Expander CNFs have Exponential DNNF Size

Simone Bova* Florent Capelli† Stefan Mengel‡ Friedrich Slivovsky*

Abstract

A negation normal form (NNF) is a circuit on unbounded fanin conjunction (AND) and disjunction (OR) gates, with fanin one negation (NOT) gates applied to input gates only. A NNF is decomposable (DNNF) if, for each AND node, the subcircuits rooted at the children of the node are variable disjoint. We prove an unconditional exponential lower bound on the DNNF size of CNF formulas; thus far, only a superpolynomial lower bound was known, under the condition that the polynomial hierarchy does not collapse. As corollaries, we derive that the negation of a DNNF has in general size exponential in the original DNNF (which was known to hold if P $\neq$ NP), and that the language of prime implicates (PI) is exponentially more succinct than DNNFs (which was unknown). These results settle three open problems in the area of knowledge compilation [DM02].

1 Introduction

Boolean circuits are a basic model of computation that has been central to computer sciences since the very beginnings of the field, see [Vol99]. Many computational models and complexity classes can be captured by Boolean circuits and this perspective often yields to insights that would otherwise be hard to come by. A completely different use of Boolean circuits can be found in the field of artificial intelligence. Here circuits are often not seen as means of computation but to encode *knowledge bases* or *propositional theories*. One standard way of encoding propositional facts are CNF-formulas which is one of the reasons for the importance of SAT-solvers in practice. Unfortunately, encoding knowledge bases in CNF has some drawbacks: Many queries on knowledge bases encoded this way are intractable. For example, clause entailment queries which are central in practice are intractable for CNFs unless $\mathsf{P} \neq \mathsf{NP}$ [DM02].

For this reason, many other representation languages for propositional theories have been studied which generally correspond to different classes of Boolean circuits, see e.g. the seminal work of Darwiche and Marquis [DM02]. The general finding is that in the choice of representation there is a trade-off between succinctness and usefulness of the encoding: One the one end of the spectrum there are classes of representation languages that allow very succinct representation but for which most queries on the knowledge base are hard; on the other end of the spectrum there are classes which allow many queries on them to be answered quickly but which encode knowledge in a very verbose way.

In this paper we study one very important class of Boolean circuits used in knowledge representation, so-called circuits in decomposable negation normal form (short DNNF). These circuits generalize most other models studied in knowledge representation, in particular classical models like OBDDs, FBDDs and DNFs, but are in general more succinct than those. Still, DNNFs are well-behaved in the sense that they allow efficient answering of clause entailment queries. Since these queries are of central importance in practice, answering them is, following [DM02], the minimal criterion a class must satisfy to be useful in the context of knowledge representation.

Since one often encounters CNF encodings of propositional theories but for the reasons outlined above it would be preferable to have a DNNF representations it is natural to ask if CNFs can be transformed into DNNFs—a process called *knowledge compilation* in the artificial intelligence literature. Since DNNFs

*Vienna University of Technology
†IMJ UMR 7586 - Logique, Université Paris Diderot, France
‡LIX UMR 7161, Ecole Polytechnique, France

are generalizations of DNFs which can express all Boolean functions, it is clear that such a translation is possible in principle. Thus the question becomes if the translation is possible in such a way that the size of the representation does not blow up too much. Using the classical Karp-Lipton theorem [KL80], one can show that, unless the polynomial hierarchy collapses, in general the size the DNNF-representation of CNFs must be superpolynomial [DM02]. But this does not rule out representations of, say, quasipolynomial size. The main result of this paper is the following unconditional, truly exponential lower bound.

Theorem 1. *There exist a class $\mathcal{C}$ of CNF formulas and a constant $c > 0$ such that, for every formula F in $\mathcal{C}$, every DNNF equivalent to F has size at least $2^{c \cdot \mathsf{size}(F)}$. In fact, $\mathcal{C}$ is a class of read 3 times, monotone, 2-CNF formulas.*

Since it is easy to find for every CNF formula a representation of F in DNF of size at most $(\mathsf{size}(F) + 1)2^{2 \cdot \mathsf{size}(F)} \leq 2^{3 \cdot \mathsf{size}(F)+1}$, and thus this is also true for the more general notion of DNNFs, Theorem 1 is essentially tight; we refer to Section 2 for the definition of $\mathsf{size}(\cdot)$.

We use Theorem 1 to show some other results. First, by inspecting the formulas in Theorem 1, we can see that the class $\mathcal{C}$ consists of formulas in so-called *prime implicates form*, a restricted form of CNFs. Up to this paper, there were not even conditional lower bounds for encoding formulas in prime implicates form in DNNFs. From the proof of Theorem 1, it follows that there is an unconditional, exponential separation between these two representations, answering an open question of Darwiche and Marquis [DM02].

The second additional result we prove concerns the closure of DNNFs under negation. It is not hard to see that, under the condition $\mathsf{P} \neq \mathsf{NP}$, the negation of a DNNF cannot be encoded into a DNNF in general [DM02]. Using Theorem 1, we strengthen this result by showing that negation may in fact lead to an exponential blow-up of the encoding size.

Related work. We are not aware of any unconditional lower bounds for general DNNFs. However, there is a rich literature on lower bounds for more restricted models. For example, there are many lower bounds on OBDDs, FBDDs and generalizations thereof, see e.g. the textbooks [Weg00, Juk12]. Moreover, certain subclasses of DNNFs, so-called *decision-DNNFs* have been recently considered in database theory in the context of probabilistic databases. In this setting, lower bounds could be shown by proving a simulation of deterministic DNNFs by FBDDs and then showing lower bounds for FBDDs [BLRS13, BLRS14]. Finally, CNF and DNF cannot be translated into each other without an exponential blow-up [GKPS95].

Pipatsrisawat and Darwiche [PD10] have proposed a framework for showing lower bounds on *structured* DNNFs, a subclass in which the variables respect a common tree-ordering. The difference between DNNFs and structured DNNFs is roughly similar that of FBDDs and OBDDs and as the former are exponentially more succinct than the latter (see e.g. [Weg00]), one can expect general DNNFs to be far stronger than structured DNNFs.

As already stated in the introduction, conditional lower bounds for DNNFs were already shown in [DM02] but we stress that the techniques used there cannot show exponential lower bounds as we do in this paper.

The techniques in this paper build on recent work by Razgon [Raz14a]: Theorem 3 is taken from that paper and plays an important role in our proof combined with the usage of expander graphs which have already recently been taken to a good effect by two of the authors of this paper in establishing lower bounds for OBDDs [BS14]. Expander graphs have found applications throughout many fields of computer science [HLW06], and here we show that they can be used to show very strong lower bounds in the knowledge representation context.

2 Preliminaries

For every integer $n \geq 1$, we let $[n]$ denote the set $\{1, \dots, n\}$. Let X be a countable set of elements, called *variables*, and call the elements of $\{0, 1\}$ *constants*. A *literal* is a variable in X (a *positive* literal) or the negation, $\neg$, of a variable in X (a *negative* literal). For a literal $l \in \{x, \neg x\}$ for some $x \in X$, we let $\mathsf{var}(l) = x$. An *atom* is a literal or a constant.

Negation Normal Forms. A *negation normal form* (in short, *NNF*) D on a finite set of variables $\mathsf{vars}(D) \subseteq X$ is a finite node labelled directed acyclic graph (DAG) with a designated sink node, referred to as *the* sink of the NNF and denoted by $\mathsf{sink}(D)$. The source nodes of D, denoted by $\mathsf{sources}(D)$, are labelled by a constant or a literal in $\{x, \neg x \colon x \in \mathsf{vars}(D)\}$. The non source nodes of D are labelled by $\wedge$ or $\vee$. The label of a node v is denoted by $\mathsf{label}(v)$. We let $\mathsf{atoms}(D) = \{\mathsf{label}(v) \colon v \in \mathsf{sources}(D)\}$. The *size* of D, in symbols $\mathsf{size}(D)$, is the number of arcs in the DAG underlying D.

