Electronic Colloquium on Computational Complexity, Report No. 73 (2020)

Lower Bounds on OBDD Proofs with Several Orders

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

[1]St. Petersburg Department of Steklov Institute of Mathematics of the Russian Academy of Sciences
[2]St. Petersburg State University
[3]University of California, San Diego
sbuss@ucsd.edu, dmitrits@pdmi.ras.ru, aknop@ucsd.edu, aariazanov@gmail.com, sokolov.dmt@gmail.com

May 5, 2020

Abstract

This paper is motivated by seeking lower bounds on $\mathrm{OBDD}(\land, \mathrm{w}, \mathrm{r})$ refutations, namely OBDD refutations that allow weakening and arbitrary reorderings. We first work with 1-NBP($\land$) refutations based on read-once nondeterministic branching programs. These generalize $\mathrm{OBDD}(\land, \mathrm{r})$ refutations. There are polynomial size 1-NBP($\land$) refutations of the pigeonhole principle, hence 1-NBP($\land$) is strictly stronger than $\mathrm{OBDD}(\land, \mathrm{r})$. There are also formulas that have polynomial size tree-like resolution refutations but require exponential size 1-NBP($\land$) refutations. As a corollary, $\mathrm{OBDD}(\land, \mathrm{r})$ does not simulate tree-like resolution, answering a previously open question.

The system 1-NBP($\land, \exists$) uses projection inferences instead of weakening. 1-NBP($\land, \exists_k$) is the system restricted to projection on at most k distinct variables. We construct explicit constant degree graphs G_n on n vertices and an $\epsilon > 0$, such that 1-NBP($\land, \exists_{\epsilon n}$) refutations of the Tseitin formula for G_n require exponential size.

Second, we study the proof system $\mathrm{OBDD}(\land, \mathrm{w}, \mathrm{r}_\ell)$ which allows ℓ different variable orders in a refutation. We prove an exponential lower bound on the complexity of tree-like $\mathrm{OBDD}(\land, \mathrm{w}, \mathrm{r}_\ell)$ refutations for $\ell = \epsilon \log n$, where n is the number of variables and $\epsilon > 0$ is a constant. The lower bound is based on multiparty communication complexity.

1 Introduction

Ordered Binary Decision Diagrams (OBDD's) are a flexible way to represent Boolean predicates. Proof systems based on OBDD's were introduced by Atserias, Kolaitis and Vardi [4]. Their proof system was used to refute sets of clauses and allowed only conjunctions (the "$\land$" rule), projections (the "$\exists$" rule) and weakenings (the "w" rule). By default, an OBDD proof must use the same variable order for all OBDD's in the proof. However, the variable reordering rule (the "r" rule) of [18] can be used to dynamically change variable orderings. We use notations such as $\mathrm{OBDD}(\land)$, $\mathrm{OBDD}(\land, \exists)$, $\mathrm{OBDD}(\land, \mathrm{r})$, $\mathrm{OBDD}(\land, \mathrm{w}, \mathrm{r})$ to denote OBDD-based proof systems with the indicated rules of inference. See Section 2 for definitions of these systems; throughout the present paper, all systems are presumed to be dag-like unless we explicitly mention that they are tree-like. The known inclusions for the main OBDD proof systems are shown in Figure 1, where an arrow indicates p-simulation (polynomial time simulation). Given that projection ($\exists$) is a special case of weakening (w), most of these inclusions follow immediately from the definitions.

Atserias et al. [4] showed that $\mathrm{OBDD}(\land, \exists)$ p-simulates resolution. They also showed that $\mathrm{OBDD}(\land, \mathrm{w})$ p-simulates cutting planes with unary coefficients (CP^*), but left whether $\mathrm{OBDD}(\land, \exists)$ p-simulates CP^* as an open problem. Also by [4], $\mathrm{OBDD}(\land, \exists)$ has polynomial size proofs of formulas that encode unsatisfiable linear systems over GF(2), and by [9], it also has polynomial size proofs of the pigeonhole principle. Neither of these principles have short resolution proofs; hence $\mathrm{OBDD}(\land, \exists)$ is strictly stronger than resolution; i.e.,

ISSN 1433-8092

it p-simulates resolution, but it is not p-simulated by resolution. The last two $\mathrm{OBDD}(\land, \exists)$ proofs (for GF(2) systems and the pigeonhole principle) can also be carried out in the OBDD-based symbolic quantifier elimination algorithm of Pan and Vardi [24]. For those algorithms, the join ($\land$) rules are required to download the initial clauses sequentially and the projection rule ($\exists$) can be used only if all the clauses containing the variable are alread downloaded: the resulting $\mathrm{OBDD}(\land, \exists)$ proofs are linear and consequently tree-like.

The projection rule ($\exists$) is a special case of the weakening rule, so $\mathrm{OBDD}(\land, \mathrm{w})$ certainly p-simulates $\mathrm{OBDD}(\land, \exists)$. It is open whether the two systems are p-equivalent. Buss et al. [7] proved that $\mathrm{OBDD}(\land, \mathrm{w})$ has polynomial size proofs of the Clique-Coloring tautologies. On the other hand, Pudlák [25] showed these tautologies require exponential size cutting plane (CP) proofs.[1] From this, CP does not p-simulate $\mathrm{OBDD}(\land, \mathrm{w})$. In particular, $\mathrm{OBDD}(\land, \mathrm{w})$ p-simulates CP^*, but not vice versa.

Although $\mathrm{OBDD}(\land, \mathrm{w})$ is a powerful proof system, we know several known exponential lower bounds. Segerlind [27] proved an exponential lower bound on the size of tree-like $\mathrm{OBDD}(\land, \mathrm{w})$ refutations, and Krajíček [21] proved an exponential lower bound on the size of dag-like refutations. Both lower bounds used a similar approach. They first established principles that require exponential size $\mathrm{OBDD}(\land, \mathrm{w})$ under some particular variable ordering π. This part of their arguments were based on lower bounds using communication complexity; randomized communication complexity in [28] and deterministic communication complexity in [21]. This by itself does not establish an exponential lower bound on $\mathrm{OBDD}(\land, \mathrm{w})$ proofs, since it only applies to the particular order π, and since an $\mathrm{OBDD}(\land, \mathrm{w})$ proof allows an arbitrary variable ordering as long as the same ordering is used for all OBDD's in the proof. The second step of the proofs of [28] and [21] used general methods for coding permutations to convert principles that are hard for OBDD proofs in a particular order π to principles that are hard for all variables orderings.

The reordering rule ("r") was proposed by Itsykson et al. [18], to relax the requirement that all OBDD's in a proof use the same variable ordering. If D and D' are a π-OBDD and a π'-OBDD that represent the same Boolean function, then the reordering rule allows inferring D' from D (even with different orders π and π'. It is open whether $\mathrm{OBDD}(\land, \mathrm{w}, \mathrm{r})$ has subexponential size, or even polynomial size, proofs for all valid formulas. The lower bound methods of [28, 21] used for $\mathrm{OBDD}(\land, \mathrm{w})$ proofs can not be applied to $\mathrm{OBDD}(\land, \mathrm{w}, \mathrm{r})$ proofs since the communication complexity arguments fail if there are multiple orders. This is because a fixed variable ordering can be used to construct a partition of the variables for the communication complexity game between two players; but this construction fails without a fixed ordering. We even do not have a good candidate for a formula conjectured to be hard for $\mathrm{OBDD}(\land, \mathrm{w}, \mathrm{r})$. The main conjectured candidate hard for $\mathrm{OBDD}(\land, \mathrm{w}, \mathrm{r})$ is random 3-CNFs; however it is even open whether these have short $\mathrm{OBDD}(\land)$ proofs. (Friedman and Xu [12] proved a lower bound for random CNFs, but only for very restricted subsystems of $\mathrm{OBDD}(\land)$.)

Buss et al. [7] showed that $\mathrm{OBDD}(\land, \mathrm{w}, \mathrm{r})$ is exponentially stronger than $\mathrm{OBDD}(\land, \mathrm{w})$. This separation was proved for a version of clique-coloring tautologies combined with coded permutations and orification.

Itsykson et al. [18] studied the proof system $\mathrm{OBDD}(\land, \mathrm{r})$. This system lacks both weakening (w) and projection ($\exists$) and this means that each line in an $\mathrm{OBDD}(\land, \mathrm{r})$ is equivalent to a conjunction of the initial hypotheses. Using this, [18] showed that the pigeonhole principle and Tseitin formulas require exponential size $\mathrm{OBDD}(\land, \mathrm{r})$ proofs. However, the pigeonhole principle has polynomial size CP^* proofs and hence polynomial size $\mathrm{OBDD}(\land, \mathrm{w})$ and $\mathrm{OBDD}(\land, \mathrm{w}, \mathrm{r})$ proofs. Thus these last three systems can have an exponential speedup over $\mathrm{OBDD}(\land, \mathrm{r})$.

In the opposite direction, [7] gave an example where the $\mathrm{OBDD}(\land, \mathrm{w})$ proofs may need to be superpolynomially longer than $\mathrm{OBDD}(\land, \mathrm{r})$ proofs. This example was based on the Tseitin tautology on a logarithmic-size complete graph, lifted with an indexing gadget and then transformed with the coded permutation method of Segerlind [28]. Their separation was only quasipolynomial, however; it is open whether there is an exponential separation or whether $\mathrm{OBDD}(\land, \mathrm{w})$ quasi-polynomially simulates $\mathrm{OBDD}(\land, \mathrm{r})$.

[18] proved that $\mathrm{OBDD}(\land, \mathrm{r})$ can have exponential speedup over $\mathrm{OBDD}(\land)$. For $\mathrm{OBDD}(\land)$, [7] showed that the Tseitin tautologies on a logarithmic-size complete graph require quasipolynomial size resolution refutations, but have polynomial size (tree-like) $\mathrm{OBDD}(\land)$ proofs. They also showed that these Tseitin tautologies lifted with an indexing gadget require quasipolynomial size CP proofs, but have polynomial size

[1] By "exponential", we mean 2^{m^δ}, where m is the size of the branching program and $\delta > 0$.

$$\begin{tikzcd}
\mathrm{OBDD}(\land, \mathrm{w}, \mathrm{r}) \arrow[r] \arrow[d] & \mathrm{OBDD}(\land, \exists, \mathrm{r}) \arrow[r] \arrow[d] & \mathrm{OBDD}(\land, \mathrm{r}) \arrow[d] \\
\mathrm{OBDD}(\land, \mathrm{w}) \arrow[r] \arrow[d] & \mathrm{OBDD}(\land, \exists) \arrow[r] \arrow[d] & \mathrm{OBDD}(\land) \\
\mathrm{CP}^* \arrow[r] & \mathrm{Res} &
\end{tikzcd}$$

Figure 1: The known polynomial simulations for the main OBDD proof systems. The top two rows follow from the definitions and the fact that projection is a special case of weakening. The simulations from the middle row to the bottom row are from [4].

OBDD($\land$) proofs. It is open whether these quasipolynomial separations can be improved to exponential separations.

The above-discussed results completely characterize the p-simulations between the four systems OBDD($\land, \mathrm{w}, \mathrm{r}$), OBDD($\land, \mathrm{w}$), OBDD($\land, \mathrm{r}$) and OBDD($\land$) (refer to Figure 1). The main open question about the relative strengths of these four systems is whether OBDD($\land, \mathrm{w}$) can quasipolynomially (or, sub-exponentially) simulate OBDD($\land, \mathrm{r}$). Figure 1 shows two proof systems using projection, OBDD($\land, \exists$) and OBDD($\land, \exists, \mathrm{r}$). This raises the question of what separations hold for these systems and the other four OBDD systems of Figure 1. Not much is known beyond the inclusions shown in the figure. The main result known is that the pigeonhole and Tseitin tautologies have polynomial size OBDD($\land, \exists$)) proofs by [4, 9] (since the Tseitin formulas are a special case of unsatisfiable linear equations over GF(2)), but require exponential size OBDD($\land, \mathrm{r}$) proofs by [18].

The present paper is motivated by the problem of giving superpolynomial lower bounds for OBDD($\land, \mathrm{w}, \mathrm{r}$) proofs. We certainly expect such bounds to hold, as otherwise NP = coNP by [11]. We are unable to solve this problem, but the present paper gives superpolynomial lower bounds for systems that are connected to OBDD($\land, \mathrm{w}, \mathrm{r}$).

We also establish that there are formulas where (tree-like) resolution has an exponential speedup over OBDD($\land, \mathrm{r}$). Consequently it has the same speedup over OBDD($\land$).

Our first collection of results are for proof systems which reason with formulas that are either deterministic or non-deterministic read-once branching programs, called 1-BP's or 1-NBP's. (Figure 2 shows the main systems studied in the paper, and which ones are known to have superpolynomial or exponential lower bounds.) Since the branching programs are read-once, any achievable path through the branching program can read each variable only once; however, different paths may query the variables in different orders. Thus 1-BP's and 1-NBP's generalize OBDD's in that any OBDD is already a 1-BP. On the other hand, it is well-known that 1-BP's are more expressive than OBDD's, see [35].

The inference rules $\land$, $\exists$ and w can also be used in 1-BP and 1-NBP proofs. A major difference however is that checking the validity of an inference is no longer (known to be) polynomial time checkable. For instance, for a 1-BP or 1-NBP) proof, the $\land$ rule can be used to derive any formula E from D_1 and D_2 provided that E is expresses the conjunction of D_1 and D_2. Here D_1, D_2 and E must be 1-BP's or 1-NBP's, respectively). Similarly E is derivable from D by the projection rule if E is equivalent to $\exists x\, D$ for some variable x, and by the weakening rule if E is implied by D. Note that these inference rules are all *semantic*, that is the validity of rule depends only on the functions computed by the formulas. As is shown in Section 2.2, there is no polynomial time algorithm for checking the validity of $\land$ inferences for 1-BP;s unless P = NP.

Many of new results of the present paper are lower bounds for the 1-BP and 1-NBP proof systems. These lower bounds immediately translate to lower bounds for OBDD proof systems with reordering.

Section 3 goes onto to show an exponential lower bound on the size of 1-NBP proofs of the perfect matching principle and of the Tseitin tautology, both based on graphs which are algebraic expanders. A common proof is used for both lower bounds; the arguments are rather involved, but use heavily the fact that every formula in a 1-NBP($\land$) refutation is equivalent to a conjunction of the initial clauses. Certain initial clauses are designated as "special", and it is shown that some formula in the proof must be a conjunction of a large (linearly many) special initial clauses. Then it is shown that any 1-NBP representation of such a conjunction must be exponentially large. See the proof below for details.

Section 4 uses the exponential lower bound for 1-NBP proofs to resolve the question of whether OBDD($\wedge$) simulates resolution. The history of this question is somewhat entangled; [33] claimed that OBDD($\wedge$) does not simulate resolution, and [19] claimed that OBDD($\wedge$) does not simulate even tree-like resolution. However, [7] noticed that proofs in the mentioned papers were incomplete, leaving the question about dag-like OBDD($\wedge$) open. Section 4 starts by showing that it is not just that the proofs from [33, 19] are incomplete but some of the main statements are incorrect. The constructions in [33] foundered because applications of the extension rule can make the OBDD($\wedge$) proofs shorter. (In contrast, it is well known that adding the extension rule to resolution makes it equivalent to the extended Frege.) This problem can be fixed by working with 1-NBP's instead of OBDD's: it turns out that the extension does not make 1-NBP($\wedge$) proofs shorter. This observation together with the lower bounds for 1-NBP($\wedge$) proofs allows us close the gaps in the previous proofs, by showing examples of formulas with polynomial size tree-like resolution proof which require exponential size 1-NBP($\wedge$) proofs. This immediately gives examples of exponential speedup of resolution over the systems OBDD($\wedge$) and OBDD($\wedge, \mathrm{r}$).

