Electronic Colloquium on Computational Complexity, Report No. 41 (2018)

Reordering Rule Makes OBDD Proof Systems Stronger

Sam Buss[3], Dmitry Itsykson[2], Alexander Knop[2, 3], and Dmitry Sokolov[1, 2]

[1]KTH Royal Institute of Technology
[2]St. Petersburg Department of V.A. Steklov Institute of Mathematics of the Russian Academy of Sciences
[3]University of California, San Diego
sbuss@ucsd.edu, dmitrits@pdmi.ras.ru, aknop@ucsd.edu, sokolovd@kth.se

February 26, 2018

Abstract

Atserias, Kolaitis, and Vardi [AKV04] showed that the proof system of Ordered Binary Decision Diagrams with conjunction and weakening, OBDD($\wedge$, weakening), simulates CP* (Cutting Planes with unary coefficients). We show that OBDD($\wedge$, weakening) can give exponentially shorter proofs than dag-like cutting planes. This is proved by showing that the Clique-Coloring tautologies have polynomial size proofs in the OBDD($\wedge$, weakening) system.

The reordering rule allows changing the variable order for OBDDs. We show that OBDD($\wedge$, weakening, reordering) is strictly stronger than OBDD($\wedge$, weakening). This is proved using the Clique-Coloring tautologies, and by transforming tautologies using coded permutations and orification. We also give CNF formulas which have polynomial size OBDD($\wedge$) proofs but require superpolynomial (actually, quasipolynomial size) resolution proofs, and thus we partially resolved open question proposed by Groote and Zantema [GZ03].

Applying dag-like and tree-like lifting techniques to the mentioned results we completely investigate the mutual strength for every pair of systems among CP*, OBDD($\wedge$), OBDD($\wedge$, weakening) and OBDD($\wedge$, weakening, reordering). For dag-like proof systems, some of our separations are quasipolynomial and some are exponential; for tree-like systems, all of our separations are exponential.

1 Introduction

An Ordered Binary Decision Diagram (OBDD) is a type of branching program such that variables are queried in the same order on every path from the source to a sink. OBDDs were defined by Bryant [Bry85] and has been shown to be useful in a variety of domains, such as hardware verification, model checking, and other CAD applications [McM93, BCM+92]. Perhaps their most important property is that it is possible to carry out operations on OBDDs efficiently, including Boolean operations, projects, and testing satisfiability.

OBDDs have been used for several approaches to SAT-solving [US94]. The first such algorithms [US94] worked by computing an OBDD for bigger and bigger subformulas of the input formula until obtaining an OBDD for the entire input formula, and then testing the resulting OBDD for satisfiability. A more attractive algorithm, called symbolic quantifier elimination, was proposed by Pan and Vardi [PV05]. Symbolic quantifier elimination loads clauses of the input formula into the current OBDD one by one and applies projection by a variables which do not appear in the remaining clauses. In contrast with DPLL algorithms, symbolic quantifier elimination can solve Tseitin formulas [IKRS17] and the pigeonhole principle [CZ09] in polynomial time.

Atserias-Kolaitis-Vardi [AKV04] defined a proof system based on OBDDs for proving unsatisfiability of CNFs, which is now called OBDD($\wedge$, weakening). An OBDD($\wedge$, weakening) proof is a sequence of π-OBDDs with the ordering π of the variables held fixed. The initial lines are π-OBDD's expressing the input clauses; the final line is the constant false. Each step of the proof applies one of the two rules:

ISSN 1433-8092

join or $\wedge$: A conjunction of any two previously derived π-OBDDs is inferred;

weakening: A π-OBDD is inferred that is semantically implied by some earlier derived π-OBDD.

The paper [AKV04] shows that Cutting Planes with unary coefficients (CP^*) is simulated by $\mathrm{OBDD}(\wedge, \text{weakening})$. This was proved by showing that any linear inequality there is a short π-OBDD representation (under any ordering π) and that addition of two inequalities may be simulated by join and weakening. Hence $\mathrm{OBDD}(\wedge, \text{weakening})$ is strictly stronger than resolution; however, Segerlind [Seg08] have shown that tree-like $\mathrm{OBDD}(\wedge, \text{weakening})$ does not simulate (dag-like) resolution. Additionally, [AKV04] showed that any unsatisfiable system of linear equation modulo two has a short refutation in $\mathrm{OBDD}(\wedge, \text{weakening})$, while it is open, whether linear systems have short CP refutations. It is still open whether CP is strictly stronger than CP^*, and corresponding, it is open whether $\mathrm{OBDD}(\wedge, \text{weakening})$ simulates CP.

Krajíček [Kra08] proved the first exponential lower bound for $\mathrm{OBDD}(\wedge, \text{weakening})$. His lower bound consisted of two parts.

1. If a function f is computed by a π-OBDD D, the communication complexity of f under a partition Π_0, Π_1 of the variables where the variables in Π_0 precede (in the sense of π) the variables from Π_1 is at most $\lceil \log |D| \rceil + 1$. Since every proof system that operates with proof lines with small communication complexity admits monotone feasible interpolation [Kra97], there is an ordering π of the variables so that any π-$\mathrm{OBDD}(\wedge, \text{weakening})$ proof of the Clique-Coloring principle has exponential size. (This was already proven by Atserias et al. [AKV04]).

2. Second, there is a transformation that transforms any "good" formula that is hard for $\mathrm{OBDD}(\wedge, \text{weakening})$ in *some* order into a formula that is hard for $\mathrm{OBDD}(\wedge, \text{weakening})$ in *all* orders. (We use similar transformation proposed by Segerlind [Seg08] to prove Lemma 3.1 and Theorem 3.4).

In Theorem 3.2, we give short (polynomial size) $\mathrm{OBDD}(\wedge, \text{weakening})$ proofs of the Clique-Coloring principle. Since any CP proof of the Clique-Coloring principle has exponential size [Pud97], it follows that CP does not simulate $\mathrm{OBDD}(\wedge, \text{weakening})$ and moreover, that $\mathrm{OBDD}(\wedge, \text{weakening})$ is strictly stronger than CP^*. The existence of the small proofs of the Clique-Coloring principle implies that $\mathrm{OBDD}(\wedge, \text{weakening})$ does not have the feasible interpolation property. This is very curious, because the monotone feasible interpolation property nonetheless helps to prove lower bounds for this system.

Our short proofs of the Clique-Coloring principles are based on Grigoriev et. al [GHP02], who gave short proofs of Clique-Coloring in LS^4, a proof system that uses inequalities of degree 4. Unfortunately, even inequalities of degree 2 do not have short OBDD representation, in contrast to inequalities of degree 1. Nevertheless, the proof of [GHP02] may be simulated in $\mathrm{OBDD}(\wedge, \text{weakening})$ in some order over the variables.

An interesting subsystem of $\mathrm{OBDD}(\wedge, \text{weakening})$ is the system $\mathrm{OBDD}(\wedge)$ that uses only the join rule; this system is connected with early OBDD algorithms for SAT-solving [US94]. Tveretina et al. [TSZ10] proved that $\mathtt{PHP}_n^{n+1}$ is hard for $\mathrm{OBDD}(\wedge)$. Grut and Zantema [GZ03] showed that there is an unsatisfiable formula (not in CNF) such that its translation in to OBDDs has short $\mathrm{OBDD}(\wedge)$ proofs, any resolution proof of its Tseitin transformation has exponential size. Because of the different translations, the question of an actual separation between $\mathrm{OBDD}(\wedge)$ and resolution was left open. In Corollary 4.1 and Lemma 4.2, we improve their result by giving CNF formulas which have polynomial size $\mathrm{OBDD}(\wedge)$ proofs but require superpolynomial (actually, quasipolynomial size) resolution proofs.

Järvisalo [Jär11] claimed an exponential separation between tree-like resolution proof and (dag-like) $\mathrm{OBDD}(\wedge)$ proofs. Unfortunately, as is discussed in Section 5, the proof for the last claim was erroneous. We correct the proof and establish an even stronger result: the proof of Theorem 5.4 shows that there is a formula ψ_n such that in some order π any tree-like π-$\mathrm{OBDD}(\wedge, \text{weakening})$ proof of ψ_n has exponential size, but there is a short $\mathrm{OBDD}(\wedge)$ proof of ψ_n in another order. Note that tree-like π-$\mathrm{OBDD}(\wedge, \text{weakening})$ simulates tree-like resolution for any order π.

So far, we have only discussed OBDD proof systems for which proofs consists of π-OBDDs in the same fixed order π. This constraint is somewhat artificial since there is an algorithm polynomial to transforms an OBDD in one order into an OBDD in another order which runs in time polynomially bounded by the combined sizes of the input and output OBDDs. Accordingly, Itsykson et al. [IKRS17] introduced the proof system OBDD($\wedge$, reordering). This system includes a *reordering* rule which allows changing an OBDD to a different variable ordering. It also includes the join ($\wedge$) rule, but with the condition that the two conjoined OBDDs use the same variable ordering. They showed that OBDD($\wedge$, reordering) does not have short proofs of $\mathtt{PHP}_n^{n+1}$ or of Tseitin formulas based on expanders. Additionally, they showed that OBDD($\wedge$, reordering) is strictly stronger than OBDD($\wedge$). In Theorem 3.4, we resolve an open question of [IKRS17] by showing that OBDD($\wedge$, weakening, reordering) is strictly stronger than OBDD($\wedge$, weakening).

Theorem 4.5 constructs formulas that have small size tree-like OBDD($\wedge$, reordering) proofs but require superpolynomially larger size (dag-like) OBDD($\wedge$, weakening) proofs. The proof uses a result of [GGKS17] and formulas that have short OBDD($\wedge$) refutations but require superpolynomial size resolution proofs. This method also allows constructing formulas that are hard for CP but easy for OBDD($\wedge$), see Theorem 4.4. In Theorem 5.4 further, we give CNF formulas which have polynomial size tree-like OBDD($\wedge$, reordering) proofs but require exponential size for tree-like OBDD($\wedge$, weakening) proofs.

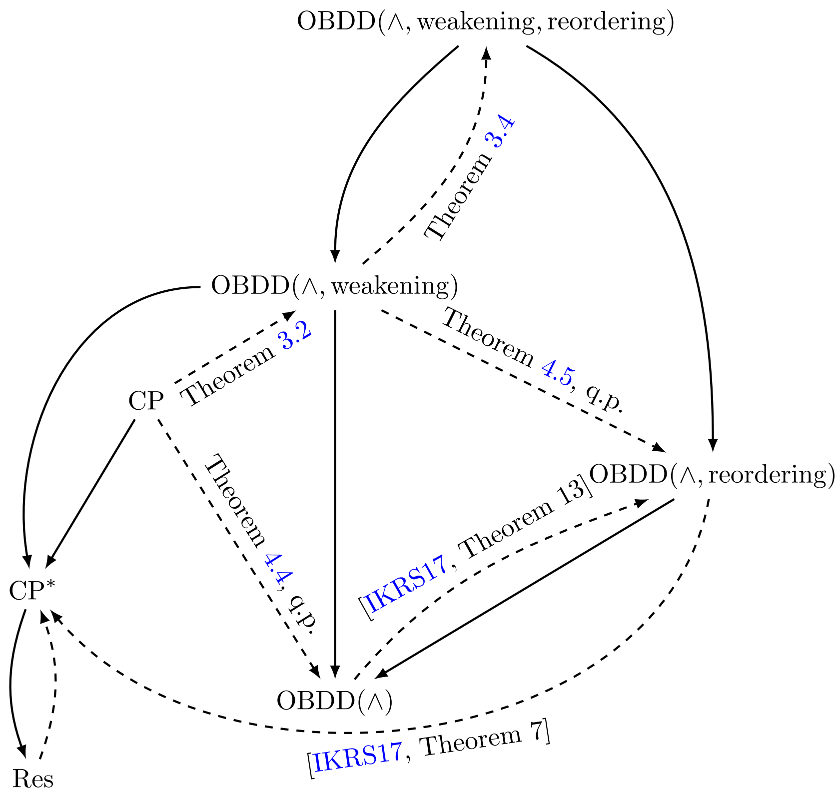

Figure 1: $C_1 \longrightarrow C_2$ denotes C_1 p-simulates C_2, and $C_1 \dashrightarrow C_2$ denotes C_1 does not p-simulate C_2. The results are for the dag-like versions of the systems. New results are labelled with the relevant theorem. All the separations on the picture are exponential, except the two separations labeled by "q.p" for "quasipolynomial".

