_{\mathrm{ER}}$;
   1.3. $\mathrm{Comp}_{(\infty,\infty)} = \mathrm{Comp}_{\mathrm{GER}}$.
2. $\mathrm{Comp}_{(a,b)}(F) = \min\{\mathrm{Comp}_{\mathrm{R}}(F \cup F')\colon F'$ blocked (a,b)-extension for $F\}$.
3. $a \geqslant a',\ b \geqslant b' \Rightarrow \mathrm{Comp}_{(a,b)} \leqslant \mathrm{Comp}_{(a',b')}$.
4. To obtain more general concepts of proofs, the concept of a kernel $K(F)$ could be generalized to any polynomially computable $K : \mathscr{CLS} \to \mathscr{CLS}$ such that $\forall F \in \mathscr{CLS}\colon K(F) \overset{\text{sat}}{\equiv} F$ holds.

7. Polynomial simulation of GER by ER

Definition 7.1. For clause-sets $F_1,\ F_2 \in \mathscr{CLS}$ we define $F_1 \cong F_2$ iff there is a renaming σ with $\sigma(F_1) = F_2$.

Definition 7.2. $F' \in \mathscr{CLS}$ is called a *Normal Extension* for $F \in \mathscr{CLS}$ iff there is $m \geqslant 0$ with $|F'| = 3m$ and there is an order $F' = \{C_0, \ldots, C_{3m-1}\}$ such that for $i \in \{0, \ldots, m-1\}$ the following holds:

1. $C_{3i} = \{\overline{v_i}, \overline{a_i}, \overline{b_i}\}$, $a_i,\ b_i \in \mathscr{VA}$, $a_i \neq b_i$, $v_i \in \mathscr{VA} \setminus (\mathrm{Var}(F \cup \{C_0, \ldots, C_{3i-1}\}) \cup \{a_i, b_i\})$.
2. $C_{3i+1} = \{v_i,\ a_i\}$, $C_{3i+2} = \{v_i,\ b_i\}$.

An *Extended Resolution proof* $\mathscr{P}$ (for short: *ER proof*) of $F_2 \in \mathscr{CLS}$ from $F_1 \in \mathscr{CLS}$ is a pair $\mathscr{P} = (F_1',\ (C_1, \ldots, C_n))$ such that F_1' is a normal extension for F_1, and $(C_1, \ldots, C_n)$ is a resolution proof of F_2 from $F_1 \cup F_1'$. The *length* of $\mathscr{P}$ is n. An *ER proof for* $F \in \mathscr{CLS}$ is an ER proof of $\{\bot\}$ from F. For $F \in \mathscr{CLS}$ we define

$$\mathrm{Comp}_{\mathrm{ER}}(F) := \inf\{n\colon \exists \mathscr{P}\ \textit{ER} \text{ proof for } F \text{ of length } n\}.$$

Additionally we define for $F_1,\ F_2 \in \mathscr{CLS}$:

$$\mathrm{Comp}_{\mathrm{ER}}(F_1,\ F_2) := \inf\{n\colon \exists F_2^* \cong F_2\ \exists \mathscr{P}\ \textit{ER} \text{ proof of } F_2^* \text{ from } F_1 \text{ with length } n\}.$$

Remark 1. The restrictions of signs in the new clauses is due to [21]. Every other distribution of signs has the same effect (e.g. we could have introduced the new clauses $\{\overline{v},\ a,\ b\}, \{v,\ \overline{a}\},\ \{v,\ \overline{b}\}$, which are the CNF of $v \leftrightarrow a \vee b$).

2. Every normal extension is a blocked extension, every ER proof is a GER proof, but not vice versa.

3. If F' is a normal extension for F, then F' is also a normal extension for every F^+ with $\mathrm{Var}(F^+) \subseteq \mathrm{Var}(F)$. This is not the case for blocked extension, if they contain blocked clauses which are blocked only for literals whose variables are already in F: These blocked clauses depend on the special shape of F.

4. "$\mathrm{Comp}_{\mathrm{ER}}(F_1, F_2)$" is introduced for the purpose of simulation: F_2 shall contain (in renamed form) F_1 together with the blocked extension F_1' (which shall be simulated).

5. $\mathrm{Comp}_{\mathrm{ER}}(F) = \mathrm{Comp}_{\mathrm{ER}}(F, \{\bot\})$.

Lemma 7.1. *For every $F \in \mathscr{CLS}$ and every blocked extension F' for F we have*

$$\mathrm{Comp}_{\mathrm{ER}}(F, F \cup F') \leqslant \mathrm{O}(\ell(F \cup F')^5),$$

where $\ell(F) := \sum_{C \in F} |C|$.

Corollary 7.2. *For $F \in \mathscr{CLS}$ we have* $\mathrm{Comp}_{\mathrm{ER}}(F) \leqslant \mathrm{O}(\mathrm{Comp}_{\mathrm{GER}}(F)^{10})$.

Proof. By Lemma 7.1 and point 3 of Remark that follows Definition 3.1. □

Proof of Lemma 7.1. We proceed by showing in I–IV how to handle F''s of increasing generality. In order to increase the strength of the induction hypothesis in fact we derive *exactly* $F \cup F'$ (i.e., for condition 1 of Definition 3.1 here we have "$=$" instead of "$\subseteq$").

I. $F' = \{\{\overline{v}, \overline{a}, \overline{b}\}, \{v, a\}, \{v, b\}\}$, $v \in \mathscr{VA} \setminus (\mathrm{Var}(F) \cup \mathrm{Var}(\{a, b\}))$, $a, b \in \mathscr{LIT}$. (If $a = b$ holds, then actually we have $F' = \{\{\overline{v}, \overline{a}\}, \{v, a\}\}$.) Here we have $\mathrm{Comp}_{\mathrm{ER}}(F, F') \leqslant \mathrm{O}(1)$.

Proof. *Case A*: $a \neq b$

A.1: $a,\ b \in \mathscr{VA}$. $\surd$

A.2: $a \in \mathscr{VA}$, $b \in \overline{\mathscr{VA}}$.

Introduce

$$\{\overline{w},\ \overline{a},\ b\}, \quad \{w, a\}, \quad \{w, \overline{b}\},$$
$$\{\overline{v},\ \overline{w},\ \overline{a}\}, \quad \{v, w\}, \quad \{\boldsymbol{v}, \boldsymbol{a}\}.$$

Infer

$$\{\overline{v}, \overline{w}, \overline{a}\},\ \{w, \overline{b}\} \vdash \{\boldsymbol{\overline{v}}, \boldsymbol{\overline{a}}, \boldsymbol{\overline{b}}\},$$

$$\{\overline{w}, \overline{a}, b\},\ \{v, a\} \vdash \{v, \overline{w}, b\};\ \{v, \overline{w}, b\},\ \{v, w\} \vdash \{\boldsymbol{v}, \boldsymbol{b}\}.\ \surd$$

A.3: $a \in \overline{\mathscr{VA}}$, $b \in \mathscr{VA}$. Use A.2. $\surd$

A.4: $a, b \in \overline{\mathscr{VA}}$.