Let G be a DAG and let v be a node in G. The *subgraph of G sinked at v* is the DAG whose node set is

$$V' = \{v\} \cup \{u \colon \text{there exists a directed path from } u \text{ to } v \text{ in } G\},$$

and whose arcs are exactly the arcs of G among the nodes in V'. Let D be a NNF, and let v be a node in the DAG G underlying D. The *subgraph of G sinked at v* is the DAG whose node set is

$$\{v\} \cup \{u \colon \text{there exists a directed path from } u \text{ to } v \text{ in } G\},$$

and whose arcs are exactly the arcs of G among the nodes in the above node set. We let $\mathsf{sub}(D, v)$ denote the *subcircuit of D sinked at v*, that is, the NNF on variables $\mathsf{vars}(D)$ whose underlying DAG is the subgraph of D sinked at v, with the same labels (the designated sink is v).

An *assignment* is a function from a subset of X to the constants. We equivalently view an assignment f as the set of atoms $\{\neg x \colon f(x) = 0\} \cup \{x \colon f(x) = 1\}$; as a set of atoms, an assignment does not contain the constant 0 nor a pair of complementary literals.

Let D be a NNF, v be a node in D, and f be an assignment of $\mathsf{vars}(D)$. The *value of* $\mathsf{sub}(D, v)$ *under* f, in symbols $\mathsf{sub}(D, v)(f)$, is defined inductively on the structure of $\mathsf{sub}(D, v)$ as follows:

- if v is a 0-node or an $\vee$-node having indegree 0, then $\mathsf{sub}(D, v)(f) = 0$;
- if v is a 1-node or a $\wedge$-node having indegree 0, then $\mathsf{sub}(D, v)(f) = 1$;
- if v is an x-node, then $\mathsf{sub}(D, v)(f)$ equals $f(x)$;
- if v is an $\neg x$-node, then $\mathsf{sub}(D, v)(f)$ equals $1 - f(x)$;
- if v is a $\wedge$-node with ingoing arcs from nodes $v_1, \dots, v_m$, then $\mathsf{sub}(D, v)(f)$ is equals to the minimum over the $\mathsf{sub}(D, v_i)(f)$ for $i \in [m]$;
- if v is a $\vee$-node with ingoing arcs from nodes $v_1, \dots, v_m$, then $\mathsf{sub}(D, v)(f)$ is equal to the maximum of the $\mathsf{sub}(D, v_i)(f)$ for $i \in [m]$.

The *value of D under f*, in symbols $D(f)$, is equal to $\mathsf{sub}(D, \mathsf{sink}(D))(f)$. We let $\mathsf{sat}(D)$ denote the set of all *satisfying* assignments of D, that is, assignments $f \colon \mathsf{vars}(D) \to \{0, 1\}$ such that $D(f) = 1$. The NNF D is *trivial* if the set of its satisfying assignments is the empty set or the set of all assignments of $\mathsf{vars}(D)$, and *nontrivial* otherwise. Two NNFs D and D' are *equivalent* if $\mathsf{sat}(D) = \mathsf{sat}(D')$; in particular, $\mathsf{vars}(D) = \mathsf{vars}(D')$.

Let D be a NNF. A *certificate* of D is a NNF T on variables $\mathsf{vars}(D)$ whose labelled nodes and arcs are subsets of the labelled nodes and arcs of D satisfying the following: $\mathsf{sink}(D)$ is the sink node of T, so $\mathsf{sink}(D) = \mathsf{sink}(T)$; if a node v is in $D \cap T$, and v is an $\wedge$-node in D with ingoing arcs from nodes $v_1, \dots, v_i$, then the nodes $v_1, \dots, v_i$ and the arcs $(v_1, v), \dots, (v_i, v)$ are in T ($i \geq 0$); if a node v is in $D \cap T$, and v is an $\vee$-node in D with ingoing arcs from nodes $v_1, \dots, v_i$, then exactly one node $w \in \{v_1, \dots, v_i\}$ is in T and the arc (w, v) is in T v ($i \geq 0$). We let $\mathsf{cert}(D)$ denote the set of certificates of D.

The models and the certificates of an NNF are nicely related as follows.

Proposition 1. *Let D be a NNF and let f be an assignment. The following are equivalent.*

- $f \in \mathsf{sat}(D)$.
- *There exists* $T \in \mathsf{cert}(D)$ *such that* $f \in \mathsf{sat}(T)$.

Proof. For the forward direction, let $f \in \mathsf{sat}(D)$. Call an arc (u, v) in the DAG underlying D *activated* by f if f satisfies the subcircuit of D sinked at u, in symbols $\mathsf{sub}(D, u)(f) = 1$. It is readily verified that there exists a certificate T for D containing only arcs activated by f. Moreover, $f \in \mathsf{sat}(T)$ because by construction $\mathsf{sub}(T, v)(f) = 1$ for all source nodes v of T, and therefore $\mathsf{sub}(T, \mathsf{sink}(T))(f) = 1$.

For the backward direction, let T be a certificate of D such that $f \in \mathsf{sat}(T)$. By induction on the structure of D, we prove that for all $t \in T$ it holds that $\mathsf{sub}(T, t)(f) = \mathsf{sub}(D, t)(f) = 1$. Since $\mathsf{sink}(\mathsf{D}) = \mathsf{sink}(\mathsf{T})$, we conclude that $D(f) = \mathsf{sub}(D, \mathsf{sink}(\mathsf{D}))(f) = \mathsf{sub}(T, \mathsf{sink}(\mathsf{T}))(f) = 1$. Then, $f \in \mathsf{sat}(D)$.

If t is a source node of T, then $\mathsf{sub}(T, t)(f) = 1$ because otherwise f does not satisfy T. We also have that t is a source node of D, hence $\mathsf{sub}(D, t)(f) = 1$. Let t be a $\vee$-node in T, with ingoing arcs from nodes $t_1, \ldots, t_i$ in D; say without loss of generality that t_1 is chosen in T. By the induction hypothesis, $\mathsf{sub}(T, t_1)(f) = \mathsf{sub}(D, t_1)(f) = 1$; hence, $\mathsf{sub}(T, t)(f) = \mathsf{sub}(D, t)(f) = 1$. The case where t is a $\wedge$-node is similar. $\square$

Decomposable Negation Normal Forms (DNNFs). A NNF D is *decomposable* (in short, a *DNNF*) if for all $\wedge$-nodes v with ingoing arcs from nodes $v_1, \ldots, v_i$ and all $j, j' \in [i]$, $j \neq j'$, the variable sets of the subcircuits of D sinked at v_j and $v_{j'}$ are disjoint, in symbols,

$$\mathsf{vars}(\mathsf{sub}(D, v_j)) \cap \mathsf{vars}(\mathsf{sub}(D, v_{j'})) = \emptyset.$$

Conjunctive Normal Forms (CNFs). A *conjunctive normal form* (in short, *CNF*) is a finite set of clauses, where a *clause* is a finite set of literals. Equivalently, we regard a CNF as a NNF where the maximum number of arcs on a path from a source to the sink is 2, and each $\vee$-node has ingoing arcs only from sources. A CNF F is *monotone* if its labels do not contain negative literals, a *k-CNF* if the indegree of $\vee$-nodes is at most k, and a *read k times* CNF if, for every $x \in \mathsf{vars}(F)$, the number of arcs leaving nodes whose label contain the variable x is at most k.

Graph CNFs. We refer to a standard reference for basic notions and facts in graph theory [Die05]. Let $G = (V, E)$ be a graph. A *vertex cover* in G is a subset C of the vertices V such that $\{u, v\} \cap C \neq \emptyset$ for all $\{u, v\} \in E$. If G has no isolated vertices, we view E as a CNF on the variables $\mathsf{vars}(E) = V$ (a trivial CNF, if $E = \emptyset$), which we call a *graph CNF*. The satisfying assignments of E are essentially vertex covers of G: If f is a satisfying assignment of E, then $\{x \colon f(x) = 1\}$ is a vertex cover of G, and if C is a vertex cover of G, then any assignment f of $\mathsf{vars}(E)$ containing C satisfies E.