A summary of the (non-)simulation results for dag-like systems is shown in Figure 1. There are still a few questions left open about the systems shown there. First, it is a long-standing open problem whether CP* simulates CP. Second, it is open whether OBDD($\wedge$, weakening) simulates CP. Third, we do not whether resolution is simulated by OBDD($\wedge$, reordering). In fact, we do not know whether resolution is simulated by OBDD($\wedge$). A couple earlier papers have claimed that resolution is not simulated by OBDD($\wedge$), see

Theorem 5 of [TSZ10] and Corollary 4 of [Jär11], but we have been unable to verify their proofs.[1]

All the other missing arrows in Figure 1 follow from the arrows shown. For instance, OBDD($\wedge$) does not simulate CP*, since OBDD($\wedge$, reordering) does not simulate CP*.

Further research. Segerlind showed [Seg08] that dag-like resolution does not simulate tree-like OBDD($\wedge$, weakening), hence dag-like OBDD($\wedge$, weakening) is strictly stronger than tree-like OBDD($\wedge$, weakening). It is interesting to compare OBDD($\wedge$), OBDD($\wedge$, reordering) and OBDD($\wedge$, weakening, reordering) with their tree-like versions.

It is interesting open question, whether resolution quasipolynomially simulates OBDD($\wedge$). Any improving of our separation will automatically improve separations between CP vs. OBDD($\wedge$) and OBDD($\wedge$, weakening) vs. OBDD($\wedge$, reordering).

The major open question is to prove superpolynomial lower bound on the size of OBDD($\wedge$, weakening, reordering) refutations.

2 Preliminaries

2.1 Ordered Binary Decision Diagrams

An ordered binary decision diagram (OBDD) is a data structure that is used to represent a Boolean function [Bry85]. Let $\Gamma = \{x_1, \ldots, x_n\}$ be a set of propositional variables. A binary decision diagram (BDD) is a directed acyclic graph with one source. Each vertex of the graph is labeled by a variable from Γ or by a constant 0 or 1. If a vertex is labeled by a constant, then it is a sink (has out-degree 0). If a vertex is labeled by a variable, then it has exactly two outgoing edges: one edge is labeled by 0 and the other edge is labeled by 1. Every binary decision diagram defines a Boolean function $\{0,1\}^n \to \{0,1\}$. The value of the function for given values of $x_1, \ldots, x_n$ is computed as follows: we start a path at the source and at every step follow the edge that corresponds to the value of the variable labelling the current vertex. Every such path reaches a sink, which is labelled either 0 or 1: this constant is the value of the function.

Let π be a permutation of the set $[n] = \{1, \ldots, n\}$. A π-ordered binary decision diagram (π-OBDD) is a binary decision diagram such that on every path from the source to a sink every variable has at most one occurrence and the variable $x_{\pi(i)}$ can not appear before $x_{\pi(j)}$ if $i > j$. An ordered binary decision diagram (OBDD) is a π-ordered binary decision diagram for some permutation π. By convention, every OBDD is associated with a single fixed permutation π. This π puts a total order on all the variables, even if the OBDD does not query all variables.

OBDD's have a number of nice properties. The size of an OBDD is the number of vertices in it, and for a fixed ordering π of variables, every Boolean function has a unique minimal π-OBDD. Furthermore, the minimal π-OBDD of a function f may be constructed in polynomial time from any π-OBDD for the same f. There are also polynomial-time algorithms which act on π-OBDD's and efficiently perform the operations of conjunction, negation, disjunction, and projection [MS94]. (Projection is the operation that maps a π-OBDD D computing the Boolean function $f(x, y_1, \ldots, y_n)$ to a π-OBDD D' computing the Boolean function $\exists x\, f(x, y_1, \ldots, y_n)$.) In addition, there is an algorithm running in time polynomial in the combined sizes of the input and the output which takes as input a π-OBDD D and a permutation ρ, and returns the minimal ρ-OBDD that represents the same function as D [MS94].

[1] The difficult point in the proofs is in Lemma 8 of [TSZ10] and in Lemma 4 of [Jär11]. In the former, it is shown that two distinct nodes in an OBDD $B(F, \prec)$ correspond to two distinct nodes in another OBDD $B(F \cup G, \prec)$; however, it does not follow from this that n distinct nodes in $B(F, \prec)$ correspond to n distinct nodes in $B(F \cup G, \prec)$. A similar technique is implicitly used in the latter paper.

2.2 Proof Systems

2.2.1 Resolution

For an unsatisfiable CNF formula φ, a resolution refutation or a proof of its unsatisfiability in the resolution proof system is a sequence of clauses with the following properties: the last clause is an empty clause, denoted $\square$; and every clause is either a clause of the initial formula φ, or can be obtained from previous ones by the resolution rule. The resolution rule allows to infer a clause $(B \vee C)$ from clauses $(x \vee B)$ and $(\neg x \vee C)$. The size of a resolution refutation is the number of clauses in it. It is well known that the resolution proof system is sound and complete. Soundness means that if a formula has a resolution refutation then it is unsatisfiable. Completeness means that every unsatisfiable CNF formula has a resolution refutation. The size of such a proof is a number of clauses in it. If every clause is used as a premise of the inference rule at most once, then the proof is *tree-like*.

2.2.2 Cutting Planes

Before we give a definition of this proof system let us define the translation of clauses into linear inequalities by the following rule: if $C = \bigvee\limits_{i=1}^{n} x_i^{b_i}$, then $L(C)$ is the following inequality $\sum\limits_{i=1}^{n}(-1)^{1-b_i}x_i \geq 1 - \sum\limits_{i=1}^{n}(1-b_i)$ where x^0 denotes $\neg x$ and x^1 denotes x. For an unsatisfiable CNF formula φ over the variables $x_1, \ldots, x_n$, a Cutting Planes refutation or a proof of its unsatisfiability in the Cutting Planes proof system is a sequence of inequalities $I_1, \ldots, I_t$ of the type $\sum\limits_{i=1}^{n} a_i x_i \geq c$ (where $a_i, c \in \mathbb{Z}$) such that I_t is an inequality $0 \geq 1$ and every inequality I_j is either $L(C)$ where C is some clause of the initial formula φ or can be obtained from previous inequalities by the following rules:

linear combination: I_j is an inequality $\sum\limits_{i=1}^{n}(\alpha \cdot a_i + \beta \cdot b_i)x_i \geq \alpha c + \beta d$ where for some $\alpha, \beta > 0$ and $1 \leq k, \ell < j$, I_k is an inequality $\sum\limits_{i=1}^{n} a_i x_i \geq c$ and I_ℓ is an inequality $\sum\limits_{i=1}^{n} b_i x_i \geq d$;

division: I_j is an inequality $\sum\limits_{i=1}^{n} a_i x_i \geq \lceil c/d \rceil$, where for some $k < j$, I_k is an inequality $\sum\limits_{i=1}^{n} d a_i x_i \geq c$.

The size of such a refutation is a number of inequalities.

Additionally, we say that an unsatisfiable CNF formula φ has CP* refutation of size S iff there is a CP refutation of φ such that the sum of absolute values of coefficients in the inequalities in this proof is at most S.

We say that an unsatisfiable CNF formula φ has a semantic CP refutation (semantic CP* refutation) of size S if there is a CP refutation of φ of size S such that instead of these rules we allow to derive any semantic implication of at most two previously derived inequalities. Note that semantic CP (semantic CP*) is not a Cook–Reckhow proof system since it is NP-hard to check the correctness of the semantic rule.

Also, If every inequality is used as a premise of the inference rule at most once, then we call this proof a *tree-like* proof.

2.2.3 OBDD-based Proof Systems

Let φ be an unsatisfiable CNF formula. An OBDD proof of φ is a sequence $D_1, D_2, \ldots, D_t$ of OBDD's and permutations $\pi_1, \ldots, \pi_t$ such that D_t is an π_t-OBDD that represents the constant false function, and such that each D_i is either an π_i-OBDD which represents a clause of φ or can be obtained from previous OBDD's by one of the following inference rules:

join or $\wedge$: D_i represents the Boolean function $D_k \wedge D_\ell$ for $1 \leq \ell, k < i$, where D_i, D_k, D_ℓ have the same order $\pi_i = \pi_k = \pi_\ell$;

weakening: there exists a $j < i$ such that D_i and D_j have the same order $\pi_i = \pi_j$, and D_j semantically implies D_i. The latter means that every assignment that satisfies D_j also satisfies D_i;

reordering: D_i is an π_i-OBDD that is equivalent to a π_j-OBDD D_j with $j < i$.

Note that although we use terminology "OBDD proof", it is actually a *refutation* of φ. By the discussion in the previous section, there is a polynomial time algorithm which recognizes whether a given $D_1, \dots, D_t$ and $\pi_1, \dots, \pi_t$ is a valid OBDD proof of a given φ. The size of this proof is equal to $\sum_{i=1}^{t} |D_i|$.

We use several different OBDD proof systems with different sets of allowed rules. For example, the OBDD($\wedge$, weakening) proof system uses conjunction and weakening rules; hence, all OBDDs in such a proof have the same order π. We use the notation π-OBDD($\wedge$) proof and π-OBDD($\wedge$, weakening) proof to explicit indicate the ordering. If every D_i is used as a premise of the inference rule at most once, then the proof *tree-like.*

3 $\mathrm{OBDD}(\wedge, \mathrm{weakening}, \mathrm{reordering})$ is Strictly Stronger Than $\mathrm{OBDD}(\wedge, \mathrm{weakening})$

This section constructs formulas which are easy for OBDD($\wedge$, weakening, reordering) (have short OBDD($\wedge$, weakening, reordering) proofs) and hard for OBDD($\wedge$, weakening) (do not have short OBDD($\wedge$, weakening) proofs). For this, we construct a transformation $\mathcal{T} = \mathcal{T}(\varphi)$ such that

- If a formula φ is hard for π-OBDD($\wedge$, weakening) for some order π, then $\mathcal{T}(\varphi)$ is hard for OBDD($\wedge$, weakening); i.e., $\mathcal{T}(\varphi)$ is hard for any order.
- If a formula φ is easy for π-OBDD($\wedge$, weakening) for some order π, then $\mathcal{T}(\varphi)$ is easy OBDD($\wedge$, weakening, reordering).

Then we construct a formula φ such that there are two orders π_1 and π_2 such that φ is hard for π_1-OBDD($\wedge$, weakening) but easy for π_2-OBDD($\wedge$, weakening). As a corollary, we get that $\mathcal{T}(\varphi)$ separates OBDD($\wedge$, weakening, reordering) and OBDD($\wedge$, weakening).

The formula φ expresses Clique-Coloring principle ($\texttt{Clique-Coloring}_{n,m}$) that encodes that any $(m-1)$-colorable graph on n vertices does not contain a clique of the size m for $m \approx \sqrt{n}$. Atserias, Kolaitis, and Vardi [AKV04] proved (see also Krajíček [Kra08]) that $\texttt{Clique-Coloring}_{n,m}$ is hard for π-OBDD($\wedge$, weakening) for some order π. However, in Section 6 we show that there is an order π such that $\texttt{Clique-Coloring}_{n,m}$ has an π-OBDD($\wedge$, weakening) proof of size polynomially bounded by n and m.

3.1 Construction of $\mathcal{T}$

The transformation $\mathcal{T}$ is based on a construction of Segerlind [Seg08]. We develop the definition of $\mathcal{T}$ in stages. As a first approximation, we define how to transform a formula $\varphi(x_1, \dots, x_n)$ into a formula $\mathrm{perm}_{S_n}(\varphi)(z_1, \dots, z_\ell, x_1, \dots, x_n)$ where $\ell = \lceil \log(n!) \rceil$. Fix an injective map $\mathrm{rep} : S_n \to \{0,1\}^\ell$ that maps the set of permutations of $[n]$ into binary strings of length ℓ. The formula $\mathrm{perm}_{S_n}(\varphi)$ is defined by:

$$\mathrm{perm}_{S_n}(\varphi)(z_1, \dots, z_\ell, x_1, \dots, x_n) = \bigwedge_{\sigma \in S_n} \left[\left(\bigwedge_{i=1}^{\ell} z_i = \mathrm{rep}(\sigma)_i \right) \to \varphi\left(x_{\sigma(1)}, \dots, x_{\sigma(n)}\right) \right] \wedge \bigwedge_{t \in \{0,1\}^\ell \setminus \mathrm{rep}(S_n)} \neg(z_1 = t_1 \wedge z_2 = t_2 \wedge \dots \wedge z_\ell = t_\ell).$$

Note that it is easy to convert $\mathrm{perm}_{S_n}(\varphi)$ into a formula in CNF. We just add to each clause of $\varphi(x_{\sigma(1)}, \dots, x_{\sigma(n)})$ the literals $z_1^{1-\mathrm{rep}(\sigma)_1}, z_1^{1-\mathrm{rep}(\sigma)_2}, \dots, z_\ell^{1-\mathrm{rep}(\sigma)_\ell}$, where z_i^0 denotes $\neg z_i$, z_i^1 denotes z_i, and also add the clauses $\neg(z_1 = t_1 \wedge z_2 = t_2 \wedge \cdots \wedge z_\ell = t_\ell)$. It is easy to see that the formula $\mathrm{perm}_{S_n}(\varphi)$ is unsatisfiable since if a substitution to variables $z_1, z_2, \ldots, z_\ell$ does not correspond to a representation of some permutation, then this substitution falsifies the constraint $\neg(z_1 = t_1 \wedge z_2 = t_2 \wedge \cdots \wedge z_\ell = t_\ell)$ and if a substitution to the variables $z_1, z_2, \ldots, z_\ell$ corresponds to a permutation σ, then the formula $\left(\bigwedge_{i=1}^{\ell} z_i = \mathrm{rep}(\sigma)_i\right) \to \varphi(x_{\sigma(1)}, \ldots, x_{\sigma(n)})$ is falsified by this substitution, since φ is unsatisfiable.