Introduce (by A.2)

$$\{\overline{w}, a, \overline{b}\}, \quad \{w, \overline{a}\}, \quad \{w, b\},$$
$$\{\overline{v}, \overline{w}, \overline{b}\}, \quad \{v, w\}, \quad \{\boldsymbol{v}, \boldsymbol{b}\}.$$

Infer

$$\{\overline{v},\overline{w},\overline{b}\},\ \{w,\overline{a}\} \vdash \{\overline{\boldsymbol{v}},\overline{\boldsymbol{a}},\overline{\boldsymbol{b}}\},$$
$$\{\overline{w},a,\overline{b}\},\ \{v,w\} \vdash \{v,a,\overline{b}\};\ \{v,a,\overline{b}\},\{v,b\} \vdash \{\boldsymbol{v},\boldsymbol{a}\}.\ \surd$$

Case B: $a = b$

Introduce (by A)

$$\{\overline{w},a,\overline{x}\},\quad \{w,\overline{a}\},\quad \{w,x\},$$
$$\{\overline{v},\overline{w},\overline{a}\},\quad \{v,w\},\quad \{\boldsymbol{v},\boldsymbol{a}\}.$$

Infer

$$\{\overline{v},\overline{w},\overline{a}\},\quad \{w,\overline{a}\} \vdash \{\overline{\boldsymbol{v}},\overline{\boldsymbol{a}}\}.\ \surd$$

II. $F' = \{\{\overline{v},\overline{a_1},\ldots,\overline{a_n}\},\{v,a_1\},\ldots,\{v,a_n\}\}$, $n \geqslant 0$, $v \in \mathscr{V}\mathscr{A} \setminus (\mathrm{Var}(F) \cup \mathrm{Var}(\{a_1,\ldots,a_n\}))$, $a_1,\ldots,a_n \in \mathscr{L}\mathscr{I}\mathscr{T}$. Here we have $\mathrm{Comp}_{\mathrm{ER}}(F,F') \leqslant \mathrm{O}(n^2)$.

Proof (By induction)**.** $n = 0$

Introduce (by I)

$$\{\overline{w},\overline{x}\},\quad \{w,x\},$$
$$\{\overline{u},\overline{w},\overline{x}\},\quad \{u,w\},\quad \{u,x\},$$
$$\{\overline{v},\overline{u}\},\quad \{v,u\}.$$

Infer

$$\{\overline{w},\overline{x}\},\{u,x\} \vdash \{\overline{w},u\};\{\overline{w},u\},\{w,u\} \vdash \{u\};\{u\},\{\overline{v},\overline{u}\} \vdash \{\overline{\boldsymbol{v}}\}.\ \surd$$

$n \geqslant 1$:

Introduce (by induction hypothesis)

$$\{\overline{w},\overline{a_1},\ldots,\overline{a_{n-1}}\},\{w,a_1\},\ldots,\{w,a_{n-1}\}$$
$$\{\overline{v},w,\overline{a_n}\},\{v,\overline{w}\},\{\boldsymbol{v},\boldsymbol{a}_n\}$$

Infer

$$\{\overline{w},\overline{a_1},\ldots,\overline{a_{n-1}}\},\{\overline{v},w,\overline{a_n}\} \vdash \{\overline{\boldsymbol{v}},\overline{\boldsymbol{a}_1},\ldots,\overline{\boldsymbol{a}_n}\};$$
$$\text{for } i \in \{1,\ldots,n-1\}:\ \{v,\overline{w}\},\{w,a_i\} \vdash \{\boldsymbol{v},\boldsymbol{a}_i\}.\ \surd$$

III. $F' = \{C_0\}$, C_0 blocked w.r.t. F. Here we have $\mathrm{Comp}_{\mathrm{ER}}(F, F \cup F') \leqslant \mathrm{O}(\ell(F)^3 + |C_0|^2)$.

Proof. Let $C_0 = \{x,x_1,\ldots,x_m\}$ ($|C_0| = m+1, m \geqslant 0$), C_0 blocked for x w.r.t. F. Define

$$p := |\{C \in F: x \in C\}|,$$
$$n := |\{C \in F: \overline{x} \in C\}|.$$

Let $C_1,\ldots,C_p$ be the x-occurrences in F ($\{C_1,\ldots,C_p\} = \{C \in F: x \in C\}$), and let $D_1,\ldots,D_n$ be the $\overline{x}$-occurrences in F ($\{D_1,\ldots,D_n\} = \{C \in F: \overline{x} \in C\}$). Furthermore we use

$$C_i \setminus \{x\} = \{c_{i,j}\}_{j \in \{1,\ldots,|C_i|-1\}} \quad (i \in \{1,\ldots,p\}),$$
$$D_i \setminus \{\overline{x}\} = \{d_{i,j}\}_{j \in \{1,\ldots,|D_i|-1\}} \quad (i \in \{1,\ldots,n\}).$$

Case A: $n = 0$

By II introduce

$$\{\overline{v_i}, c_{i,1}, \dots, c_{i,|C_i|-1}\}, \quad \{v_i, \overline{c_{i,1}}\}, \dots, \quad \{v_i, \overline{c_{i,|C_i|-1}}\} \quad \text{for } i \in \{1, \dots, p\};$$
$$\{\overline{v_{p+1}}, x_1, \dots, x_m\}, \quad \{v_{p+1}, \overline{x_1}\}, \dots, \quad \{v_{p+1}, \overline{x_m}\};$$
$$\{\overline{v}, \overline{v_1}, \dots, \overline{v_{p+1}}\}, \quad \{v, v_1\}, \dots, \quad \{v, v_{p+1}\}.$$

Infer for $i \in \{1, \dots, p\}$

$$\{\overline{v_i}, c_{i,1}, \dots, c_{i,|C_i|-1}\}, \{v, v_i\} \vdash \{\boldsymbol{v}, \boldsymbol{c}_{i,1}, \dots, \boldsymbol{c}_{i,|C_i|-1}\}.$$

And

$$\{\overline{v_{p+1}}, x_1, \dots, x_m\}, \{v, v_{p+1}\} \vdash \{\boldsymbol{v}, \boldsymbol{x}_1, \dots, \boldsymbol{x}_m\}.$$

Thus we inferred $\langle x \leftrightarrow v\rangle$ $(\{C_1, \dots, C_p, C_0\})$, where $\langle x \leftrightarrow v\rangle$ denotes the renaming $\sigma : \{v, x\} \to \{v, x\}$ with $\sigma(v) = x$ and $\sigma(x) = v$.$\surd$

Case B: $n \neq 0$

Step (*a*): For $i \in \{1, \dots, n\}$ introduce

$$\{\overline{v_i}, d_{i,1}, \dots, d_{i,|D_i|-1}\}, \{v_i, \overline{d_{i,1}}\}, \dots, \{v_i, \overline{d_{i,|D_i|-1}}\}.$$

And infer for $i \in \{1, \dots, n\}$:

$$\{\overline{x}, d_{i,1}, \dots, d_{i,|D_i|-1}\}, \{v_i, \overline{d_{i,1}}\}, \dots, \{v_i, \overline{d_{i,|D_i|-1}}\} \vdash \{\overline{x}, v_i\}.$$