Section 5 shows that certain formulas "based on a bipartite graphs" have short 1-BP($\wedge$) refutations. For example, the pigeonhole principle and Tseitin formulas based on bipartite graphs have short 1-BP($\wedge$) refutations. Note this does not contradict the exponential lower bounds of Section 3 since algebraic expanders are not bipartite. It follows that 1-BP($\wedge$), and hence 1-NBP($\wedge$), has exponential speedup over OBDD($\wedge, \mathrm{r}$) for these formulas. (And, this gives another proof that 1-BP's are more expressive than OBDD's.) As a corollary, we get formulas that have polynomial size 1-NBP($\wedge$) proofs, but require exponentially long CP proofs. By comparison, as mentioned above, the best known speedup for OBDD($\wedge, \mathrm{r}$) over CP is quasi-polynomial [7].

Section 6 works with a proof system OBDD($\wedge, \exists_k, \mathrm{r}$); proofs in this system are restricted to use the projection ($\exists$) rule on at most k distinct variables. We prove that there is a constant $\epsilon > 0$ and a family of Tseitin formulas $\mathrm{TS}_{G,f}$, based on constant degree graphs G on n vertices with vertex labelings f, such that the formulas $\mathrm{TS}_{G,f}$ require exponential size 1-NBP($\wedge, \exists_{\epsilon n}$) proofs. From this, using a padding argument, we show that OBDD($\wedge, \exists_{\epsilon n}$) has exponential speedup over OBDD($\wedge$), for each fixed $\epsilon > 0$.

The final part of the paper, Section 7, considers the fragment OBDD($\wedge, \mathrm{w}, \mathrm{r}_\ell$) of OBDD($\wedge, \mathrm{w}, \mathrm{r}$) in which at most ℓ many different variables orderings are permitted. We prove an exponential lower bound on OBDD($\wedge, \mathrm{w}, \mathrm{r}_\ell$) proofs with $\ell = \epsilon \log n$ for $\epsilon > 0$ sufficiently small. The argument is based on the problem Search_φ of searching for a falsified clause under a given truth assignment to an unsatisfiable CNF formula φ. The proof uses lower bounds on k party communication (for $k = \ell + 1$). We show that a short OBDD($\wedge, \mathrm{w}, \mathrm{r}_\ell$) refutation of φ means that Search_φ has small k-party communication complexity for $k = \ell + 1$ under some balanced partition of the inputs. Then results of [15, 26] imply that the search problem for lifted pebbling formulas requires almost linear k-party communication complexity in at least one partition of the inputs. Finally a construction from [20], based on [28], uses a further lifting-like construction to form formulas whose search problems have almost linear k-party communication complexity under every balanced partition of their inputs. This implies exponential lower bounds on OBDD($\wedge, \mathrm{w}, \mathrm{r}_\ell$) refutations. These formulas do have polynomial size OBDD($\wedge, \mathrm{r}$) refutations however.

Section 8 concludes with some discussion about open problems for future research.

2 Preliminaries

2.1 Branching program and OBDD proof systems

A deterministic branching program (BP) is a representation of a Boolean function $\{0,1\}^n \to \{0,1\}$ by a directed acyclic graph with exactly one source and two sinks. Every node other than the sinks is labeled with an input variable and has exactly two outgoing edges: one labeled with 1 (*True*) and the other with 0 (*False*). One of the sinks is labeled with 1 and the other with 0. The value of the Boolean function for a given assignment of values to the input variables is evaluated by traversing a path starting at the source

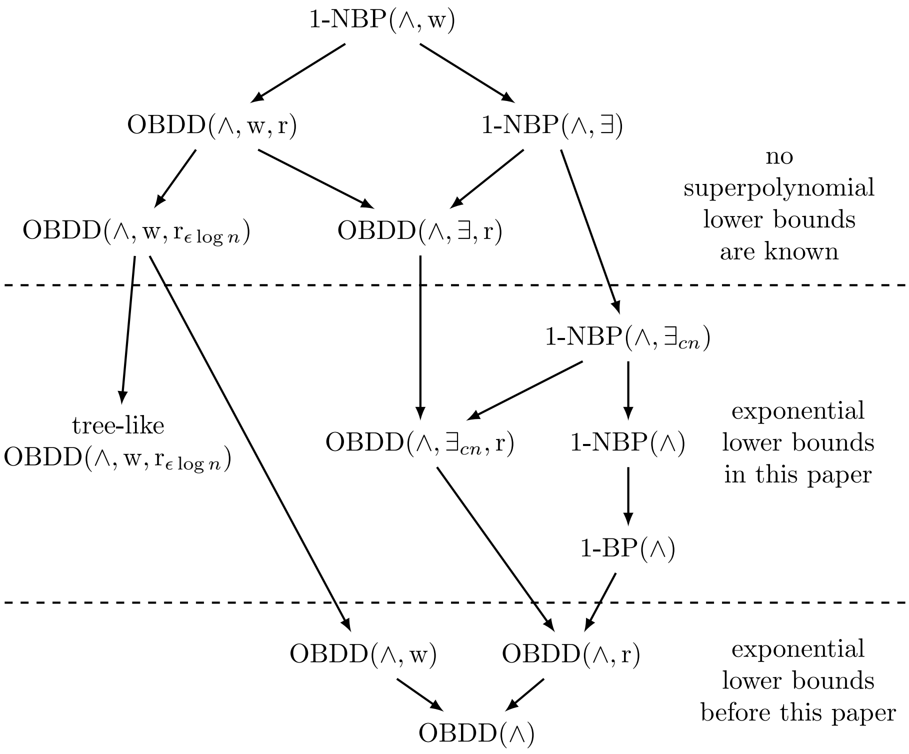
1-NBP(∧, w)
OBDD(∧, w, r)
1-NBP(∧, ∃)
no superpolynomial lower bounds are known
OBDD(∧, w, $\mathrm{r}_{\epsilon \log n}$)
OBDD(∧, ∃, r)
1-NBP(∧, $\exists_{cn}$)
tree-like OBDD(∧, w, $\mathrm{r}_{\epsilon \log n}$)
OBDD(∧, $\exists_{cn}$, r)
1-NBP(∧)
exponential lower bounds in this paper
1-BP(∧)
OBDD(∧, w)
OBDD(∧, r)
exponential lower bounds before this paper
OBDD(∧)

Figure 2: A summary of the systems for which exponential lower bounds are known. The systems are dag-like, except for the one labelled tree-like. The arrows indicate p-simulations.

node and, at every node labeled with an input variable, extending the path along the edge that is labeled with the value of the variable. This path ends in a sink; the label of the sink is the value of the Boolean function.

A *nondeterministic* branching program (NBP) differs from a deterministic one in that the directed acyclic graph may include non-sink nodes which are unlabeled and have outdegree two; these are called *guessing nodes*. That is the directed acyclic graph for a NBP can have three kinds of nodes: labeled nodes (labeled with a variable), guessing nodes, and sink nodes. A path that reaches a guessing node may be extended along either outgoing edge. The values of a function represented by a nondeterministic branching program equals 1 exactly when there exists at least one such path from the source to the sink labeled with 1. Note that deterministic branching programs are a special case of nondeterministic branching programs.

A deterministic or nondeterministic branching program is (syntactic) read-once if every path from the source to a sink queries each variable at most once. Read-once BP's and NBP's are denoted 1-BP's and 1-NBP's.

We generally denote the input variables by $x_1, \ldots, x_n$, so there are n inputs. We let $[n] = \{1, \ldots, n\}$. An *order* on the input variables $x_1, \ldots, x_n$ is a bijection π from $[n]$ to $\{x_1, \ldots, x_n\}$. We often call π a *variable ordering* for clarity. We let $\pi[\leq s] := \{x_{\pi(1)}, \ldots, x_{\pi(s)}\}$, namely the set of the first s variables in the order. And $\pi[>s] := \{x_{\pi(s+1)}, \ldots, x_{\pi(n)}\}$, namely, the set of remaining variables.

An ordered binary decision diagram (OBDD) is a special case of a 1-BP, and thereby a special case of a 1-NBP. For π a variable ordering, a π-OBDD is a 1-BP φ such that, for every path p in φ, the order in which the variables are queried respects the order π. More generally, a 1-BP φ is an OBDD provided there is some order π such that φ is a π-OBDD. OBDD's were first defined by Bryant, see [6].

Definition 2.1. Let φ be an (unsatisfiable) CNF formula; i.e, φ is a set of clauses. An OBDD proof (also called a *refutation*) of φ consists of a sequence $D_1, \ldots, D_t$ of OBDD's with associated permutations $\pi_1, \ldots,$

π_t such that each D_i is a π_i-OBDD, the last OBDD, D_t, represents the constant false, and each D_i either represents one of the clauses of φ or is inferred from earlier OBDD's in the proof by one of the following rules of inference:

conjunction or join ($\wedge$): D_i represents the Boolean function $D_j \wedge D_k$ where $j, k < i$ and where D_i, D_j D_k use the same order $\pi_i = \pi_j = \pi_k$.

weakening (w): D_i is semantically implied by D_j where $j < i$ and where D_i and D_j use the same order $\pi_i = \pi_j$.

projection ($\exists$): D_i represents the Boolean function $\exists x\, D_j$ where $j < i$, where x is a variable, and where D_i and D_j use the same order $\pi_i = \pi_j$.

reordering (r): D_i is equivalent to D_j for some $j < i$. Here, π_i and π_j are different orders.

Although we usually use the terminology "OBDD proof of φ", it is actually a *refutation* of φ. It is well-known that there are polynomial time algorithms for recognizing the validity of each the above inference rules (see [6, 18]). Therefore, OBDD proof systems are propositional proof systems in the sense of Cook and Reckhow [10, 11]. The *size* of an OBDD proof is equal to the sum of the sizes of the OBDD's in the proof; the size of an OBDD is the number of vertices in the graph. There is also a well-known algorithm for converting an π-OBDD to a minimal π-OBDD; hence, we may assume without loss of generality that all π-OBDD's have minimal size (for that order).

As discussed in the introduction, we use different OBDD proof systems with different sets of allowed rules. For example, an OBDD($\wedge$, w) proof is an OBDD proof in which only the $\wedge$ and w rules are used. Since the projection rule is a special case of the weakening rule, we never use both in the same system. An OBDD proof is *tree-like* if each OBDD in the proof is used as a hypothesis at most once.

If the reordering rule is not allowed, then w.l.o.g., all OBDD's in a proof use a common order π, since only the reordering rule allows changing to a different order. When we want make the order explicit, we use terminology such as "π-OBDD($\wedge$, w) proof" to mean that OBDD's in the proof are restricted to use the order π.

An OBDD($\wedge, \mathrm{w}, \mathrm{r}_\ell$) proof is an OBDD($\wedge$, w, r) which uses at most ℓ distinct variable orderings. An OBDD($\wedge, \mathrm{r}, \exists_k$) proof is an OBDD($\wedge$, r, $\exists$) which applies the projection inferences with at most k distinct variables.

1-BP proofs and 1-NBP proofs are defined similarly to OBDD proofs; however, there is no reordering rule. (And in fact, 1-BP's and 1-NBP's do not in general have a fixed order for variable queries.)

Definition 2.2. Let φ be an (unsatisfiable) CNF formula. A 1-BP proof, respectively a 1-NBP proof, of φ consists of a sequence $D_1, \ldots, D_t$ of 1-BP's, respectively 1-NBP's, such that D_t represents the constant false, and each D_i either represents one of the clauses of φ or is inferred from earlier lines in the proof by one of the following rules of inference:

conjunction or join ($\wedge$): D_i represents the Boolean function $D_j \wedge D_k$ where $j, k < i$.

weakening (w): D_i is semantically implied by D_j where $j < i$.

projection ($\exists$): D_i represents the Boolean function $\exists x\, D_j$ where $j < i$ and x is a variable.

As discussed in the introduction, 1-BP and 1-NBP proof systems can be defined with different rules of inference. For example, NBP($\wedge$, w) means that the conjunction rule and weakening rules are permitted. Since projection is a special case of weakening, this is the strongest of the systems considered in the present paper. The 1-BP and 1-NBP systems are not propositional proof systems in the sense of Cook and Reckhow. Indeed, as is discussed in Section 2.2, there is no polynomial time algorithm to decide the validity of conjunction inferences or weakening inferences unless P = NP. However, note that OBDD proofs are also 1-BP and 1-NBP proofs. Therefore lower bounds for the last two systems imply lower bounds for the corresponding OBDD proof system.

Definitions 2.1 and 2.2 defined refutations. A *derivation* of D is defined exactly the same as a refutation except the final line is D instead of the constant false. Derivations are *implicationally sound* in that any truth assignment that satisfies φ also satisfies D.

Note that if D has a derivation from φ in which the only inference rules are conjunction ($\wedge$) and reordering (r), then each line D_i in the derivation expresses the conjunction of a subset of the clauses of φ. Thus, the above derivation systems without the weakening rule are not semantically complete in that not all consequences of φ can be derived.

Section 4 will discuss the relative strengths of 1-NBP and resolution. For this, an important tool is the use of the extension rule. An extension rule allows the introduction of a new variable z along with an axiom $z = f(\vec{x})$ where f is a Boolean function, and $\vec{x}$ are the already-used variables. In applications, $f(\vec{x})$ will be expressible as a conjunction of clauses, or of branching programs.

Definition 2.3. Let $\mathcal{P}$ be a proof system used to derive refutations of CNF formulas. Let $\varphi(\vec{x})$ be a CNF formula. A set E of extension axioms for $\varphi(\vec{x})$ is a set of clauses expressing $z_i \leftrightarrow \psi_i(\vec{x}, z_1, \ldots, z_{i-1})$, where each ψ_i is a conjunction of literals and $z_1, \ldots, z_\ell$ are new variables. Then, an *extension-$\mathcal{P}$* refutation of φ is by definition a refutation of $\varphi \wedge E$ where E is a set of extension axioms for φ.

For example, a definition by extension of the form $z \leftrightarrow (y_1 \wedge y_2)$ is represented by the three clauses $\neg z \vee y_1$, $\neg z \vee y_2$, and $\neg y_1 \vee \neg y_2 \vee z$. The canonical example of extension is the "extended resolution" refutation system of Tseitin [31]. The pigeonhole principle gives an exponential separation between resolution and extended resolution [17, 11]. It is surprising that extension-1-NBP($\wedge$) is no stronger than 1-NBP($\wedge$); this is proved in Section 4.

Definition 2.3 required the defining formulas ψ_i to be conjunctions of literals. This can be generalized so that φ_i can be represented by an OBDD, a BP or even a Boolean circuit, at the cost of introducing additional extension variables for the nodes in the branching program or circuit.

2.2 Weirdness with reordering, weakening and extension

This section explores some undesirable situations arise when too strong rules of inferences are allowed. We first show that if conditions on OBDD inferences are removed to allow dynamically changing the variable order, then it becomes NP-hard to check the validity of inferences. We second show that using the extension rule together with weakening can make some of the just discussed systems so powerful that they have polynomial size proofs for all tautologies (that is, to be more accurate, polynomial size refutations of unsatisfiable sets of clauses). This section is purely foundational, and the rest of the paper does not depend on it.

We start with generalized versions of the OBDD inferences that allow dynamic reorderings. Define the "$\wedge$&r" inference rule to be the same as the $\wedge$ (conjunction) rule for OBDD's, but without the restriction that $\pi_i = \pi_j = \pi_k$. Likewise define the "$\exists$&r" and the w&r inference rules to be the same as the $\exists$ rule and w rules but without the restriction that $\pi_i = \pi_j$.