In light of the above observation, the following statement, to which we will essentially appeal in proving the main result, says that the Hamming weight of any model of a graph CNF is directly proportional to the cardinality of the underlying graph (with a proportionality constant inversely proportional to the degree of the graph).

Proposition 2. *Let C be a vertex cover of a connected graph G of degree at most d. Then, $\frac{|V|}{d+1} \leq |C|$.*

Proof. Let C be a vertex cover of a connected graph G of degree at most d. Then, $V \setminus C$ is an independent set. Since G is connected, each vertex in $V \setminus C$ is incident to at least one edge with a vertex in C. Hence, there are at least $|V \setminus C|$ edges between C and $V \setminus C$. Since each vertex in C has degree at most d,

$$|V| - |C| = |V \setminus C| \leq d|C|,$$

and we are done. $\square$

3 Nice DNNFs for Graph CNFs

In this section, we observe that a DNNF computing a graph CNFs is equivalent to a DNNF that is nice, in that its non source nodes have indegree at most 2, and its source nodes are labelled by positive literals only (Corollary 1). We also observe that nice DNNFs have nice certificates, in the sense that their underlying graphs are rooted binary trees whose leaves are labelled by variables (Proposition 6).

Proposition 3. *Let D be a DNNF. There exists a DNNF D' equivalent to D, having indegree 2, and such that* $\mathsf{size}(D') \leq 2 \cdot \mathsf{size}(D)$.

Proof. Let D be a DNNF. A NNF D' equivalent to D and having indegree 2 is obtained by editing D as follows, until no node of indegree larger than 2 exists: Let v be a $\wedge$-node with ingoing arcs from nodes $v_1, \ldots, v_i$ with $i > 2$; delete the arcs (v_j, v) for $j \in \{2, \ldots, i\}$; create a fresh $\wedge$-node w, and the arcs (w, v) and (v_j, w) for $j \in \{2, \ldots, i\}$. The case where v is an $\vee$-node is similar.

It is readily verified that D' is decomposable. Moreover, each arc in D is processed at most once (when it is an ingoing arc of a node having indegree larger than 2) and it generates at most 2 arcs in D', hence the size of D' is at most twice the size of D. □

A Boolean function $F\colon \{0,1\}^{X'} \to \{0,1\}$ is called *monotone* if for all assignments $f, f' \in X' \to \{0,1\}$ such that $f(x) \leq f'(x)$ for all $x \in X'$, it holds that $F(f) \leq F(f')$. A NNF is called *negation free* if its labels do not contain negative literals.

Proposition 4. *Let D be a DNNF computing a monotone Boolean function. There exists a DNNF D' equivalent to D, negation free, and such that* $\mathsf{size}(D') \leq \mathsf{size}(D)$*. Moreover, D' has the same indegree as D.*

Proof. Suppose D contains a node u labelled with literal $\neg x$. Let D' be the DNNF on $\mathsf{vars}(D)$ obtained from D by relabelling u with the constant 1. We claim that an assignment satisfies D if and only if it satisfies D'. Let f be a satisfying assignment of D. By Proposition 1 there is a certificate T of D such that f satisfies T. We obtain a certificate T' of D' by relabelling u with the constant 1 (if u appears in T). It is straightforward to verify that f is a satisfying assignment of T'. We apply Proposition 1 once more to conclude that f must be a satisfying assignment of D'. For the converse, let f be a satisfying assignment of D', and let T' be a certificate of D' such that f satisfies T'. If T' does not contain the node u then T' is also a certificate of D, and f is a satisfying assignment of D by Proposition 1. Otherwise, we obtain a certificate T of D from T' by relabelling u with the literal $\neg x$. Let f' be the assignment such that $f(y) = f'(y)$ for all $y \in \mathsf{vars}(D) \setminus \{x\}$, and such that $f'(x) = 0$. Since D is decomposable and T contains the node u labelled with $\neg x$, no node of T can be labelled with the literal x. Thus T' cannot contain such a node either, and f' satisfies T'. Since $\neg x$ evaluates to 1 under f', the certificate T is satisfied by f' as well. By Proposition 1, the assignment f' is a satisfying assignment of D. Because the function computed by D is monotone, we conclude that f must satisfy D as well. Clearly D and D' have the same size and maximum indegree. It follows that the desired negation free DNNF can be obtained by replacing every negative literal by the constant 1 in the labels of D. □

A NNF is called *constant free* if its labels do not contain the constants 0 and 1. Let D be a nontrivial DNNF. A constant free DNNF, denoted by $\mathsf{elimconst}(D)$, is obtained by editing D as follows, until all nodes labelled by a constant are deleted: Let v be a 0-node, and let v have arcs to nodes $v_1, \ldots, v_r$. For all $j \in [r]$: if v_j is a $\wedge$-node, relabel v_j by 0, and delete all the ingoing arcs of v_j (possibly creating some undesignated sink nodes); if v_j is a $\vee$-node, then delete the arc (v, v_j); relabel v_j by 0 if it becomes indegree 0; finally, delete v. The case where v is a 1-node is similar.

Proposition 5. *Let D be a nontrivial DNNF and let $D' = \mathsf{elimconst}(D)$. Then, D and D' are equivalent, and* $\mathsf{size}(D') \leq \mathsf{size}(D)$*; moreover, the indegree and negation freeness of D are preserved in D'.*

Proof. By construction. □

Corollary 1. *Let E be a graph CNF and let D be a DNNF equivalent to E. There exists a indegree 2, constant free, and negation free DNNF D' equivalent to D such that* $\mathsf{size}(D') \leq 2 \cdot \mathsf{size}(D)$.

Proof. Let E be a graph CNF and let D be a DNNF equivalent to E. By Proposition 3, there exists a DNNF D_1, equivalent to D, having indegree 2 whose size is at most twice the size of D. Since E is a monotone Boolean function, by Proposition 4, there exists an indegree 2 and negation free DNNF D_2, equivalent to D_1, whose size is at most the size of D_1. Since E is a non constant Boolean function, by Proposition 5, there exists a indegree 2, negation free, and constant free DNNF D_3, equivalent to D_2, whose size is at most the size of D_3. Let $D' = D_3$. Then, D' is a indegree 2, constant free, and negation free DNNF, equivalent to D, whose size is at most twice the size of D. $\square$

Proposition 6. *Let D be a DNNF and let $T \in \mathsf{cert}(D)$.*

- *If D is constant free, then the underlying graph of T is a tree.*
- *If D is constant free and indegree 2, then the tree underlying T has degree 3.*
- *If D is constant free and negation free, then the $\mathsf{atoms}(T) \subseteq X$.*

Proof. We prove the first item (the rest is clear). Assume that the underlying graph of T is cyclic, so that there exist two distinct nodes v and w in T and two arc disjoint directed paths from v to w; in particular, w has at least two ingoing arcs in T, hence by construction w is a $\wedge$-node in D. By decomposability, no variables occur as labels of source nodes in $\mathsf{sub}(D, v)$, which is impossible since D is constant free. $\square$

4 Models of DNNFs for Graph CNFs

In this section, we prove two lemmas to which we critically appeal in the proof of the main result. The first lemma allows to derive, given a DNNF D and a node v in D, a DNNF D' whose models are exactly the models of D minus those models that correspond (in the sense of Proposition 1) to certificates of D using the node v (Lemma 1). The second lemma essentially establishes that if the models of a DNNF D computing a graph CNF have to satisfy a clause $\{x, x'\}$, and for some node v in D it is the case that a source labelled x has a directed path to v in the DAG underlying D but no source labelled x' has such a directed path to v, then all certificates of D using the node v hit x or all certificates of D using the node v hit x' (Lemma 2).