Step (*b*): Because C_0 is blocked for x w.r.t. F, for every $i \in \{1, \dots, n\}$ there is $z_i \in \{x_1, \dots, x_m\}$ such that $\overline{z_i} \in \{d_{i,1}, \dots, d_{i,|D_i|-1}\}$ holds. Introduce

$$\{\overline{v}, \overline{v_1}, \dots, \overline{v_n}\}, \{v, v_1\}, \dots, \{v, v_n\}.$$

Infer

$$\{\overline{v}, \overline{v_1}, \dots, \overline{v_n}\}, \{v_1, z_1\}, \dots, \{v_n, z_n\} \vdash \{\overline{v}, z_1, \dots, z_n\}.$$

And for $i \in \{1, \dots, n\}$

$$\{\overline{v_i}, d_{i,1}, \dots, d_{i,|D_i|-1}\}, \{v, v_i\} \vdash \{\boldsymbol{v}, \boldsymbol{d}_{i,1}, \dots, \boldsymbol{d}_{i,|D_i|-1}\}.$$

Step (*c*): Introduce $\{\overline{w}, \overline{v}, x_1, \dots, x_m\}, \{w, v\}, \{w, \overline{x_1}\}, \dots, \{w, \overline{x_m}\}$. Infer (remember: $\{z_1, \dots, z_n\} \subseteq \{x_1, \dots, x_m\}$):

$$\{\overline{v}, z_1, \dots, z_n\}, \{w, \overline{z_1}\}, \dots, \{w, \overline{z_n}\} \vdash \{\overline{v}, w\}.$$

And

$$\{\overline{v}, w\}, \{\overline{w}, \overline{v}, x_1, \dots, x_m\} \vdash \{\overline{\boldsymbol{v}}, \boldsymbol{x}_1, \dots, \boldsymbol{x}_m\}.$$

Step (*d*): Infer (using (a) and (b))

$$\{\overline{v}, \overline{v_1}, \dots, \overline{v_n}\}, \{\overline{x}, v_1\}, \dots, \{\overline{x}, v_n\} \vdash \{\overline{v}, \overline{x}\}.$$

And for $i \in \{1, \dots, p\}$:

$$\{\overline{v}, \overline{x}\}, \{x, c_{i,1}, \dots, c_{i,|C_i|-1}\} \vdash \{\overline{\boldsymbol{v}}, \boldsymbol{c}_{i,1}, \dots, \boldsymbol{c}_{i,|C_i|-1}\}.$$

Altogether we inferred $\langle x \leftrightarrow \overline{v}\rangle(\{C_1, \dots, C_p, D_1, \dots, D_n, C_0\})$. $\surd$

IV. $F' = \{C\}, \{C\}$ blocked extension for F. Here we have $\mathrm{Comp}_{\mathrm{ER}}(F, F \cup F') \leqslant \mathrm{O}(\ell(F)^4 + |C|^2)$.

Proof. Let $C_1, \ldots, C_m$ be a maximal blocking sequence for $F \cup \{C\}$. There is $i \in \{1, \ldots, m\}$ with $C_i = C$. Now by III add subsequently C_j to $(F \setminus \{C_m, \ldots, C_{i+1}\}) \cup \{C_i, C_{i-1}, \ldots, C_{j+1}\}$ for $j = i, i-1, \ldots, 1$. $\surd$

Finally Lemma 7.1 is an immediate consequence of IV.

8. 1- and 2-clauses in Blocked Extensions

Lemma 8.1. *Assume that F' is a blocked extension for F with $\{l\} \in F'$ such that l is pure for F. Then also $\langle l \to 1 \rangle * F'$ is a blocked extension for F, and we have*

$$\mathrm{Comp}_{\mathrm{R}}(F \cup (\langle l \to 1 \rangle * F')) \leqslant \mathrm{Comp}_{\mathrm{R}}(F \cup F').$$

Proof.

$$\begin{aligned}
\mathrm{Comp}_{\mathrm{R}}(F \cup F') &\geqslant \mathrm{Comp}_{\mathrm{R}}(\langle l \to 1 \rangle * (F \cup F')) \quad \text{(by Lemma 3.1 part 2)} \\
&= \mathrm{Comp}_{\mathrm{R}}(\langle l \to 1 \rangle * F \cup \langle l \to 1 \rangle * F') \\
&= \mathrm{Comp}_{\mathrm{R}}(\langle l \to 1 \rangle * F \cup \{C \in F:\ \in C\} \cup \langle l \to 1 \rangle * F') \\
&\qquad \text{(by Lemma 3.1 part 3(a))} \\
&= \mathrm{Comp}_{\mathrm{R}}(F \cup \langle l \to 1 \rangle * F') \quad \text{(since } l \text{ is pure for } F\text{)}.
\end{aligned}$$

To show that $\langle l \to 1 \rangle * F'$ is a blocked extension for F, consider a maximal blocking sequence $C_1, \ldots, C_n$ for $F \cup F'$ with blocking literals $l_1, \ldots, l_n$.

Because of $\{l\} \in F'$ we have

$$\forall i \in \{1, \ldots, n\}:\ l_i \neq \bar{l}. \tag{‡}$$

(Before $\{l\}$ is eliminated, no clause can be blocked for $\bar{l}$, and for $\{l\}$ being blocked l must be pure.)

Define $I := \{i \in \{1, \ldots, n\}:\ l \notin C_i\}$, and for $i \in I$: $C_i' := \langle l \to 1 \rangle * C_i$. Now $(C_i')_{i \in I}$ is a blocking sequence for $\langle l \to 1 \rangle * (F \cup F')$ because of (‡) (with blocking literals l_i). By definition $\langle l \to 1 \rangle * F' \subseteq \{C_i'\}_{i \in I}$ holds, and therefore $\langle l \to 1 \rangle * F'$ is a blocked extension for $\langle l \to 1 \rangle * F$, and thus also for F (clauses with pure literals are blocked).

Corollary 8.2. 1. $\mathrm{Comp}_{(1,\infty)} = \mathrm{Comp}_{(1,0)}$.

2. *For $F \in \mathscr{CLS}$ with $K(F) = F (\Leftrightarrow B^{\mathrm{in}}(F) = \emptyset)$ we only have to consider blocked extension without any 1-clause*:

$$\mathrm{Comp}_{(a,b)}(F) = \min\{\mathrm{Comp}_{\mathrm{R}}(F \cup F'):\ F' \text{ blocked } (a,b)\text{-extension for } F \text{ and } \forall C \in F'[|C| \geqslant 2]\}.$$

Proof. For part 2 note that if $\{l\}$ is blocked for F, then l must be pure for F. □

The assumption $K(F)=F$ is fulfilled for example, if F is minimally unsatisfiable. But for F with $K(F)\neq F$ in general even $(1,0)$-resolution proofs can cause exponential speed ups (compared to $(0,0)$-resolution). To prove this we need the following lemma:

Lemma 8.3 (Cook [2] or Cook and Reckhow [6]). $\mathrm{Comp}_{\mathrm{ER}}(\mathrm{PHP})\leqslant \mathrm{O}(n^4)$.