Applying the partial substitution $z_i := \mathrm{rep}(\sigma)_i$ for all i to $\mathrm{perm}_{S_n}(\varphi)(z_1, \ldots, z_\ell, x_1, \ldots, x_n)$ yields the formula $\varphi(x_{\sigma(1)}, \ldots, x_{\sigma(n)})$. This implies that if φ requires π-OBDD($\wedge$, weakening) proof of size S for some order π, then $\mathrm{perm}_{S_n}(\varphi)$ requires a OBDD($\wedge$, weakening) proof of size S in any order. Indeed, let τ be an order on the variables $z_1, z_2, \ldots, z_\ell$, $x_1, x_2, \ldots, x_n$ and let σ be the order on the variables $x_1, \ldots, x_n$ induced by τ. The substitution $z_1 z_2 \ldots z_\ell := \mathrm{rep}(\pi\sigma^{-1})$ transforms a τ-OBDD($\wedge$, weakening) proof of $\mathrm{perm}_{S_n}(\varphi)$ to a π-OBDD($\wedge$, weakening) proof of φ with no increase in size. Hence the size of the minimal OBDD($\wedge$, weakening) proof of $\mathrm{perm}_{S_n}(\varphi)$ is at least S.

The problem with the transformation perm_{S_n} is that $\mathrm{perm}_{S_n}(\varphi)$ can be exponentially big. So the next idea for a transformation is to consider a small "good" set of permutations $\Pi \subseteq S_n$ instead of all of S_n. Letting $\ell = \lceil \log |\Pi| \rceil$ and letting rep now be some injective map $\mathrm{rep} : \Pi \to \{0,1\}^\ell$, we define analogously

$$\mathrm{perm}_\Pi(\varphi)(z_1, \ldots, z_\ell, x_1, \ldots, x_n) = \bigwedge_{\sigma \in \Pi} \left[\left(\bigwedge_{i=1}^{\ell} z_i = \mathrm{rep}(\sigma)_i \right) \to \varphi\left(x_{\sigma(1)}, \ldots, x_{\sigma(n)}\right) \right] \wedge \bigwedge_{t \in \{0,1\}^\ell \setminus \mathrm{rep}(\Pi)} \neg(z_1 = t_1 \wedge z_2 = t_2 \wedge \cdots \wedge z_\ell = t_\ell).$$

The problem with this is that it is possible that $\pi\sigma^{-1}$ does not belong to Π.

To solve this problem we "or-ify" variables: each variable x_i is replaced by the disjunction of m fresh variables $y_{i,1}, \ldots y_{i,m}$; i.e., instead of $\varphi(x_1, x_2, \ldots, x_n)$ we consider $\varphi^{\vee m}(y_{1,1}, \ldots, y_{n,m}) = \varphi\left(\bigvee_{j=1}^{m} y_{1,j}, \ldots, \bigvee_{j=1}^{m} y_{n,j}\right)$. Now let $\Pi \subseteq S_{mn}$ and consider $\mathrm{perm}_\Pi(\varphi^{\vee m})$. As in previous case we want to substitute variables to a proof of $\mathrm{perm}_\Pi(\varphi^{\vee m})$ in some order and get a proof of φ in order π. However, in this case we substitute not only for the variables $z_1, \ldots, z_\ell$, but also for each $k \in [n]$ we substitute zero for all variables $y_{k,i}$ except one. This increases the number of different permutations of the variables $x_1, \ldots, x_n$ that we can obtain. The only problem with this transformation is that for some formulas φ, the size of $\varphi^{\vee m}$ may be exponentially bigger than the size of φ. However, if each clause of φ there is only one negated literal or only $O(1)$ negated literals, then the size of $\varphi^{\vee m}$ will be polynomially bounded.

Our "good" set of permutations is a set of pairwise independent permutations. Let $t = \lceil \log(n) \rceil$ and $N = 2^t$, and $\mathbb{F}$ be the field $\mathrm{GF}(N)$. Define Π_n to be the set of all mappings given by $x \mapsto ax + b$ with $a, b \in \mathbb{F}$ and $a \neq 0$. Elements of Π_n may be represented by binary strings of length $\ell = 2t$ such that the first t bits are not all zero. Note that $\Pi_n \subseteq S_N$ so we have to add new variables, $x_{n+1}, \ldots, x_N$. Then define

$$\mathrm{perm}(\varphi)(z_1, \ldots, z_\ell, x_1, \ldots, x_N) = \bigwedge_{\sigma \in \Pi_n} \left[\left(\bigwedge_{i=1}^{2t} z_i = \mathrm{rep}(\sigma)_i \right) \to \varphi(x_{\sigma(1)}, \ldots, x_{\sigma(n)}) \right] \wedge \bigvee_{i=1}^{t} z_i.$$

Now we can define the transformation $\mathcal{T}$. Let φ be a formula on n variables and m be the least integer such that $\frac{2n^3}{m} + \frac{n^2}{mn-1} < 1$, so $m = O(n^3)$. Then $\mathcal{T}(\varphi) = \mathrm{perm}(\varphi^{\vee m})$. The first property of $\mathcal{T}$ given at the beginning of Section 2.2 was established by Segerlind [Seg08]:

Lemma 3.1 ([Seg08]). *Let φ be an unsatisfiable formula in CNF on the variables $x_1, \ldots, x_n$. Suppose there is an* OBDD($\wedge$, weakening) *proof (respectively, an* OBDD($\wedge$) *proof) of the formula $\mathcal{T}(\varphi)$ of size S. Then for every order π on $x_1, \ldots, x_n$ there is a π-*OBDD($\wedge$, weakening) *proof (respectively, a π-*OBDD($\wedge$) *proof) of φ of size at most S.*

The idea of the proof of lemma is as follows. Suppose $\tau \in \Pi_n$ is an order on $z_1, \ldots, z_\ell, x_1, \ldots, x_N$, and let π be an order on $x_1, \ldots, x_n$. Then there are $j_1, \ldots, j_n$ such the order τ restricted to $y_{1,j_1}, \ldots, y_{n,j_n}$ is the same as the order π on $x_1, \ldots, x_n$. Replacing the variables z_i with the constants $(\mathrm{rep}(\tau)_i$, renaming the variables y_{i,j_i} to x_i, and replacing all other variables $y_{i,j}$ with 0 thus transforms the OBDD($\wedge$, weakening) or OBDD($\wedge$) proof of $\mathcal{T}(\varphi)$ into a proof of φ. For details, consult Segerlind [Seg08].

The second property of $\mathcal{T}$ states that if φ is easy for OBDD($\wedge$, weakening) in some order, then $\mathcal{T}(\varphi)$ is easy for OBDD($\wedge$, weakening, reordering). Its proof consists of two parts: First, Lemma 3.2 shows that if φ is easy for OBDD($\wedge$, weakening), then $\mathrm{perm}(\varphi)$ is easy for OBDD($\wedge$, weakening, reordering); then Section 3.2 shows that if φ is easy for OBDD($\wedge$, weakening), then $\varphi^{\vee m}$ is easy for OBDD($\wedge$, weakening).

Lemma 3.2. *Let $\varphi_n(x_1, x_2, \ldots, x_n)$ be a family of unsatisfiable formulas such that there is an order τ such that φ_n has a τ-OBDD($\wedge$, weakening) proof P_1 of size $t(n)$ then the formula $\mathrm{perm}(\varphi_n)$ has an OBDD($\wedge$, weakening, reordering) proof P_2 of size $t(n)\mathrm{poly}(n)$. If P_1 is tree-like, then so is P_2. In addition, if P_1 does not use the weakening rule, then neither does P_2.*

Proof. Let τ be an order on $x_1, x_2, \ldots, x_n$ such that $\varphi(x_1, x_2, \ldots, x_n)$ has a short τ-OBDD($\wedge$, weakening) proof of size $t(n)$. We describe a short OBDD($\wedge$, weakening, reordering) proof of $\mathrm{perm}(\varphi_n)$. Let μ_σ be an order on $z_1, z_2, \ldots, z_\ell, x_1, x_2, \ldots, x_n$ such that $x_1, x_2, \ldots, x_n$ are ordered by $\tau\sigma^{-1}$ and the variables $z_1, z_2, \ldots, z_\ell$ appear before the variables $x_1, x_2, \ldots, x_n$. In other words, μ_σ orders variables as follows: $z_1, z_2, \ldots, z_\ell, x_{\tau\sigma^{-1}(1)}, x_{\tau\sigma^{-1}(2)}, \ldots, x_{\tau\sigma^{-1}(n)}$.

Consider a τ-OBDD($\wedge$, weakening) proof of φ_n of size $t(n)$. It is easy to see that this proof may be transformed into a μ_σ-OBDD($\wedge$) proof of a diagram that represents $\neg\left(\bigwedge_{i=1}^{\ell} z_i = \mathrm{rep}(\sigma)_i\right)$ from the formula $\left(\bigwedge_{i=1}^{\ell} z_i = \mathrm{rep}(\sigma)_i\right) \to \varphi(x_{\sigma(1)}, \ldots, x_{\sigma(n)})$. Indeed, the variables $z_1, z_2, \ldots, z_\ell$ in the order μ_σ appear in the beginning, hence every diagram D from the original proof will be transformed into a diagram that represents $D \vee \neg\left(\bigwedge_{i=1}^{\ell} z_i = \mathrm{rep}(\sigma)_i\right)$ and the latter diagram has the size at most $|D| + O(\ell)$, where $|D|$ is the size of D.

This derives all the diagrams which represent $\bigwedge_{i=1}^{\ell} z_i = \mathrm{rep}(\sigma)_i$ for $\sigma \in \Pi_n$ and a diagram encodes $\bigvee_{i=1}^{\ell} z_i$. Formally these diagrams use different orders μ_σ but in fact the diagrams depend essentially only on the variables $z_1, z_2, \ldots, z_\ell$, and all μ_σ order them in the same way. Thus, the reordering rule can be used to change the orders in all of these diagrams to some "standard" one, without changing the diagrams. Applying the conjunction rule to these diagrams yields the constant false diagram since $z_1 z_2 \ldots z_\ell$ is a binary representation of $\mathrm{rep}(\sigma)$ for some $\sigma \in \Pi_n$ or $z_1 = z_2 = \cdots = z_t = 0$. All intermediate diagrams have the size at most $2^\ell \cdot \mathrm{poly}(n)$. Hence, we get a proof of size $t(n)\mathrm{poly}(n)$.

The construction preserves the tree-like property, and whether the weakening rule is used, so Lemma 3.2 is proved. □

3.2 Complexity of Composition

We now prove that if φ has a small OBDD($\wedge$, weakening) proof, then $\varphi^{\vee m}$ has a small OBDD($\wedge$, weakening) proof. In fact, we prove more a general statement. Let φ be a CNF formula with n variables, and $g : \{0,1\}^k \to \{0,1\}$ be a Boolean function. Then $\varphi \circ g$ denotes a CNF formula on kn variables that represents $\varphi(g(\vec{x}_1), g(\vec{x}_2), \ldots, g(\vec{x}_n))$, where $\vec{x}_i$ denotes a vector of k new variables. $\varphi \circ g$ is constructed by applying the substitution to every clause C of φ and converting the resulting function $C \circ g$ to CNF in some fixed way.

We need the following technical definition. Consider a CNF formula $\varphi = \bigwedge_{i=1}^{m} C_i$. We say φ is S-constructible with respect to (w.r.t.) the order π if there is a binary tree with vertices labeled by π-OBDD's such that: (1) the root is labeled by an π-OBDD-representation of φ, (2) the tree contains m leaves labeled by π-OBDD-representations of the clauses C_i, each clause appears in exactly one leaf, (3) each vertex is labelled by an π-OBDD that represents the conjunction of labels of its children, and (4) the size of each label is at most S.