Theorem 2.4 ([23, Lemma 8.14])**.** *The problem of verifying* $\wedge$&r*, or* w&r*, inferences is* NP*-hard.*

Proof. The property of being a valid inference is in coNP since it is an assertion about all truth assignments and since OBDD's can be evaluated in polynomial time (even, in logspace). Let φ be an instance of satisfiability. Let $x_1, \ldots, x_n$ be the variables in φ, and suppose x_i appears k_i times in φ. We introduce new variables $x_{i,j}$ for $1 \le j \le k_i$. Form φ^* by replacing the j-th copy of each x_i with $x_{i,j}$. (If the j-th copy of x_i in φ is negated, then $x_{i,j}$ is negated in φ^*.) Note every variable $x_{i,j}$ occurs exactly once in φ^*. Let $E(x_{1,1}, \ldots, x_{n,k_n})$ be the Boolean function which is true if and only if $x_{i,j}$ is equal to $x_{i,j'}$ for all i and all $1 \le j, j' \le k_i$. The formula φ^* can easily be written as a π_0-OBDD where π_0 is the order in which variables appear in φ. And, E can be written as a π_1-OBDD where π_1 takes the variables $x_{i,j}$ is the lexicographic order of (i, j).

Clearly $\varphi^* \wedge E$ is satisfiable if and only if φ is satisfiable. Therefore, the inference inferring $\perp$ (falsity) from φ^* and E is a valid $\wedge$&r inference if and only if φ is unsatisfiable. Similarly, inferring $\neg E$ from φ^* is a valid w&r inference if and only if φ is unsatisfiable. □

Note however, that there is a polynomial time algorithm for verifying the correctness of an ∃&r inference of D_i from D_j. This is done, by first forming $\exists x\, D_j$ as π_j-OBDD and then checking if the result is equivalent to D_i. The former is in polynomial time by [6]; the latter is by the fact that the correctness of reordering can be verified in polynomial time [18].

We next consider systems that have weakening and extension, and no restrictions on reordering. These turn out to be "super" in the sense of [11], namely to have polynomial size refutations of all unsatisfiable sets of clauses.

Theorem 2.5. *Extension-*OBDD(∧&r, w) *is super. Thus, the same holds for extension-*1-BP(∧, w) *and extension-*1-NBP(∧, w)*.*

Proof. We modify the construction in the proof of Theorem 2.4. Let $\varphi(x_1, \ldots, x_n)$ be an unsatisfiable set of clauses. We must show φ has a polynomial size extension-OBDD(∧&r, w) refutation. Let E' be the set of extension axioms $x_{i,j} \leftrightarrow x_i$. Let φ_ℓ denote the conjunction of the first ℓ clauses of φ. Let φ^*_ℓ be the result of replacing each j-th occurrence of x_i in φ_ℓ with $x_{i,j}$.

The refutation proceeds as follows. It first derives successively, an OBDD representing each φ^*_ℓ, for $\ell = 1, 2, \ldots$, by using conjunction (∧) and then weakening (w) to combine $\varphi^*_{\ell-1}$ and, for each variable x_i appearing in the ℓ-th clause, the two clauses in E' expressing $x_{i,j} \leftrightarrow x_i$ for the appropriate value of j. At the end, an OBDD representing φ^* has been derived. Note that each φ^*_ℓ is polynomial size (in fact, linear size).

The refutation then derives E from E' (E is defined in the proof of Theorem 2.4). It does this by deriving for each pair (I, J), an OBDD $E_{I,J}$ representing the conjunction of the equalities $x_{i,j} \leftrightarrow x_i$ for all (i, j) equal to or lexicographically before (I, J). For example, $E_{I,J+1}$ is derived by conjunction from $E_{I,J}$ and two clauses from E'. Note E_{n,k_n} is just E. The $E_{I,J}$'s are all linear size OBDD's.

The refutation concludes by using an ∧&r inference to derive ⊥. □

It was crucial for Theorem 2.5 that reordering, extension and weakening are present. On the other hand, Theorem 3.1 and Lemma 4.5 together give exponential lower bounds for extension-1-NBP(∧) proofs, so that system is not super. It is open whether either of extension-OBDD(∧, w) or 1-NBP(∧, w) is super.

2.3 Graph based formulas

Section 3 proves lower bounds on 1-NBP(∧) for a general class of formulas that includes perfect matching principles and Tseitin formulas. These formulas are said to be "based on" a graph, as defined next.

Definition 2.6. Let $G(V, E)$ be an undirected graph without loops, but possibly with multi-edges. Let $E_G(v)$ denote the set of edges incident to v in G. Each edge $e \in E$, has an associated propositional variable x_e. A formula is *based on* G if it has the form $\bigwedge_{v \in V} \varphi_v$, where each φ_v is a CNF formula depending only on the variables x_e for $e \in E_G(v)$.

A formula based on G is *matching-like* if in addition, each φ_v satisfies the following, letting $E_G(v) = \{e_1, \ldots, e_d\}$:

- Each φ_v has the form $(x_{e_1} \vee x_{e_2} \vee \cdots \vee x_{e_d}) \wedge \varphi'_v$ where φ'_v has value 1 if all the variables x_{e_i} are set to 0. Thus each clause in φ'_v contains a negated edge variable $\neg x_{e_i}$. Note φ_v has value 0 if all variables are set to 0.

- φ_v is satisfied by any assignment that sets exactly one of $x_{e_1}, \ldots, x_{e_d}$ to 1.

We call such formulas matching-like because the local constraint φ_v is satisfied only if at least one edge incident to v has value 1, and φ_v is satisfied whenever exactly one edge set to 1. However, the behavior of φ_v in all other cases is not specified.

The following formulas are most important examples of formulas based on graphs.

Tseitin formulas. The Tseitin formula $\mathrm{TS}_{G,f}$ is a formula based on an undirected graph $G(V,E)$ parameterized by a function $f : V \to \{0,1\}$ labelling vertices with 0 or 1. $\mathrm{TS}_{G,f} = \bigwedge_{v \in V} \varphi_v$, where φ_v is a CNF formula expressing $\sum_{e \in E_G(v)} x_e \equiv f(v) \pmod 2$. $\mathrm{TS}_{G,f}$ is satisfiable iff $\sum_{v \in S} f(v) \equiv 0 \pmod 2$ for every connected component S of G [34]. Note that $\mathrm{TS}_{G,f}$ is matching-like if $f(v) = 1$ for all $v \in V$.

Perfect matching principle. The formula PMP_G is a matching-like formula based on an undirected graph $G(V,E)$. $\mathrm{PMP}_G = \bigwedge_{v \in V} \varphi_v$, where φ_v is a CNF formula expressing $|\{e \in E_G(v) \mid x_e = 1\}| = 1$. The formula PMP_G is satisfiable iff G has a perfect matching.

Graph pigeonhole principle. The formula PHP_G is based on a bipartite graph $G(V,E)$ with the vertices partitioned into two parts, P (pigeons) and H (holes). $\mathrm{PHP}_G = \bigwedge_{v \in V} \varphi_v$, where for $v \in P$, φ_v expresses $|\{e \in E_G(v) \mid x_e = 1\}| \geq 1$ (each pigeon is mapped to at least one hole) and for $v \in H$, φ_v expresses $|\{e \in E_G(v) \mid x_e = 1\}| \leq 1$ (at most one pigeon is mapped to each hole).

The standard pigeonhole principle PHP^{n+1}_n is equal to $\mathrm{PHP}_{K_{n+1,n}}$, where $K_{n+1,n}$ is the complete bipartite graph with $n+1$ and n vertices in the parts.

Sometimes we want to use the above formulas, but based on a graph G with loops. In this case, we ignore the loops in G.

2.4 Expander graphs

The hard formulas from Theorem 3.1 are based on algebraic expanders.

Definition 2.7. Let $G(V,E)$ be an undirected graph possibly with loops and multiple edges. The graph G is an (n,d,α)-*algebraic expander* if G is d-regular, $|V| = n$, and the absolute value of the second largest eigenvalue of the adjacency matrix of G is at most αd.

It is well-known that for all $1 > \alpha > 0$ and all large enough constants d, there exist a natural number n_0 and a family $\{G_n\}_{n=n_0}^{\infty}$ of (n,d,α)-algebraic expanders. With a high probability, a random graph is an (n,d,α)-algebraic expander. In addition, there are explicit constructions such that G_n can be constructed in time polynomial in n [22].

For $A, B \subseteq V$, $E_G(A,B)$ denotes the multiset of edges of G that have one end in A and another end in B. Note that in the case where both ends of an edge are simultaneously in A and in B, we count this edge twice. We write $E(A,B)$ instead of $E_G(A,B)$ when G is clear from the context.

Lemma 2.8 (Expander mixing lemma [2])**.** *Let $G(V,E)$ be an (n,d,α)-algebraic expander, and $A, B \subseteq V$. Then*

$$\left| |E(A,B)| - \frac{d|A||B|}{n} \right| \leq \alpha d \sqrt{|A||B|}.$$

Lemma 2.9 ([14])**.** *Let $G(V,E)$ be an (n,d,α)-algebraic expander. Then for every set $S \subseteq V$,*

$$|E(S, V \setminus S))| \geq d|S| \left(1 - \alpha - \frac{|S|}{n}\right).$$

Proof. $|E(S, V \setminus S)| = d|S| - |E(S,S)| \geq d|S| - \frac{d}{n}|S|^2 - \alpha d\sqrt{|S|^2} = d|S|(1 - \alpha - \frac{|S|}{n})$. The inequality follows from Lemma 2.8. □

We also use *edge expanders*. An undirected graph $G(V,E)$ is an *edge (r,c)-expander* if for every set $A \subseteq V$, if $|A| \leq r$, then $|E(A, V \setminus A)| \geq c|A|$.

Lemma 2.10 ([1], [3, Theorem 21.9])**.** *Let $G(V,E)$ be a d-regular edge $(n/2,c)$-expander with at least one loop at each vertex, where $n = |V|$. Then G is a $(n,d,1-\gamma)$-algebraic expander, where $\gamma = \min\left\{\frac{2}{d}, \frac{c^2}{2d^2}\right\}$.*

The expansion quality, as measured by α, can be improved using the following method. For an undirected graph $G(V,E)$ we let G^k denote the graph that vertices V and has edges between vertices u and v for each path in G from u and v of length exactly k. If G is an (n,d,α)-algebraic expander, then G^k is an (n,d^k,α^k)-algebraic expander.

2.5 Communication complexity

Section 7 will prove a lower bound on $\mathrm{OBDD}(\wedge, \mathrm{w}, \mathrm{r}_\ell)$ proofs. The lower bound is based on "number on forehead" (NOF) communication complexity.

Let $f : \{0,1\}^n \to \{0,1\}$ be a Boolean function. We have k players who cooperate to compute $f(s)$. The function f is known by all of them; in the number on forehead model, each bit is known to all but one of the players. For this, let $\Pi = (\Pi_1, \Pi_2, \ldots, \Pi_k)$ be a partition of $[n]$. (So $\Pi_i \cap \Pi_j = \emptyset$ for every $i \neq j \in [k]$.) The i-th player knows only bits of s with indices from $[n] \setminus \Pi_i$. The players have a common communication channel. In each round of play, one player broadcasts a string to everyone else. Their goal is to compute $f(s)$ with the minimum number of bits sent during all rounds.

In a more general situation, instead of the function f, there is a relation $R \subseteq \{0,1\}^n \times Z$. Now the players' goal is to find $z \in Z$ such that $(s, z) \in R$.

More formally, a communication protocol with respect to a partition Π is a tree T where each internal node v is labeled by a function $d_v : \{0,1\}^{[n]\setminus\Pi_i} \to \{0,1\}$, each leaf is labeled by an element $z \in Z$, each node has two children, and the edges from a node to its children are labeled by different Boolean values. A node of this type corresponds to the i-th player broadcasting one bit. The value of the protocol T on an input s is the label of the leaf reached starting from the root, and traversing a branch of the tree; at each internal node labeled by $d : \{0,1\}^{[n]\setminus\Pi_i} \to \{0,1\}$, the traversal uses the edge labeled by $f\big(s\big|_{[n]\setminus\Pi_i}\big)$. The cost of the protocol is the depth of T.

The communication complexity, $\mathbf{D}(R, \Pi)$, of a relation R is the minimum cost among the protocols for this relation with respect to the partition Π.

This paper works with the communication complexity of search problems corresponding to unsatisfiable CNFs with respect to balanced partitions. Let $\varphi = \bigwedge_{i=1}^m C_i$ be an unsatisfiable CNF on $x_1, \ldots, x_n$. Then $\mathrm{Search}_\varphi \subseteq \{0,1\}^n \times [m]$ is the relation such that

$$(\vec{x}, i) \in \mathrm{Search}_\varphi \iff C_i(\vec{x}) = 0.$$

Then $\mathbf{D}(\mathrm{Search}_\varphi, \Pi)$ denotes the communication complexity of finding an i such that $C_i(\vec{x}) = 0$, relative to the partition Π of the inputs x. A partition Π of $[n]$ into k subsets is *balanced* if $|\Pi_i| \geq \lfloor n/k \rfloor$ for every $i \in [k]$.

3 Lower bounds for 1-NBP($\wedge$)

The next theorem gives a lower bound for 1-NBP($\wedge$) proofs of general matching-like formulas.

Theorem 3.1. *Let $d \in \mathbb{N}$ and $\alpha < \frac{1}{400}$ be constants. Let Φ be an unsatisfiable matching-like formula based on an (n, d, α)-algebraic expander $G(V, E)$. Then any 1-NBP($\wedge$) refutation of Φ has size at least $2^{\Omega(n)}$.*

The theorem will be proved with the aid of Lemmas 3.3 and 3.4 below. Recall the matching-like formula Φ has the form $\bigwedge_{v\in V} \varphi_v$. Let Ψ be a conjunction of a subset of the clauses of Φ, so Ψ is $\bigwedge_{v\in V} \psi_v$, where each ψ_v is the conjunction of zero or more of the clauses of φ_v. A vertex $v \in V$ is *active* in Ψ if ψ_v has value 0 if all variables are set to 0. By the first condition in the definition of "matching-like", a vertex $v \in V$ that is incident to edges $e_1, \ldots, e_d$ is active if and only if ψ_v contains the clause $x_{e_1} \vee x_{e_2} \vee \cdots \vee x_{e_d}$.

The most important role in the proof of Theorem 3.1 is played by the following observation. Suppose $S \subseteq V$ is the set of active vertices in Ψ. Then Ψ can be satisfied by any matching that covers S (by substituting 1 to the edges in the matching and 0 to other edges) provided that such a matching exists.

The proof of the theorem will use several constants. Fix $\alpha \leq 1/400$. Set $\beta_0 = \frac{1-\alpha(\sqrt{8}+1)}{10}$, and $\beta_1 = \frac{9}{10}(\frac{1}{2} - \alpha)$. It is easy to check the following properties.

Remark 3.2. *The following inequalities are true:*

(i) $\beta_1 \geq \beta_0$,

(ii) $2\alpha + 2\beta_1 < 1$,

(iii) $(\sqrt{8}+1)\alpha + 9\beta_0 < 1$, *and*

(iv) $\beta_0 \geq 64\alpha/3$.

Lemma 3.3. *Let Ψ be a conjunction of some of the clauses of $\Phi = \bigwedge_{v \in V} \varphi_v$. If at most $\beta_1 n$ vertices are active in Ψ, then Ψ is satisfiable.*

Thus any unsatisfiable subformula of Φ (i.e., any unsatisfiable conjunction of clauses of Φ) must contain at least $\beta_1 n$ many active vertices. Recall that, since $\wedge$ is the only rule allowed for 1-NBP($\wedge$) proofs, each line in an 1-NBP($\wedge$) refutation of Φ represents a conjunction of clauses of Φ. Lemma 3.3 implies that any refutation of Φ contains a line representing a conjunction in which at least $\beta_1 n$ many vertices are active.

Lemma 3.4. *Let Ψ be a conjunction of some of the clauses of the formula $\Phi = \bigwedge_{v \in V} \varphi_v$. Suppose Ψ is satisfiable. Finally suppose the number of active vertices in Ψ is at least $\beta_0 n/2$ and at most $\beta_0 n$. Then every* 1-NBP *representation of Ψ has size at least $2^{\Omega(n)}$.*

The idea for the proof of Theorem 3.1 is that Lemma 3.3 states the presence of a line for which Lemma 3.4 gives an exponential size lower bound.

