

Lower Bounds for Approximate Knowledge Compilation

Alexis de Colnet and Stefan Mengel

CRIL, CNRS & Univ Artois
 decolnet@cril.fr, mengel@cril.fr

Abstract

Knowledge compilation studies the trade-off between succinctness and efficiency of different representation languages. For many languages, there are known strong lower bounds on the representation size, but recent work shows that, for some languages, one can bypass these bounds using approximate compilation. The idea is to compile an approximation of the knowledge for which the number of errors can be controlled. We focus on circuits in deterministic decomposable negation normal form (d-DNNF), a compilation language suitable in contexts such as probabilistic reasoning, as it supports efficient model counting and probabilistic inference. Moreover, there are known size lower bounds for d-DNNF which by relaxing to approximation one might be able to avoid. In this paper we formalize two notions of approximation: *weak approximation* which has been studied before in the decision diagram literature and *strong approximation* which has been used in recent algorithmic results. We then show lower bounds for approximation by d-DNNF, complementing the positive results from the literature.

1 Introduction

Knowledge compilation is a subarea of artificial intelligence which studies different representations for knowledge [Darwiche and Marquis, 2002]. The basic idea is that different types of representation are more useful when solving reasoning problems than others. One general observation that has been made is that often representations that allow many reasoning tasks to be solved efficiently, such as the classical OBDDs, are necessarily large in size whereas more succinct representations often make reasoning hard. This trade-off between succinctness and usefulness is studied systematically in knowledge compilation.

One canonical area where the representation languages introduced in knowledge compilation are applied is probabilistic reasoning. For example, one can translate or *compile* classifiers based on graphical models, e.g. Bayesian networks, into such representations and then reason about the classifiers by querying the compiled representation [Chan and Dar-

wiche, 2003]. If the representation is of reasonable size and can be computed efficiently, then the overall reasoning process is efficient. The most important representation language in this setting are circuits in deterministic, decomposable negation normal form, short d-DNNF, which allow for efficient (weighted) model counting and probability computation and are thus particularly well suited for probabilistic reasoning [Darwiche, 2001]. d-DNNFs are generalizations of other important languages like OBDD [Bryant, 1986] and SDD [Darwiche, 2011] which have also found applications in probabilistic reasoning [Chan and Darwiche, 2003; Choi *et al.*, 2013; Shih *et al.*, 2019]. Due to their importance, essentially all practical implementations of knowledge compilers create d-DNNFs or sub-classes thereof [Darwiche, 2004; Muise *et al.*, 2012; Darwiche, 2011; Oztok and Darwiche, 2015; Lagniez and Marquis, 2017]. For these reasons we focus on d-DNNFs in this paper.

Unfortunately, in general, representations of knowledge in d-DNNF are large. This had been known under standard complexity theoretical assumptions for a long time [Darwiche and Marquis, 2002] and more recently there has been a series of papers showing exponential, unconditional lower bounds for many representation languages [Bova *et al.*, 2016; Beame *et al.*, 2017; Pipatsrisawat and Darwiche, 2010; Capelli, 2017; Beame and Liew, 2015]. Moreover, [Bova *et al.*, 2016] gave an explicit connection between DNNF lower bounds and communication complexity, a subarea of theoretical computer science. This makes it possible to use known results from communication complexity to get strong unconditional lower bounds in knowledge compilation. As one consequence, it is now known that the representation of many problems in d-DNNF is infeasible.

Fortunately, this bad news is not necessarily a fatal problem for probabilistic reasoning. Since graphical models like Bayesian networks are almost exclusively inferred by learning processes, they are inherently not exact representations of the world. Thus, when reasoning about them, in most cases the results do not have to be exact but approximate reasoning is sufficient, assuming that the approximation error can be controlled and is small. It is thus natural in this context to consider *approximate knowledge compilation*: the aim is no longer to represent knowledge exactly as one authorizes a small number of errors. Very recently, Chubarian and Turán [2020] showed, building on [Gopalan *et al.*, 2011],

that this approach is feasible in some settings: it is possible to compile approximations of so-called Tree Augmented Naive Bayes classifiers (TAN) (or more generally bounded pathwidth Bayes classifiers) into OBDDs efficiently. Note that efficient exact compilation is ruled out in this setting due to strong lower bounds for threshold functions from [Take-naga *et al.*, 1997] which imply lower bounds for TANs.