Remark 3.1. *If φ is S-constructible CNF w.r.t. the order π, then there is a tree-like π-$\mathrm{OBDD}(\wedge)$-derivation of size $(2m-1)S$ of an π-OBDD that represents φ from the clauses of φ.*

Proposition 3.1. *Let $F = G_1 \vee G_2$, where G_1 and G_2 are Boolean functions that depend on disjoint sets of variables. If the variables of G_1 precede variables of G_2 in the order π, then the smallest size of π-OBDD representation of F is at most the sum of sizes of the smallest π-OBDD representations of G_1 and G_2.*

Proof. This is obvious. The π-OBDD for F can be obtained by the identifying the source of the π-OBDD for G_2 with the sink of π-OBDD for G_1 labeled by 0. □

Lemma 3.3. *Let $F_1, F_2, \ldots, F_k$ be CNF formulas with disjoint sets of variables, where $F_j = \bigwedge_{i \in I_j} C_i$ for all $j \in [k]$. Let $\pi_1, \ldots, \pi_k$ be orders such that each F_j is S-constructible w.r.t. π_j. Define the order π to order the variables of each F_i according to π_i and so that all variables of F_i precede all variables of F_{i+1}. Let F be the CNF representation of the function $F_1 \vee F_2 \vee \cdots \vee F_k$, namely, $F = \bigwedge_{i_1 \in I_1, \ldots, i_k \in I_k} \bigvee_{j=1}^{k} C_{i_j}$. Then F is kS-constructible w.r.t. π.*

Proof. We prove this lemma by induction on k. The basis case is trivial: if $k = 1$, then $F = F_1$, hence F is S-constructible. For the induction hypothesis, let $G = F_1 \vee F_2 \vee \cdots \vee F_{k-1}$. By the induction hypothesis G is $(k-1)S$-constructible w.r.t. π. For each clause D of G and each $i \in I_k$, the clause $D \vee C_i$ is a clause of F. The formula F_k is S-constructible w.r.t. π by a tree T_k with $|I_k|$ leaves which are labeled by C_i for $i \in I_\ell$. We wish to replace each leaf of T_k labelled with a C_i with a tree for $G \vee C_i$. Since G is $(k-1)S$-constructible and since the variables of C_i are disjoint from those of G, Proposition 3.1 implies that $G \vee C_i$ is kS-constructible w.r.t. π, since we can incorporate the clause C_i into all clauses the tree giving the $(k-1)S$-constructibility of G. In addition, replace all the diagrams D labelling vertices in the tree T_k by $D \vee G$; by Proposition 3.1 the size of the updated diagrams is at most kS. This gives a tree witnessing the kS-constructibility of $F_1 \vee \cdots \vee F_k$ as desired. □

Theorem 3.1. *Let π be an order on $z_1, \ldots, z_m$. Let f and g be Boolean functions of $z_1, \ldots, z_m$ such that $f = \neg g$ and that both f and g have S-constructible CNF representations w.r.t. π. If $\varphi(x_1, \ldots, x_n)$ is a CNF formula that has an $\mathrm{OBDD}(\wedge, \mathrm{weakening})$ proof of size L, then $\varphi \circ g$ has an $\mathrm{OBDD}(\wedge, \mathrm{weakening})$ proof of size $\mathrm{poly}(|\varphi \circ g|, S, L)$.*

The statement is also true for $\mathrm{OBDD}(\wedge)$, tree-like $\mathrm{OBDD}(\wedge)$, and tree-like $\mathrm{OBDD}(\wedge, \mathrm{weakening})$.

Proof. Let φ have an $\mathrm{OBDD}(\wedge, \mathrm{weakening})$ proof of size L using the order σ on $x_1, \ldots, x_n$. Define the order τ on the variables $z_{i,j}$ as follows. The variables are grouped into blocks, the i-th block is $z_{i,1}, \ldots, z_{i,m}$. The blocks are ordered according to σ so all variables of block i precede those of block j iff x_i precedes x_j according to σ. Within the i-th block, the variables $z_{i,1}, \ldots, z_{i,m}$ are ordered according to the order π. We construct the desired $\mathrm{OBDD}(\wedge, \mathrm{weakening})$ proof using the order τ.

Lemma 3.3 implies that, for any clause C, the CNF $C \circ g$ is $S|C|$-constructible in order τ. Note that we need that both g and $\neg g$ are S-constructible to apply Lemma 3.3, since variables can appear both positively and negatively in C.

Consider the following τ-$\mathrm{OBDD}(\wedge, \mathrm{weakening})$ proof of $\varphi \circ g$: First we create τ-OBDD's that represent the functions $C \circ g$ for each clause C of the formula φ. Then we repeat the $\mathrm{OBDD}(\wedge, \mathrm{weakening})$ proof for φ, but we do it for $\varphi \circ g$. Each a diagram D from the proof of φ is replaced by a diagram for $D \circ g$. It is not hard to see that the definition of τ allows us to replace a splitting over a variable x_i in the diagram D by a subdiagram splitting over the value of the function $g(\vec{z}_i)$, where $\vec{z}_i$ is the vector of the variables $z_{i,1}, \ldots, z_{i,m}$. This increases the proof size by at most a factor of S. The resulting proof is a correct $\mathrm{OBDD}(\wedge, \mathrm{weakening})$ proof and its size is at most $L \cdot |S| + |\varphi \circ g| \cdot S$. □

The clause $\bigvee_{i=1}^{m} y_i$ and the CNF $\bigwedge_{i=1}^{m} \neg y_i$ are both m-constructible, thus we obtain:

Corollary 3.1. *If there is a short $\mathrm{OBDD}(\wedge, \mathrm{weakening})$ proof (tree-like $\mathrm{OBDD}(\wedge)$ proof) of a formula φ, then there is a short $\mathrm{OBDD}(\wedge, \mathrm{weakening})$ proof (tree-like $\mathrm{OBDD}(\wedge)$ proof) of the formula $\varphi^{\vee m}$.*

3.3 Separation

We have shown that if a formula φ is hard for $\mathrm{OBDD}(\wedge, \text{weakening})$ in one order, but is easy for $\mathrm{OBDD}(\wedge, \text{weakening})$ in another, then $\mathcal{T}(\varphi)$ is hard for $\mathrm{OBDD}(\wedge, \text{weakening})$ but it is easy for $\mathrm{OBDD}(\wedge, \text{weakening}, \text{reordering})$. We will prove this holds for φ the Clique-Coloring principle.

Definition 3.1. *The Clique-Coloring principle is a formula encoding the statement that it is impossible that a graph both is $(m-1)$-colorable and has a m-clique. The Clique-Coloring principle uses the variables $\{p_{i,j}\}_{i\neq j\in[n]}$, $\{r_{i,l}\}_{i\in[n],l\in[m-1]}$, and $\{q_{k,i}\}_{k\in[m],i\in[n]}$. Informally $p_{i,j}=1$ if there is an edge between vertices i and j, $r_{i,l}=1$ if vertex i has color l, and $q_{k,i}=1$ if vertex i is the kth vertex in the clique.*

More formally, the Clique-Coloring principle is the conjunction of the following statements written as clauses. For technical reasons we also express the clauses as inequalities with integer coefficients:

1. $\bigvee_{i=1}^{n} q_{k,i}$ $(\sum_{i=1}^{n} q_{k,i} \geq 1)$ *for any $k\in[m]$. This states that the clique has a vertex with number k.*
2. $\neg q_{k,i} \vee \neg q_{k',j} \vee p_{i,j}$ $(q_{k,i}+q_{k',j} \leq p_{i,j}+1)$ *for all $i\neq j\in[n]$ and $k\neq k'\in[m]$. This states that there is an edge between the i-th and j-th vertices of the clique.*
3. $\neg q_{k,i} \vee \neg q_{k,j}$ $(q_{k,i}+q_{k,j}\leq 1)$ *for any $k\in[m]$ and $i\neq j\in[n]$. This states that at most one element in the clique with number k.*
4. $\neg q_{k,i} \vee \neg q_{k',i}$ $(q_{k,i}+q_{k',i}\leq 1)$ *for all $i\in[n]$ and $k\neq k'\in[m]$. This states that the n vertices in clique are distinct.*
5. $\bigvee_{l=1}^{m-1} r_{i,l}$ $(\sum_{l=1}^{m-1} r_{i,l}\geq 1)$ *for all $i\in[n]$. This states that the i-th vertex has a color.*
6. $\neg p_{i,j} \vee \neg r_{i,l} \vee \neg r_{j,l}$ $(p_{i,j}+r_{i,l}+r_{j,l}\leq 2)$ *for all $i\neq j$ and l. This states that if vertices i and j have the same color l, there there is no edge between them.*

$\texttt{Clique-Coloring}_{n,m}$ *denotes the Clique-Coloring principle for n and m. This formula has size polynomially bounded by m and n.*

Note that, usually Clique-Coloring principle is defined without constraints 3 and 4.

We prove the next Theorem in Section 6.

Theorem 3.2. *There is an* $\mathrm{OBDD}(\wedge, \text{weakening})$ *proof of the* $\texttt{Clique-Coloring}_{n,m}$ *principle of size polynomial in n and m.*

An exponential lower bound on the size of proofs of the formula $\texttt{Clique-Coloring}_{n,m}$ has been given by Atserias-Kolaitis-Vardi and by Krajíček.

Theorem 3.3 ([AKV04, Kra08])**.** *There is an order π such that any* $\mathrm{OBDD}(\wedge, \text{weakening})$ *proof of* $\texttt{Clique-Coloring}_{n,\sqrt{n}}$ *has size at least $2^{n^{1/5}}$.*

These two theorems let us separate the $\mathrm{OBDD}(\wedge, \text{weakening}, \text{reordering})$ and $\mathrm{OBDD}(\wedge, \text{weakening})$ proof systems.

Theorem 3.4. *There are a family of CNF formulas φ_n and a constant $c>0$ such that:*

- *φ_n has size* $\mathrm{poly}(n)$*;*
- *there is an* $\mathrm{OBDD}(\wedge, \text{weakening}, \text{reordering})$ *proof of φ_n of size* $\mathrm{poly}(n)$*;*
- *any* $\mathrm{OBDD}(\wedge, \text{weakening})$ *proof of φ_n has size $\Omega(2^{n^c})$.*

Proof. Let us consider $\psi_n = \texttt{Clique-Coloring}_{n,\sqrt{n}}$. By Theorem 3.3 there is an order π such that any π-OBDD($\wedge$, weakening) proof of the formula ψ_n has size at least 2^{n^ϵ}. Since all clauses of $\texttt{Clique-Coloring}_{n,\sqrt{n}}$ that contain a negation have constant width, the CNF encoding of $\texttt{Clique-Coloring}^{\vee m}_{n,\sqrt{n}}$ has size poly(n, m). By Lemma 3.1 any OBDD($\wedge$, weakening) proof of the formula $\mathcal{T}(\psi)$ has size 2^{n^ϵ}. In the definition of $\mathcal{T}(\psi_n)$, we choose m that is polynomially bounded in the number of variables in $\texttt{Clique-Coloring}_{n,\sqrt{n}}$. Hence by Theorem 3.2 and Theorem 3.1 there is an OBDD($\wedge$, weakening) proof of $\psi_n^{\vee m}$ of size polynomial in n. As a result, by Lemma 3.2 there is an OBDD($\wedge$, weakening, reordering) proof of $\mathcal{T}(\psi_n) = \text{perm}(\psi_n^{\vee m})$ of size poly(n, m). Thus, we can use the formula $\mathcal{T}(\psi_n)$ as φ_n. □

4 Quasipolynomial Separations for Dag-like Case

4.1 Resolution Does Not Polynomially Simulate OBDD($\wedge$)

In this section we prove that resolution does not polynomially simulate OBDD($\wedge$). Then we will apply to this result a lifting technique recently developed by Garg et al. [GGKS17] and get as a corollary that Cutting Planes does not polynomially simulate OBDD($\wedge$), and that OBDD($\wedge$, weakening) does not polynomially simulate OBDD($\wedge$, reordering).

A Tseitin formula $\texttt{TS}_{G,c}$ is based on an undirected graph $G(V, E)$ and a labelling function $c : V \to \{0, 1\}$. In this formula for every edge $e \in E$ there is the corresponding propositional variable p_e. For every vertex $v \in V$ we write down a formula in CNF encoding $\sum\limits_{u \in V:(u,v)\in E, u\neq v} p_{(u,v)} \equiv c(v) \pmod 2$. The conjunction of the formulas described above is called a Tseitin formula. If $\sum\limits_{v\in U} c(v) \equiv 1 \pmod 2$ for some connected component $U \subseteq V$, then the Tseitin formula is unsatisfiable. Indeed, if we sum up (modulo 2) all equalities corresponding to the vertices from U we get $0 \equiv 1 \pmod 2$ since each variable has exactly 2 occurrences. If $\sum\limits_{v\in U} c(v) \equiv 0 \pmod 2$ for every connected component U, then the Tseitin formula is satisfiable ([Urq87, Lemma 4.1]).