Proof of Theorem 3.1. Let D_1, D_2, ..., D_s be a 1-NBP($\wedge$) refutation of Φ. Each D_i represents the conjunction of a subset of clauses of Φ. The final line D_s is an unsatisfiable conjunction of clauses; Lemma 3.3 states that there are more than $\beta_1 n$ active vertices in this unsatisfiable conjunction. We have $\beta_1 \geq \beta_0$. Hence $\beta_1 n \geq \beta_0 n$.

Let Ψ_1 and Ψ_2 be conjunctions of some of the clauses of Φ. The number of active vertices in $\Psi_1 \wedge \Psi_2$ is at most the sum of the numbers of active vertices in Ψ_1 and in Ψ_2. Indeed, a vertex v is active in a formula Ψ if and only if Ψ contains the clause $\bigvee_{e \in E(v)} x_e$, where $E(v)$ is set of edges that are adjacent to v. Hence there is a $j < s$ such that D_j represents a conjunction of clauses with at least $\beta_0 n/2$ and at most $\beta_0 n$ active vertices. By Lemma 3.3, D_j is satisfiable. Hence by Lemma 3.4 the size of D_j is at least $2^{\Omega(n)}$. □

We still need to prove Lemmas 3.3 and 3.4.

3.1 Expanders, matchings, and the proof of Lemma 3.3

To prove Lemma 3.3 we shall show that for any $S \subseteq V$ such that $|S| < \beta_1 n$, there is a matching in G (i.e., a perfect matching on a subgraph of G) which covers S. We use Tutte's classical criterion for the existence of a matching.

Theorem 3.5 ([32]). *A graph G has a perfect matching if and only if, for any set $A \subseteq V$,*

$$o(G - A) \leq |A|,$$

where $G - A$ denotes the graph G without the vertices from the set A and $o(G - A)$ denotes the number of connected components with odd cardinality in $G - A$.

For $G(V, E)$ a graph and $S \subseteq V$, we let $\delta_G(S)$ denote the set of vertices from $V \setminus S$ adjacent to vertices in S.

Lemma 3.6. *Let $G(V, E)$ be an undirected graph with n vertices. Let $S \subseteq V$. Assume that for all subsets $U \subseteq S$ the inequality $|\delta_G(U)| > |U|$ holds. Then there exists a matching in G that covers all the vertices from S.*

Proof. We define a new graph G'. If $|S \cup \delta_G(S)|$ is even, set $B = \delta_G(S)$; otherwise, let v_0 be a new vertex and set $B = \delta_G(S) \cup \{v_0\}$. The vertices of G' are $S \cup B$; the edges of G' are the induced edges from G plus all possible edges between vertices in B. We will show that G' has a perfect matching; since v_0 has no edge to S, this gives a matching in G covering S, after dropping edges between vertices of B.

We show that G' satisfies the conditions of Tutte's theorem (Theorem 3.5). First consider Tutte's criterion with $A = \emptyset$. For this, we must show that G' does not contain a connected component of odd cardinality. In fact, we claim G' is connected, and by construction G' has an even number of vertices. Let v be a vertex of G', and U be the connected component of G' containing v. If $U \cap B = \emptyset$, then U is a connected component in G as well, so $|\delta_G(U)| = 0 < |U|$ which is a contradiction. Thus v is connected to B. Since B is a clique, G' is connected.

Now let $A \subseteq S \cup B$ be non-empty. For the sake of contradiction assume that $G' - A$ has at least $|A| + 1$ odd connected components. Since B is a clique, all vertices in $B \setminus A$ are in the same connected component of $G' - A$. Hence there are at least $|A|$ connected components of $G' - A$ that contain only vertices from S. Let U be the union of these connected components. We have $|U| \geq |A|$. No member of $\delta_G(U)$ can lie in the connected component containing the vertices of $B \setminus A$; hence $\delta_G(U) \subseteq A$. The contradicts the hypothesis that $\delta_G(U) > |U|$ since $|U| \geq |A|$.

Thus G' satisfies the conditions of Tutte's theorem (Theorem 3.5) and G' has a perfect matching. □

The next lemma will let us prove the existence of sets S satisfying the conditions of Lemma 3.6.

Lemma 3.7. *Let $G(V, E)$ be an (n, d, α)-algebraic expander. Let $k > 0$ and $\beta \in (0, 1)$ satisfy $\alpha(\sqrt{k} + 1) + \beta(k + 1) < 1$. Then for every set $S \subseteq V$, if $|S| \leq \beta n$, then $|\delta(S)| > k|S|$.*

Proof. Assume that $|S| \leq \beta n$ and $|\delta(S)| \leq k|S|$. Since $E(S, V \setminus S) = E(S, \delta_G(S))$, Lemmas 2.8 and 2.9 say that

$$|E(S, V \setminus S)| \leq \frac{d}{n}|S||\delta_G(S)| + \alpha d\sqrt{|S||\delta_G(S)|} \leq \frac{d}{n}k|S|^2 + \alpha d\sqrt{k}|S|$$

and $|E(S, V \setminus S)| \geq d|S|(1 - \frac{|S|}{n} - \alpha)$. Thus $\alpha(\sqrt{k} + 1) + \frac{|S|}{n}(k + 1) \geq 1$. This is a contradiction by $\frac{|S|}{n} \leq \beta$ and by the hypothesis $\alpha(\sqrt{k} + 1) + \beta(k + 1) < 1$. □

The following special cases of Lemma 3.7 are important for us.

Corollary 3.8. *Let $G(V, E)$ be an (n, d, α)-algebraic expander.*

1. *Then for every set $S \subseteq V$, if $|S| \leq \beta_1 n$, then $|\delta(S)| > |S|$.*
2. *Then for every set $S \subseteq V$, if $|S| \leq \beta_0 n$, then $|\delta(S)| > 8|S|$.*

Now we are ready to prove Lemma 3.3.

Proof of Lemma 3.3. The graph G is an (n, d, α)-algebraic expander; hence by Corollary 3.8, we get that for any set $S \subseteq V$, if $|S| \leq \beta_1 n$, then $|\delta(S)| > |S|$. Furthermore, by Lemma 3.6, there is a matching in G that covers S.

The formula Ψ has the form $\bigwedge_{v \in V} \psi_v$ where each ψ_v is a (possibly empty) conjunction of clauses in φ_v. Let S be the set of active vertices in Ψ, and fix a matching in G that covers S. Let ρ be the assignment to variables corresponding to this matching; namely, edges from the matching get value 1 and all other edges get value 0. For any vertex v covered by the matching, exactly one incident edge is assigned the value 1; hence ψ_v is true under ρ. For any other vertex v, all incident edges get value 0. Since v is not active, ψ_v is a subformula of φ'_v, so again by the definition of matching-like formulas, ψ_v is satisfied by ρ. Therefore, ρ satisfies Ψ. □

3.2 The proof of Lemma 3.4

We continue to have $G(V, E)$ an (n, d, α)-algebraic expander, Φ a matching-like formula based on G, and Ψ a satisfiable conjunction of clauses of Φ. Let S be the set of vertices active in Ψ, and assume S has size between $\beta_0 n/2$ and $\beta_0 n$. Finally, D is a 1-NBP representation of Ψ. We wish to show D has size $2^{\Omega(n)}$.

Note that for any matching M covering S, D must accept the truth assignment ρ_M corresponding to M. Let p_M be the (accepting) path traversed in D on input ρ_M. Fix a vertex v; p_M may query some of the inputs x_e for e an edge incident v. Since M is a matching at most one these x_e's has value 1 (for each fixed v). If $v \in S$, then v is active in Ψ, hence p_M must find an x_e to have value 1 for exactly one edge e incident to v.

As p_M is traversed, more-and-more edges e are found such that x_e has value 1. Each such edge e is incident to either one or two members of S: we say these members of S have been *covered* by p_M. At some point, p_M will have covered $\lceil S/2 \rceil - 1$ or $\lceil S/2 \rceil$ many members of S. Let u_M be the first node in D along p_M where this many vertices of S are covered; and let S_M be these covered vertices.

To prove Lemma 3.4, we lower bound the number of vertices equal to u_M for some matching M. For this, Lemma 3.9 shows that if two matchings M and M' lead to the same $u_M = u_{M'}$, then $S_M = S_{M'}$. We then exploit the fact that M cannot contain any edge between S_M and $S \setminus S_M$. Theorem 3.10 gives a distribution on matchings such that, for fixed M and randomly chosen M', the probability that $S_M = S_{M'}$ is exponentially small; it follows that there are exponentially many distinct nodes in D of the form u_M.

Lemma 3.9. *Let M_1 and M_2 be two matchings in G. If $u_{M_1} = u_{M_2}$, then $S_{M_1} = S_{M_2}$.*

Proof. For $i = 1, 2$, we let p'_{M_i} be the first portion of p_{M_i} leading up to u_{M_i} and p''_{M_i} be the remainder of p_{M_i}. Let ρ'_{M_i} and ρ''_{M_i} denote the partial assignments of values set along the paths p'_{M_i} and p''_{M_i}. Note that the path p'_{M_1} followed by p''_{M_2} is an accepting path. The values set along p'_{M_1} cover exactly S_{M_1} among the vertices of S. Since D is read-once, the values set along p''_{M_2} cover exactly $S \setminus S_{M_2}$ among the vertices of S, and cover nothing from S_{M_1}. Overall, the path covers all vertices of S; hence $S_{M_1} \cup (S \setminus S_{M_2}) = S$. Thus, $S_{M_2} \subseteq S_{M_1}$. Analogously, $S_{M_1} \subseteq S_{M_2}$. Hence, $S_{M_1} = S_{M_2}$. □

In the proof of Lemma 3.4 we will use the following theorem:

Theorem 3.10. *There is a probability distribution $\mathcal{D}$ on matchings covering S such that for every fixed subset $A \subseteq S$ with $|A| \in \{\lceil |S|/2 \rceil, \lceil |S|/2 \rceil - 1\}$, a randomly chosen matching from $\mathcal{D}$ contains an edge connecting a vertex of A with a vertex from $S \setminus A$ with probability $1 - 2^{-\Omega(n)}$.*

We will prove Theorem 3.10 in the next section.

Proof of Lemma 3.4. Let us fix a matching M covering S. It is clear that M does not have an edge connecting S_M and $S \setminus S_M$. Therefore

$$\begin{aligned}\Pr_{M' \sim \mathcal{D}}[u_M = u_{M'}] &\leq \Pr_{M' \sim \mathcal{D}}[S_M = S_{M'}] \\ &\leq \Pr_{M' \sim \mathcal{D}}[M' \text{ has no edge connecting } S_M \text{ and } S \setminus S_M].\end{aligned}$$

However,

$$\Pr_{M' \sim \mathcal{D}}[M' \text{ has no edge connecting } S_M \text{ and } S \setminus S_M] \leq 2^{-\Omega(n)}.$$

As a result, for any node u of D, $\Pr_{M' \sim \mathcal{D}}[u = u_{M'}] \leq 2^{-\Omega(n)}$; i.e., there are at least $2^{\Omega(n)}$ different nodes in D. □

3.3 The proof of Theorem 3.10

We start with an informal idea of the proof. Since S is large, we would expect that a random matching has $\Omega(n)$ edges with both endpoints in S. Likewise, since A is approximately half the size of S, we would expect that the set of edges between vertices in A and $S \setminus A$ forms a constant fraction of the edges which have both endpoints in S. Consider the following first attempt at generating a random matching:

```
1. Let $I$ be the set of edges which have both endpoints in $S$.
2. $M := \emptyset$.
3. While $I$ is not empty
     • Take $e \leftarrow I$ at random;
```

  - $M := M \cup \{e\}$;
  - Remove e and all the edges that share an endpoint with e from I.

4. Try to cover all the still uncovered vertices of S using a matching N joining them to vertices in $V \setminus S$.
5. Return $M \cup N$.

The described algorithm has the following problem: it may be impossible to implement the above step 4 since there may be no such matching N. The idea to fix this is as follows. We first choose a set of *bad* vertices $B \subseteq S$ so that there are a relatively small number of edges connecting B and $V \setminus S$. We argue that B is small, and then use Lemma 3.6 to generate a matching N covering B. Let B' be the set of vertices covered by N. We have $|B'| \leq 2|B|$, hence B' is also small. Since B' is small, step 3 of the random process described below will generate $\Omega(n)$ many edges $A \setminus B'$ and $S \setminus (A \cup B')$. The next proof carries out the details.

Proof of Theorem 3.10. Fix a maximal set of vertices $B \subseteq S$ such that at most $|B|$ many vertices in $V \setminus S$ have edges to vertices in B. Note that B exists, since the empty set satisfies this property. By the maximality of B, any $X \subseteq S \setminus B$ has more than $|X|$ many neighbors in $V \setminus S$, since otherwise $X \cup B$ would contradict the maximality of B.

Since $B \subseteq S$, $|B| \leq |S| \leq \beta_0 n$. Then by Corollary 3.8, $|\delta(B)| > 8|B|$. Since $|\delta(B) \cap (V \setminus S)| \leq |B|$, $|S \cap \delta(B)| \geq 7|B|$. Hence, $|S| \geq 8|B|$, i.e. $|B| \leq |S|/8$.

By Lemma 3.6, there exists a matching N that covers B, and w.l.o.g., every edge in N is incident to a vertex of B. Let B' be the set of vertices covered by N. Let $S' = S \setminus B'$. We have $|B'| \leq 2|B|$, so $|S'| \geq |S| - 2|B| \geq \frac{3}{4}|S|$.

The distribution $\mathcal{D}$ is defined by the following random process.

1. Let I be the set of the edges with both endpoints in S'.
2. $M := \emptyset$
3. While I is not empty
   - Take $e \leftarrow I$ uniformly at random;
   - $M := M \cup \{e\}$;
   - Remove e and all edges that have a common endpoint with e from I;
4. Choose a matching N' so that $M \cup N \cup N'$ is a matching which covers all the vertices of S.
5. Return $M \cup N \cup N'$.

It does not matter which N' is chosen in step 4.; however, we must show it exists. Let T be the subset of S' not covered by $M \cup N$. We show the desired N' exists using Hall's Theorem, applied to the bipartite subgraph of G induced by the two parts T and $V \setminus S$. Let $X \subseteq T$. We must show that X has at least $|X|$ many neighbors in $V \setminus S$. However, if this does not hold, the set $B \cup X$ violates the maximality of B.

Now fix $A \subseteq S$ such that $|A| \in \{\lceil |S|/2 \rceil, \lceil |S|/2 \rceil - 1\}$. We must show that a matching chosen by the distribution $\mathcal{D}$ has an edge between A and $S \setminus A$ with high probability. Setting $A' = S' \cap A$, it will suffice to show that M contains an edge between A' and $S' \setminus A'$ with high probability.

We have $|A'| \leq |A| \leq |S|/2 \leq \frac{2}{3}|S'|$ and $|A'| = |A \setminus B'| \geq |S|/2 - 1 - |B'| \geq |S|/4 - 1 \geq |S'|/4 - 1$. Let $\tau = |S'|/n$. It follows that, for sufficiently large n, $\tau n/5 \leq |A'| \leq 2\tau n/3$. Therefore, $\tau n/3 \leq |S' \setminus A'| \leq 4\tau n/5$. By the Expander Mixing Lemma,

$$|E(A', S' \setminus A')| \geq \frac{d}{n}|A'||S' \setminus A'| - \alpha d\sqrt{|A'||S' \setminus A'|} \geq dn(\tau^2/15 - 8\alpha\tau/15)$$

for large enough n. We need that $\tau^2/15 - 8\alpha\tau/15$ is positive and bounded away from zero. This holds since $\tau \geq \frac{3}{4}\frac{\beta_0}{2} > 8\alpha$ by Remark 3.2(iv), and hence $\tau - 8\alpha$ is positive and bounded away from zero. Thus

$|E(A', S' \setminus A')| = \Omega(n)$, so the edges in I between vertices in A' and $S' \setminus A'$ form a constant fraction of the edges in I.