Let D be a constant free DNNF and let $v \in D \setminus \{\mathsf{sink}(D)\}$. Let $\mathsf{elimgate}(D, v)$ denote the DNNF obtained from D as follows. Let $\mathsf{paths}(D, v)$ be the set of directed paths from v to $\mathsf{sink}(D)$; display each P in $\mathsf{paths}(D, v)$ by $(w_0, w_1, \ldots, w_p)$, where $w_0 = v$, $w_p = \mathsf{sink}(D)$, and $p \geq 1$.

If there exists $P = (w_0, w_1, \ldots, w_p)$ in $\mathsf{paths}(D, v)$ such that w_j is an $\wedge$-node for every $j \in [p] = \{1, \ldots, p\}$, then $\mathsf{elimgate}(D, v)$ is the DNNF formed by a single 0-node (computing the constant 0). Otherwise, for every $P = (w_0, w_1, \ldots, w_p)$ in $\mathsf{paths}(D, v)$ there exists $j \in [p]$ such that w_j is an $\vee$-node; let $e_P = (w_{j-1}, w_j)$ denote the arc in P such that j is the smallest index in $[p]$ where w_j is an $\vee$-node. Then, $\mathsf{elimgate}(D, v)$ is obtained from D by deleting the arcs $\{e_P : P \in \mathsf{paths}(D, v)\}$ in the DAG underlying D. The process might create 0-nodes in the form of indegree-0 $\vee$-nodes, and undesignated sinks.

Lemma 1. *Let D be a DNNF and let $v \in D \setminus \{\mathsf{sink}(D)\}$. Then,*

$$\mathsf{sat}(\mathsf{elimgate}(D, v)) = \mathsf{sat}(D) \setminus \left(\bigcup_{\{T \in \mathsf{cert}(D) \colon v \in T\}} \mathsf{sat}(T) \right) .$$

Proof. Let f be a satisfying assignment of D such that f does not satisfy any certificate T of D with $v \in T$. Let T be a certificate of D such that f satisfies T; such a certificate exists by Proposition 1. Clearly, $v \notin T$. We claim that all arcs in T are kept in passing from D to $\mathsf{elimgate}(D, v)$; therefore, T is a certificate of $\mathsf{elimgate}(D, v)$, and since f satisfies T, by Proposition 1, it also holds that f satisfies $\mathsf{elimgate}(D, v)$.

We prove the claim. Assume for a contradiction that an arc (u', u) of T is deleted in passing from D to $\mathsf{elimgate}(D, v)$. Then, by definition of $\mathsf{elimgate}(D, v)$, there is a path from v to u' whose nodes are all labeled by $\wedge$. Since (u', u) is an arc of T and thus u' is a node of T, we have, by definition of certificates, that v is in T, which is a contradiction.

Conversely, let f be a satisfying assignment of $\mathsf{elimgate}(D, v)$. By Proposition 1, there exists a certificate T of $\mathsf{elimgate}(D, v)$ such that f satisfies T; then, T does not contain 0-nodes. In particular, T does not contain 0-nodes that arise, in passing from D to $\mathsf{elimgate}(D, v)$, as indegree-0 $\vee$-nodes on some directed path in D from v to $\mathsf{sink}(D)$. Therefore, every node $w \in T$ is either a $\wedge$-node in D, so that the node itself and all its ingoing arcs are kept in $\mathsf{elimgate}(D, v)$ by construction, or is a $\vee$-node in D for which at least one of its ingoing arcs is kept in $\mathsf{elimgate}(D, v)$, and is used by T. Hence, T is a certificate for D. The node v does not have a directed path to $\mathsf{sink}(D)$ in $\mathsf{elimgate}(D, v)$, because at least one arc is deleted on each such path in passing from D to $\mathsf{elimgate}(D, v)$; hence, $v \notin T$. Since f satisfies T, by Proposition 1, f satisfies D. $\square$

Lemma 2. *Let D be a constant free and negation free DNNF, $v \in D \setminus \{\mathsf{sink}(D)\}$, $x \in \mathsf{atoms}(\mathsf{sub}(D, v))$, and $x' \in \mathsf{vars}(D) \setminus \mathsf{atoms}(\mathsf{sub}(D, v))$. If $\{x, x'\} \cap \mathsf{atoms}(T) \neq \emptyset$ for all $T \in \mathsf{cert}(D)$, then at least one of the following two statements holds:*

- *$x \in \mathsf{atoms}(T)$ for all $T \in \mathsf{cert}(D)$ such that $v \in T$.*
- *$x' \in \mathsf{atoms}(T)$ for all $T \in \mathsf{cert}(D)$ such that $v \in T$.*

Proof. Let $\{T, T'\} \subseteq \mathsf{cert}(D)$ be such that $v \in T \cap T'$. As D is constant free, by Proposition 6, the underlying graphs of the certificates of D are trees. By hypothesis, $\{x, x'\} \cap \mathsf{atoms}(T) \neq \emptyset$ and $\{x, x'\} \cap \mathsf{atoms}(T') \neq \emptyset$. We want to show that $x \in \mathsf{atoms}(T) \cap \mathsf{atoms}(T')$ or $x' \in \mathsf{atoms}(T) \cap \mathsf{atoms}(T')$.

Assume towards a contradiction that $x \in \mathsf{atoms}(T) \setminus \mathsf{atoms}(T')$ and $x' \in \mathsf{atoms}(T') \setminus \mathsf{atoms}(T)$; the case where $x' \in \mathsf{atoms}(T) \setminus \mathsf{atoms}(T')$ and $x \in \mathsf{atoms}(T') \setminus \mathsf{atoms}(T)$ is symmetric.

First, we observe that $x \notin \mathsf{atoms}(T) \setminus \mathsf{atoms}(\mathsf{sub}(T, v))$. Otherwise, assume that $x \in \mathsf{atoms}(T) \setminus \mathsf{atoms}(\mathsf{sub}(T, v))$, and let w be the node of D lying in $\mathsf{sources}(T) \setminus \mathsf{sources}(\mathsf{sub}(T, v))$ labelled x. By hypothesis, $x \in \mathsf{atoms}(\mathsf{sub}(D, v))$, and let w' be the node of D in $\mathsf{sources}(\mathsf{sub}(D, v))$ labelled x. Then, it is possible to construct a certificate S for the constant free DNNF $\mathsf{sub}(D, v)$ such that the node w' is a source of S, and therefore $x \in \mathsf{atoms}(S)$; the underlying graph of S is a tree by Proposition 6. Now, the tree R obtained by replacing in T the subtree rooted at v by S is a certificate of D; in R, there exists a non source node u such that both w and w' have a directed path to u in R; hence, u has indegree at least 2, so that u is a $\wedge$-node in D, which contradicts the decomposability of D (at u by variable x).

Second we observe that $x' \notin \mathsf{atoms}(\mathsf{sub}(T', v))$, because $\mathsf{atoms}(\mathsf{sub}(T', v)) \subseteq \mathsf{atoms}(\mathsf{sub}(D, v))$ and $x' \notin \mathsf{atoms}(\mathsf{sub}(D, v))$ by hypothesis. Therefore,

$$\{x, x'\} \cap \mathsf{atoms}(T) \setminus \mathsf{atoms}(\mathsf{sub}(T, v)) = \emptyset$$

and

$$\{x, x'\} \cap \mathsf{atoms}(\mathsf{sub}(T', v)) = \emptyset.$$

Now, the tree S obtained by replacing in T the subtree rooted at v by the subtree rooted at v in T' is a certificate of D; moreover, $\{x, x'\} \cap \mathsf{atoms}(S) = \emptyset$, contradicting the hypothesis that all certificates of D have a nonempty intersection with $\{x, x'\}$. $\square$

5 Lower Bound

In this section, we prove the main result, an exponential separation between CNFs and DNNFs (Theorem 1). First, we introduce the two pieces of theory on which the proof relies, namely the existence of infinite families of expander graphs (Theorem 2), and the combinatorial techniques developed by Razgon to prove exponential lower bounds for the representation of graph CNFs within classes of branching programs, including prominent subclasses of DNNFs such as FBDDs [Raz14a]. Next, we provide a proof idea and a full proof of the main result.