In this paper, we complement the positive results of [Chubarian and Turán, 2020] by extending lower bounds for exact representations to lower bounds for approximations. Similar questions have been treated before for OBDDs and some extensions such as *read-k branching programs*, see e.g. [Krause *et al.*, 1999; Bollig *et al.*, 2002]. We extend this line of work in two ways: we show that the techniques used in [Bollig *et al.*, 2002] can be adapted to show lower bounds for the approximation by d -DNNFs and prove that there are functions for which any d -DNNF computing a non-trivial approximation must have exponential size.

As a second contribution, we refine the approximation notion used in [Bollig *et al.*, 2002] which we call *weak approximation*. For this notion, the approximation quality is measured as the probability of encountering an error when comparing a function and its approximation on a random input. It follows that all families of Boolean functions for which the probability of encountering a model on a random input is not bounded by a constant, can be approximated trivially by constant functions (see Section 4 for details). This makes weak approximation easy for rather uninteresting reasons for many functions, e.g. most functions given by CNF-formulas. Moreover, it makes the approximation quality sensitive to encodings, in particular the use of auxiliary variables that functionally depend on the input. In general, the space of satisfying assignments is arguably badly described by weak approximations. In particular, the relative error for model counting and probability evaluation is unbounded which makes that notion useless for probabilistic reasoning.

We remedy the situation by formalizing a new notion of approximation for knowledge compilation which we call *strong approximation*. It is modeled to allow efficient counting with approximation guarantees and is insensitive to addition of functionally dependent auxiliary variables, see Section 4 for the definition and detailed discussion. While not formalized as such, it can be verified that the OBDDs of [Chubarian and Turán, 2020; Gopalan *et al.*, 2011] are in fact strong approximations in our sense. We then show that weak and strong approximations differ by exhibiting a family of functions that has trivial weak approximations but any d -DNNFs approximating it non-trivially must be of exponential size.

We remark that approximation in knowledge compilation has been considered before – in fact one of the earliest lines of work in the setting was approximating Boolean functions by Horn formulas [Selman and Kautz, 1996]. However, the focus was different in this setting: on the one hand, Horn formulas are not fully expressive so the question becomes that of understanding the formulas that are the best out of all Horn formulas approximating a function instead of requesting error guarantees for the approximation. On the other hand, that line of work was less concerned with the size of the approximating formulas and more with their existence. Our work

is different in these respects: since we deal with a fully expressive representation language, the main concern becomes that of a trade-off between the quality of approximation (measured in the number of inputs in which the function at hand and its approximation differ) and the representation size of the approximation.

Outline of the paper. We give some preliminaries in Section 2. We then introduce the notion of weak approximation and show our lower bound for it in Section 3. We introduce and discuss strong approximations next in Section 4 and show that weak and strong approximations differ in Section 5. We close the paper with some conclusions and open questions in Section 6. Due to space constraints some of the proofs are not contained in this version of the paper and will appear in the upcoming full version.