We claim that with probability $1 - 2^{-\Omega(n)}$ the random process puts an edge from $E(A', S' \setminus A')$ into M. Initially I has $\Omega(n)$ many edges from $E(A', S' \setminus A')$. Each iteration of the loop removes at most $2d - 1$ many edges from I. Hence there are $\Omega(n)$ many iterations of the loop before half the edges of $E(A', S' \setminus A')$ are removed. Each of these iterations chooses an edge from $E(A', S' \setminus A')$ with probability $\Omega(1)$. Thus the probability that the final set M does not contain an edge from $E(A', S' \setminus A')$ is at most $2^{-\Omega(n)}$. □

3.4 Tseitin formulas

Theorem 3.11. *There exists $\alpha_0 < 1$ such that if $G(V, E)$ is an (n, d, α_0)-algebraic expander, then the size of any 1-NBP($\wedge$) refutation of an unsatisfiable Tseitin formula $\mathrm{TS}_{G,c}$ is $2^{\Omega(n)}$.*

Proof. A Tseitin formula is a matching-like formula if all vertex labels are equal to 1. Such a Tseitin formula is unsatisfiable if the number of vertices n is odd. Thus Theorem 3.1 implies the statement if c is identically one and n is odd.

To prove the lower bound for arbitrary labelling functions c, first suppose n is odd. It is well known that any two unsatisfiable Tseitin formulas based on the same connected graph can be obtained from each other by replacing some subset of the variables with their negations. (The idea is that the labels of two vertices can be flipped, by following a path connecting the vertices, and replacing edge variables along the path with their negations.)

Now suppose n is even. It is well known that a connected graph contains a vertex v which can be removed without disconnecting the graph. Thus it is sufficient to prove a lower bound for the graph $G - \{v\}$. However, $G - \{v\}$ may not be an expander so Theorem 3.1 does not directly apply. The proof of Theorem 3.1, however, does not use the algebraic expansion property directly. Instead, it uses the fact that algebraic expanders have good edge expansion properties, which are used to prove Corollary 3.8 and Theorem 3.10. It is possible to verify that these properties hold also for a graph that can be obtained from an algebraic expander by the removal of a vertex, by very slightly adjusting constants. Thus, the proof of Theorem 3.1 gives an exponential lower bound for proofs of the Tseitin formulas for the graph $G - \{v\}$. □

4 1-NBP($\wedge$) does not simulate tree-like resolution

This section proves results about how extension can affect the size of OBDD representations and OBDD($\wedge$, r) and 1-NBP($\wedge$) refutations. The first result, Theorem 4.1 is that extensions can provide an exponential improvement in the size of OBDD's; this shows a counterexample to [19, Lemma 4]. Then we adapt the idea of [19] and the earlier [33] and show in Lemma 4.5 that extension does not shorten 1-NBP($\wedge$) proofs. Consequently, we obtain examples where resolution has exponential speedup over 1-NBP($\wedge$) and hence over BP($\wedge$) and OBDD($\wedge$, r).

[19, Lemma 4] claimed that if φ is a CNF formula, E is a set of extension axioms for φ, then the minimal size of an τ-OBDD for φ is bounded by the size of any π-OBDD for $\varphi \wedge E$ where the order π extends τ. [33, Lemma 8] made a similar claim for a special case, but did not use any specific properties of the special case.

Theorem 4.1 gives a counterexample by exhibiting a Boolean function f and a set E of extension axioms so that $f \wedge E$ is representable by a short π-OBDD, but f requires an exponentially long τ-OBDD representation, where τ is the restriction of π to the original variables.

Theorem 4.1. *There are functions $f_n : \{0,1\}^m \to \{0,1\}$ with $m = O(n)$, and orders τ_n such that*

- *any τ_n-OBDD representation of f_n has size $2^{\Omega(n)}$, and*
- *for each n, there is a set E of extension axioms and an order π extending τ_n such that $f_n \wedge E$ has a π-OBDD representation of size $\mathrm{poly}(n)$.*

Proof. Let $t = \lfloor \log n \rfloor$ and $\ell = 2^t$. Let $f_n(y_1, \ldots, y_\ell, x_1, \ldots x_t)$ be the *index* function defined as

$$f_n(y_1, \ldots, y_\ell, x_1, \ldots x_t) \;=\; y_{\text{bin}(\vec{x})},$$

where $\text{bin}(\vec{x})$ means the integer with binary representation given by $x_1, \ldots, x_t$. (We could have also called f_n a "selection" function or "look up" function.) Note that $m = \lfloor \log n \rfloor + 2^{\lfloor \log n \rfloor} \leq 2n$.

We add extension variables $z_1, \ldots, z_t$ with the (rather trivial) set E of extension axioms $z_i \leftrightarrow x_i$. Let τ_n be the linear order placing all y_i's before all x_i's. Let π be the linear order placing all z_i's before all y_i's, and all y_i's before all x_i's, so that π extends τ_n.

It is easy to prove the first statement by observing that once the τ_n-OBDD has queried all the y_i's, it must remember all ℓ of the values of $y_1, \ldots, y_\ell$. To prove this, note that each setting to $y_1, \ldots, y_\ell$ gives a different function of $x_1, \ldots, x_t$. So the size of any τ_n-OBDD representation of f has size $2^{2^t} = 2^{\Omega(n)}$.

The second statement is proved by constructing the π-OBDD. The intuition is that the OBDD remembers all the z_j values, checks the appropriate y_i value, and compares the z_j values to the x_j values. The first stage uses $2^{t+1} - 1$ many nodes to query the variables z_j and remember all their values. The second stage uses exactly 2^t nodes, one per y_i, to query the needed value of y_i with $i = \text{bin}(\vec{z})$. If the queried y_i has value 0 (False), the OBDD outputs 0. Otherwise, the third stage checks the values of each x_j to see if it is equal to the corresponding z_j. It is obvious that this can be done with $t \cdot 2^t$ nodes, but by collapsing nodes, the third stage can even be done with $2^{t+1} - 1$ many nodes. The overall size of the π-OBDD is less than $5 \cdot 2^t = O(2^t) = O(n)$. □

We now switch to working with 1-NBP's, and show that tree-like resolution can have exponential speedup over 1-NBP proofs.

Theorem 4.2. *There are formulas Ψ_n of size* $\text{poly}(n)$ *such that any* 1-NBP($\wedge$) *refutation of Ψ_n has size at least $2^{\Omega(n)}$ and there is a tree-like resolution refutation of Ψ_n of size* $\text{poly}(n)$.

Since 1-NBP($\wedge$) proofs trivially simulate OBDD($\wedge$, r) proofs, we get:

Corollary 4.3. *There are formulas Ψ_n of size* $\text{poly}(n)$ *such that any* OBDD($\wedge$, r) *refutation of Ψ_n has size at least $2^{\Omega(n)}$ and there is a tree-like resolution refutation of Ψ_n of size* $\text{poly}(n)$.

Before proving Theorem 4.2, we show how to eliminate extension from 1-NBP's and 1-NBP($\wedge$) proofs. We do this in a fairly general way: recall that an extension axiom has the form $z \leftrightarrow h(\vec{x})$ where $h(\vec{x})$ is a conjunction of literals. Let $g(z, \vec{x})$ be $z \leftrightarrow h(\vec{x})$ so that $g(z, \vec{x})$ expresses the extension condition. In fact, the next lemma does not need any conditions on g and h except that $g(h(\vec{x}), \vec{x})$ is the constant 1, i.e., is true for $\vec{x}$. In particular, there is no requirement that g and h are easy to compute.

Lemma 4.4. *Let $f(\vec{x})$, $g(z, \vec{x})$ and $h(\vec{x})$ be Boolean functions. Assume that $g(h(\vec{x}), \vec{x})$ is the constant* 1. *If $f(\vec{x}) \wedge g(z, \vec{x})$ has a* 1-NBP *representation of size S, then $f(\vec{x})$ has a* 1-NBP *representation of size at most S.*

Proof. Let D be a 1-NBP for $f(\vec{x}) \wedge g(z, \vec{x})$. Modify D by changing all nodes labeled with z to guessing nodes; let D' be the resulting 1-NBP. We claim that D' is a 1-NBP for $f(\vec{x})$. First note that D' is still read-once. Consider fixed values $\vec{\alpha}$ for $\vec{x}$. Suppose $f(\vec{\alpha}) = 1$. Then, for $\beta = h(\vec{\alpha})$, $D(\beta, \vec{\alpha})$ accepts. The accepting path in D is also an accepting path for D', as desired. Now suppose $f(\vec{\alpha}) = 0$, but that D' has an accepting path. Since D' is read-once, it non-deterministically branches on z (at most) once. The corresponding accepting path in D requires z to have some value β (not necessarily equal to $h(\vec{\alpha})$). This path witnesses that $f(\vec{\alpha}) \wedge g(\beta, \vec{\alpha})$ is true, contradicting the assumption. □

Lemma 4.5. *Let φ be an unsatisfiable CNF formula and E be a set of extension axioms. Suppose $\varphi \wedge E$ has a* 1-NBP($\wedge$) *refutation of size S. Then φ has a* 1-NBP($\wedge$) *refutation of size at most S.*

Proof. Let φ use the variables $\vec{x}$. It is sufficient to assume E contains a single extension axiom, $z \leftrightarrow h(\vec{x})$. Assume that $\varphi \wedge (z \leftrightarrow h(\vec{x}))$ has a 1-NBP($\wedge$) refutation $D_1, \ldots, D_\ell$. Since the only rule is $\wedge$, each D_i is a conjunction of clauses from φ and possibly clauses from $z \leftrightarrow h(\vec{x})$. Let D_i' be a minimal 1-NBP expressing the conjunction of the clauses from φ that are used by D_i. By Lemma 4.4, D'_i has size at most the size of D_i. We claim that, after discarding any 1-NBP's equal to the constant 1, the remaining lines among $D'_1, \dots, D'_\ell$ form a valid 1-NBP($\wedge$) refutation for φ. It is easily checked that $\wedge$ inferences remain valid. Furthermore, since D_ℓ is the constant 0, so is D'_ℓ. □

It is well known that for every constant-degree graph $G(V, E)$ on n vertices and function $f : V \to \{0, 1\}$, if the Tseitin formula $\mathrm{TS}_{G,f}$ is unsatisfiable, then there exists a tree-like derivation of $\neg\mathrm{TS}_{G,f}$ in Extended Frege proof system of poly(n) size. Hence, there is a tree-like refutation of $\mathrm{TS}_{G,f}$ in extended resolution of poly(n) size. This can be formulated as:

Lemma 4.6. *Let G_n be an undirected graph with n vertices, with all vertices of degree at most d. Let f_n be a labeling function for G_n such that TS_{G_n,f_n} is unsatisfiable. Then there is a set E of extension axioms for TS_{G_n,f_n} of size* poly(n) *such that there is a tree-like resolution refutation of $\mathrm{TS}_{G_n,f_n} \wedge E$ of size* poly(n).

Proof of Theorem 4.2. Let G_n be an (n, d, α)-algebraic expander with $\alpha < 1/400$ and let f_n be a labeling function for G_n such that TS_{G_n,f_n} is unsatisfiable. Then by Theorem 3.1, the size of any 1-NBP($\wedge$) refutation of TS_{G_n,f_n} has size at least $2^{\Omega(n)}$. Let E be the set of extension axioms for TS_{G_n,f_n} from Lemma 4.6. By Lemma 4.5, any 1-NBP($\wedge$) refutation of $\mathrm{TS}_{G_n,f_n} \wedge E$ has size at least $2^{\Omega(n)}$. However, by Lemma 4.6, the formula $\mathrm{TS}_{G_n,f_n} \wedge E$ has a tree-like resolution refutation of size poly(n). □

5 Upper bounds for 1-NBP($\wedge$)

This section gives examples of formulas which require long OBDD($\wedge$, r) refutations, but have short 1-NBP($\wedge$) refutations, and even short 1-BP($\wedge$) refutations.

Theorem 5.1. *Let $\varphi = \bigwedge_{v \in V} \varphi_v$ be an unsatisfiable formula based on a bipartite graph $G(V, E)$. Suppose that, for all $v \in V$ there is a* 1-BP($\wedge$) *derivation of φ_v from its clauses of size at most S. Then there exists a* 1-BP($\wedge$) *refutation of φ of size* poly($|V|, S$).

Proof. Let V_1 and V_2 be the two parts of the bipartite graph G. We show that there are a 1-BP($\wedge$) derivation of $\bigwedge_{v \in V_i} \varphi_v$ of size poly($|V|, S$), for $i \in \{1, 2\}$. Note that for a fixed $i \in \{1, 2\}$, distinct formulas φ_v for $v \in V_i$ do not share variables. For each $v \in V_i$ we derive φ_v; the total size of these derivations is at most $|V_i|S$. Then we consequently derive the conjunction of the formulas φ_v for $v \in V_i$: by forming the conjunction of the first two, of the first three, etc. If two 1-BP's do not share variables, the size of their conjunction is at most the sum of sizes of initial 1-BPs. Hence size of the derivation of $\bigwedge_{v \in V_i} \varphi_v$ is at most $O(|V_i|^2 S)$.

After deriving $\bigwedge_{v \in V_i} \varphi_v$ for $i = 1$ and $i = 2$, we apply the conjunction rule and get the constant 0. □

Corollary 5.2. *Unsatisfiable instances of* PMP_G, $\mathrm{TS}_{G,f}$, PHP_G *over bipartite graphs G have polynomial size* 1-BP($\wedge$) *refutations.*

Theorem 5.3. *There are formulas ψ_n of size* poly(n) *such that*

- *any* OBDD($\wedge$, r) *refutation of ψ_n has size at least $2^{\Omega(n)}$ but*
- *there is a* 1-BP($\wedge$) *refutation of ψ_n of size* poly(n).

Proof. Let $K_{n,n+1}$ denote the complete bipartite graph on parts of size n and $n + 1$. Itsykson et al. [18] showed that OBDD($\wedge$, r) refutations of $\mathrm{PHP}_{K_{n,n+1}}$ require size $2^{\Omega(n)}$. However by Corollary 5.2, there are 1-BP($\wedge$) refutations of $\mathrm{PHP}_{K_{n,n+1}}$ of size poly(n). □

The formula $\mathrm{TS}_{K_{\log n},c} \circ \mathrm{Ind}^n_m$ is the composition of the Tseitin formulas on a complete graph on $\log n$ vertices with the standard indexing gadget. Theorem 23 of [7] showed that, for appropriate parameters, the lifted Tseitin formulas $\mathrm{TS}_{K_{\log n},c} \circ \mathrm{Ind}^n_m$ have OBDD($\wedge$) refutations which are polynomial size (in the size of the formula), but require quasipolynomial size cutting planes proofs. The lower bound for the cutting planes proofs was obtained using the translation from resolution width lower bounds to cutting planes refutation size lower bounds, based on the triangular-dag complexity arguments of Garg et al. [13].

A similar argument, applied to the graph pigeonhole principle, can improve this quasipolyonomial separation to an exponential separation. It is known that there are bipartite graphs G_n such that G_n has linearly many edges, and PHP_{G_n} is a k-CNF formula for $k = O(1)$ and requires $\Omega(n)$ resolution width [5]. By the same argument as in [7], we obtain lower bounds for the formulas ψ_n equal to $\mathrm{PHP}_{G_n} \circ \mathrm{Ind}^n_m$.

Proposition 5.4. *There are formulas ψ_n of size* $\mathrm{poly}(n)$ *such that*

- *cutting planes refutations of ψ_n require size at least $2^{n^{\Omega(1)}}$, but*
- *there are* 1-BP($\wedge$) *refutations of ψ_n of size* $\mathrm{poly}(n)$.