Elements. Let $G = (V, E)$ be a graph. For every $S \subseteq V$, we let $\mathsf{neigh}(S, G)$ denote the *open neighbourhood* of S in G, in symbols,

$$\mathsf{neigh}(S, G) = \{v \in V : v \in V \setminus S \text{ and there exists } u \in S \text{ such that } \{u, v\} \in E\}.$$

Let $d \geq 3$ and $c > 0$. A graph $G = (V, E)$ is a (c, d)-*expander* if $\mathsf{degree}(G) \leq d$ and for all $S \subseteq V$ such that $|S| \leq |V|/2$ it holds that

$$|\mathsf{neigh}(S, G)| \geq c|S|. \tag{1}$$

Note that a (c,d)-expander is connected; otherwise, if S is the vertex set of a (nonempty) connected component of G of minimum size, then $0 < |S| \leq |V|/2$ and $0 = |\mathsf{neigh}(S,G)| \geq c|S| > 0$.

Theorem 2 (Section 9.2 in [AS00])**.** *For all $d \geq 3$, there exists $c > 0$ and a sequence of graphs $\{G_i \mid i \in \mathbb{N}\}$ such that $G_i = (V_i, E_i)$ is a (c,d)-expander and $|V_i| \to \infty$ as $i \to \infty$ ($i \in \mathbb{N}$).*

An *independent set* in G is a subset I of the vertices V such that $\{u,v\} \notin E$ for all $u,v \in I$. An independent set $I \subseteq V$ in a graph G is called *distant* if for all $\{u,v\} \in I$,

$$\mathsf{neigh}(\{u\},G) \cap \mathsf{neigh}(\{v\},G) = \emptyset.$$

A family $\mathcal{I}$ of distant independent sets in G *covers* the vertex covers of G if for all vertex covers C of G, there exists $I \in \mathcal{I}$ such that $I \subseteq C$.

Theorem 3 (Lemma 1 in [Raz14a])**.** *Let $d \geq 3$, let G be a graph such that of degree bounded by d, and let $\mathcal{I}$ be a family of distant independent sets in G. If $\mathcal{I}$ covers the vertex covers of G, then*

$$|\mathcal{I}| \geq 2^{\min\{|I| \colon I \in \mathcal{I}\} \cdot f(d)},$$

where $f(d) = \log_2(\frac{1}{1-2^{-(d+1)}}) = \log_2(\frac{2^{d+1}}{2^{d+1}-1}) > 0$.

Proof. Before turning to prove the main result, we provide an informal outline of the proof. The class of CNFs we use to separate CNFs and DNNFs is a class of graph CNFs naturally corresponding to a family of expander graphs (Theorem 2). Without loss of generality, the DNNF size of a graph CNF can be measured on DNNFs having fanin 2 and not containing negations or constants (Corollary 1); in particular, the labels on source nodes are just variables (or vertices of the underlying graphs). Let E be a graph CNF whose underlying graph (V,E) is an expander, and let D be a DNNF equivalent to E. In view of an application of Theorem 3, we construct a family of distant independent sets covering the set of satisfying assignments of E, as follows. In each step, we pick a node in D, and then we edit D so as to remove all certificates through that node. As long as D has certificates, it follows from Proposition 2 that it must have at least $|V|/(d+1)$ distinct variables (vertices) labelling its source nodes. We find a node v in D such that the variables occuring below v in D are at least $|V|/(d+1)$ and at most $|V|/2$ (Lemma 3, Claim 1); this is sufficient and necessary to appeal to the expansion of the graph underlying E, and find in such a graph a large distant matching, of size proportional in $|V|$, between variables occuring below v in D and variables not occuring below v in D (Lemma 3, Claim 2). We then argue using Lemma 2 that there is an equally large distant independent set covering the satisfying assignments corresponding (in the sense of Proposition 1) to certificates through v (Lemma 3, Claim 3). We now remove the certificates through v from D, by following every path from v to the sink of D until we find an edge that leads to an $\vee$-node, and by deleting this edge. By Lemma 1, performing these deletions eliminates a satisfying assignment of D if and only if it corresponds to a certificate through v. The iteration terminates once the resulting DNNF is unsatisfiable. At termination, we have found a family of distant independent sets, each of size proportional in $|V|$, that covers all satisfying assignments of E; by Theorem 3, this family has size exponential in $|V|$. Since there is a bijection between the distant independent sets in the family and a subset of nodes in the original DNNF D, the latter must have exponential size in $|V|$ as well.

We now give a full proof of the main result. The following statement is the main technical lemma.

Lemma 3. *Let $G = (V,E)$ be a (c,d)-expander ($d \geq 3$, $c > 0$), and let D be an indegree 2, constant free, and negation free DNNF computing the CNF E. There exist $K \subseteq D$ and a family $\mathcal{I}$ of distant independent sets of G such that:*

(i) $|K| \geq |\mathcal{I}|$;

(ii) *$\mathcal{I}$ covers the vertex covers of G;*

(*iii*) $\min\{|I| \colon I \in \mathcal{I}\} \geq \frac{\min\{1,c\}}{4d^3}|V|$.

Proof. We construct $K \subseteq D$ and a family $\mathcal{I}$ of distant independent sets in G as follows.

Let $D_0 = D$, $K_0 = \emptyset$, and $\mathcal{I}_0 = \emptyset$. For $i \geq 0$, we define D_i, K_i, and $\mathcal{I}_i$ maintaining the following invariants:

(I1) D_i is an indegree 2, constant free, and negation free nontrivial DNNF;

(I2) $\mathsf{sat}(D_i) = \mathsf{sat}(D) \setminus \Big(\bigcup_{\{T \in \mathsf{cert}(D) \colon T \cap K_i \neq \emptyset\}} \mathsf{sat}(T)\Big)$.

At $i = 0$, the invariant (I1) holds by the hypotheses on D and the fact that the CNF E is neither unsatisfiable nor valid; the invariant (I2) holds trivially because $K_0 = \emptyset$. Let $i \geq 1$, and assume that the invariant holds at $i - 1$.

Claim 1. *There exists a node v_i in D_{i-1} such that*

$$\frac{|V|}{d+1} \leq |\mathsf{atoms}(\mathsf{sub}(D_{i-1}, v_i))| \leq \frac{|V|}{2}.$$

Proof of Claim 1. Let $(w_1, \ldots, w_s)$ be a directed path in D from a source w_s to the sink $w_1 = \mathsf{sink}(D_{i-1})$ such that for all $j \in [s-1]$ if w_j has ingoing arcs from nodes w_{j+1} and w in D_{i-1} (by hypothesis, w_j has indegree at most 2), then

$$|\mathsf{atoms}(\mathsf{sub}(D_{i-1}, w))| \leq |\mathsf{atoms}(\mathsf{sub}(D_{i-1}, w_j))|.$$

By (I1), it holds that D_{i-1} is nontrivial, so that there exists a satisfying assignment $f \in \mathsf{sat}(D_{i-1})$. By (I2), $\mathsf{sat}(D_{i-1}) \subseteq \mathsf{sat}(D)$. By Proposition 1, there exists a certificate $T \in \mathsf{cert}(D_{i-1})$ such that f extends T. Since D_{i-1} is constant free and negation free, $\mathsf{atoms}(T) \subseteq V$.