Tseitin formulas based on constant degree expanders are known to be hard for resolution [Urq87]; Itsykson et al. [IKRS17] showed that they are also hard for OBDD($\wedge$, reordering). Now we consider Tseitin formulas based on the complete graph on $\lfloor \log n \rfloor$ vertices $K_{\log n}$.

By the definition of a Tseitin formula, $\texttt{TS}_{K_{\log_n},c}$ is a system of $\lfloor \log n \rfloor$ linear equations and every equation depends on $\lfloor \log n \rfloor - 1$ variables. Hence, $\texttt{TS}_{K_{\log_n},c}$ is a $(\lfloor \log n \rfloor - 1)$-CNF formula with $O(\log^2 n)$ variables and $O(n \log n)$ clauses.

Lemma 4.1. *Let F be a canonical CNF representation of an unsatisfiable linear system A over $\mathbb{F}_2$ that contains m equations and n variables. Then for every order of variables, F has a tree-like* OBDD($\wedge$) *proof of size at most $8m|F|^2 + mn2^m + 2m$.*

Proof. First of all, for every linear equation of A we deduce an OBDD representing this equation. Assume that a linear equation contains r variables, then its canonical CNF representation contains 2^{r-1} clauses, hence $|F| \geq 2^{r-1}$. We deduce an OBDD representation of the equation by joining all the clauses that represent this equation. The conjunction of several clauses that represent the equation is a Boolean function from r variables, hence it has an OBDD representation of size at most $2^{r+1} + 1$ (this is the size of an OBDD that corresponds to the complete decision tree). Hence, the size of the derivation is at most $8|F|^2$. And the size of the derivation of all OBDDs for all equations is at most $8m|F|^2$.

Finally, we join all OBDDs representing linear equations one by one and we get the constant false OBDD. The size of the described derivation may be estimated using the following claim.

Claim 4.1. *For any order over the variables there is an* OBDD *of size at most $n2^m + 2$ that represents the system of m linear equations with n variables.*

Proof of Claim 4.1. Let us fix some order on the variables. The described OBDD will have n levels. Nodes on the i-th level are labeled with i-th variable in the chosen order.

Assume that we already tested the values of the first $i-1$ variables. For every equation we compute the sum modulo 2 of the values of these $i-1$ variables that occur in the equation. So we will have a vector of m parities. The i-th level of the OBDD contains 2^m nodes corresponding to the all possible values of the vector of parities that we get after the reading of the first $i-1$ edges. Each node on the i-th level has two outgoing edges to nodes on the $(i+1)$-th level corresponding to the way how values of variables change the partial sum. The node on the first level corresponding to all zero values of parities is the source of the OBDD (all nodes that are not reachable from the source should be removed). Outgoing edges for every node on the last level lead to a sink labelled 1 or 0 depending whether or not all the equations are satisfied. □

□

Corollary 4.1. *If $\mathtt{TS}_{K_{\log_n},c}$ is unsatisfiable Tseitin formula, then there is a tree-like* OBDD($\wedge$) *proof of $\mathtt{TS}_{K_{\log_n},c}$ of size at most* poly(n).

Lemma 4.2. *Every resolution proof of $\mathtt{TS}_{K_{\log_n},c}$ has size at least $2^{\Omega(\log^2 n)}$.*

The proof of Lemma 4.2 is based on the width based lower bound by Ben-Sasson and Wigderson [BSW01]. The *width* of a clause is the number of literals in it. For a CNF formula φ, the *width* $w(\varphi)$ *of* φ is the maximum width of its clauses. The *width of a resolution refutation* is a width of the largest used clause. $w(\vdash \varphi)$ denotes the minimum width of any resolution proof of φ.

Theorem 4.1 ([BSW01])**.** *Size of the shortest resolution refutation of any CNF formula φ with n variables is at least $2^{\Omega((w(\vdash\varphi)-w(\varphi))^2/n)}$.*

Theorem 4.2 ([BSW01])**.** *The minimal width of a resolution proof of a Tseitin formula based on a graph $G(V,E)$ is at least $e(G)$, where $e(G)$ is the size of the minimal number of edges between U and $V\setminus U$ over all set of vertices U of size between $|V|/3$ and $2|V|/3$.*

Corollary 4.2. *If $\mathtt{TS}_{K_{\log n},c}$ is an unsatisfiable Tseitin formula, then $w(\vdash \mathtt{TS}_{K_{\log n},c}) = \Omega(\log^2 n)$.*

Proof. It is straightforward that $e(K_{\log n}) = \Omega(\log^2 n)$. So by Theorem 4.2, $w(\vdash \mathtt{TS}_{K_{\log n},c}) = \Omega(\log^2 n)$. □

Proof of Lemma 4.2. It is easy to see that $w(\mathtt{TS}_{K_{\log n},c}) = O(\log n)$ and $\mathtt{TS}_{K_{\log n},f}$ contains $O(\log^2 n)$ variables. Thus by Theorem 4.1 and by Corollary 4.2 the size of the shortest resolution proof of $\mathtt{TS}_{K_{\log n},f}$ is at least $2^{\Omega(\log^2 n)}$. □

Corollary 4.1 and Lemma 4.2 give a superpolynomial separation between resolution and tree-like $\wedge$-OBDD. The next sections describe how to lift this to separate cutting planes and tree-like $\wedge$-OBDD.

4.2 Lifting from Resolution Width

In this subsection we briefly describe the results by Garg et al. [GGKS17] that allows maps formulas with large resolution width to formulas that are hard for several stronger proof systems.

Let $\mathcal{G}$ be a family of functions $\{0,1\}^n \to \{0,1\}$ and φ be an unsatisfiable formula over n variables. The $\mathcal{G}$-refutation of φ is a directed acyclic graph of fan-out at most 2 with each node v labeled by a function $g_v \in \mathcal{G}$ such that the following constraints are satisfied.

Source: There is a distinguished source node r with fan-in 0, and g_r is constant 0 function.

Non-sinks: For each non-sink node v with children u_1 and u_2, we have $g_v^{-1}(0) \subseteq g_{u_1}^{-1}(0) \cup g_{u_2}^{-1}(0)$. And if v has only one child u, then $g_v^{-1}(0) \subseteq g_u^{-1}(0)$.

Sinks: Each sink node v is labeled with a clause C of φ such that $g_v^{-1}(0) \subseteq C^{-1}(0)$ (i.e. every assignment that satisfies C also satisfies g_v).

The size of a $\mathcal{G}$-refutation is the size of the graph.

The notion of $\mathcal{G}$-refutation extends several proof systems including resolution (if functions from $\mathcal{G}$ are represented by clauses), Cutting Planes (if functions from $\mathcal{G}$ are represented by linear inequalities) and OBDD($\wedge$, weakening) (if functions from $\mathcal{G}$ are represented by OBDDs). $\mathcal{G}$-refutations are commonly called "semantic refutations".

Let $\Pi = (X, Y)$ be a partition of $[n]$ into two disjoint parts. We say that $\mathcal{G}$ is Π-rectangular if for every $g \in \mathcal{G}$, the set $g^{-1}(0)$ is a rectangle, i.e. $g^{-1}(0) = A \times B$, where $A \subseteq \{0,1\}^X$ and $B \subseteq \{0,1\}^Y$. We say that $\mathcal{G}$ has Π-communication complexity at most c iff for every $g \in \mathcal{G}$ the communication complexity of g with respect to the partition Π is at most c. Notice that if $\mathcal{G}$ is Π-rectangular, then it has Π-communication complexity at most 2.

Lemma 4.3 ([Sok17])**.** *Let φ be an unsatisfiable CNF formula with n variables and $\Pi = (X, Y)$ be a partition of $[n]$ into two disjoint parts. Assume that π has a $\mathcal{G}$-refutation of size S and $\mathcal{G}$ has Π-communication complexity at most c. Then there is a Π-rectangular set $\mathcal{G}'$ such that φ has a $\mathcal{G}'$-refutation of size at most $2^{3c}S$.*

Notice that the set of all clauses is Π-rectangular for every partition Π. The set of π-OBDDs of size S has Π-communication complexity $\log S + 1$ for partitions $\Pi = (X, Y)$ where the variables of X precede the variables of Y in the order π.

In order to capture Cutting Planes we say that $\mathcal{G}$ is Π-triangular if for every $g \in \mathcal{G}$ there are functions $a : \{0,1\}^X \to \mathbb{R}$ and $b : \{0,1\}^Y \to \mathbb{R}$ such that $g^{-1}(0) = \{x \in \{0,1\}^X, y \in \{0,1\}^y \mid a(x) < b(y)\}$. Note that the set of all linear inequalities with integer coefficients over Boolean variables is Π-triangular for every partition Π.

Let $\mathrm{Ind}_m : \{0,1\}^{\lfloor \log m \rfloor} \times \{0,1\}^m \to \{0,1\}$ be a Boolean function such that $\mathrm{Ind}_m(z_1, \ldots, z_{\lfloor \log m \rfloor}, y_1, \ldots, y_m) = y_b$, where b is the integer with binary representation $z_1 \ldots z_{\lfloor \log m \rfloor}$.

Theorem 4.3 ([GGKS17])**.** *Let φ be an unsatisfiable CNF formula φ with n variables. Let $m = n^\delta$, where δ is some global constant. Let $\Pi = (X, Y)$ be the following partition of variables of $\varphi \circ \mathrm{Ind}_m$: all z-variables go to X, all y-variables go to Y. If $\mathcal{G}$ is Π-rectangular or $\mathcal{G}$ is Π-triangular, then every $\mathcal{G}$-refutation of $\varphi \circ \mathrm{Ind}_m$ has size at least $n^{\Omega(w(\vdash \varphi))}$.*

Corollary 4.3. *Under the conditions of Theorem 4.3, if $\mathcal{G}$ has Π-communication complexity at most c, then every $\mathcal{G}$-refutation of $\varphi \circ \mathrm{Ind}_m$ has size at least $2^{-3c} n^{\Omega(w(\vdash \varphi))}$.*

Proof. By Lemma 4.3, if there is a $\mathcal{G}$-refutation of $\varphi \circ \mathrm{Ind}_m$ of size S, there exists an $\mathcal{G}'$-refutation of $\varphi \circ \mathrm{Ind}_m$ of size at most $2^{3c}S$ such that $\mathcal{G}'$ is Π-rectangular. By Theorem 4.3, $2^{3c}S \geq n^{\Omega(w(\vdash \varphi))}$, hence $S \geq 2^{-3c} n^{\Omega(w(\vdash \varphi))}$. ☐

Corollary 4.4. *Under the conditions of Theorem 4.3 every Cutting Planes proof of $\varphi \circ \mathrm{Ind}_m$ has size at least $n^{\Omega(w(\vdash \varphi))}$.*

Proof. By Theorem 4.3, since the set of linear inequalities is Π-triangular for every partition Π. ☐

4.3 Cutting Planes Does Not Polynomially Simulates OBDD($\wedge$)

Lemma 4.4. *Both functions* Ind_m *and* $\neg\mathrm{Ind}_m$ *have* $\mathrm{poly}(m)$*-constructible CNF representations.*

Proof. Let us consider the following formula for Ind_m,

$$\bigwedge_{i=1}^{m} (\mathrm{bin}(z_1, \ldots, z_{\lfloor \log m \rfloor}) = i) \to y_i,$$

where $\mathrm{bin}(z_1, \ldots, z_{\lfloor \log m \rfloor}) = i$ is the conjunction of literals stating that $z_1, \ldots, z_{\lfloor \log m \rfloor}$ is the binary representation of i. For $\ell \in [m]$, let φ_ℓ be the formula $\bigwedge_{i=1}^{\ell} (\mathrm{bin}(z_1, \ldots, z_{\lfloor \log m \rfloor}) = i) \to y_i$, and let $\varphi_m = \mathrm{Ind}_m$.

We claim that for all $\ell \in [m]$ the formula φ_ℓ has an OBDD representation of size $\mathrm{poly}(m)$ in the order $z_1, \ldots, z_{\lfloor \log m \rfloor}, y_1, \ldots, y_m$. Indeed, such OBDD has the following structure: it starts with the complete decision tree over all variables z_i; consider a leaf of this decision tree that corresponds to a number i. If $i \le \ell$, then we add to this leaf a node of OBDD labeled with y_i and the outgoing edge labeled with 0 going to the 0-sink and the outgoing edge labeled with 1 going to the 1-sink. If $i > \ell$, then we identify this leaf with 1-sink. Hence, there is a $\mathrm{poly}(m)$-constractible CNF representation of Ind_m.