2 Preliminaries

We describe some conventions of notation for Boolean algebra. In our framework, a Boolean variable takes value 0 (*false*) or 1 (*true*), we see it as a variable over \mathbb{F}_2 , the field with two elements. Assignments of n Boolean variables are vectors from \mathbb{F}_2^n and operations on vectors and matrices are considered in this field. We use the notation $\mathbf{0}^n$ to denote the 0-vector from \mathbb{F}_2^n . For clarity we also use the operators \neg , \vee and \wedge for negation, disjunction and conjunction in \mathbb{F}_2 . The conjunction of Boolean variables and the product in \mathbb{F}_2 are equivalent and used interchangeably. Single variables are written in plain style “ x ” while assignments of $n > 1$ variables use bold style “ \mathbf{x} ”. A Boolean function on n variables is a mapping $f : \mathbb{F}_2^n \rightarrow \mathbb{F}_2$ and its models are given by $f^{-1}(1)$. Given a set of assignments S , we sometimes denote $\mathbb{1}_S$ the Boolean function whose set of models is exactly S . We write $f \leq g$ when $f^{-1}(1) \subseteq g^{-1}(1)$, which corresponds to logical entailment. A distribution on truth assignments is a probabilistic distribution \mathcal{D} on \mathbb{F}_2^n . We write $\Pr_{\mathbf{x} \sim \mathcal{D}}[\cdot]$ to denote the probability measure when sampling an assignment \mathbf{x} according to \mathcal{D} . For clarity, the uniform distribution on \mathbb{F}_2^n is denoted \mathcal{U} (regardless of n), $\mathbf{x} \sim \mathcal{U}$ means that any assignment is sampled with probability $1/2^n$.

Deterministic decomposable NNF. Let X be a finite set of Boolean variables. A circuit in *negation normal form* (NNF) over X is a single output Boolean circuit whose inputs gates are labeled with Boolean variables x from X and their negations $\neg x$ and whose internal gates are fanin-2 AND and OR-gates. The *size* of a circuit is the number of its gates. A circuit over X is said to *accept* a truth assignment \mathbf{x} of the variables if it outputs 1 (*true*) when its inputs are set as in \mathbf{x} . In this case \mathbf{x} is a *model* of the function represented by the circuit. An NNF is *decomposable* if, for any AND-gate g , the two sub-circuits rooted at g share no input variable, i.e., if x or $\neg x$ is an input gate of the circuit rooted at the left input of g , then neither x nor $\neg x$ is an input gate of the subcircuit rooted at the right input, and vice versa. An NNF is *deterministic* if, for any OR-gate g , the sets of assignments accepted by the two subcircuits rooted at the children of g are disjoint. A decomposable NNF is called a DNNF; if in addition it is deterministic, then it is called a d -DNNF.

Rectangle covers. Let X be a finite set of Boolean variables. A *combinatorial rectangle* over X (more succinctly a *rectangle*) is a Boolean function r defined as the conjunction of two Boolean functions ρ_1 and ρ_2 over disjoint variables of X . That is, there is a partition (X_1, X_2) of X such that ρ_1 and ρ_2 are defined over X_1 and X_2 , respectively, and $r = \rho_1 \wedge \rho_2$. We call (X_1, X_2) the *partition* of r . The rectangle is *balanced* if $|X|/3 \leq |X_1| \leq 2|X|/3$ (the same bounds hold for $|X_2|$). A *rectangle cover* of a Boolean function f is any disjunction of rectangles over X (possibly for different partitions of X) equivalent to f , i.e., $f = \bigvee_{i=1}^K r_i$ where the r_i are rectangles. The *size* of a cover is the number K of its rectangles. A rectangle cover is called *balanced* if its rectangles are balanced and it is said *disjoint* if no two rectangles share a model. Note that any function f has at least one balanced disjoint rectangle cover, because it can be written as a DNF in which every term contains all variables. There is a tight link between the smallest size of a balanced disjoint rectangle cover of a function and the size of any equivalent d-DNNF.

Theorem 1. [Bova *et al.*, 2016] *Let D be a d-DNNF encoding a function f . Then f has a balanced disjoint rectangle cover of size at most the size of D .*

Theorem 1 implies that, to show a lower bound on the size of any d-DNNF encoding f , it is sufficient to find a lower bound on the size of any balanced disjoint rectangle cover of f .

3 Large d-DNNFs for Weak Approximations

In this section, we start by considering the notion of approximation that has been studied for different forms of branching programs before, see e.g. [Krause *et al.*, 1999; Bollig *et al.*, 2002]. To differentiate it from other notions, we give it the name *weak approximation*.