Proof. We give the proof in some detail because later we can make use of it. The idea is simple:

Suppose an assignment for the variables $v_{i,j}$ ($i\in\{1,\dots,n\}$, $j\in\{1,\dots,n-1\}$) is given fulfilling PHP. We want to derive a fulfilling assignment for PHP_{n-1} from this. Therefore, we introduce new variables $v^1_{i,j}$ for $i\in\{1,\dots,n-1\}$, $j\in\{1,\dots,n-2\}$ and "project" the assignment from domain $(v_{i,j})$ to domain $(v^1_{i,j})$ by

$$v^1_{i,j}\leftrightarrow v_{i,j}\vee(v_{n,j}\wedge v_{i,n-1}) \qquad (\S)$$

(critical are only the pigeons in the nth row or the $(n-1)$th column: either there is a pigeon at $(n,n-1)$, and then we simply forget this pigeon, or there are uniquely determined pigeons at (n,j) and $(i,n-1)$ for $i\in\{1,\dots,n-1\}$, $j\in\{1,\dots,n-2\}$, and in this case these two pigeons are collapsed into a new one at (i,j)).

In this way one gets a fulfilling assignment for PHP_{n-1} (but with variables $(v^1_{i,j})$ instead of $(v_{i,j})$), and by iteration of this process we eventually reach a contradiction. This idea is put into work as follows:

Consider new variables $v^r_{i,j}$, $x^r_{i,j}$ for $r\in\{1,\dots,n-2\}$, $i\in\{1,\dots,n-r\}$, $j\in\{1,\dots,n-r-1\}$. Define

$$\begin{aligned}
&C^r_{i,j}(1):=\{x^r_{i,j},\overline{v^{r-1}_{n-r+1,j}},\overline{v^{r-1}_{i,n-r}}\},\\
&C^r_{i,j}(2):=\{\overline{x^r_{i,j}},v^{r-1}_{i,n-r}\},\ C^r_{i,j}(3):=\{\overline{x^r_{i,j}},v^{r-1}_{n-r+1,j}\};\\
&C^r_{i,j}(4):=\{\overline{v^r_{i,j}},v^{r-1}_{i,j},\ x^r_{i,j}\},\\
&C^r_{i,j}(5):=\{v^r_{i,j},\overline{v^{r-1}_{i,j}}\},\ C^r_{i,j}(6):=\{v^r_{i,j},\ \overline{x^r_{i,j}}\}
\end{aligned}$$

where $v^0_{i,j}:=v_{i,j}$. And

$$E_n:=\bigcup_{r=1}^{n-2}E^r_n,$$
$$E^r_n:=\{C^r_{i,j}(1),\dots,C^r_{i,j}(6)\}_{\substack{i\in\{1,\dots,n-r\}\\ j\in\{1,\dots,n-r-1\}}}.$$

E_n is a normal extension for PHP (the order of introduction is: levels $E^1_n,\dots,E^{n-2}_n$; within the levels the order of the 6-clause blocks is arbitrary, but within the blocks choose numerical order), and we want to show $\mathrm{Comp}_{\mathrm{R}}(\mathrm{PHP}\cup E_n)\leqslant \mathrm{O}(n^4)$.

To that end first derive for $r\in\{1,\dots,n-2\}$, $i\in\{1,\dots,n-r\}$, $j\in\{1,\dots,n-r-1\}$ the following four clauses from the corresponding 6-clause block in E^r_n (using three resolution steps for each triple (r,i,j)):

$$D^r_{i,j}(1):=\{\overline{v^r_{i,j}},v^{r-1}_{i,j},v^{r-1}_{n-r+1,j}\},\qquad D^r_{i,j}(2):=\{\overline{v^r_{i,j}},v^{r-1}_{i,j},v^{r-1}_{i,n-r}\}$$

$$D^r_{i,j}(3) := \{v^r_{i,j}, \overline{v^{r-1}_{n-r+1,j}}, \overline{v^{r-1}_{i,n-r}}\}, \qquad D^r_{i,j}(4) := \{v^r_{i,j}, \overline{v^{r-1}_{i,j}}\}$$

(these four clauses are a CNF of the generalization of (§)).

Because of the completeness of resolution we have for $r \in \{1, \ldots, n-2\}$:

$$\sigma_{r-1}(\mathrm{PHP}_{n-r+1}) \cup E^{r'}_n \vdash \sigma_r(\mathrm{PHP}_{n-r}), \qquad (\|)$$

where

$$E^{r'}_n := \{D^r_{i,j}(1), \ldots, D^r_{i,j}(4)\}_{\substack{i \in \{1,\ldots,n-r\} \\ j \in \{1,\ldots,n-r-1\}}}$$

and the renamings σ_r are defined by

$$\sigma_r := \langle v_{i,j} \leftrightarrow v^r_{i,j} \rangle_{\substack{i \in \{1,\ldots,n-r\} \\ j \in \{1,\ldots,n-r-1\}}}$$

(σ_0 is the identity).

Fortunately in ($\|$) each single resolution proof of a "long" (positive) clause is of length $\mathrm{O}(n)$, and of a "short" (negative) clause is of length $\mathrm{O}(1)$, and thus altogether we obtain

$$\mathrm{Comp}_{\mathrm{R}}(\mathrm{PHP} \cup E_n) \leqslant \mathrm{O}(1)\mathrm{O}(n^3) + (\mathrm{O}(n)\mathrm{O}(n) + \mathrm{O}(1)\mathrm{O}(n^3))\mathrm{O}(n) = \mathrm{O}(n^4). \qquad \square$$

Lemma 8.4. *There is a sequence* $(F_n)_{n \in \mathbb{N}}$ *of (unsatisfiable) clause-sets* ($\lim_{n\to\infty} |(F_n)| = \infty$) *with* $\mathrm{Comp}_{(1,0)}(F_n) \leqslant \mathrm{O}(n^4)$, *but* $\mathrm{Comp}_{\mathrm{R}}(F_n) = \mathrm{Comp}_{\mathrm{R}}(\mathrm{PHP}) \geqslant c^n$ (*c as defined in Theorem* 1).

Proof. Consider E_n from the proof of the previous lemma. Choose a new variable $v \in \mathcal{VA} \setminus \mathrm{Var}(\mathrm{PHP} \cup E_n)$. Define

$$E'_n := \{C \cup \{v\}: C \in E_n\},$$
$$F_n := \mathrm{PHP} \cup E'_n.$$

With the help of Lemma 3.1 part 3(a) we get

$$\mathrm{Comp}_{\mathrm{R}}(F_n) = \mathrm{Comp}_{\mathrm{R}}(\mathrm{PHP}).$$

But $\{\{\overline{v}\}\}$ is a blocked extension for F_n, and thus

$$\mathrm{Comp}_{(1,0)}(F_n) \leqslant \mathrm{O}(n^4).$$

(We used the simple fact, that if F' is a blocked extension for F and $v \in \mathcal{VA} \setminus \mathrm{Var}(F \cup F')$, then also $\{C \cup \{v\}: C \in F'\} \cup \{\{\overline{v}\}\}$ is a blocked extension for F.) $\square$

8.1. 2-clauses in blocked extensions

Lemma 8.5. *Assume that $\{C_0\}$ is a blocked extension for $F \in \mathcal{CLS}$ and that $C_1, \ldots, C_n$ is a maximal blocking sequence for $F \cup \{C_0\}$ with blocking literals $l_1, \ldots, l_n$. There is $i \in \{1, \ldots, n\}$ with $C_i = C_0$. If now there is $x \in C_0 \setminus \{l_i\}$ with $\forall C \in F$: $\{\overline{l_i}, \overline{x}\} \nsubseteq C$, then also $\{C_0 \setminus \{x\}\}$ is a blocked extension for F.*

Proof. $C_1,\dots,C_{i-1}$, $C_0\backslash\{x\}$ is a blocking sequence for $F\cup\{C_0\backslash\{x\}\}$ with blocking literals $l_1,\dots,l_i$ (the assumption $\{\overline{l_i},\overline{x}\}\not\subseteq C$ for $C\in F$ ensures that no original "blockade" has been destroyed). □

Corollary 8.6. *If $\{\{a,b\}\}$ is a blocked extension for $F\in\mathscr{CLS}$ such that*

$$\forall C\in F\colon\ \{\overline{a},\overline{b}\}\not\subseteq C$$

holds, then at least one of $\{\{a\}\}$ or $\{\{b\}\}$ is also a blocked extension for F.