The same argument works also for $\neg\mathrm{Ind}_m$, since $\neg\mathrm{Ind}_m(z_1, \ldots, z_{\lfloor \log m \rfloor}, y_1, y_2, \ldots, y_m) = \mathrm{Ind}_m(z_1, \ldots, z_{\lfloor \log m \rfloor}, \neg y_1, \neg y_2, \ldots, \neg y_m)$. ☐

Lemma 4.5. *The formula $\mathtt{TS}_{K_{\log_n}, c} \circ \mathrm{Ind}_m$ has at most $m^{O(\log n)}$ clauses of size $O(\log n \log m)$ and $O(m \log^2 n)$ variables.*

Proof. Note that each clause of $\mathtt{TS}_{K_{\log_n}, c}$ consists of $\lceil \log n \rceil - 1$ literals and by Lemma 4.4 there is CNF representations of Ind_m and $\neg\mathrm{Ind}_m$ with m clauses. Hence, for each clause C of $\mathtt{TS}_{K_{\log_n}, c}$, the formula $C \circ \mathrm{Ind}_m$ has $m^{\lceil \log n \rceil - 1}$ clauses each of length $(\lceil \log n \rceil - 1)(\lfloor \log m \rfloor + 1)$. ☐

Theorem 4.4. *Let $\mathtt{TS}_{K_{\log_n}, c}$ be unsatisfiable Tseitin formula based on a complete graph $K_{\log n}$ on $\lfloor \log n \rfloor$ vertices.*

Let $m = (\log n)^{2\delta}$, where δ is the constant from Theorem 4.3. Then

1. *$\mathtt{TS}_{K_{\log_n}, c} \circ \mathrm{Ind}_m$ has a tree-like $\mathrm{OBDD}(\wedge)$ proof of size $(\log n)^{O(\log n)}$ and*
2. *every Cutting Planes proof of $\mathtt{TS}_{K_{\log_n}, c} \circ \mathrm{Ind}_m$ has size at least $(\log n)^{\Omega(\log^2 n)}$.*

Proof.

1. By Lemma 4.4 both Ind_m and $\neg\mathrm{Ind}_m$ are $\mathrm{poly}(m)$-constructible. By Corollary 4.1 there is a tree-like $\mathrm{OBDD}(\wedge)$ refutation of $\mathtt{TS}_{K_{\log_n}, c}$ of size $\mathrm{poly}(n)$. By Lemma 4.5 the size of the formula $\mathtt{TS}_{K_{\log_n}, c} \circ \mathrm{Ind}_m$ is at most $m^{O(\log n)}$. Hence by Theorem 3.1 there is a tree-like $\mathrm{OBDD}(\wedge)$ refutation of $\mathtt{TS}_{K_{\log_n}, c} \circ \mathrm{Ind}_m$ of size $\mathrm{poly}(\mathrm{poly}(n), m^{O(\log n)}, \mathrm{poly}(n)) = (\log n)^{O(\log n)}$.
2. By Corollary 4.2, $w(\vdash \mathtt{TS}_{K_{\log n}, c}) = \Omega(\log^2 n)$. Hence, by Corollary 4.4, every Cutting Planes proof of $\mathtt{TS}_{K_{\log n}, c} \circ \mathrm{Ind}_m$ has size at least $(\log^2 n)^{\Omega(\log^2 n)} = (\log n)^{\Omega(\log^2 n)}$.

☐

4.4 $\mathrm{OBDD}(\wedge, \mathrm{weakening})$ Does Not Polynomially Simulate $\mathrm{OBDD}(\wedge, \mathrm{reordering})$

Theorem 4.5. *There is a family of formulas φ_n such that:*

- *the size of φ_n is $(\log n)^{O(\log n \log\log n)}$ and the number of variables in φ_n is $\mathrm{poly}(\log n)$;*
- *there is a tree-like $\mathrm{OBDD}(\wedge, \mathrm{reordering})$ proof of φ_n of size $(\log n)^{O(\log n \log\log n)}$;*
- *every $\mathrm{OBDD}(\wedge, \mathrm{weakening})$ proof of φ_n has size at least $(\log n)^{\Omega(\log^2 n)}$.*

Lemma 4.6. *Let $\mathtt{TS}_{K_{\log n}, c}$ be an unsatisfiable Tseitin formula. Let $m = (\log n)^{2\delta}$, where δ is the constant from Theorem 4.3.*

There is a family of orders $\{\pi_n\}_{n \in \mathbb{N}}$ over the variables of the formulas $\mathtt{TS}_{K_{\log n}, c} \circ \mathrm{Ind}_m$ such that every π_n-$\mathrm{OBDD}(\wedge, \mathrm{weakening})$ proof of $\mathtt{TS}_{K_{\log n}, c} \circ \mathrm{Ind}_m$ has size at least $(\log n)^{\Omega(\log^2 n)}$.

Proof. Let π_n be an order on variables of $\mathtt{TS}_{K_{\log n}, c} \circ \mathrm{Ind}_m$, where all z-variables precedes all y-variables. Consider some π-$\mathrm{OBDD}(\wedge, \mathrm{weakening})$ proof of $\mathtt{TS}_{K_{\log n}, c} \circ \mathrm{Ind}_m$; let S denote its total size. Hence, the number of proof lines and sizes of all OBDDs are at most S. Consider a partition $\Pi = (X, Y)$ of the variables of $\mathtt{TS}_{K_{\log n}, c} \circ \mathrm{Ind}_m$ such that X contains all z-variables and Y contains all y-variables. The communication complexity of computing an OBDD of size S w.r.t. the partition Π is at most $\log S + 1$. Therefore, the π-$\mathrm{OBDD}(\wedge, \mathrm{weakening})$ proof can be viewed as a $\mathcal{G}$-refutation, where $\mathcal{G}$ has Π-communication complexity at most $\log S + 1$. Hence, by Corollary 4.3, $S \ge 2^{-3 \log S - 3} (\log^2 n)^{\Omega(\log^2 n)}$. Thus, $S \ge (\log^2 n)^{\Omega(\log^2 n)} = (\log n)^{\Omega(\log^2 n)}$. ☐

Proof of Theorem 4.5. Let $\mathtt{TS}_{K_{\log n},c}$ be an unsatisfiable Tseitin formula. Let $m = (\log n)^{2\delta}$, where δ is the constant from Theorem 4.3.

Let us consider $\varphi_n = \mathcal{T}(\mathtt{TS}_{K_{\log n},c} \circ \mathrm{Ind}_m)$, where $\mathcal{T}$ is the transformation defined in Section 3.1. By Corollary 4.1 there is a tree-like OBDD($\wedge$) proof of $\mathtt{TS}_{K_{\log n},c}$ of size $\mathrm{poly}(n)$. By Lemma 4.5, $\mathtt{TS}_{K_{\log n},c} \circ \mathrm{Ind}_m$ has $\log n^{O(\log n)}$ clauses of size $O(\log n \log\log n)$ and $\mathrm{poly}(\log n)$ variables By Lemma 4.4, Ind_m is $\mathrm{poly}(m)$-constructible; hence, by Theorem 3.1 there is a tree-like OBDD($\wedge$) proof of $\mathtt{TS}_{K_{\log n},c} \circ \mathrm{Ind}_m$ of size $\log n^{O(\log n)}$.

Recall that $\varphi_n = \mathcal{T}(\mathtt{TS}_{K_{\log n},c} \circ \mathrm{Ind}_m) = \mathrm{perm}((\mathtt{TS}_{K_{\log n},c} \circ \mathrm{Ind}_m)^{\vee_k})$, where $k = \mathrm{poly}(\log n)$.

The formula $(\mathtt{TS}_{K_{\log n},c} \circ \mathrm{Ind}_m)^{\vee_k}$ has size $(\log n)^{O(\log n \log\log n)}$; by Theorem 3.1 there is a tree-like OBDD($\wedge$) proof of $(\mathtt{TS}_{K_{\log n},c} \circ \mathrm{Ind}_m)^{\vee_k}$ of size $(\log n)^{O(\log n \log\log n)}$.

Thus, by Lemma 3.2, there is a tree-like OBDD($\wedge$, reordering) proof of $\mathcal{T}(\mathtt{TS}_{K_{\log n},c} \circ \mathrm{Ind}_m)$ of size $(\log n)^{O(\log n \log\log n)}$.

Note that, by Lemma 4.6 and Lemma 3.1, every OBDD($\wedge$, weakening) proof of $\mathcal{T}(\mathtt{TS}_{K_{\log n},c} \circ \mathrm{Ind}_m)$ has size at least $(\log^2 n)^{\Omega(\log^2 n)} = (\log n)^{\Omega(\log^2 n)}$. $\square$

5 Exponential Separations for Tree-like Case

In this section we exhibit a formula which is hard for tree-like OBDD($\wedge$, weakening) and easy for tree-like OBDD($\wedge$, reordering) in another order. An example of such a formula can be obtained from a construction of Göös and Pitassi [GP14]. We use a pebbling contradiction as the base of our example.

Definition 5.1. *Let G be a directed acyclic graph with one sink t. The CNF formula $\mathtt{Peb}_G$ (*pebbling contradiction *for a graph G), uses a variable x_v for each vertex v of G and has the following clauses:*

- $\neg x_t$;
- *for each vertex v, the clause $x_v \vee \bigvee_{i=1}^{d} \neg x_{p_i}$ where $p_1, \dots, p_d$ are all the immediate predecessors of v ($d = 0$ if v is a source).*

It is not hard to see that $\mathtt{Peb}_G$ has short tree-like OBDD($\wedge$) proofs.

Theorem 5.1. *For any directed acyclic graph $G(V, E)$ with n vertices and maximum in-degree d there is a tree-like* OBDD($\wedge$) *proof of $\mathtt{Peb}_G$ of size* $\mathrm{poly}(n)$.

Proof. For a vertex $v \in V$, we let $p_{v,1}, \dots, p_{v,l_v}$ be the immediate predecessors of v. For any set $S \subseteq V$ such that if $v \in S$, then $p_{v,1}, \dots, p_{v,l_v}$ are also in S (we call such a set closed under predecessors), the formula $\bigwedge_{v\in S}\left(x_v \vee \bigvee_{i=1}^{l_v} \neg x_{p_{v,i}}\right)$ is equivalent to $\bigwedge_{v\in S} x_v$. Thus $\bigwedge_{v\in S}\left(x_v \vee \bigvee_{i=1}^{l_v} \neg x_{p_{v,i}}\right)$ has an OBDD representation of size $\mathrm{poly}(n, d)$.

Let $v_1, \dots, v_n$ be a topological ordering of vertices of the graph G. Consider an order π and a sequence $D_1, \dots, D_{n+1}$ of π-OBDD's such that D_i represents the formula $\bigwedge_{j=1}^{i}\left(x_{v_i} \vee \bigvee_{k=1}^{l_{v_i}} \neg x_{p_{v_i,k}}\right)$ for all $1 \le i \le n$ and D_{n+1} is the constant false diagram. We claim that, together with π-OBDD's representing the initial clauses, $D_1, \dots, D_{n+1}$ is an OBDD($\wedge$) refutation of $\mathtt{Peb}_G$ of total size $O(n^2)$. Indeed, since for all $i \in [n]$ the set $\{v_1, v_2, \dots, v_i\}$ is closed under predecessors, then $D_i = \bigwedge_{j=1}^{i} x_{v_i}$ has size $2i + 2$. It is easy to see that D_{i+1} is equal to $D_i \wedge \left(x_{v_{i+1}} \vee \bigvee_{i=1}^{l_{v_{i+1}}} \neg x_{p_{v_{i+1},i}}\right)$. $\square$

Corollary 5.1 (Lemma 2, [Jär11])**.** *For any directed acyclic graph $G(V, E)$ with n vertices and maximum in-degree d there is a tree-like* OBDD($\wedge$) *proof of $\mathtt{Peb}_G^{\vee_2}$ of size* $\mathrm{poly}(n, 2^d)$.

Proof. Since $\texttt{Peb}_G$ is a formula in $(d+1)$-CNF, size of the formula $\texttt{Peb}_G^{\vee_2}$ is at most $O(|\texttt{Peb}_G|2^d)$. The Corollary follows from Theorem 5.1 and Theorem 3.1. $\square$

Corollary 5.1 was presented earlier as [Jär11, Lemma 2], however, there was a flaw in previous proof. The proof of [Jär11, Lemma 2] was based on the following statement ([Jär11, Lemma 1]): Let G be a dag on n nodes, and j be a node in G with parents $i_1, \ldots, i_k$ where $k = O(\log n)$. Consider the clauses $(x_{i_1,0} \vee x_{i_1,1}), \ldots, (x_{i_k,0} \vee x_{i_k,1})$ and $(\neg x_{i_1,a_1} \vee \cdots \vee \neg x_{i_k,a_k} \vee x_{j,0} \vee x_{j,1})$ for all $(a_1, \ldots, a_k) \in \{0,1\}^k$. For any variable order π, there is a polynomial-size π-OBDD($\wedge$) derivation of $x_{j,0} \vee x_{j,1}$ from these clauses. However, [Jär11, Lemma 1] is incorrect, for example for $k = 1$ it claims that it is possible to derive $(a \vee b)$ from $A = \{(\neg x \vee a \vee b), (\neg y \vee a \vee b), (x \vee y)\}$ in OBDD($\wedge$). Assume that $(a \vee b)$ is the conjunction of clauses from $B \subseteq A$. Notice that $(x \vee y) \notin B$, since otherwise it would be possible to satisfy $(a \vee b)$ by substitution $x := 0, y := 0$. It is easy to see that B can not be empty, hence B is non empty subset of $\{(\neg x \vee a \vee b), (\neg y \vee a \vee b)\}$. In this case it should be possible to satisfy $a \vee b$ by substitution $x := 0, y := 0$. Thus, [Jär11, Lemma 1] is incorrect.