We observe that $\mathsf{atoms}(T)$ contains a vertex cover of G. To see this, assume by way of contradiction that no vertex cover of G is contained in $\mathsf{atoms}(T)$. Let g be the assignment defined by $g(x) = 1$ if and only if $x \in \mathsf{atoms}(T)$. Since g extends T, we have that $g \in \mathsf{sat}(D_{i-1})$ by Proposition 1. Consequently, $g \in \mathsf{sat}(D)$. Since D computes E, it holds that $g \in \mathsf{sat}(D)$ if and only if $\{x \colon g(x) = 1\}$ is a vertex cover of G. But by definition $\{x \colon g(x) = 1\} \subseteq \mathsf{atoms}(T)$, a contradiction.

As G is a (c, d)-expander, G is connected and has degree at most d, we have by Proposition 2, that any vertex cover of G has size at least $|V|/(d+1)$. Hence, $|\mathsf{atoms}(T)| \geq |V|/(d+1)$. Therefore, we have that

$$|\mathsf{atoms}(\mathsf{sub}(D_{i-1}, w_1))| = |\mathsf{atoms}(D_{i-1})| \geq |\mathsf{atoms}(T)| \geq \frac{|V|}{d+1}.$$

If we have $|\mathsf{atoms}(D_{i-1})| \leq |V|/2$, we may choose $v_i = w_1 = \mathsf{sink}(D_{i-1})$ and are done. Otherwise,

$$\frac{|V|}{d+1} < \frac{|V|}{2} < |\mathsf{atoms}(D_{i-1})|,$$

where the first inequality holds as $d \geq 3$. Hence, there exists an index j in $[s-1]$ such that

$$|\mathsf{atoms}(\mathsf{sub}(D_{i-1}, w_{j+1}))| < \frac{|V|}{d+1} \leq |\mathsf{atoms}(\mathsf{sub}(D_{i-1}, w_j))|;$$

by construction,

$$|\mathsf{atoms}(\mathsf{sub}(D_{i-1}, w_j))| < 2 \cdot |\mathsf{atoms}(\mathsf{sub}(D_{i-1}, w_{j+1}))| \leq \frac{2|V|}{d+1} \leq \frac{|V|}{2},$$

where the last inequality holds as $d \geq 3$. We choose $v_i = w_j$. $\square$

Let v_i be the node in D_{i-1} given by Claim 1. If $\mathsf{elimconst}(\mathsf{elimgate}(D_{i-1}, v_i))$, that is, the result of first eliminating gate v_i and then constants in D_{i-1}, computes the constant 0 function, then the construction terminates with $K = K_{i-1}$ and $\mathcal{I} = \mathcal{I}_{i-1}$. Otherwise, put

$$D_i = \mathsf{elimconst}(\mathsf{elimgate}(D_{i-1}, v_i)).$$

Note that D_i satisfies (I1) by inspection of the elimination operation. Moreover, D_i satisfies (I2) as follows:

$$\begin{aligned}
\mathsf{sat}(D_i) &= \mathsf{sat}(\mathsf{elimconst}(\mathsf{elimgate}(D_{i-1}, v_i))) \\
&= \mathsf{sat}(\mathsf{elimgate}(D_{i-1}, v_i)) \\
&= \mathsf{sat}(D_{i-1}) \setminus \left(\bigcup_{\{T \in \mathsf{cert}(D_{i-1}) :\, v_i \in T\}} \mathsf{sat}(T) \right) \\
&= \mathsf{sat}(D) \setminus \left(\bigcup_{\{T \in \mathsf{cert}(D) :\, T \cap K_{i-1} \neq \emptyset\} \cup \{T \in \mathsf{cert}(D_{i-1}) :\, v_i \in T\}} \mathsf{sat}(T) \right) \\
&= \mathsf{sat}(D) \setminus \left(\bigcup_{\{T \in \mathsf{cert}(D) :\, T \cap K_{i-1} \neq \emptyset\} \cup \{T \in \mathsf{cert}(D) :\, v_i \in T, T \cap K_{i-1} = \emptyset\}} \mathsf{sat}(T) \right) \\
&= \mathsf{sat}(D) \setminus \left(\bigcup_{\{T \in \mathsf{cert}(D) :\, T \cap K_i \neq \emptyset\}} \mathsf{sat}(T) \right),
\end{aligned}$$

using the fact that $\mathsf{elimconst}(\cdot)$ is equivalence preserving for the second equation, Lemma 1 for the third equation, the fact that (I2) is satisfied by D_{i-1} for the fourth equation, and, by Lemma 1 and $(I2)$,

$$\bigcup_{\{T \in \mathsf{cert}(D_{i-1}) :\, v_i \in T\}} \mathsf{sat}(T) = \bigcup_{\{T \in \mathsf{cert}(D) :\, v_i \in T, T \cap K_{i-1} = \emptyset\}} \mathsf{sat}(T)$$

for the fifth equation.

A *matching* in G is a subset M of the edges E such that $\{u, v\} \cap \{u', v'\} \neq \emptyset$ for every two distinct edges $\{u, v\}$ and $\{u', v'\}$ in M. For disjoint subsets V' and V'' of the vertices of G, a matching M in G is said *between* V' *and* V'' if every edge in M intersects both V' and V''. A matching in a graph is *distant* if no two vertices incident to distinct edges of the matching are neighbours or share a neighbour.

Claim 2. *There exists a distant matching M_i in G between $\mathsf{atoms}(\mathsf{sub}(D_i, v_i))$ and $V \setminus \mathsf{atoms}(\mathsf{sub}(D_i, v_i))$ such that*

$$|M_i| \geq \frac{\min\{1, c\}}{4d^3} |V|.$$

Proof of Claim 2. Let $S = \mathsf{atoms}(\mathsf{sub}(D_i, v_i))$. Since $|S| \leq |V|/2$ and G is a (c, d)-expander, by (1) we have that

$$|\mathsf{neigh}(S, G)| \geq c|S| \geq \frac{c}{d+1} |V|.$$

We construct a distant matching M_i between between S and $V \setminus S$ in G as follows. Pick an edge $\{v, w\} \in E$ with $v \in S$ and $w \in \mathsf{neigh}(S, G) \subseteq V \setminus S$, add $\{v, w\}$ to M_i, delete the vertices at distance at most 2 from v or w in $S \cup \mathsf{neigh}(S, G)$, and iterate on the updated S and $\mathsf{neigh}(S, G)$, until either $S = \emptyset$ or $\mathsf{neigh}(S, G) = \emptyset$. At each step, we delete at most $2(d^2 + d + 1)$ vertices in S and at most $2(d^2 + d + 1)$ vertices in $\mathsf{neigh}(S, G)$. Hence, we iterate for at least

$$s \geq \min \left\{ \frac{S}{2(d^2 + d + 1)}, \frac{\mathsf{neigh}(S, G)}{2(d^2 + d + 1)} \right\} \geq \frac{\min\{1, c\}}{2(d+1)(d^2 + d + 1)} |V|$$

steps, so we have that

$$|M_i| \geq s \geq \frac{\min\{1, c\}}{4d^3}|V|$$

for $d \geq 3$. $\square$

Let M_i be the distant matching in G given by Claim 2.

Claim 3. *There exists a distant independent set I_i in G such that $|I_i| = |M_i|$ and, for all certificates T of D such that $v_i \in T$, and all vertex covers C of G such that $\mathsf{atoms}(T) \subseteq C$, it holds that $I_i \subseteq C$.*

Proof of Claim 3. We construct a set $I_i \subseteq V$ as follows. Let $\{x, x'\} \in M_i$. By construction, $x \in \mathsf{atoms}(\mathsf{sub}(D_i, v_i))$ and $x' \in \mathsf{neigh}(\mathsf{atoms}(\mathsf{sub}(D_i, v_i)), G)$. Since $M_i \subseteq E$ and D computes E, it holds that $\{x, x'\} \cap \mathsf{atoms}(T) \neq \emptyset$ for all $T \in \mathsf{cert}(D)$; otherwise, by Proposition 1, an assignment extending T and sending both x and x' to 0 would satisfy E, contradiction. By the invariant, any certificate of D_i is also a certificate of D; hence $\{x, x'\} \cap \mathsf{atoms}(T) \neq \emptyset$ for all $T \in \mathsf{cert}(D_i)$. Then, by applying Lemma 2 to D_i, $x \in \mathsf{atoms}(T)$ for all $T \in \mathsf{cert}(D_i)$ such that $v_i \in T$, or $x' \in \mathsf{atoms}(T)$ for all $T \in \mathsf{cert}(D_i)$ such that $v_i \in T$. If $x \in \mathsf{atoms}(T)$ for all $T \in \mathsf{cert}(D_i)$ such that $v_i \in T$, then add x to I_i; otherwise, add x' to I_i.