Corollary 8.7. *Assume a blocked extension F' for $F\in\mathscr{CLS}$ with $\{a,b\}\in F'$ such that $\forall C\in F\cup F'$: $\{\overline{a},\overline{b}\}\not\subseteq C$ holds. Then for $x=a$ or $x=b$ also $F'':=(F'\backslash\{a,b\})\cup\{x\}$ is a blocked extension for F with* $\mathrm{Comp}_{\mathrm{R}}(F\cup F'')\leqslant\mathrm{Comp}_{\mathrm{R}}(F\cup F')$.

Proof. Use Lemmas 8.6 and 6.4 and part 1 of Lemma 3.1. □

Hence we can restrict (a,b)-resolution proofs for $F\in\mathscr{CLS}$ without affecting $\mathrm{Comp}_{(a,b)}(F)$ to such blocked extensions F' which fulfill:
1. $\overline{\mathrm{Lit}(F')}\subseteq\mathrm{Lit}(F\cup F')$,
2. $\{l\}\in F'\Rightarrow\bar{l}\in\mathrm{Lit}(F)$,
3. $\{a,b\}\in F'\Rightarrow\exists C\in F\cup F'\colon\ \{\overline{a},\overline{b}\}\subseteq C$.

Now consider a $(2,\infty)$-extension F' for F fulfilling restrictions 1–3. Suppose $\{l,x\}\in F'$ with $\mathrm{Var}(l)\notin\mathrm{Var}(F)$. Then by 3 also $\{\bar{l},\overline{x}\}\in F'$ holds, and furthermore (also by 3 and by $F'\in\mathscr{SAT}$): $\{l,\overline{x}\},\{\bar{l},x\}\notin F'$. The next lemma shows how to eliminate the new variable $\mathrm{Var}(l)$ in this situation.

Lemma 8.8. *Assume $F\in\mathscr{CLS}$, $n\geqslant 1$, $C_i\in\mathscr{CL}$, $C_i=\{l,x_i\}$ for $i\in\{1,\dots,n\}$,* $\mathrm{Var}(l)\notin\mathrm{Var}(F)$, $\mathrm{Var}(x_i)\neq\mathrm{Var}(x_j)$ *for $i\neq j$. Assume that $\{C_1,\dots,C_n,\overline{C_1},\dots,\overline{C_n}\}$ is a blocked extension for F. Then also*

$$\langle l\leftrightarrow\overline{x_1}\rangle(\{C_1,\dots,C_n,\overline{C_1},\dots,\overline{C_n}\})=\{\{\overline{x_1},x_2\},\dots,\{\overline{x_1},x_n\},\{x_1,\overline{x_2}\},\dots,\{x_1,\overline{x_n}\}\}$$

is a blocked extension for F.

Proof. Consider a maximal blocking sequence $D_1,\dots,D_m$ for $F\cup\{C_1,\dots,C_n,\overline{C_1},\dots,\overline{C_n}\}$ with blocking literals $l_1,\dots,l_m$. We use $\sigma:=\langle l\leftrightarrow\overline{x_1}\rangle$.

There are $p,q\in\{1,\dots,m\}$ with $D_p=\{l,x_1\}$, $D_q=\{\bar{l},\overline{x_1}\}$. Define $I:=\{1,\dots,m\}\backslash\{p,q\}$. In the following we prove that $\sigma(D_i)$, $i\in I$ is a blocking sequence for

$$F\cup\sigma(\{C_1,\dots,C_n,\overline{C_1},\dots,\overline{C_n}\})=F\cup\{\sigma(C_2),\dots,\sigma(C_m),\sigma(\overline{C_2}),\dots,\sigma(\overline{C_m})\}$$

with blocking literals $\sigma(l_i)$ (and thus the assertion follows).

Consider $i\in I$. We have to show that $\sigma(D_i)$ is blocked for $\sigma(l_i)$ w.r.t.

$$(F\cup\{\sigma(C_2),\dots,\sigma(C_m),\sigma(\overline{C_2}),\dots,\sigma(\overline{C_m})\})\backslash\{\sigma(D_j)\}_{j\in I,\,j<i}.$$

I. $D_i \in F (\Rightarrow \sigma(D_i) = D_i)$: We have two critical cases: $l_i = x_1$ or $l_i = \overline{x_1}$. W.l.o.g. $l_i = x_1$. Because of $\mathrm{Var}(l) \notin \mathrm{Var}(F)$ we have $q \leqslant i-1$. Thus $l_q = \bar{l}$ holds, and hence $\{C_2, \dots, C_n\} \subseteq \{D_1, \dots, D_{q-1}\}$, yielding $\{\sigma(C_2), \dots, \sigma(C_n)\} \subseteq \{\sigma(D_1), \dots, \sigma(D_{q-1})\}$. Therefore, all additional $\overline{x_1}$-clauses have been deleted before step i. $\surd$

II. $D_i = C_j$ for one $j \in \{2, \dots, n\}$: Here only $l_i = l$ is critical ($\sigma(l_i) = \overline{x_1}$).

Now $\{\overline{C_2}, \dots, \overline{C_n}\} \setminus \{\overline{C_j}\} \subseteq \{D_1, \dots, D_{i-1}\}$ holds, and hence, as in I, it follows that all additional x_1-clauses have been eliminated before step i.

Furthermore we have $q \leqslant i-1$, and thus here $l_q = \overline{x_1}$ must hold. Hence $\overline{x_1}$ is pure for $F \setminus \{D_1, \dots, D_{i-1}\} = F \setminus \sigma(\{D_1, \dots, D_{i-1}\})$. $\surd$

III. $D_i = \overline{C_j}$ for one $j \in \{2, \dots, n\}$: Analogous to II. □

Lemma 8.9. $\mathrm{Comp}_{(2,\infty)} = \mathrm{Comp}_{(2,0)}$.

Proof. By the above argumentation and by Lemma 3.4. □

We conclude this section by showing that $(2,0)$-resolution cannot be bounded polynomially by $(1,0)$-resolution:

Lemma 8.10. *There is a sequence* $(F_n)_{n \in \mathbb{N}}$ *of* (*unsatisfiable*) *clause-sets* ($\lim_{n \to \infty} |F_n| = \infty$) *with* $\mathrm{Comp}_{(2,0)}(F_n) \leqslant O(n^4)$, *but* $\mathrm{Comp}_{(1,0)}(F_n) = \mathrm{Comp}_{\mathrm{R}}(\mathrm{PHP}) \geqslant c^n$.