Järvisalo [Jär11] used Corollary 5.1 in order to give a family of formulas that are easy for OBDD($\wedge$) but hard for tree-like Resolution. The lower bound was proved by Buresh-Oppenheim and Pitassi [BOP07], namely Buresh-Oppenheim and Pitassi who proved that there is a family of graphs $\{G_n\}_{n\in\mathbb{N}}$ with n vertices and maximum in-degree 2 such that any tree-like resolution proof of $\varphi_n = \texttt{Peb}_{G_n}^{\vee_2}$ has size at least $2^{\Omega(n/\log(n))}$.

Let $\varphi(x_1, \ldots, x_n, y_1, \ldots, y_n) = \bigwedge_{i=1}^{m} C_i(x_1, \ldots, x_n, y_1, \ldots, y_n)$. The relation $\text{Search}_\varphi \subseteq \{0,1\}^n \times \{0,1\}^n \times [m]$ is defined by

$$(x, y, i) \in \text{Search}_\varphi \text{ iff } C_i(x_1, \ldots, x_n, y_1, \ldots, y_n) = 0.$$

Consider the following communication game: Alice knows values of variables $x_1, x_2, \ldots, x_n$ and Bob knows variables $y_1, y_2, \ldots, y_n$. The goal of the communication game is to compute some $i \in [m]$ such that $(x_1, \ldots, x_n, y_1, \ldots, y_n, i) \in \text{Search}_\varphi$.

Göös and Pitassi [GP14] proved the following theorem:

Theorem 5.2 ([GP14])**.** *There are a family of directed acyclic graphs $\{G_n\}_{n\in\mathbb{N}}$ with constant degree such that G_n has n vertices, and a CNF formula g on variables x_1, x_2, y_1, y_2 such that deterministic communication complexity of $\text{Search}_{\texttt{Peb}_{G_n} \circ g}$ is at least $\Omega(\sqrt{n})$ if Alice knows variables $\{x_{1,1}, x_{1,2}, \ldots, x_{n,1}, x_{n,2}\}$ and Bob knows variables $\{y_{1,1}, y_{1,2}, \ldots, y_{n,1}, y_{n,2}\}$.*

In fact Theorem 5.2 is true even for randomized communication complexity, but the deterministic version is enough for our applications.

Lemma 5.1. *Let a function f be computed by a π-OBDD D, the communication complexity of f under a partition Π_0, Π_1 of the variables where the variables in Π_0 precede (in the sense of π) the variables from Π_1 is at most $\lceil \log |D| \rceil + 1$.*

Proof. Alice starts the computation of f according D using her variables. Finally Alice reaches vertex v of D reading all her variables. Alice sends to Bob number of the vertex v, it has at most $\lceil \log |D| \rceil$ bits. Bob continues computing f starting from v using his variables and sends the result of the computation (it is 1 bit) to Alice. $\square$

Theorem 5.3. *Let $\varphi(x_1, \ldots, x_n, y_1, \ldots, y_n)$ be an unsatisfiable CNF formula. Suppose the the communication complexity of the relation $Search_\varphi$ is equal to t if Alice knows the values of variables x_i and Bob knows the variables y_i. Let π be an ordering of the variables of φ such that variables x_i precede variables y_i. Then the size of any tree-like* π-OBDD($\wedge$, weakening) *refutation of φ is at least $2^{O(\sqrt{t})}$.*

Proof. Consider a tree-like π-OBDD($\wedge$, weakening) proof $D_1, \ldots, D_\ell$ of the formula φ of size S. Based on this proof we construct a communication protocol for Search_φ of complexity at most $O(\log^2 S)$. The protocol consists of $\ell = O(\log S)$ steps. At each step we consider some tree T_i that is known by both players. The inner vertices of the tree are labelled with π-OBDD's and the leaves are labelled with clauses of φ or with trivially satisfied clauses. In the first step, the tree T_1 is the tree of our tree-like proof. $T_i \subseteq T_{i-1}$. At each step, the two players know that the clause at the root of T_i is falsified by the input assignment, and that there exists some clause at a leaf of T_i that is falsified. In the end, the tree T_ℓ consists of a single vertex; hence it provides clause of φ. that is falsified by the input assignment.

Now we describe how we obtain the tree T_{i+1} from the tree T_i. Let v be a vertex of tree T_i such that a subtree T' with root v satisfies the following condition: $\frac{1}{3}|T_i| \le |T'| \le \frac{2}{3}|T_i|$ (such a vertex v players can find without communication). Let D be the OBDD labelling v; if the input assignment evaluates diagram D to zero, then T_{i+1} equals T'. (The players can evaluate the π-OBDD D on the input assignment with at most $\lceil \log |D| + 1 \rceil \le 2 \log S$ bits of communication by Lemma 5.1). Otherwise, $T_{i+1} := T_i \setminus T'$.

It is easy to see that if the value of D equals zero then there is a leaf with falsified clause in the tree T'. Otherwise there is a leaf with falsified clause in the tree $T_i \setminus T'$. Also, at each step the players use at most $2\log(S)$ bits of communication and there are at most $O(\log(S))$ steps (since $|T_i| \le \frac{2}{3}|T_{i+1}|$). Hence, the players use at most $O(\log^2 S)$ bits of communication. Therefore $S = 2^{\Omega(\sqrt{t})}$. $\square$

As a result we obtain the following separation.

Theorem 5.4. *There are a family of formulas φ_n in CNF and a constant $c > 0$ such that:*

- *size of φ_n and number of variables in φ_n are polynomially bounded by n;*
- *there is a tree-like* $\mathrm{OBDD}(\wedge, \mathrm{reordering})$ *proof of φ_n of size polynomial in n;*
- *any tree-like* $\mathrm{OBDD}(\wedge, \mathrm{weakening})$ *proof of φ_n has size at least $2^{\Omega(n^{1/4})}$.*

Proof. Let g be a CNF formula on the variables x_1, x_2, y_1, y_2 and let $\{G_n\}_{n\in\mathbb{N}}$ be a family of graphs so that Theorem 5.2 holds. Consider the formula $\psi_n = \mathtt{Peb}_{G_n} \circ g$. By Theorem 5.2 and Theorem 5.3 there exist an order π such that the size of every tree-like π-$\mathrm{OBDD}(\wedge, \mathrm{weakening})$ refutation of ψ_n has size at least $2^{O(n^{1/4})}$. By Lemma 3.1 any tree-like $\mathrm{OBDD}(\wedge, \mathrm{weakening})$ proof of the formula $\varphi_n := \mathcal{T}(\psi_n)$ has size $2^{\Omega(n^{1/4})}$.

By Theorems 5.1 and 3.1, ψ_n has a tree-like $\mathrm{OBDD}(\wedge)$ proof of size $\mathrm{poly}(n)$. Then, by Lemma 3.2, there is a $\mathrm{OBDD}(\wedge, \mathrm{reordering})$ proof of $\mathcal{T}(\psi_n)$ of size $\mathrm{poly}(n)$. $\square$

6 Clique-Coloring is Easy for $\mathrm{OBDD}(\wedge, \mathrm{weakening})$

In this section we prove Theorem 3.2. Let π be the following order on the variables of $\texttt{Clique-Coloring}_{n,m}$:

$$p_{1,1}, \ldots, p_{n,n}, q_{1,1}, \ldots, q_{m,1}, r_{1,1}, \ldots, r_{1,m}, q_{1,2} \ldots, q_{m,2}, r_{2,1}, \ldots, r_{2,m}, \ldots, q_{1,n}, \ldots, q_{m,n}, r_{n,1} \ldots, r_{n,m}. \quad (1)$$

This order places at the beginning the variables encoding a graph, after them variables encoding the number of the first vertex in clique, after them variables encoding the color of the first vertex and so on. All OBDD's used in this section are π-OBDD's.

Lemma 6.1. *For any integer constants c, c_q, c_r, and sets $I \subseteq [n]$, $K \subseteq [m]$, and $L \subseteq [m-1]$ the inequality*

$$\sum_{i\in I}\Big(\sum_{k\in K} q_{k,i} - c_q\Big)\Big(\sum_{l\in L} r_{i,l} - c_r\Big) \ge c \quad (2)$$

has an π-OBDD representation of size polynomial in c_r, c_q, m, and n.

Proof. The order π was picked to make it convenient to evaluate the left hand side of (2) with a π-OBDD. The OBDD is constructed in levels, one level per variable. Each level has vertices corresponding to the values of partial sums used to compute the left hand side of (2). Specifically, let $Q_{i,k} = \sum\limits_{k'\in K, k'\le k}(q_{k',i} - c_q)$, let $R_{i,l} = \sum\limits_{l'\in L, l'\le l}(r_{i,l'} - c_r)$, and let $S_i = \sum\limits_{i'\in I, i'<i} Q_{i,m+1}R_{i,m}$. Note $S_{1+\max(I)}$ equals the left hand side of (2).

The vertices of the OBDD at the level corresponding to a variable $q_{k,i}$ encode the values of S_i and $Q_{i,k}$. The vertices at the level corresponding to a variable $r_{i,l}$ encode the values of S_i, $Q_{i,m+1}$, and $R_{i,l}$. The number of possible values at each level is polynomially bounded by c_r, c_q, m, n. To finalize the π-OBDD for evaluating (2), the vertices in the final level that correspond to a value $\geq c$ are sinks labeled with 1, and the remaining vertices in the final level are sinks with label 0. $\square$

Proof of Theorem 3.2. The idea of the proof is to first derive an π-OBDD which represents the inequality $\sum_{k,i,l} q_{k,i}r_{i,l} \geq m$, stating that every vertex of clique is colored, and second to derive an π-OBDD which represents the inequality $\sum_{k,i,l} q_{k,i}r_{i,l} \leq m-1$ stating roughly that there is at most one vertex per color. Combining this these with conjunction derives a contradiction.

1. We first describe the derivation of the OBDD representing $\sum_{k,i,l} q_{k,i}r_{i,l} \geq m$. For $i \in [n]$, the derivation starts with an OBDD representing the inequality $\sum_{l=1}^{m-1} r_{i,l} \geq 1$; note that $\texttt{Clique-Coloring}_{n,m}$ has such a clause. For each $k \in m$, using the weakening rule (in fact multiplying the inequality by $q_{k,i}$) gives an OBDD that represents the inequality
$$\sum_{l=1}^{m-1} q_{k,i}r_{i,l} \geq q_{k,i}. \tag{3}$$
Since this is equivalent to $q_{k,i} \sum_{l=1}^{m-1}(r_{i,l}-1) \geq 0$, Lemma 6.1 implies that the OBDD representing (3) has polynomial size. Summing the inequalities (3) for all $i \in [n]$ gives
$$\sum_{i=1}^{n}\sum_{l=1}^{m-1} q_{k,i}r_{i,l} \geq \sum_{i=1}^{n} q_{k,i}. \tag{4}$$
To derive an OBDD representation of the inequality (4) for a fixed value of k, we add the inequalities (3) for $i \in [n]$ one by one. The addition of two inequalities may be expressed by a conjunction followed by a weakening rule. The intermediate inequalities can be expressed as $\sum_{i=1}^{u} q_{k,i} \sum_{l=1}^{m-1}(r_{i,l}-1) \geq 0$; hence by Lemma 6.1, they have OBDD representations of size $\mathrm{poly}(n,m)$. This allows the derivation of polynomial size OBDD's representing (4) for each k.

   The inequality $\sum_{i=1}^{n} q_{k,i} \geq 1$ is expressed by a clause of $\texttt{Clique-Coloring}_{n,m}$; combining this with the inequality (4) using the conjunction and weakening rules gives an OBDD representing
$$\sum_{i=1}^{n}\sum_{l=1}^{m-1} q_{k,i}r_{i,l} \geq 1. \tag{5}$$
The size of an OBDD representation of (5) is polynomially bounded, again by Lemma 6.1.

   Finally, to get the desired inequality $\sum_{k,i,l} q_{k,i}r_{i,l} \geq m$ we sum the inequalities (5) for all $k \in [m]$. As in the previous cases, we do this iteratively, combining the inequalities (5) one by one with the conjunction and weakening rules. The intermediate OBDD's are $\sum_{k<u}\sum_{i,l} q_{k,i}r_{i,l} \geq u$ and are polynomially bounded by Lemma 6.1.