Theorem 24 of [7] used the lifted formula $\mathrm{TS}_{K_{\log n},c} \circ \mathrm{Ind}^n_m$, further transformed with the Segerlind transformation [28], to obtain formulas φ_n with polynomial size tree-like OBDD($\wedge$, r) refutations, but which require exponential size OBDD($\wedge$, w) refutations. The only role that the Tseitin principle plays in their constructions is that they require $\log^2 n$ width resolution refutations. By modifying their construction to use the graph pigeonhole formulas PHP_{G_n} instead of $\mathrm{TS}_{K_{\log n},c}$, we obtain formulas φ_n giving an exponential separation between OBDD($\wedge$, r) and OBDD($\wedge$, w):

Proposition 5.5. *There are formulas φ_n of size* $\mathrm{poly}(n)$ *such that*

- OBDD($\wedge$, w) *refutations of φ_n require size at least $2^{n^{\Omega(1)}}$, but*
- *there are* 1-BP($\wedge$) *refutations of φ_n of size* $\mathrm{poly}(n)$.

6 Lower bounds for 1-NBP($\wedge, \exists_\ell$)

The main result of this section is an exponential lower bound for 1-NBP($\wedge, \exists_{\epsilon n}$). First, however, we use a padding construction to give an example of formulas with short OBDD($\wedge, \exists_{\epsilon n}$) refutations, and hence short 1-NBP($\wedge, \exists_{\epsilon n}$) refutations, that require exponential size 1-NBP($\wedge$) refutations and thus exponential size OBDD($\wedge$, r) refutations. For $N > 0$, let Φ_N be a Tseitin formula based on an (N, d, α)-algebraic expander G for $\alpha < \alpha_0$, where α_0 is a constant satisfying Theorem 3.11. The number of edges in G is $\leq d \cdot N$; let $n = dN/\epsilon$ so that ϵn is greater than the number of variables in Φ_N. Let Ψ_N be Φ_N with $n - N$ many additional dummy variables. Then, by Theorem 3.11, 1-NBP($\wedge$) refutations of the formulas Ψ_N require size $2^{\Omega(n)}$. On the other hand, by [4, 9], there are polynomial size OBDD($\wedge, \exists$) refutations of Φ_N, and hence of Ψ_N. Since there are $\leq \epsilon n$ distinct variables in Φ_N, there are also polynomial size OBDD($\wedge, \exists_{\epsilon n}$) refutations of Ψ_N.

The construction of the exponential lower bounds on 1-NBP($\wedge, \exists_{\epsilon n}$) uses the following two steps. The first step is to show (in Lemma 6.1) that if a CNF φ has a 1-NBP($\wedge, \exists$) refutation of size S that uses projection ($\exists$) on only $\ell = \epsilon n$ many distinct variables, there is a way to fix the values of those ℓ variables so that the resulting formula has a 1-NBP($\wedge$) refutation of size at most S. The second, and more technical part of the proof, constructs Tseitin formulas $\mathrm{TS}_{G',f}$ which can remain hard for 1-NBP($\wedge$) refutations (by virtue of containing an expander as a subgraph) after any ϵn many vertices are removed from the graph G'. This second part is proved using a graph G' with suitably robust expansion properties.

Lemma 6.1. *Let φ be a CNF formula with a* 1-NBP($\wedge, \exists$) *refutation of size S and let X be the set of variables that are used in projection rules. Let ψ be the conjunction of the clauses of φ which contain at least one variable of X. Suppose ρ satisfies ψ. Then there exists a* 1-NBP($\wedge$) *refutation of $\varphi|_\rho$ of size at most S.*

Proof. Express φ in the form $\psi \wedge \theta$, where θ is the clauses of φ not containing variables of X. Consider a 1-NBP($\wedge, \exists$) refutation $D_1, D_2, \ldots, D_s$ of φ of size S. We claim that each line D_i is equivalent to a conjunction $E_i \wedge F_i$ where E_i is a conjunction of clauses of ψ, and F_i is true under the assignment ρ. (We allow the case that E_i or F_i is the constant true.) The claim is proved by induction on i. The base case, where D_i is a clause of either ψ or θ is trivial. The case where D_i is inferred by a $\wedge$ inference is also trivial. The case of a projection inference is also simple, since the projection acts on a variable x of X that does not appear in ψ and since applying projection preserves the property of being made true by ρ.

The sequence $D_1|_\rho$, $D_2|_\rho$, $\ldots$, $D_s|_\rho$ is a 1-NBP($\wedge, \exists$) refutation of $\psi|_\rho \wedge \theta|_\rho$; note $\psi|_\rho$ is the constant true. The projection rule no longer has any effect since ρ has fixed values for the variables X. In addition, the initial lines of this refutation are either clauses of $\varphi|_\rho$ or constant true clauses. Thus, it can be simplified to be a 1-NBP($\wedge$) refutation of just $\varphi|_\rho$. □

Now we state the main theorem of the section. Recall that, if F is a graph, then F^k denotes the k-th power of F, namely the graph on the vertices of F in which edges correspond to paths in F of length exactly k.

Theorem 6.2. *There are constants $k > 0$ and $\epsilon > 0$ such that the following holds. If $G(V, E)$ is an $(n, d, \frac{1}{24})$-algebraic expander, the graph F is formed from G by adding d self-loops to each vertex and f is an edge labelling of $G' = F^k$, then any* 1-NBP($\wedge, \exists_{\epsilon n}$) *refutation of* $\mathrm{TS}_{G',f}$ *has size* $2^{\Omega(n)}$.

Corollary 6.3. *Let $G(V, E)$ be an $(n, d, \frac{1}{24})$-algebraic expander. Let G' and ϵ satisfy the conditions of Theorem 6.2. Then any* OBDD($\wedge, \exists_{\epsilon n}$, r) *refutation of* $\mathrm{TS}_{G',f}$ *has size* $2^{\Omega(n)}$.

Corollary 6.3 follows immediately from the theorem. Theorem 6.2 will be proved from Theorem 6.6 below that, loosely speaking, shows how to form expander graphs that still have good expansion properties after removing ϵn vertices. The next Lemma 6.4 and its corollary are the heart of Theorem 6.6. Those are proved before we prove Theorem 6.2.

Lemma 6.4. *Let $G(V, E)$ be an (n, d, α)-algebraic expander with $\alpha \leq 1/24$. Let $A \subseteq V$ with $|A| \leq n/8$. Then there is a set U of size at most $\frac{1}{8}n$ such that $G - (A \cup U)$ is an $(n/2, d/8)$-edge expander.*

Proof. Let U be a maximal subset of V of size $\leq 5n/8$ such that

$$|E(U, V \setminus (A \cup U))| \leq \frac{1}{8} d|U|. \tag{1}$$

This condition means, roughly, that U fails to have large expansion within $G - A$. (It is permitted that U and A intersect.) Note that $U = \emptyset$ satisfies (1), so such a U exists.

We shall first prove U is not too large, namely $|U| \leq n/8$. Let $B = V \setminus (A \cup U)$. By Lemma 2.8,

$$|E(U, B)| \ \geq \ \frac{d}{n}|U||B| - \alpha d\sqrt{|U||B|}.$$

By the size bounds on A and U, we have $|B| \geq n/4$. Thus,

$$|E(U, B)| \ \geq \ \frac{d}{4}|U| - \alpha d\sqrt{|U|n}.$$

Also, by (1), $|E(U, B)| \leq d|U|/8$. Therefore, $\frac{1}{8}|U| \geq \frac{1}{4}|U| - \alpha\sqrt{|U|n}$. Simplifying this gives $\alpha\sqrt{n} \geq \sqrt{|U|}/8$, hence $|U| \leq 64\alpha^2 n < n/8$.

We now prove that $G - (A \cup U)$, namely G restricted to the vertices B, has the desired edge expansion. Let $S \subseteq B$ have size at most $n/2$. Assume, for sake of contradiction, that $|E(S, B \setminus S)| < d|S|/8$. Set $H = U \cup S$, so that $|H| = |U| + |S| \leq \frac{1}{8}n + \frac{1}{2}n = \frac{5}{8}n$. Then

$$\begin{aligned} |E(H, V \setminus (A \cup H))| &= |E(U, V \setminus (A \cup H)| + |E(S, V \setminus (A \cup H))| \\ &\leq |E(U, V \setminus (A \cup U))| + |E(S, B \setminus S)| \\ &< \frac{1}{8}d|U| + \frac{1}{8}d|S| \ = \ \frac{1}{8}d|H|. \end{aligned}$$

The property of H contradicts the maximality of U. □

Corollary 6.5. *Let $G(V,E)$ be an $(n,d,1/24)$-algebraic expander. Let $A \subseteq V$ and $|A| \le n/8$, and U be as in Lemma 6.4. Form a $(d{+}1)$-regular graph H from $G-(A\cup U)$ by adding up to $d+1$ self loops to every vertex. Then H is a $(n-|A\cup U|, d+1, 1{-}\gamma)$-algebraic expander, where $\gamma = \min\{\frac{1}{128}, \frac{2}{d+1}\}$.*

The corollary is proved by noting that each vertex in H has at least one self loop, and applying Lemma 2.10.

Theorem 6.6. *Let $G(V,E)$ be an $(n,d,1/24)$-algebraic expander and k be a positive integer. Form the graph F from G by adding $d+1$ self loops to each vertex. Assume that $A\subseteq V$ has size at most $n/8$. Then there is a $U\subseteq V$ of size at most $n/8$, such that $F^k-(A\cup U)$ has a subgraph (obtained by removing edges) which is an $(n-|A\cup U|, (d+1)^k, (1{-}\gamma)^k)$-algebraic expander, for $\gamma=\min\{\frac{1}{128},\frac{2}{d+1}\}$.*

Proof of Theorem 6.6. Fix G, k, and A. Let $U\subseteq V$ and H be as in Lemma 6.4 and Corollary 6.5. Since H was formed by adding up to $d{+}1$ self loops to each vertex, H is a subgraph of $F-(A\cup U)$. Therefore H^k is a subgraph of $(F-(A\cup U))^k$, which in turn is a subgraph of $F^k-(A\cup U)$. Since H is an $(n-|A\cup U|, d{+}1, 1{-}\gamma)$-algebraic expander, H^k is an $(n-|A\cup U|, (d+1)^k, (1{-}\gamma)^k)$-algebraic expander. Thus H^k is the desired subgraph of $F^k-(A\cup U)$. □

Proof of Theorem 6.2. Let $\epsilon = 1/16$. Consider a 1-NBP$(\wedge, \exists_{\epsilon n})$ refutation of $\mathrm{TS}_{G',f}$ of size S. We wish to show S is $2^{\Omega(n)}$. There are at most ϵn many edge variables of $\mathrm{TS}_{G',f}$ used in projection inferences. Let A be the set of vertices incident to these edges, so $|A|$ is at most $2\epsilon n = n/8$.

Let $\gamma = \min\left\{\frac{1}{128},\frac{2}{d+1}\right\}$ as in Theorem 6.6. Let k be the minimal integer number such that $(1{-}\gamma)^k \le \alpha_0$. Recall α_0 is the constant from Theorem 3.11. By Theorem 6.6, there exists a set of vertices $U\subseteq V$ of size at most $n/8$ such that the graph $G'-(A\cup U)$ has a subgraph H' which is an $(n-|A\cup U|, (d+1)^k, (1{-}\gamma)^k)$-algebraic expander. (Note that H' is the H^k of the proof of Theorem 6.6.)

Since G' is connected and $A\cup U$ is not all of V, there is a truth assignment ρ to the variables incident to vertices in $A\cup U$ which satisfies the the clauses of $\mathrm{TS}_{G',f}$ expressing the parity conditions for the vertices in $A\cup U$. By Lemma 6.1, there is a 1-NBP$(\wedge)$-refutation of $(\mathrm{TS}_{G',f})|_\rho$ of size most S. Furthermore, $(\mathrm{TS}_{G',f})|_\rho$ is identical to $\mathrm{TS}_{G'-(A\cup U),f'}$ where f' is the labelling of the edges of $G'-(A\cup U)$ obtained by updating the parities of vertices according to the partial assignment ρ.

Finally, by assigning value 0 to all edges in $G'-(A\cup U)$ but not in H', we obtain a refutation of $\mathrm{TS}_{H',f'}$ of size S. By Theorem 3.11, $S=2^{\Omega(n)}$. □

7 Lower bounds on $\mathrm{OBDD}(\wedge, \mathrm{w}, \mathrm{r}_\ell)$

Theorem 7.1. *For $\ell > 0$, there are formulas φ_n of size $n^{\mathrm{poly}(\ell)}$ and using $\mathrm{poly}(\ell n)$ variables such that tree-like $\mathrm{OBDD}(\wedge,\mathrm{w},\mathrm{r}_\ell)$ refutations of the formulas φ_n have size $2^{\Omega\left(n^{1/8}/(2^{\ell/2}\sqrt{\ell})\right)}$, and such that the φ_n's have tree-like $\mathrm{OBDD}(\wedge,\mathrm{r})$ refutations of size $\mathrm{poly}(|\varphi_n|)$.*

The theorem is meaningful only for $\ell \le \epsilon\log n$. In this case, φ_n has quasipolynomial size, and the lower bound $2^{\Omega\left(n^{1/8}/(2^{\ell/2}\sqrt{\ell})\right)}$ is $2^{\Omega\left(n^{1/8-\epsilon}\right)}$ which is exponential in the size of φ_n. Also, the number ℓ of orders allowed in the refutations can be as large as $(\log|\varphi_n|)^\delta$ for some constant $\delta>0$.

The proof of Theorem 7.1 is based on the next two theorems. Theorem 7.2 converts a refutation using ℓ orders to an $(\ell{+}1)$-party communication protocol of small complexity. Theorem 7.3 constructs formulas, such that for every balanced partition of size k, the associated search problem has large communication complexity. The latter theorem will be proved by combining lifting techniques from [15] and [20].

Theorem 7.2. *Suppose φ has a tree-like $\mathrm{OBDD}(\wedge,\mathrm{w},\mathrm{r}_\ell)$ refutation of size S. Then there is a balanced partition Π of the variables of φ into $\ell+1$ subsets such that $\mathbf{D}(\mathrm{Search}_\varphi,\Pi) = O(\log^2 S)$.*

Theorem 7.3. *Let $k>0$. There are formulas φ_n on $\mathrm{poly}(kn)$ variables of size $n^{\mathrm{poly}(k)}$ such that $\mathbf{D}(\mathrm{Search}_{\varphi_n},\Pi) = \Omega(\sqrt[4]{n}/(2^k k))$ for every balanced partition Π into k subsets.*

Later, Proposition 7.11 will show that these formulas φ_n have polynomial size tree-like OBDD($\wedge$, r) refutations.

Theorem 7.1 is proved by noting that, with $k = \ell + 1$, the formulas φ_n of Theorem 7.3 satisfy the desired properties. Namely, since $\mathbf{D}(\mathrm{Search}_{\varphi_n}, \Pi) = \Omega(\sqrt[4]{n}/2^k k)$, Theorem 7.2 implies that any tree-like refutation of φ_n has size S satisfying $(\log S)^2 = \Omega(\sqrt[4]{n}/2^k k)$. From this, $S = 2^{\Omega\left(n^{1/8}/(2^{k/2}\sqrt{k})\right)}$.

7.1 Lower bounds for multiparty complexity

This section proves Theorem 7.3. We start with a lower bound for communication complexity proven by Göös and Pitassi [15]. Their construction uses a pebbling formula.

Definition 7.4. Let G be a directed acyclic graph with a single sink t. The *pebbling formula* Peb_G for G is the CNF formula which uses the variables x_v for v a vertex of G and has the clauses:

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

It is not hard to see that Peb_G has short tree-like OBDD($\wedge$) proofs, even though G is a dag:

Theorem 7.5 ([7]). *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* Peb_G *of size* $\mathrm{poly}(|\mathrm{Peb}_G|)$*, i.e., of size* $\mathrm{poly}(n, d)$*.*

Additionally, we need the concept of composition of CNF's; this is central to the notion of "lifting" query complexity bounds to communication complexity bounds. Let $\varphi(x_1, \ldots, x_n)$ and $g(y_1, \ldots, y_s)$ be CNF formulas. Then $\varphi \circ g^n$ denotes the CNF formula obtained from φ by replacing each variable x_i with $g(y_{i,1}, \ldots, y_{i,s})$, and using De Morgan's rules and distributivity to make it a CNF formula.