Proof. Consider E_n from the proof of Lemma 8.3 and define

$$F_n := \mathrm{PHP} \cup \{C \in E_n^1 : |C| = 3\} \cup \bigcup_{r=2}^{n-2} E_n^r.$$

Now $\{C \in E_n^1 : |C| = 2\}$ is a $(2,0)$-extension for F_n, and thus we can estimate

$$\mathrm{Comp}_{(2,0)}(F_n) \leqslant \mathrm{Comp}_{\mathrm{R}}(\mathrm{PHP} \cup E_n) \leqslant O(n^4).$$

On the other hand consider a $(1,0)$-extension F' for F_n. Since the clauses

$$C_{i,j}^1 = \{x_{i,j}^1, \overline{v_{n,j}}, \overline{v_{i,n+1}}\}$$

are not blocked for $\overline{v_{n,j}}$ or $\overline{v_{i,n+1}}$ w.r.t. PHP, and $\mathrm{Var}(x_{i,j}^1) \notin \mathrm{Var}(\bigcup_{r=2}^{n-2} E_n^r)$, the literals $x_{i,j}^1$ are pure in $F_n \cup F'$. Thus, using Lemma 3.1, part 3(a), we get

$$\mathrm{Comp}_{\mathrm{R}}(F_n \cup F') = \mathrm{Comp}_{\mathrm{R}}\left(\mathrm{PHP} \cup \bigcup_{r=2}^{n-2} E_n^r \cup F'\right) = \mathrm{Comp}_{\mathrm{R}}(\mathrm{PHP}). \qquad \square$$

9. An exponential lower bound for GER without new variables

In order to obtain a lower bound for $\mathrm{Comp}_{(\infty,0)}$ we have to strengthen Theorem 1 by adding such clauses to PHP which are fulfilled by every “critical assignment”:

Definition 9.1. Let

$$V_n := \{v_{i,j}\}_{\substack{i\in\{1,\ldots,n\}\\ j\in\{1,\ldots,n-1\}}}$$

denote the set of variables of PHP (see Section 3).

An *i-critical assignment* ($i \in \{1,\ldots,n\}$) for the pigeonhole formula PHP is a partial assignment φ with variables $\mathrm{Var}(\varphi) = V_n$ such that for $L_i := \{v_{i,j}\}_{j\in\{1,\ldots,n-1\}} \in \mathrm{PHP}_n^p$ we have

$$\varphi(L_i) = 0 \quad \text{and} \quad \varphi(\mathrm{PHP}\setminus\{L_i\}) = 1.$$

φ is simply called a *critical assignment* if φ is i-critical for any i. By $\mathscr{CASS}_n$ we denote the set of all critical assignments for PHP.

Critical assignments correspond to the different ways of distributing $n-1$ pigeons on $n-1$ holes such that no hole contains two pigeons.

Theorem 2. *Let E_n be the set of clauses $C \in \mathscr{CL}$ with $\mathrm{Var}(C) \subseteq V_n$ such that for all $\varphi \in \mathscr{CASS}_n$ we have $\varphi(C) = 1$.*

Then for $c := 2^{1/20}$ and $n \geqslant 1676$ the lower bound $\mathrm{Comp}_{\mathrm{R}}(\mathrm{PHP} \cup E_n) \geqslant c^n$ holds. [6]

Proof. The point here is, that the lower bound of [8] for PHP (see also [3,20,1]) immediately can be generalized to $\mathrm{PHP} \cup E_n$, since the proof follows only the trace of "long clauses", and this by means of critical assignments, i.e., only clauses which are falsified by some critical assignment are interesting here.

For readers not familiar with that proof we give a survey in the following, using the improved version of [1].

9.1. Outline of proof

We use for abbreviation: $N := \{1,\ldots,n\}$, $N' := \{1,\ldots,n-1\}$. In this subsection we consider only clause-sets F, G and clauses C, D with variables from V_n.

The next notions reflect the concentration on critical assignments.

Definition 9.2. For clause-sets F and clauses C we define:

$$F \models^{c_n} C :\Leftrightarrow \ \forall \varphi \in \mathscr{CASS}_n : \varphi(F) = 1 \Rightarrow \varphi(C) = 1.$$

A sequence $(C_1,\ldots,C_k)$ of clauses is a $\models^{c_n}$-proof of $\bot$ from F iff $C_k = \bot$ holds, and for each $i \in \{1,\ldots,k\}$ either we have $C_i \in F$ or there is $G \subseteq \{C_1,\ldots,C_{i-1}\}$ with $|G| \leqslant 2$ and $G \models^{c_n} C_i$ (or both).

Finally by $\gamma_n(C)$ we denote the minimal number of clauses from PHP_n^p implying C with respect to $\models^{c_n}$:

$$\gamma_n(C) := \min\{|F| : F \subseteq \mathrm{PHP}_n^p \wedge F \models^{c_n} C\}. \qquad \square$$

[6] Note $\mathrm{PHP} \cup E_n = \mathrm{PHP}_n^p \cup E_n$.

Note that $E_n = \{C\colon \top \models^{c_n} C\}$, and thus $\models^{c_n}$-proofs can arbitrarily introduce clauses from E_n. Every resolution proof of $\perp$ from PHP is a $\models^{c_n}$-proof of $\perp$ from PHP_n^p.

Lemma 9.1. *Consider a $\models^{c_n}$-proof $(C_1,\dots,C_k)$ of $\perp$ from PHP_n^p, where all C_i are positive (i.e., $C_i \subseteq V_n$). Then there is $r\in\{1,\dots,k\}$ with $|C_r|\geqslant 2/9n^2$.*

Proof. Consider $r\in\{1,\dots,k\}$ and $F\subseteq \mathrm{PHP}_n^p$ with $F\models^{c_n} C_r$, $|F|=\gamma(C_r)$. Let $I_r:=\{i\in\{1,\dots,n\}\colon L_i\in F\}$. Consider $i_1\in I_r$. Due to the minimality of F there is $\varphi_{i_1}\in\mathscr{CASS}_n$ such that φ_{i_1} is i_1-critical and $\varphi_{i_1}(C_r)=0$.

For $i_2\in N\backslash I_r$ consider the (uniquely determined) $k(i_1,i_2)\in N'$ with $\varphi_{i_1}(v_{i_2,k(i_1,i_2)})=1$. Note that for $i_2\neq i_2'$ we have $k(i_1,i_2)\neq k(i_1,i_2')$, and that $\varphi_{i_1}(v_{i_1,k(i_1,i_2)})=0$ holds.

Obtain $\varphi_{i_1}^{i_2}$ from φ_{i_1} by flipping the values for $v_{i_1,k(i_1,i_2)}$ and $v_{i_2,k(i_1,i_2)}$.

$\varphi_{i_1}^{i_2}$ is an i_2-critical assignment, thus $\varphi_{i_1}^{i_2}(F)=1$ holds, and we can infer $\varphi_{i_1}^{i_2}(C_r)=1$. Since C_r is a positive clause, we conclude $v_{i_1,k(i_1,i_2)}\in C_r$.