Definition 1 (Weak approximation). *Let \mathcal{D} be a distribution on the truth assignments to X and $\varepsilon > 0$. We say that \tilde{f} is a weak ε -approximation of f (or weakly ε -approximates f) with respect to \mathcal{D} if*

$$\Pr_{\mathbf{x} \sim \mathcal{D}} [f(\mathbf{x}) \neq \tilde{f}(\mathbf{x})] \leq \varepsilon.$$

When \mathcal{D} is the uniform distribution \mathcal{U} , then the condition of weak ε -approximability is equivalent to $|\{\mathbf{x} : f(\mathbf{x}) \neq \tilde{f}(\mathbf{x})\}| \leq \varepsilon 2^n$.

Note that weak ε -approximation is only useful when $\varepsilon < 1/2$. This is because every function has a trivial $(1/2)$ -approximation: if $\Pr_{\mathbf{x} \sim \mathcal{D}} [f(\mathbf{x}) = 1] > 1/2$, then the constant 1-function is a $(1/2)$ -approximation, otherwise this is the case for the constant 0-function. Note that it might be hard to decide which case is true, but in any case we know that the approximation ratio of one of the constants is good.

Bollig *et al.* [2002] used a *discrepancy* argument to show that there are classes of functions such that any ε -approximation w.r.t. \mathcal{U} requires exponential OBDD size. We lift their techniques to d-DNNF showing that the same functions are also hard for d-DNNF.

Theorem 2. *Let $0 \leq \varepsilon < 1/2$, there is a class of Boolean functions \mathcal{C} such that, for any $f \in \mathcal{C}$ on n variables, any d-DNNF encoding a weak ε -approximation of f w.r.t. \mathcal{U} has size $2^{\Omega(n)}$.*

Since d-DNNFs are strictly more succinct than OBDDs [Darwiche and Marquis, 2002], Theorem 2 is a generalization of the result on OBDDs in [Bollig *et al.*, 2002]. However, since the proof is almost identical, differing near the end only, we defer the technical details to the full version. We here only introduce the notion of discrepancy that is central to the proof and will be useful later.

The discrepancy method. We want to use Theorem 1 to bound the size of a d-DNNF encoding \tilde{f} a weak ε -approximation of f w.r.t. some distribution. To this end we study disjoint balanced rectangle covers of \tilde{f} . Let r be a rectangle from such a cover. r can make *false positives* on f , i.e., have models that are not models of f . Similarly, *true positives* are models shared by r and f . The *discrepancy* $\text{Disc}(f, r)$ of f on r is the difference between the number of false positives and true positives, normalized by the total number of assignments: $\text{Disc}(f, r) := \frac{1}{2^n} |r^{-1}(1) \cap f^{-1}(1)| - |r^{-1}(1) \cap f^{-1}(0)|$. A small discrepancy indicates that r has few models or that it makes roughly as many false positives as true positives on f . Discrepancy bounds have been used before to prove results in distributional communication complexity [Kushilevitz and Nisan, 1997, Chapter 3.5]. Here we show that when there is an upper bound on $\text{Disc}(f, r)$ for any rectangle r from a cover of \tilde{f} , one can obtain a lower bound on the size of the cover of \tilde{f} .

Lemma 1. *Let f be a Boolean function on n variables and let \tilde{f} be a weak ε -approximation of f w.r.t. \mathcal{U} . Let $\tilde{f} = \bigvee_{k=1}^K r_k$ be a disjoint balanced rectangle cover of \tilde{f} and assume that there is an integer $\Delta > 0$ such that $\text{Disc}(f, r_k) \leq \Delta/2^n$ for all r_k . Then $K \geq (|f^{-1}(1)| - \varepsilon 2^n)/\Delta$.*