In our applications, g will depend on only a small number s of inputs. In this case, both $g(y_1, \ldots, y_s)$ and its negation can be written as a conjunction of at most 2^s disjunctions of size at most s. Thus, if φ is an r-CNF, the size of the CNF $\varphi \circ g^n$ is bounded by $|\varphi| \cdot s \cdot (2^s)^r$.

Theorem 7.6. *Fix $k > 0$. There are constant degree, directed acyclic graphs G_n on n vertices, a CNF formula g on $s = s(k) = k^{1+o(1)}$ variables, and partitions Π_n such that*

- $|\mathrm{Peb}_{G_n} \circ g^n| = \mathrm{poly}(n, 2^s)$*;*
- Π_n *is a partition of the $n \cdot s$ many variables of* $\mathrm{Peb}_{G_n} \circ g^n$ *into k subsets; and*
- $\mathbf{D}(\mathrm{Search}_{\mathrm{Peb}_{G_n} \circ g^n}, \Pi_n) = \Omega(\sqrt[4]{n}/(2^k k))$.

Proof. This a direct corollary of results of Göös and Pitassi [15] and Sherstov [29] (see also Rao and Yehudayoff [26]). The *disjointness* function $\mathrm{DISJ}_{k,b} : (\{0,1\}^n)^k \to \{0,1\}$ is

$$\mathrm{DISJ}_{k,n}(\vec{x}_1, \ldots, \vec{x}_k) \;=\; \bigwedge_{j=1}^{n} \bigvee_{i=1}^{k} \neg x_{i,j}$$

where each $\vec{x}_i$ is a tuple of n bits; this value is 1 if the k sets coded by the $\vec{x}_i$'s have empty intersection (k-party disjointness). Unique disjointness, $\mathrm{UDISJ}_{k,n}$, is the same function, but with the promise that the intersection of the k sets has cardinality at most one. This is expressed as a NOF k-party communication problem using the partition Γ where the ith player sees all but the inputs x_i: the players only need to succeed on inputs satisfying the promise of uniqueness. Sherstov [29] proved that this has randomized communication complexity $\mathbf{D}^{\mathrm{rand}}(\mathrm{UDISJ}_{k,n}, \Gamma) = \Omega(\sqrt{n}/2^k k)$.

Göös and Pitassi [15, Theorem 2 and Theorem 4] showed there are graphs G_n, a CNF formula g and partitions Π_n satisfying the conditions of the theorem, except with $\mathbf{D}^{\mathrm{rand}}(\mathrm{Search}_{\mathrm{Peb}_{G_n} \circ g^n}, \Pi_n) \geq \mathbf{D}^{\mathrm{rand}}(\mathrm{UDISJ}_{k,b}, \Gamma)$, where $b = \Theta(\sqrt{n})$. Theorem 7.6 follows immediately. Göös and Pitassi give several possible g's, called "versatile gadgets"; their best asymptotic for g is $s(k) = k^{1+o(1)}$. □

In order to prove Theorem 7.3, we need to extend Theorem 7.6 to give lower bounds that hold for *all* partitions. For this, we recall a transformation introduced by Segerlind [28]. Let $t = \lceil \log n \rceil$ and $\mathbb{F}$ be the field $\mathrm{GF}(2^t)$. $\mathbb{F}$ has 2^t elements that are identified with the elements of $[2^t]$. Let $\mathcal{P}_t$ be the set of all mappings given by $x \mapsto ax + b$ with $a, b \in \mathbb{F}$ and $a \neq 0$. Elements of $\mathcal{P}_t$ may be represented by binary strings of length $2t$ such that the first t bits are not all zero; the i-th bit of the representation of $\sigma \in \mathcal{P}_t$ is denoted $\mathrm{rep}(\sigma)_i$. The idea is that $\mathcal{P}_t$ is a set of pairwise independent permutations of $\mathbb{F}$. A permutation σ in $\mathcal{P}_t$ acts on the input variables by mapping x_i to $x_{\sigma(i)}$. For this, since $n \leq 2^t = |\mathbb{F}|$, we need to add new dummy variables $x_{n+1}, \ldots, x_{2^t}$. For φ a CNF on n variables, define

$$\mathrm{perm}_{\varphi}(z_1, \ldots, z_{2t}, x_1, \ldots, x_{2^t}) \ = \ \bigwedge_{\sigma \in \mathcal{P}_t} \left[\left(\bigwedge\nolimits_{i=1}^{2t} z_i = \mathrm{rep}(\sigma)_i \right) \to \varphi(x_{\sigma(1)}, \ldots, x_{\sigma(n)}) \right] \ \wedge \ \bigvee\nolimits_{i=1}^{t} z_i.$$

Further let $\varphi^{\vee_m}(y_{1,1}, \ldots, y_{n,m})$ denote the CNF formula that is obtained from φ by replacing each x_i by the disjunction of m fresh variables $y_{i,1}, \ldots, y_{i,m}$. (We could equivalently denote $\varphi^{\vee_m}$ using "lifting" notation as the composition $\varphi \circ (\vee_m)^n$.)

Now we can define the transformation $\mathcal{T}_k$ of a CNF formula $\varphi(x_1, \ldots, x_n)$. Let $k > 0$ be an integer (k will be the number of players), and define $m = m(k, n)$ to be the least integer satisfying $\frac{2k^2 n}{m} + \frac{kn}{mn-1} < 1$; note $m = O(k^2 n)$. Then $\mathcal{T}_k(\varphi)$ is the formula $\mathrm{perm}_{\varphi^{\vee_m}}$.

Theorem 7.7. *Let $k > 0$. For every φ with sufficiently many variables, every k-partition Π of the variables of φ, and every balanced k-partition Γ of the variables of $\mathcal{T}_k(\varphi)$, we have* $\mathbf{D}(\mathrm{Search}_{\mathcal{T}_k(\varphi)}, \Gamma) \geq \mathbf{D}(\mathrm{Search}_{\varphi}, \Pi)$.

We need the following two standard lemmas; the first formalizes the fact that the set $\mathcal{P}_t$ consists of pairwise independent permutations.

Lemma 7.8 ([8])**.** *$\mathcal{P}_t$ contains $2^t(2^t - 1)$ permutations. For any $x_1 \neq x_2$ and $y_1 \neq y_2$ in $\mathbb{F}$, then* $\Pr_{\pi \in \mathcal{P}_t}[\pi(x_1) = y_1 \text{ and } \pi(x_2) = y_2] = \frac{1}{2^t(2^t-1)}$.

Lemma 7.9 (Chebyshev's inequality)**.** *If $X_1, \ldots, X_t$ are random Boolean variables and $Y = \sum_{i=1}^{t} X_i$, then*

$$\Pr[Y = 0] \ \leq \ \frac{\mathbb{E}Y + \sum_{i \neq j \in [t]} \mathrm{Cov}(X_i, X_j)}{(\mathbb{E}Y)^2}.$$

Proof of Theorem 7.7. Let φ be a CNF with variables x_i for $i \in [n]$. Let $m = m(k, n)$. The formula $\varphi^{\vee_m}$ has $N = n \cdot m$ many variables $y_{i,j}$. We also name these variables as $v_1, \ldots, v_N$, and set $t = \lceil \log N \rceil$. Then $\mathcal{T}_k(\varphi)$ is a CNF formula on the variables z_i for $i \in [2t]$ and the variables v_i for $i \in [2^t]$.

Let $\Pi = (\Pi_1, \ldots, \Pi_k)$ be an arbitrary k-partition of the variables of φ. Suppose $\Gamma = (\Gamma_1, \ldots, \Gamma_k)$ is a balanced k-partition of the variables of $\mathcal{T}_k(\varphi)$, and there is a k-party protocol for $\mathrm{Search}_{\mathcal{T}_k(\varphi)}$ with respect to Γ, which has communication complexity S. We need to show there is a protocol for $\mathrm{Search}_{\varphi}$ with respect to Π also of communication complexity S.

The variables of $\varphi^{\vee_m}$ are grouped into the blocks: the i-th block is the variables $y_{i,j}$. The next lemma states that we can find a permutation in $\mathcal{P}_t$ that sends representatives from every block to every member of the partition.

Claim 7.10. *There is a permutation $\pi \in \mathcal{P}_t$ such that, for any $i \in [n]$ and $\ell \in [k]$, there is a $j \in [m]$ so that $y_{i,j}$ is mapped to Γ_ℓ by π.*

Theorem 7.7 follows almost immediately from the claim. Fix a π satisfying Claim 7.10. Let $v_{r(i,\ell)}$ be the variable $y_{i,j}$ given by the claim. Define a substitution ρ on the variables of $\mathcal{T}_k(\varphi)$ by setting the values of $\rho(z_i)$ to encode the permutation π, and setting

- $\rho(v_{r(i,\ell)}) = x_i$ if $x_i \in \Pi_\ell$, and

- $\rho(v_j) = 0$ in all other cases.

The protocol for Search_φ with respect to Π runs as follows: It runs the protocol for $\text{Search}_{\mathcal{T}_k(\varphi)}$ on the values of the z_i's and v_i's as set by ρ. This is a valid k-party protocol with respect to Π since the variables $v_{r(i,\ell)}$ and x_i are both in the ℓ-th member of their respective partitions, and all other variables are set to 0. It correctly solves Search_φ since the value of $y_{i,1} \vee \cdots \vee y_{i,m}$ under ρ is equal to x_i; therefore $\mathcal{T}_k(\varphi)|_\rho$ is equivalent to φ. Since π and ρ can be computed without any communication, the two protocols have the same communication complexity. Thus, $\mathbf{D}(\text{Search}_\varphi, \Pi) \le \mathbf{D}(\text{Search}_{\mathcal{T}_k(\varphi)}, \Gamma)$.

We now prove Claim 7.10. Γ is a partition of the z_i's and v_i's; it induces a k-partition $\Gamma' = (\Gamma'_1, \ldots, \Gamma'_k)$ of the v_i's. Γ' is not balanced, but is near-balanced since each Γ'_ℓ has size at least $\lfloor N/k \rfloor - 2t$ since Γ is balanced and there are $2t$ many z_i's.

Choosing $\sigma \in \mathcal{P}_t$ uniformly at random, let $\chi^\ell_{i,j}$ be the Boolean random variable such that $\chi^\ell_{i,j} = 1$ iff $y_{i,j}$ is mapped by π into Γ'_ℓ. Set $Y^k_i = \sum_{j=1}^m \chi^k_{i,j}$. By Lemma 7.8, $\chi^\ell_{i,j}$ has expectation equal to $\frac{|\Gamma'_\ell|}{N}$, so by additivity of expectation, the expected value of Y^ℓ_i is equal to $\frac{m|\Gamma'_\ell|}{N}$. For $j_0 \neq j_1$,

$$\begin{aligned}
\text{Cov}(\chi^\ell_{i,j_0}, \chi^\ell_{i,j_1}) &= \mathbb{E}\big(\chi^\ell_{i,j_0} \cdot \chi^\ell_{i,j_1}\big) - \mathbb{E}\chi^\ell_{i,j_0}\mathbb{E}\chi^\ell_{i,j_1} \\
&= \sum_{v_{i'} \neq v_{i''} \in \Gamma'_\ell} \Pr[v_{\sigma(i')}{=}y_{i,j_0} \text{ and } v_{\sigma(i'')}{=}y_{i,j_1}] - \frac{|\Gamma'_\ell|^2}{N^2} \\
&= \frac{|\Gamma'_\ell|(|\Gamma'_\ell|{-}1)}{N(N{-}1)} - \frac{|\Gamma'_\ell|^2}{N^2} \\
&< \frac{|\Gamma'_\ell|^2}{N}\left(\frac{1}{N-1} - \frac{1}{N}\right) \;=\; \frac{|\Gamma'_\ell|^2}{N^2(N-1)} \;=\; \frac{\big(\mathbb{E}Y^\ell_i\big)^2}{m^2(N-1)}.
\end{aligned}$$

Hence, by Lemma 7.9,

$$\begin{aligned}
\Pr[Y^\ell_i = 0] &\le \frac{\mathbb{E}Y^\ell_i + \sum\limits_{j_0 \neq j_1 \in [m]} \text{Cov}(\chi^\ell_{i,j_0}, \chi^\ell_{i,j_1})}{\big(\mathbb{E}Y^\ell_i\big)^2} \;\le\; \frac{N}{m|\Gamma'_\ell|} + \frac{m(m-1)}{m^2(N-1)} \\
&\le \frac{N}{m\left(\lfloor N/k \rfloor - 2t\right)} + \frac{1}{N-1} \;\le\; \frac{N}{m\frac{N}{2k}} + \frac{1}{N-1} \;=\; \frac{2k}{m} + \frac{1}{nm-1},
\end{aligned}$$

where the fourth inequality used $N/(2k)) > 2t$ for n sufficiently large.

By the union bound, the probability that some Y^ℓ_i is equal to zero is at most $\frac{2k^2 n}{m} + \frac{kn}{nm-1} < 1$. Therefore, there is π in $\mathcal{P}_t$ such that no Y^ℓ_i is zero. This π satisfies Claim 7.10. □

Proof of Theorem 7.3. We let φ_n be the formula $\mathcal{T}_k(\text{Peb}_{G_n} \circ g^n)$ with G_n and g as in Theorem 7.6. Letting Π_n be the partition from that theorem, $\mathbf{D}(\text{Search}_{\text{Peb}_{G_n} \circ g^n}, \Pi_n) = \Omega(\sqrt[4]{n}/2^k k)$. Thus, by Theorem 7.7, for every balanced k-partition Γ, $\mathbf{D}(\text{Search}_{\varphi_n}, \Gamma) = \Omega(\sqrt[4]{n}/2^k k)$.

Next we estimate the size of φ_n. G_n has constant degree and n vertices, so Peb_{G_n} is an r-CNF of size $O(rn)$ on n variables for constant r. The versatile gadget g can be chosen to use only $s = k^{1+o(1)} = \text{poly}(k)$ variables. Therefore $\text{Peb}_{G_n} \circ g^n$ is an (sr)-CNF of size $2^{sr} s \cdot |\text{Peb}_{G_n}|$ on $s \cdot n$ variables. Since $s = \text{poly}(k)$ and r is constant, $\text{Peb}_{G_n} \circ g^n$ is a $\text{poly}(k)$-CNF of size $S' = 2^{\text{poly}(k)} n$ on $n' = \text{poly}(k) \cdot n$ variables.

Define the formula ψ_n to be $(\text{Peb}_{G_n} \circ g^n)^{\vee m}$ with $m = O(k^2 \cdot n')$ as used for the transformation $\mathcal{T}_k$. The formula ψ_n is a $m \cdot \text{poly}(k)$-CNF of size $m^{\text{poly}(k)} \cdot S'$ on $m \cdot n'$ variables. Thus ψ_n is a CNF of size $n'' = n^{\text{poly}(k)}$ and uses $\text{poly}(k) \cdot n^2$ many variables. Finally, φ_n is perm_{ψ_n}. Thus φ_n is a CNF formula of size $O((n'')^2 \cdot \psi_n) = n^{\text{poly}(k)}$ using $\text{poly}(kn)$ variables. □

Proposition 7.11. *The formulas φ_n of Theorem 7.3 have tree-like* $\text{OBDD}(\wedge, \text{r})$ *refutations of the formulas φ_n of size* $\text{poly}(n, k)$ *and thus of size polynomial in $|\varphi_n|$.*

The proposition is proved using the following result of [7, Lemma 2, Corollary 6, Theorem 5]:

Theorem 7.12. *Let φ_n be unsatisfiable r-CNFs on $\ell = \mathrm{poly}(n)$ variables such that there are tree-like $\mathrm{OBDD}(\wedge)$ refutations of φ_n of size S. Then:*

(i) There is a tree-like $\mathrm{OBDD}(\wedge, \mathrm{r})$ refutation of $\mathrm{perm}_{\varphi_n}$ of size $\mathrm{poly}(n) \cdot S$.