It follows that $|C_r|\geqslant|I_r||N\backslash I_r|=\gamma(C_r)(n-\gamma(C_r))$. Due to $\forall C\in\mathrm{PHP}_n^p:\gamma(C)=1$, $\gamma(\perp)=n$, and

$$G\models^{c_n} C\Rightarrow \gamma(C)\leqslant\sum_{D\in G}\gamma(D),$$

there is $r\in\{1,\dots,k\}$ with $1/3n\leqslant\gamma(C_r)\leqslant 2/3n$. Using elementary calculus we get $|C_r|\geqslant n/3(n-n/3)=2/9n^2$. $\square$

The restriction to positive clauses in the previous lemma is justified by the next definition (and lemma).

Definition 9.3. For a clause C we define C^+ as the outcome of the following procedure:

$C^+:=C$;
FOR $j\in N'$ DO
  IF $\exists^{=1} i\in N:\overline{v_{i,j}}\in C^+$ THEN
    $C^+:=(C^+\backslash\{\overline{v_{i,j}}\})\cup\{v_{i',j}\}_{i'\in N\backslash\{i\}}$
  ELSE IF $\exists^{\geqslant 2} i\in N:\overline{v_{i,j}}\in C^+$ THEN
    $C^+:=(C^+\backslash\{\overline{v_{i',j}}\}_{i'\in N})\cup\{v_{i',j}\}_{i'\in N}$
END FOR.

Lemma 9.2. *For a clause C and any critical assignment φ we have $\varphi(C)=\varphi(C^+)$. Thus, if $(C_1,\dots,C_k)$ is a $\models^{c_n}$-proof of $\perp$ from PHP_n^p, then also $(C_1^+,\dots,C_k^+)$.*

By the next definition (and lemma) we are enabled to reduce PHP_n^p to PHP_{n-1}^p.

Definition 9.4. For $i\in N$, $j\in N'$ let $\psi_{i,j}$ be the partial assignment with

$$\psi_{i,j}(v_{i,j})=1,\ \psi_{i,j}(v_{i,j'})=\psi_{i,j}(v_{i',j})=0$$

for $i'\in N\backslash\{i\}$, $j'\in N'\backslash\{j\}$, and undefined else.

Lemma 9.3. *There is a renaming* $\sigma_{i,j} : (V_n \backslash \mathrm{Var}(\psi_{i,j})) \to V_{n-1}$ *with*

$$\sigma_{i,j}(\psi_{i,j} * \mathrm{PHP}_n^p) = \mathrm{PHP}_{n-1}^p.$$

For $\varphi \in \mathscr{CASS}_{n-1}$ *we have* $(\varphi \circ \sigma_{i,j}) \cup \psi_{i,j} \in \mathscr{CASS}_n$ *and* $\varphi(\sigma_{i,j}(\psi_{i,j} * \{C\})) = ((\varphi \circ \sigma_{i,j}) \cup \psi_{i,j})(C)$ *for all clauses* C.

Thus in case of $G \models^{c_n} C$ *we either have* $\psi_{i,j}(C) = 1$ *or* $\sigma_{i,j}(\psi_{i,j} * G) \models^{c_{n-1}} \sigma_{i,j}(\psi_{i,j} * C)$.

Now we are ready to prove Theorem 2. Assume there is a resolution proof $\mathscr{P} = (C_1, \ldots, C_k)$ of $\perp$ from $\mathrm{PHP} \cup E_n$ with $k < c^n$. Consider $\mathscr{P}^+ = (C_1^+, \ldots, C_k^+)$.

By Lemma 9.2 the sequence $\mathscr{P}^+$ is a $\models^{c_n}$-proof of $\perp$ from PHP_n^p. Define the number of "large clauses" in $\mathscr{P}^+$ by

$$\#\mathrm{lc}(\mathscr{P}^+) := |\{i \in \{1, \ldots, k\} : |C| \geqslant q \cdot |V_n|\}|,$$

where $0 < q < 1$ is a parameter.

Trivially $\#\mathrm{lc}(\mathscr{P}^+) < c^n$. There must be a variable $v_{i,j}$ which appears in at least $q \cdot \#\mathrm{lc}(\mathscr{P}^+)$-many large clauses, and thus $\psi_{i,j}$ makes at least $q \cdot \#\mathrm{lc}(\mathscr{P}^+)$-many large clauses come true.

Obtain $\mathscr{P}^{+'}$ from $\mathscr{P}^+$ by deleting clauses from $\mathscr{P}^+$ which become true by $\psi_{i,j}$, and applying first $\psi_{i,j}$ and then $\sigma_{i,j}$ to the rest of the clauses (see Lemma 9.3).

By Lemma 9.3 $\mathscr{P}^{+'}$ is a $\models^{c_{n-1}}$-proof of $\perp$ from PHP_{n-1}^p. For the number of large clauses (which still refers to the size of V_n, and not of V_{n-1}) we know

$$\#\mathrm{lc}(\mathscr{P}^{+'}) \leqslant (1-q)\#\mathrm{lc}(\mathscr{P}^+) \leqslant (1-q)k.$$

By repeating this process at most $\lfloor 1 + \log_{(1-q)^{-1}} k \rfloor$ times we are sure that no large clause is left, and we obtain a $\models^{c_{n^*}}$-proof $\mathscr{P}^*$ of $\perp$ from $\mathrm{PHP}_{n^*}^p$ with

$$n^* \geqslant n - \lfloor 1 + \log_{(1-q)^{-1}} k \rfloor \geqslant n - \lfloor 1 + (1/20) n \log_{(1-q)^{-1}} 2 \rfloor.$$

Lemma 9.1 yields the existence of a clause of length at least $(2/9)n^{*2}$ in $\mathscr{P}^*$, but on the other hand, since all large clauses have been eliminated, every clause in $\mathscr{P}^*$ has length strictly less than $qn(n-1)$, which yields a contradiction for sufficiently large n, when we choose $q = 1/10$ for example. By using $q = 0.102283$ numerical calculations show that for all $n \geqslant 1676$ we have $2/9 n^{*2} \geqslant qn(n-1)$ (using $(*)$).

9.2. Applications to GER

Lemma 9.4. $\forall n \geqslant 1676 : \mathrm{Comp}_{(\infty,0)}(\mathrm{PHP}) \geqslant c^n$.

Proof. By Theorem 2 and Lemma 4.2. □

Lemma 9.5. *There is a sequence* $(F_n)_{n \in \mathbb{N}}$ *of (unsatisfiable) clause-sets* ($\lim_{n \to \infty} |F_n| = \infty$) *with* $\mathrm{Comp}_{(3,0)}(F_n) \leqslant \mathrm{O}(n^4)$, *but* $\mathrm{Comp}_{(2,0)}(F_n) \geqslant c^n$.