$$\begin{aligned} \text{Proof. We have } |f \neq \tilde{f}| &= |\{\mathbf{x} : f(\mathbf{x}) \neq \tilde{f}(\mathbf{x})\}| \\ &= |f^{-1}(1) \cap \tilde{f}^{-1}(0)| + |f^{-1}(0) \cap \tilde{f}^{-1}(1)| \\ &= |f^{-1}(1) \cap \bigcap_{k=1}^K r_k^{-1}(0)| + |f^{-1}(0) \cap \bigcup_{k=1}^K r_k^{-1}(1)| \\ &= |f^{-1}(1)| - \sum_{k=1}^K (|r_k^{-1}(1) \cap f^{-1}(1)| - |r_k^{-1}(1) \cap f^{-1}(0)|) \\ &\geq |f^{-1}(1)| - 2^n \sum_{k=1}^K \text{Disc}(f, r_k) \geq |f^{-1}(1)| - K\Delta \end{aligned}$$

where the last equality is due to the rectangles being disjoint. The weak ε -approximation w.r.t. the uniform distribution \mathcal{U} gives that $|\tilde{f} \neq f| \leq \varepsilon 2^n$, which we use to conclude. \square

Combining Lemma 1 with Theorem 1, the proof of Theorem 2 boils down to showing that there are functions such that for every balanced rectangle r , the discrepancy $\text{Disc}(f, r)$ can be suitably bounded, as shown in [Bollig *et al.*, 2002].

4 Strong Approximations

In this section, we discuss some shortcomings of weak approximation and propose a stronger notion of approximation that avoids them. Let f_0 be the constant 0-function. We say that a function is *trivially weakly ε -approximable* (w.r.t. some distribution) if f_0 is a weak ε -approximation. Considering approximations w.r.t. the uniform distribution, it is easy to find classes of functions that are trivially weakly approximable.

Lemma 2. *Let $\varepsilon > 0$ and $0 \leq \alpha < 1$. Let \mathcal{C} be a class of functions such that every function in \mathcal{C} on n variables has at most $2^{\alpha n}$ models. Then there exists a constant n_0 , such that any function from \mathcal{C} on more than n_0 variables is trivially weakly ε -approximable w.r.t. the uniform distribution.*

Proof. Take $n_0 = \frac{1}{1-\alpha} \log(\frac{1}{\varepsilon})$ and choose f any function from \mathcal{C} on $n > n_0$ variables. Then $|\{x : f(x) \neq f_0(x)\}| = |f^{-1}(1)| \leq 2^{\alpha n} < \varepsilon 2^n$. Therefore f_0 is a weak ε -approximation (w.r.t. the uniform distribution) of any function of \mathcal{C} on sufficiently many variables. \square

We remark that similar trivial approximation results can be shown for other distributions if the probability of a random assignment w.r.t. this distribution being a model is very small.

As a consequence, weak approximation makes no sense for functions with “few” (or “improbable”) models. However such functions are often encountered, for example, random k -CNF with sufficiently many clauses are expected to have few models. Furthermore, even for functions with “many” models, one often studies encodings over larger sets of variables. For instance, when using Tseitin encoding to transform Boolean circuits into CNF, one introduces auxiliary variables that compute the value of sub-circuits under a given assignment. Generally, auxiliary variables are often used in practice since they reduce the representation size of functions, see e.g. [Biere *et al.*, 2009, Chapter 2]. The resulting encodings have more variables but most of the time the same number of models as the initial function. Consequently, they are likely to be trivially weakly approximable from Lemma 2. For these reasons we define a stronger notion of approximation.

Definition 2 (Strong approximation). *Let \mathcal{D} be a distribution of the truth assignments to X and $\varepsilon > 0$. We say that \tilde{f} is a strong ε -approximation of f (or strongly ε -approximates f) with respect to \mathcal{D} if*

$$\Pr_{x \sim \mathcal{D}} [f(x) \neq \tilde{f}(x)] \leq \varepsilon \Pr_{x \sim \mathcal{D}} [f(x) = 1].$$

When \mathcal{D} is the uniform distribution \mathcal{U} , then the condition of strong approximability is equivalent to $|\{x : f(x) \neq \tilde{f}(x)\}| \leq \varepsilon |f^{-1}(1)|$. It is easy to see that strong approximation does not have the problem described in Lemma 2 for weak approximation. We also remark that strong approximation has been modeled to allow for efficient counting. In fact, a d -DNNF computing a strong ε -approximation of a function f allows approximate model counting for f with approximation factor ε .