(ii) For every positive integer m, there are tree-like $\mathrm{OBDD}(\wedge)$ refutations of $\varphi_n^{\vee_m}$ of size $\mathrm{poly}(|\varphi_n^{\vee_m}|, S, m)$.

(iii) For every formula g on m variables, there are tree-like $\mathrm{OBDD}(\wedge)$ refutations of $\varphi_n \circ g^\ell$ of size $\mathrm{poly}(|\varphi_n \circ g^\ell|, S, 2^s)$.

We apply all three parts of this theorem to the formulas $\varphi_n := \mathcal{T}_k(\mathrm{Peb}_{G_n} \circ g^n)$ with G_n and g as in Theorem 7.6. By Theorem 7.5, there are tree-like $\mathrm{OBDD}(\wedge)$ refutations of Peb_{G_n} of size $\mathrm{poly}(n)$. Hence, by Proposition 7.12(iii) there are tree-like $\mathrm{OBDD}(\wedge)$ refutations of $\mathrm{Peb}_{G_n} \circ g^n$ of size $\mathrm{poly}(n)$. By the above analysis, the formulas $\mathrm{Peb}_{G_n} \circ g^n$ have size $2^{\mathrm{poly}(k)}n$ and the formulas $(\mathrm{Peb}_{G_n} \circ g^n)^{\vee_m}$ have size $n^{\mathrm{poly}(k)}$ where $m = \mathrm{poly}(k, n)$. Hence by 7.12(ii), the formulas $(\mathrm{Peb}_{G_n} \circ g^n)^{\vee_m}$ have tree-like $\mathrm{OBDD}(\wedge)$ refutations of size $n^{\mathrm{poly}(k)}$. Finally, by 7.12(i), the formulas $\mathcal{T}_k(\mathrm{Peb}_{G_n} \circ g^n)$ have tree-like $\mathrm{OBDD}(\wedge, \mathrm{r})$ refutations of size $n^{\mathrm{poly}(k)}$. This proves Proposition 7.11.

7.2 Upper bounds for multiparty complexity

This section proves Theorem 7.2; this completes the proof of Theorem 7.1. We write $\pi[\le s]$ to denote the set containing the first s elements of an ordering π, and $\pi[>s]$ to denote the remaining elements of π.

Lemma 7.13. *Let $\pi_1, \ldots, \pi_\ell$ be orderings of the variables $x_1, \ldots, x_n$. Then there are $s_1, \ldots, s_\ell \in [n]$ and a partition $\Pi = (\Pi_1, \ldots, \Pi_{\ell+1})$ of the variables $x_1, \ldots, x_n$ into $\ell + 1$ subsets such that*

- *Π is a balanced partition (that is, $|\Pi_i| \ge \lfloor \frac{n}{\ell+1} \rfloor$ for each $i \in [\ell + 1]$);*
- *for every $i \in [\ell]$, $\pi_i[\le s_i] \cap \Pi_{i+1} = \emptyset$ and $\pi_i[> s_i] \cap \Pi_i = \emptyset$.*

Proof. The partition is constructed by the following algorithm:

- $S_1 := \{x_1, x_2, \ldots, x_n\}$;
- For $i = 1$ to ℓ
  - Let Π_i be the first $\lfloor \frac{n}{\ell+1} \rfloor$ elements of S_i in the order π_i.
  - Let s_i be the maximal index of an element of Π_i in the order π_i. That is, s_i is the minimal value such that $\Pi_i \subseteq \pi_i[\le s_i]$
  - Set $S_{i+1} := S_i \setminus \Pi_i$.
- $\Pi_{\ell+1} := S_{\ell+1}$

By the construction, $|\Pi_i| = \lfloor \frac{n}{\ell+1} \rfloor$ for $i \in [\ell]$, and hence $|\Pi_{\ell+1}| \ge \lfloor \frac{n}{\ell+1} \rfloor$. Note that Π_i and s_i are defined so that $\pi_i[>s_i] \cap \Pi_i = \emptyset$ and $\pi_i[\le s_i] \cap S_{i+1} = \emptyset$. Since $\Pi_{i+1} \subseteq S_{i+1}$, we have $\pi_i[\le s_i] \cap \Pi_{i+1} = \emptyset$. □

Lemma 7.14. *Let a function f be computed by a π-OBDD D, $s \in [n]$ be an integer, and Π be a partition of variables of f into k subsets such that $\Pi_a \cap \pi[\le s] = \Pi_b \cap \pi[> s] = \emptyset$ for some $a, b \in [k]$. Then $\mathbf{D}(f, \Pi) \le \lceil \log |D| \rceil + 1$.*

Proof. Player a knows the first s variables in the order π, and starts the computation of D according D using the variables she knows, i.e., using the variables outside of Π_a. She reaches a vertex v of D after reading all the variables $\pi[\le s]$ and sends the number of the vertex v to Player b, using $\lceil \log |D| \rceil$ bits. Player b continues computing D starting from v using now variables he knows and sends the result of the computation as a single bit. □

Proof of Theorem 7.2. Fix a tree-like OBDD$(\wedge, \mathrm{w}, \mathrm{r}_\ell)$ refutation $D_1, \ldots, D_m$ of the formula φ of size S. This proof uses only ℓ distinct orders $\pi_1, \ldots, \pi_\ell$ over the variables of φ, so each D_i is a π_j-OBDD for some $j \in [\ell]$.

Let Π be a partition satisfying Lemma 7.13. We construct a $(\ell + 1)$-party communication protocol for Search$_\phi$ with respect to the partition Π of complexity at most $O(\log^2 S)$. The protocol consists of $s = O(\log S)$ steps. At the i-th step in the protocol, there is a tree T_i that is known by all the players. The inner vertices of T_i are labelled with OBDD's from the proof of φ; the leaves of T_i are labelled with clauses of ϕ or with the constant 1. We maintain the invariant that the players know the root of T_i is labelled with an OBDD that evaluates to false under the input assignment, and therefore that some leaf is labelled with a clause that is false under the input assignment.

Initially, T_1 is the tree of the entire refutation. The protocol ends after reaching a tree that consists of a single vertex, and this will be labelled with a falsified clause. Each T_{i+1} will either be a subtree of T_i or will be obtained by pruning away some subtree. Specifically, let v be a vertex of T_i such that the subtree T of T_i rooted at v has size satisfying $\frac{1}{3}|T_i| \leq |T| \leq \frac{2}{3}|T_i|$; the players can find such a vertex v without any communication. The vertex v is labelled with a π_j-OBDD D for some $j \in [\ell]$. If D evaluates to 0 (False) under the input assignment, then T_{i+1} is T. If, however, it evaluates to 1 (True), then T_{i+1} is obtained from T_i by pruning away T and replacing it with the constant 1.

The players can determine whether the π_j-OBDD D labelling v evaluates to 0 using only $\lceil \log |D| \rceil + 1 \leq 2 \log S$ bits of communication. Namely, taking $a = j + 1$ and $b = j$, and using Lemma 7.14, the a-th and b-th players can evaluate D using only this many bits. Trivially, if the value of D equals 0, then the root of T_{i+1} evaluates to 0. Otherwise, the root of T_{i+1} is the same as the root of T_i, and it still evaluates to 0.

As each step the players use at most $2 \log S$ bits of communication and there are at most $O(\log S)$ steps (since $|T_{i+1}| \leq \frac{2}{3}|T_i|$). Hence, the players use at most $O(\log^2 S)$ bits of communication. □

8 Conclusion

Theorem 7.1 proved superpolynomial lower bounds on tree-like OBDD$(\wedge, \mathrm{w}, \mathrm{r}_{\epsilon n})$ refutations. It is open whether similar lower bounds hold for the corresponding dag-like system. It is even an open problem to give exponential bounds on (dag-like) OBDD$(\wedge, \mathrm{w}, \mathrm{r}_2)$ refutations, i.e. refutations that use at most two variable orderings. It is also an open problem to give exponential bounds on (dag-like) OBDD$(\wedge, \exists, \mathrm{r}_2)$ refutations. In fact, we do not know any OBDD or 1-NBP system for which weakening (w) is superpolynomially more efficient than projection ($\exists$).

One candidate for such a separation is the Clique-Coloring principle. [7] gave polynomial size OBDD$(\wedge, \mathrm{w})$ refutations for a version of the Clique-Coloring principle, based on a construction of [16]. It is open, however, whether this principle has polynomial size OBDD$(\wedge, \exists)$ refutations, or even polynomial size OBDD$(\wedge, \exists, \mathrm{r})$ or 1-NBP$(\wedge, \exists)$ refutations.

Acknowledgements. The research presented in Sections 3, 5 and 6 is supported by Russian Science Foundation (project 18-71-10042). The work of Sam Buss on the research presented in Sections 4 and 7 is partially supported by Simons Foundation Grant 578919.

Dmitry Itsykson is a Young Russian Mathematics award winner and would like to thank sponsors and jury of the contest.

References

[1] Noga Alon. Eigenvalues and expanders. *Combinatorica*, 6(2):83–96, June 1986.

[2] Noga Alon and Fan R. K. Chung. Explicit construction of linear sized tolerant networks. *Discrete Mathematics*, 306(10-11):1068–1071, 2006.

[3] Sanjeev Arora and Boaz Barak. *Computational Complexity: A Modern Approach.* Cambridge University Press, New York, 2009.

[4] Albert Atserias, Phokion G. Kolaitis, and Moshe Y. Vardi. Constraint propogation as a proof system. In *Proc. Tenth International Conf. on Principles and Practice of Constraint Programming*, Lecture Notes in Computer Science 3258, pages 77–91. Springer Verlag, 2004.

[5] Eli Ben-Sasson and Ave Wigderson. Short proofs are narrow — resolution made simple. *Journal of the ACM*, 48:149–169, 2001.

[6] Randal E. Bryant. Symbolic Boolean manipulation with ordered binary-decision diagram. *ACM Computing Surveys*, 24(3):293–318, 1992.

[7] Sam Buss, Dmitry Itsykson, Alexander Knop, and Dmitry Sokolov. Reordering rule makes OBDD proof systems stronger. In *Proc. 33rd Computational Complexity Conference, (CCC)*, LIPIcs 102, pages 16:1–24. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018.

[8] Larry Carter and Mark N. Wegman. Universal classes of hash functions. *Journal of Computer and System Sciences*, 18(2):143–154, 1979.

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

[10] Stephen A. Cook and Robert A. Reckhow. On the lengths of proofs in the propositional calculus, preliminary version. In *Proceedings of the Sixth Annual ACM Symposium on the Theory of Computing*, pages 135–148, 1974.

[11] Stephen A. Cook and Robert A. Reckhow. The relative efficiency of propositional proof systems. *Journal of Symbolic Logic*, 44:36–50, 1979.

[12] Luke Friedman and Yixin Xu. Exponential lower bounds for refuting random formulas using ordered binary decision diagrams. In *Computer Science — Theory and Applications, Proc. 8th Computer Science Symposium in Russia (CSR)*, Lecture Notes in Computer Science 7913, pages 127–138. Springer, 2013.

[13] Ankit Garg, Mika Göös, Pritish Kamath, and Dmitry Sokolov. Monotone circuit lower bounds from resolution. In *Proc. 50th ACM Symposium on Theory of Computing (STOC)*, pages 902–911, 2018.

[14] Ludmila Glinskih and Dmitry Itsykson. Satisfiable Tseitin formulas are hard for nondeterministic read-once branching programs. In *42nd Intl. Symp. on Mathematical Foundations of Computer Science (MFCS)*, pages 26:1–26:12, 2017.

[15] Mika Göös and Toniann Pitassi. Communication lower bounds via critical block sensitivity. *SIAM Journal on Computing*, 47(5):1778–1906, 2018.

[16] Dima Grigoriev, Edward A. Hirsch, and Dmitrii V. Pasechnik. Complexity of semi-algebraic proofs. In *Proc. 19th Symp. on Theoretical Aspects of Computer Science (STACS)*, Lecture Notes in Computer Science 2285, pages 419–430. Springer Verlag, 2002.

[17] Armin Haken. The intractability of resolution. *Theoretical Computer Science*, 39:297–308, 1985.

[18] Dmitry Itsykson, Alexander Knop, Andrei Romashchenko, and Dmitry Sokolov. On ODBB-based algorithms and proof systems that dynamically change order of variables. In *Proc. 34th Symp. on Theoretical Aspects of Computer Science (STACS 2017)*, LIPIcs 66, pages 43:1–43:14. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2017. Full version to appear in the *Journal of Symbolic Logic.*

[19] Matti Järvisalo. On the relative efficiency of DPLL and OBDDs with axiom and join. In *Proc. Principles and Practice of Constraint Programming (CP 2011)*, Lecture Notes in Computer Science 6876, pages 429–437. Springer Verlag, 2011.

[20] Alexander Knop. IPS-like proof systems based on binary decision diagrams. Technical Report ECCC-TR15-053, Electronic Colloquium on Computational Complexity, November 2017.

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

[22] Alexander Lubotzky, Ralph Phillips, and Peter Sarnak. Ramanujan graphs. *Combinatorica*, 8(3):261–277, 1988.

[23] Christoph Meinel and Thorsten Theobald. *Algorithms and Data Structures in VLSI Design: OBDD — Foundations and Applications*. Springer, 1998. Translation of the 1998 German edition.

[24] Guoqiang Pan and Moshe Y. Vardi. Symbolic techniques in satisfiability solving. *Journal of Automated Reasoning*, 35(1-3):25–50, 2005.

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

[26] Anup Rao and Amir Yehudayoff. Simplified lower bounds on the multiparty communication complexity of disjointness. In *Proc. 30th Conference on Computational Complexity (CCC)*, LIPIcs 33, pages 88–101, 2015.

[27] Nathan Segerlind. Nearly-exponential size lower bounds for symbolic quantifier elimination algorithms and OBDD-based proofs of unsatisfiability. Technical Report TR07-009, Electronic Colloquium on Computational Complexity (ECCC), January, August 2007. eccc.hpi-web.de/eccc-reports/2007/TR07-009.

[28] Nathan Segerlind. On the relative efficiency of resolution-like proofs and ordered binary decision diagram proofs. In *Proc. 23rd Annual IEEE Conference on Computational Complexity (CCC'08)*, pages 100–111, 2008.

[29] Alexander A. Sherstov. Communication lower bounds using directional derivatives. *J. ACM*, 61(6):34:1–34:71, 2014.

[30] Jorg Siekmann and Graham Wrightson. *Automation of Reasoning*, volume 1&2. Springer-Verlag, Berlin, 1983.

[31] G. S. Tsejtin. On the complexity of derivation in propositional logic. *Studies in Constructive Mathematics and Mathematical Logic*, 2:115–125, 1968. Reprinted in: [30, vol 2], pp. 466-483.

[32] William T. Tutte. The factorization of linear graphs. *Journal of the London Mathematical Society*, s1-22(2):107–111, 1947.

[33] Olga Tvertina, Carsten Sinz, and Hans Zantema. Ordered binary decision diagrams, pigeonhole principles and beyond. *Journal of Satisfiability, Boolean Modeling and Computation (JSAT)*, 7(1):35–58, 2010.

[34] Alasdair Urquhart. Hard examples for resolution. *Journal of the Association for Computing Machinery*, 34:209–219, 1987.

[35] Ingo Wegener. *Branching Programs and Binary Decision Diagrams: Theory and Applications*. Monographs on Discrete Mathematics and Applications 4. SIAM, 1987.