2. The second part derives an OBDD representation of the inequality $\sum_{k,i,l} q_{k,i}r_{i,l} \leq m-1$. If we derive
$$\sum_{k=1}^{m}\sum_{i=1}^{n} q_{k,i}r_{i,l} \leq 1 \tag{6}$$

for each $l \in [m-1]$ and sum them as we do earlier we get the desired inequality. All intermediate inequalities have small OBDD representations by Lemma 6.1.

For each l, the inequality (6) will be derived from the inequalities (7) and (10) as described below. For $k \in [m]$, we derive (an OBDD representing) the inequality (7)

$$\sum_{i=1}^{n} q_{k,i} r_{i,l} \leq 1. \tag{7}$$

stating that there is at most one vertex with number k in clique which has color l. The inequality (7) follows by weakening from the inequality

$$\sum_{i=1}^{n} q_{k,i} \leq 1. \tag{8}$$

To derive (8), we derive inequalities $\sum\limits_{i=1}^{u} q_{k,i} \leq 1$ for all $u \in [n]$. For $u = n$ this inequality is the same as (8). For $u = 1$ this inequality is the constant true statement. For $u+1$ it is a weakening of the conjunction of $\sum\limits_{i=1}^{u} q_{k,i} \leq 1$ and

$$\bigwedge_{i=1}^{u} (q_{k,i} + q_{k,u+1} \leq 1). \tag{9}$$

Each inequality $q_{k,i} + q_{k,u+1} \leq 1$ is a clause of $\texttt{Clique-Coloring}_{n,m}$ but we need to check that their u-fold conjunctions (9) have polynomial size OBDD derivations. For this, we iteratively derive $\bigwedge\limits_{i=1}^{t} (q_{k,i} + q_{k,u+1} \leq 1)$ for all $t \in [u]$. For each t, this inequality has a small OBDD representation since it is equivalent to $\left(\bigvee\limits_{i=1}^{t} q_{k,i}\right) \to \neg q_{k,u+1}$; the latter clearly has a polynomial size OBDD representation. Thus there are short refutations of constraints (9) and as a result, of inequalities (8) and (7).

To derive (6), we also need

$$\sum_{i=1}^{n} q_{k,i} r_{i,l} + \sum_{i=1}^{n} q_{k',i} r_{i,l} \leq 1 \tag{10}$$

for all $k \neq k' \in [m]$. Before deriving inequality (10) we show how to derive (6) from (7) and (10). This derivation is similar to derivation of (8) but it is slightly more complicated to show that all intermediate inequalities have polynomial size OBDD representations.

To derive (6), we derive successively the inequalities

$$\sum_{k=1}^{u} \sum_{i=1}^{n} q_{k,i} r_{i,l} \leq 1. \tag{11}$$

for all $u \in [n]$. Each inequality (11) has a polynomial size OBDD representation by Lemma 6.1. For $u = 1$, (11) is the same as (7). Let us show how to derive inequality (11) for $u+1$ from the inequality (11) for u. For this, it suffices to derive the inequality

$$\bigwedge_{k=1}^{u} \left(\sum_{i=1}^{n} q_{k,i} r_{i,l} + \sum_{i=1}^{n} q_{u+1,i} r_{i,l} \leq 1 \right) \tag{12}$$

and then use the conjunction and weakening rules. Each inequality from the conjunction is an instance of inequality (10). We must show the conjunction (12) has a small derivation. To derive (12), we iteratively derive $\bigwedge_{k=1}^{t}\left(\sum_{i=1}^{n} q_{k,i}r_{i,l} + \sum_{i=1}^{n} q_{u+1,i}r_{i,l} \leq 1\right)$ for all $t \in [u]$. This conjunction is equal to $\bigvee_{k=1}^{t}\bigvee_{i=1}^{n} q_{k,i} \wedge r_{i,l} \to \neg \bigvee_{i=1}^{n} q_{u+1,i} \wedge r_{i,l}$. Hence it has a small OBDD representation by the choice of π.

We conclude the proof of Theorem 3.2 by proving the inequality (10) for k and k'. For this we will first derive the inequalities

$$\sum_{i=1}^{t} q_{k,i}r_{i,l} = 0 \vee \sum_{i=1}^{t} q_{k',i}r_{i,l} = 0 \vee \\ \bigvee_{i\in[t]}\Big(q_{k,i}r_{i,l} = q_{k',i}r_{i,l} = 1 \wedge \bigwedge_{j\in[n]\setminus\{i\}} (q_{k,j}r_{j,l} = q_{k',j}r_{j,l} = 0)\Big) \quad (13)$$

for all $t \in [n]$. The inequality (13) for $t = n$ and the conjunction $\bigwedge_{i=1}^{n} \neg q_{k,i} \vee \neg q_{k',i}$ implies

$$\sum_{i=1}^{n} q_{k,i}r_{i,l} = 0 \vee \sum_{i=1}^{n} q_{k',i}r_{i,l} = 0. \quad (14)$$

Each clause in the conjunction $\bigwedge_{i=1}^{n} \neg q_{k,i} \vee \neg q_{k',i}$ is a clause of $\texttt{Clique-Coloring}_{n,m}$. The conjunction derived iteratively using the conjunction and weakening rules; all intermediate constraints have polynomial sized π-OBDD representations since π orders the variables $q_{k,i}$ first by i and second by k.

The constraint (14) and the two inequalities (7) for k, l and for k', l imply (10).

The constraint (13) is derived from the inequalities

$$q_{k,i}r_{i,l} + q_{k',j}r_{j,l} \leq 1 \quad (15)$$

for $i \neq j \in [n]$.

The inequality (13) is equivalent to the conjunction of inequalities (15) for all $i \neq j \in [t]$, and it is clear that these have polynomial size π-OBDD representations. We show there is a small OBDD derivation of this conjunction, that is, of (13), by deriving it for successive values of t. For $t = 0$, (13) the constant true statement. We claim there is a short derivation of (13) for $t = u + 1$ from (13) for $t = u$. Indeed, (15) together with (13) for $t = u$ implies $\bigwedge_{i=1}^{u} (q_{k,i}r_{i,l} + q_{k',u+1}r_{u+1,l} \leq 1)$. It is easy to see that this latter inequality has a small OBDD representation since it is equivalent to the constraint $\left(\bigvee_{i=1}^{u} q_{k,i}r_{i,l} = 1\right) \to q_{k',u+1}r_{u+1,l} = 0$.

Now the only thing left to derive is the inequality (15). $\texttt{Clique-Coloring}_{n,m}$ contains the clauses $\neg q_{k,i} \vee \neg q_{k',j} \vee p_{i,j}$ and $\neg p_{i,j} \vee \neg r_{i,l} \vee \neg r_{j,l}$. From these, we can derive (15) using the conjunction rule and the weakening rules.

□

Acknowledgements. The research was supported by Russian Science Foundation (project 16-11-10123).

References

[AKV04] Albert Atserias, Phokion G. Kolaitis, and Moshe Y. Vardi. Constraint propagation as a proof system. In Mark Wallace, editor, *Principles and Practice of Constraint Programming - CP 2004*, volume 3258 of *Lecture Notes in Computer Science*, pages 77–91. Springer, 2004.

[BCM+92] Jerry R. Burch, Edmund M. Clarke, Kenneth L. McMillan, David L. Dill, and L. J. Hwang. Symbolic Model Checking: 10ˆ20 States and Beyond. *Information and Computation*, 98(2):142–170, 1992.

[BOP07] Joshua Buresh-Oppenheim and Toniann Pitassi. The complexity of resolution refinements. *Journal of Symbolic Logic*, 72(4):1336–1352, 2007.

[Bry85] Randal E. Bryant. Symbolic manipulation of Boolean functions using a graphical representation. In Hillel Ofek and Lawrence A O'Neill, editors, *Proceedings of the 22nd ACM/IEEE conference on Design automation, DAC 1985, Las Vegas, Nevada, USA, 1985.*, pages 688–694. ACM, 1985.

[BSW01] Eli Ben-Sasson and Avi Wigderson. Short proofs are narrow - resolution made simple. *Journal of the ACM*, 48(2):149–169, 2001.

[CZ09] Wěi Chén and Wenhui Zhang. A direct construction of polynomial-size OBDD proof of pigeon hole problem. *Information Processing Letters*, 109(10):472–477, 2009.

[GGKS17] Ankit Garg, Mika Göös, Pritish Kamath, and Dmitry Sokolov. Monotone circuit lower bounds from resolution. *Electronic Colloquium on Computational Complexity (ECCC)*, 24:175, 2017.

[GHP02] Dima Grigoriev, Edward A. Hirsch, and Dmitrii V. Pasechnik. Complexity of semi-algebraic proofs. In Helmut Alt and Afonso Ferreira, editors, *STACS 2002, 19th Annual Symposium on Theoretical Aspects of Computer Science, Antibes - Juan les Pins, France, March 14-16, 2002, Proceedings*, volume 2285 of *Lecture Notes in Computer Science*, pages 419–430. Springer, 2002.

[GP14] Mika Göös and Toniann Pitassi. Communication lower bounds via critical block sensitivity. In *Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 - June 03, 2014*, pages 847–856, 2014.

[GZ03] Jan Friso Groote and Hans Zantema. Resolution and binary decision diagrams cannot simulate each other polynomially. *Discrete Applied Mathematics*, 130(2):157–171, 2003.

[IKRS17] Dmitry Itsykson, Alexander Knop, Andrei Romashchenko, and Dmitry Sokolov. On OBDD-based algorithms and proof systems that dynamically change order of variables. In Heribert Vollmer and Brigitte Vallée, editors, *34th Symposium on Theoretical Aspects of Computer Science, STACS 2017, March 8-11, 2017, Hannover, Germany*, volume 66 of *LIPIcs*, pages 43:1–43:14. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2017.

[Jär11] Matti Järvisalo. On the relative efficiency of DPLL and OBDDs with axiom and join. In Jimmy Ho-Man Lee, editor, *Principles and Practice of Constraint Programming - CP 2011 - 17th International Conference, CP 2011, Perugia, Italy, September 12-16, 2011. Proceedings*, volume 6876 of *Lecture Notes in Computer Science*, pages 429–437. Springer, 2011.

[Kra97] Jan Krajícek. Interpolation theorems, lower bounds for proof systems, and independence results for bounded arithmetic. *Journal of Symbolic Logic*, 62(2):457–486, 1997.

[Kra08] Jan Krajiček. An exponential lower bound for a constraint propagation proof system based on ordered binary decision diagrams. *Journal of Symbolic Logic*, 73(1):227–237, 2008.

[McM93] Kenneth L. McMillan. *Symbolic model checking*. Kluwer, 1993.

[MS94] Christoph Meinel and Anna Slobodova. On the complexity of Constructing Optimal Ordered Binary Decision Diagrams. In *Proceedings of Mathematical Foundations of Computer Science*, volume 841, pages 515–524, 1994.

[Pud97] Pavel Pudlák. Lower bounds for resolution and cutting plane proofs and monotone computations. *Journal of Symbolic Logic*, 62(3):981–998, 1997.

[PV05] Guoqiang Pan and Moshe Y. Vardi. Search vs. symbolic techniques in satisfiability solving. In *7th International Conference on Theory and Applications of Satisfiability Testing, SAT 2004, Revised Selected Papers*, volume 3542, pages 235–250, 2005.

[Seg08] Nathan Segerlind. On the relative efficiency of resolution-like proofs and ordered binary decision diagram proofs. In *Proceedings of the 23rd Annual IEEE Conference on Computational Complexity, CCC 2008, 23-26 June 2008, College Park, Maryland, USA*, pages 100–111. IEEE Computer Society, 2008.

[Sok17] Dmitry Sokolov. Dag-like communication and its applications. In *Computer Science - Theory and Applications - 12th International Computer Science Symposium in Russia, CSR 2017, Kazan, Russia, June 8-12, 2017, Proceedings*, pages 294–307, 2017.

[TSZ10] Olga Tveretina, Carsten Sinz, and Hans Zantema. Ordered binary decision diagrams, pigeonhole formulas and beyond. *JSAT*, 7(1):35–58, 2010.

[Urq87] Alasdair Urquhart. Hard examples for resolution. *Journal of the ACM*, 34(1):209–219, 1987.

[US94] Tomás E. Uribe and Mark E. Stickel. Ordered binary decision diagrams and the Davis-Putnam procedure. In Jean-Pierre Jouannaud, editor, *Constraints in Computational Logics, First International Conference, CCL'94, Munich, Germant, September 7-9, 1994*, volume 845 of *Lecture Notes in Computer Science*, pages 34–49. Springer, 1994.

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il