Proof. Consider E_n from the proof of Lemma 8.3 and define

$$G_n := \bigcup_{r=2}^{n-2} \{C \in E_n^r : |C| = 2\}$$

$$F_n := \mathrm{PHP} \cup G_n.$$

Now on the one hand $E_n \setminus G_n$ is a $(3,0)$-extension for F_n. And one the other hand, for a blocked extension F' for F_n the clause-set $F' \cup G_n$ is a blocked extension for PHP, and thus by Lemmas 9.4 and 8.9 we can conclude

$$\mathrm{Comp}_{(2,0)}(F_n) \geqslant \mathrm{Comp}_{(2,\infty)}(\mathrm{PHP}) = \mathrm{Comp}_{(2,0)}(\mathrm{PHP}) \geqslant c^n. \qquad \square$$

10. Some open problems

10.1. Some immediately ensuing questions

10.1.1. A hierarchy?

It seems to me that one has to spend some work on proving the generalization of Lemmas 8.4, 8.10 and 9.5:

"For all $i \geqslant 0$ there exists a sequence (F_n) of clause-sets with $\mathrm{Comp}_{(i+1,0)}(F_n)$ polynomial, but $\mathrm{Comp}_{(i,0)}(F_n)$ exponential".

Also it would be interesting whether for $i \geqslant 1$ the clause-sets F_n can be made minimally unsatisfiable.

10.1.2. Speed ups

Although we proved the same lower bound for $\mathrm{Comp}_{(\infty,0)}(\mathrm{PHP})$ as for $\mathrm{Comp}_{(0,0)}(\mathrm{PHP})$, it seems to be reasonable that $\mathrm{Comp}_{(\infty,0)}(\mathrm{PHP}) = o(\mathrm{Comp}_{(0,0)}(\mathrm{PHP}))$ holds.

Furthermore, it would be interesting how sharp the bound of Corollary 7.2 for the simulation of GER by ER is, that is, how much can GER speed up ER?

10.2. More general forms of extensions

While it is hard to prove lower bounds for ER (if there is any super-polynomial lower bound at all), perhaps one should look at the other end: Are there reasonable notions of extensions allowing in fact *polynomial resolution proofs* for arbitrary formulas? (Switching from proof systems to "proof systems with oracles").

Here I propose the following notion of a general form of extensions:

Definition 10.1. A clause-set $F' \in \mathscr{CLS}$ is called a *structure-preserving extension* (for short: sp-extension) *for* $F \in \mathscr{CLS}$ iff

$$\forall S \subseteq F [F \setminus S \in \mathscr{SAT} \Rightarrow (F \cup F') \setminus S \in \mathscr{SAT}]$$

holds (generalizing point 6 Remark after Lemma 6.3). For $F \in \mathscr{CLS}$ we define

$$\mathrm{Comp}_{\mathrm{SP}}(F) := \min\{\mathrm{Comp}_{\mathrm{R}}(F \cup F'): F' \text{sp-extension for } F\}.$$

For minimally unsatisfiable clause-sets F the condition for being a sp-extension states that in any proof of $\bot$ from $F \cup F'$ all clauses of F are needed.

It is conceivable that $\mathrm{Comp}_{\mathrm{SP}}$ is polynomially bounded.

11. For further reading

The following reference is also of interest to the reader: [7]

References

[1] P. Beame, T. Pitassi, Simplified and improved resolution lower bounds, In Thirtyseven Symposium on Foundations of Computer Science (FOCS' 96), 1996, pp. 274–282.

[2] S.A. Cook, A short proof of the pigeonhole principle using extended resolution, SIGACT News, October-December 1976, pp. 28–32.

[3] S.A. Cook, T. Pitassi, A feasibly constructive lower bound for resolution proofs, Inform. Process. Lett. 34 (1990) 81–85.

[4] S.A. Cook, R.A. Reckhow, Corrections for "On the lengths of proofs in the propositional calculus Preliminary version", SIGACT News 6 (1974) 15–22. Reference: [5].

[5] S.A. Cook, R.A. Reckhow, On the lengths of proofs in the propositional calculus, In sixth ACM Symposium on the Theory of Computing, May 1974, pp. 135–148. Preliminary version.

[6] S.A. Cook, R.A. Reckhow, The relative efficiency of propositional proof systems, J. Symbolic Logic 44 (1) (1979) 36–50.

[7] J. Franco, G. Gallo, H.K. Büning, E. Speckenmeyer, C. Spera (Eds.), Workshop on the Satisfiability Problem. Universität zu Köln, Report No. 96-230, 1996. Siena, 29 April – 3 May 1996.

[8] A. Haken, The intractability of resolution, Theoret. Comput. Sci. 39 (1985) 297–308.

[9] J. Krajiček, P. Pudlák, Propositional proof systems, the consistency of first order theories and the complexity of computations, J. Symbolic Logic 54 (3) (1989) 1063–1079.

[10] O. Kullmann, Obere und untere Schranken für die Komplexität von aussagenlogischen Resolutionsbeweisen und Klassen von SAT-Algorithmen. Master's thesis, Johann Wolfgang Goethe-Universität Frankfurt am Main, April 1992. (Upper and lower bounds for the complexity of propositional resolution proofs and classes of SAT algorithms (German)).

[11] O. Kullmann, An exponential lower bound for a general class of DPLL-like algorithms for SAT-decision, Preprint, March 1996.

[12] O. Kullmann, A note on a generalization of extended resolution, in: Franco et al. [7], pages 73–95. Siena, April 29 – May 3 (1996).

[13] O. Kullmann, Worst-case analysis, 3-SAT decision and lower bounds: Approaches for improved SAT algorithms. DIMACS Series in Discrete Mathematics and Theoretical Computer Science, vol. 35, American Mathematical Society, Providence, RI, 1997, 261–313.

[14] O. Kullmann, New methods for 3-SAT decision and worst-case analysis, Theoret. Comput. Sci., 1999.

[15] O. Kullmann, H. Luckhardt, Deciding propositional tautologies: algorithms and their complexity, Inform. Comput., Technical report (http://mi.informatik.uni-frankfurt.de/people/kullmann.kullmann.html). Improved version "Algorithms for SAT/TAUT decision based on various measures", submitted to Information and Computation.

[16] H. Luckhardt, Obere Komplexitätsschranken für TAUT-Entscheidungen, Frege Conference 1984, Schwerin, Akademie-Verlag, Berlin, 1984, pp. 331–337.

[17] B. Monien, E. Speckenmeyer, Solving satisfiability in less than 2^n steps, Discrete App. Math. 10 (1985) 287–295.

[18] P.W. Purdom, Solving satisfiability with less searching, IEEE Trans. on Pattern Anal. Mach. Intell. 6 (4) (1984) 510–513.
[19] I. Schiermeyer, Pure literal look ahead: An $O(1,497^n)$ 3-satisfiability algorithm, in: Franco et al. [7], pages 127–136. Siena, April 29–May 3 (1996).
[20] U. Schöning, Perlen der theoretischen Informatik, BI-Wissenschaftsverlag, 1995.
[21] G.S. Tseitin, On the complexity of derivation in propositional calculus, in: A.O. Slisenko (Ed.), Seminars in Mathematics, vol. 8. V.A. Steklov Mathematical Institute, Leningrad, 1968. English translation: Studies in Mathematics and Mathematical logic, Part II, 1970, pp. 115–125.
[22] A. Urquhart, The complexity of propositional proofs, Bull. Symbolic Logic 1 (4) (1995) 425–467.