Since M_i is a distant matching in G, the set I_i is a distant independent set in G. Moreover, by construction, I_i contains exactly one vertex for each edge in M_i, hence $|I_i| = |M_i|$. Finally, let T be a certificate for D such that $v_i \in T$. We observe that $I_i \subseteq \mathsf{atoms}(T)$. Indeed, let $w \in I_i$, and let $\{w, w'\} \in M_i$ be the corresponding edge in M_i. By construction, all certicates containing v_i have w as an atom; in particular, $w \in \mathsf{atoms}(T)$. We conclude that, for all vertex covers C of G such that $\mathsf{atoms}(T) \subseteq C$, it holds that $I_i \subseteq C$. $\square$

Put:

- $K_i = K_{i-1} \cup \{v_i\}$;
- $\mathcal{I}_i = \mathcal{I}_{i-1} \cup \{I_i\}$.

We check that the sets $K = \{v_1, \ldots, v_L\}$ and $\mathcal{I}$, returned by the above construction, satisfy the required properties.

Item (i)**:** The mapping sending $v_i \in K$ to $I_i \in \mathcal{I}$ ($i \in [L]$) is, by construction, a surjective function from K to $\mathcal{I}$. Hence, $|K| \geq |\mathcal{I}|$.

Item (ii)**:** Let C be any vertex cover of G. As D computes E, the assignment f defined by $f(v) = 1$ if and only if $v \in C$ satisfies D, that is, $f \in \mathsf{sat}(D)$. By (I2), there exists $i \in [L]$ such that $f \in \mathsf{sat}(D_{i-1})$ and $f \notin \mathsf{sat}(D_i)$. By construction, there exists a certificate $T \in \mathsf{cert}(D)$ such that $v_i \in T$ and $f \in \mathsf{sat}(T)$; that is, $\mathsf{atoms}(T) \subseteq \{v \in V : f(v) = 1\} = C$. So, by Claim 3, $I_i \subseteq C$. Thus every vertex cover of G contains some distant independent set in $\mathcal{I}$; that is, $\mathcal{I}$ covers the vertex covers of G.

Item (iii)**:** For each $i \in [L]$, $|I_i| = |M_i|$ by Claim 3, and $|M_i| \geq \min\{1, c\}|V|/4d^3$ by Claim 2.

The statement is proved. $\square$

For a CNF E, we let

$$\mathsf{DNNF}(E) = \min\{\mathsf{size}(D) : D \text{ is a DNNF equivalent to } E\}$$

denote the minimum size over all DNNFs equivalent to E.

Theorem 4. *Let $G = (V, E)$ be a (c, d)-expander with at least 2 vertices. Then*

$$\mathsf{DNNF}(E) \geq 2^{b \cdot \mathsf{size}(E) - 1},$$

where $b = \frac{\min\{1,c\} \cdot f(d)}{12d^4}$ with $f(d) = \log_2(\frac{1}{1-2^{-(d+1)}})$.

Proof. First note that $E \neq \emptyset$, because $|V| \geq 2$ and (c,d)-expanders are connected. Let D be a DNNF equivalent to the graph CNF E. By Corollary 1, there exists an indegree 2, constant free, and negation free DNNF D' equivalent to D such that $2 \cdot \mathsf{size}(D) \geq \mathsf{size}(D')$. By Lemma 3, there exist $K \subseteq D'$ and a family $\mathcal{I}$ of distant independent sets in G such that $|K| \geq |\mathcal{I}|$, every vertex cover of G contains some distant independent set in $\mathcal{I}$, and $\min\{|I| : I \in \mathcal{I}\} \geq \min\{1,c\}|V|/4d^3$. It follows that

$$\mathsf{size}(D') \geq |K| \geq |\mathcal{I}| \geq 2^{\frac{\min\{1,c\}\cdot f(d)}{4d^3}|V|},$$

where $f(d) = \log_2(\frac{1}{1-2^{-(d+1)}}) > 0$ and the last inequality follows from Theorem 3. Therefore,

$$\mathsf{size}(D) \geq 2^{\frac{\min\{1,c\}\cdot f(d)}{4d^3}|V|}/2.$$

Now observe that $|E| \leq d|V|$, because G has degree d; thus the CNF E has at most $d|V|$ clauses, each of at most 2 literals, so that $d|V| + 2d|V| = 3d|V| \geq \mathsf{size}(E)$. Therefore,

$$\frac{\min\{1,c\}\cdot f(d)}{4d^3}|V| = \frac{\min\{1,c\}\cdot f(d)}{12d^4}3d|V| \geq \frac{\min\{1,c\}\cdot f(d)}{12d^4}\cdot \mathsf{size}(E),$$

from which we conclude that $\mathsf{DNNF}(E) \geq 2^{b\cdot \mathsf{size}(E)-1}$. ☐

We can now easily prove Theorem 1.

Proof of Theorem 1. Let $d = 3$ and let $G_i = (V_i, E_i)$ be a family $\mathcal{E}$ of (c', d)-expander graphs given by Theorem 2. Since $|V_i| \to \infty$ as $i \to \infty$ and $\mathcal{E}$ satisfies (1), there exists an infinite subset $I \subseteq \mathbb{N}$ such that $1 \leq \mathsf{size}(E_i) < \mathsf{size}(E_{i'})$ for all $i, i' \in I$, $i < i'$. Let $\mathcal{C} = \{E_i : i \in I\}$. Note that $\mathcal{C}$ is a class of read 3 times, monotone, 2-CNF formulas. Moreover, for every $i \in I$, it holds that $\mathsf{DNNF}(E_i) \geq 2^{c''\cdot \mathsf{size}(E_i)-1}$ by Theorem 4, where $c'' = b$ in the statement of Theorem 4. Choosing $c > 0$ such that $c'' \cdot \mathsf{size}(E_i) - 1 \geq c \cdot \mathsf{size}(E_i)$ holds for all sufficiently large $i \in I$, we conclude that $\mathsf{DNNF}(F) \geq 2^{c\cdot \mathsf{size}(F)}$. ☐

6 Corollaries

In this section we will prove the corollaries of Theorem 1 we sketched in the introduction.

Let F be a CNF-formula. We say that a clause C is *entailed* by F if every satisfying assignment of F also satisfies C. We say that a clause C' *subsumes* C, if $C' \subseteq C$. A CNF-formula F is in prime implicates form (short PI) if every clause that is entailed by F is subsumed by a clause that appears in F and no clause in F is subsumed by another.

Note that CNF-formulas in PI-form can express all Boolean functions but it is known that encoding in PI-form may generally be exponentially bigger than general CNF, see e.g. [DM02].

Remember that the formulas of Theorem 1 are monotone 2-CNF formulas.

Lemma 4. *Every monotone 2-CNF formula is in PI-form.*

Proof. Let F be a monotone 2-CNF formula. We first note that trivially no clause in a 2-CNF formula subsumes another.

Now let C be a clause entailed by F and assume by way of contradiction that C is not subsumed by any clause of F, i.e., every clause in F contains a positive literal not in C. Let C' be the clause we get from C by deleting all negative literals. We claim that C' is entailed by F. To see this, consider a satisfying assignment f of F. Let f' be the assignment we get from f by setting the variables that are negated in C to 1. Since F is monotone, this is still a satisfying assignment of F and thus of C. Consequently, C is satisfied by one of its positive literals in f' and thus in f. Thus f satisfies C' and it follows that F entails C'.

Now let f be the assignment that sets all variables in C' to 0 and all other variables to 1. Since C and thus also C' is not subsumed by any clause of F, the assignment f satisfies F. But by construction f does not satisfy C' which is a contradiction. ☐

From Lemma 4 and Theorem 1 we directly get the promised separation.

Corollary 2. *There exist a class $\mathcal{C}$ of CNF formulas in PI-form and a constant $b > 0$ such that, for every formula F in $\mathcal{C}$, every DNNF equivalent to F has size at least $2^{b \cdot \mathsf{size}(F)}$.*

We next show that DNNFs are not closed under negation.

Lemma 5. *There exist a class $\mathcal{C}'$ of 2-DNF formulas and a constant $b > 0$ such that, for every formula D in $\mathcal{C}'$, every DNNF equivalent to $\neg D$ has size at least $2^{b \cdot \mathsf{size}(D)}$.*

Proof. Let $\mathcal{C}$ be the class of 2-CNF-formulas from Theorem 1. Let $\mathcal{C}'$ be the class of 2-DNF-formulas we get by negating the formulas in $\mathcal{C}$. Now negating $\mathcal{C}'$ gives the class $\mathcal{C}$ again for which we have the lower bound from Theorem 1. □

Observing that DNF is a restricted form of DNNF, we get the following non-closure result which was only known conditionally before.

Corollary 3. *There exist a class $\mathcal{C}'$ of DNNFs and a constant $b > 0$ such that, for every formula D in $\mathcal{C}'$, every DNNF equivalent to $\neg D$ has size at least $2^{b \cdot \mathsf{size}(D)}$.*

7 Conclusion

We have shown a strong unconditional separation between CNF and DNNF representation of knowledge bases. In doing so we could also answer an open question of Darwiche and Marquis [DM02], showing the same separation between PI representation and DNNF. Finally, we showed that negation of DNNFs leads to an exponential blow-up in size. Let us close by mentioning several directions that we feel should be explored in the future.

First, we feel that our techniques could be adapted to show more lower bounds for (restricted versions of) DNNFs. One key ingredient of our proof is that we consider certificates in DNNFs. Note that restricted to branching programs these certificates degenerate from trees to paths, and indeed basically all lower bounds for branching programs that we are aware of argue on these paths. We feel that it would be very interesting to see which results on lower bounds for branching programs can be lifted to (suitably restricted versions of) DNNFs by considering certificates instead of paths. Note that in general this will not always be true (see e.g. [Raz14c, Theorem 5]), but it would be interesting to see where our techniques apply.

One related question is showing more unconditional separations on the different classes for knowledge representation in [DM02]. Most separations there are conditioned on the non-collapse of the polynomial hierarchy, and we feel that it would be very worthwhile—and in fact possible with today's tools—to show stronger, unconditional separations. Note that some of these separations have already been shown unconditionally in the meantime (see e.g. [BLRS13]), but many cases remain to be shown. One case that is in our opinion particularly interesting, is separating so-called deterministc DNNF from DNF. Obviously, every DNF is also a DNNF, but we conjecture that DNF cannot be compiled efficiently into the restricted class of deterministic DNNF. Note that this separation is true conditinally (see [DM02]), but showing it unconditionally would for the first time prove lower bounds on deterministic DNNFs that are not true for general DNNFs.

Finally, there is a long line of research proving upper bounds for DNNFs and restrictions (see [Raz14b, OD14a, OD14b] for some recent contributions). We feel that this approach should be continued and, in particular, also be complemented by lower bounds as in [Raz14b, Raz14a]. We hope that the ideas developed in this paper will contribute to this work.

Acknowledgements The first author was supported by the European Research Council (Complex Reason, 239962) and the FWF Austrian Science Fund (Parameterized Compilation, P26200). The third author has received partial support by a Qualcomm grant administered by École Polytechnique. The results of this paper were conceived during a research stay of the second and third author at the Vienna University of Technology. The stay of the second author was made possible by financial support by the ANR Blanc COMPA. The stay of the third author was made possible by financial support by the ANR Blanc International ALCOCLAN.

References

[AS00] Noga Alon and Joel H. Spencer. *The Probabilistic Method*. Wiley, 2000.

[BLRS13] Paul Beame, Jerry Li, Sudeepa Roy, and Dan Suciu. Lower Bounds for Exact Model Counting and Applications in Probabilistic Databases. In *Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence, Bellevue, WA, USA, August 11-15, 2013*, 2013.

[BLRS14] Paul Beame, Jerry Li, Sudeepa Roy, and Dan Suciu. Counting of Query Expressions: Limitations of Propositional Methods. In *Proc. 17th International Conference on Database Theory (ICDT), Athens, Greece, March 24-28, 2014.*, pages 177–188, 2014.

[BS14] Simone Bova and Friedrich Slivovsky. On Compiling Structured CNFs to OBDDs. submitted, 2014.

[Die05] Reinhard Diestel. *Graph Theory (Graduate Texts in Mathematics)*. Springer, August 2005.

[DM02] Adnan Darwiche and Pierre Marquis. A Knowledge Compilation Map. *J. Artif. Intell. Res. (JAIR)*, 17:229–264, 2002.

[GKPS95] Goran Gogic, Henry A. Kautz, Christos H. Papadimitriou, and Bart Selman. The Comparative Linguistics of Knowledge Representation. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence, IJCAI 95, Montréal Québec, Canada, August 20-25 1995, 2 Volumes*, pages 862–869, 1995.

[HLW06] Shlomo Hoory, Nathan Linial, and Avi Wigderson. Expander graphs and their applications. *Bull. Amer. Math. Soc. (N.S.)*, 43(4):439–561 (electronic), 2006.

[Juk12] Stasys Jukna. *Boolean Function Complexity - Advances and Frontiers*, volume 27 of *Algorithms and combinatorics*. Springer, 2012.

[KL80] Richard M. Karp and Richard J. Lipton. Some Connections between Nonuniform and Uniform Complexity Classes. In *Proceedings of the 12th Annual ACM Symposium on Theory of Computing, April 28-30, 1980, Los Angeles, California, USA*, pages 302–309, 1980.

[OD14a] Umut Oztok and Adnan Darwiche. CV-width: A New Complexity Parameter for CNFs. In *ECAI 2014 - 21st European Conference on Artificial Intelligence, 18-22 August 2014, Prague, Czech Republic - Including Prestigious Applications of Intelligent Systems (PAIS 2014)*, pages 675–680, 2014.

[OD14b] Umut Oztok and Adnan Darwiche. On Compiling CNF into Decision-DNNF. In *Principles and Practice of Constraint Programming - 20th International Conference, CP 2014, Lyon, France, September 8-12, 2014. Proceedings*, pages 42–57, 2014.

[PD10] Thammanit Pipatsrisawat and Adnan Darwiche. A Lower Bound on the Size of Decomposable Negation Normal Form. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2010, Atlanta, Georgia, USA, July 11-15, 2010*, 2010.

[Raz14a] Igor Razgon. No small nondeterministic read-once branching programs for CNFs of bounded treewidth. *CoRR*, abs/1407.0491v1, 2014.

[Raz14b] Igor Razgon. On OBDDs for CNFs of Bounded Treewidth. In *Principles of Knowledge Representation and Reasoning: Proceedings of the Fourteenth International Conference, KR 2014, Vienna, Austria, July 20-24, 2014*, 2014.

[Raz14c] Igor Razgon. On the read-once property of branching programs and CNFs of bounded treewidth. *ArXiv e-prints*, November 2014.

[Vol99] Heribert Vollmer. *Introduction to circuit complexity - a uniform approach*. Texts in theoretical computer science. Springer, 1999.

[Weg00] Ingo Wegener. *Branching Programs and Binary Decision Diagrams*. SIAM, 2000.