Strong approximation has implicitly already been used in knowledge compilation. For instance it has been shown in [Gopalan *et al.*, 2011] – although the authors use a different terminology – that for $\varepsilon > 0$, any Knapsack functions on n variables has a strong ε -approximation w.r.t. \mathcal{U} that can be encoded by an OBDD of size polynomial in n and $1/\varepsilon$. The generalization to TANs [Chubarian and Turán, 2020] is also for strong approximations. These results are all the more significant since we know from [Takenaga *et al.*, 1997] that there exist threshold functions for which exact encodings by OBDD require size exponential in n .

Obviously, a strong approximation of f w.r.t. some distribution is also a weak approximation. Thus the statement of Theorem 2 can trivially be lifted to strong approximation. However the hard functions from Theorem 2 necessarily have sufficiently many models: if we are to consider only functions with few models, then they all are trivially weakly approximable. Yet we prove in the next section that there exist such functions whose exact encoding and strong ε -approximation encodings by d -DNNF require size exponential in n . Our proof follows the discrepancy method but relies on the following variant of Lemma 1 for strong approximation.

Lemma 3. *Let f be a Boolean function on n variables and let \tilde{f} be a strong ε -approximation of f w.r.t. \mathcal{U} . Let $\tilde{f} = \bigvee_{k=1}^K r_k$ be a disjoint balanced rectangle cover of \tilde{f} and assume that there is an integer $\Delta > 0$ such that $\text{Disc}(f, r_k) \leq \Delta/2^n$ for all r_k . Then $K \geq (1 - \varepsilon) |f^{-1}(1)| / \Delta$.*

Proof. The proof is essentially the same as for Lemma 1, differing only in the last lines where we use $|f \neq \tilde{f}| \leq \varepsilon |f^{-1}(1)|$ rather than $|\tilde{f} \neq f| \leq \varepsilon 2^n$. \square

5 Large d -DNNFs for Strong Approximations

In this section, we show a lower bound for strong approximations of some functions that have weak approximations by small d -DNNFs. The functions we consider are characteristic functions of linear codes which we introduce now: a *linear code* of length n is a linear subspace of the vector space \mathbb{F}_2^n . Vectors from this subspace are called *code words*. A linear code is characterized by a *parity check matrix* H from $\mathbb{F}_2^{m \times n}$ as follows: a vector $x \in \mathbb{F}_2^n$ is a code word if and only if $Hx = \mathbf{0}^m$ (operations are modulo 2 in \mathbb{F}_2^n). The *characteristic function* of a linear code is a Boolean function which accepts exactly the code words. Note that the characteristic function of a length n linear code of check matrix H has $2^{n-\text{rk}(H)}$ models, where $\text{rk}(H)$ denotes the rank of H . Following ideas developed in [Duris *et al.*, 2004], we focus on linear codes whose check matrices H have the following property: H is called *s-good* for some integer s if any submatrix obtained by taking at least $n/3$ columns¹ from H has rank at least s . The existence of s -good matrices for $s = m - 1$ is guaranteed by the next lemma.

Lemma 4. [Duris *et al.*, 2004] *Let $m = n/100$ and sample a parity check matrix H uniformly at random from $\mathbb{F}_2^{m \times n}$. Then H is $(m - 1)$ -good with probability $1 - 2^{-\Omega(n)}$.*

Our interest in linear codes characterized by s -good matrices is motivated by another result from [Duris *et al.*, 2004] which states that the maximal size of any rectangle entailing the characteristic function of such a code decreases as s increases.

Lemma 5. [Duris *et al.*, 2004] *Let f be the characteristic function of a linear code of length n characterized by the s -good matrix H . Let r be a balanced rectangle such that $r \leq f$. Then $|r^{-1}(1)| \leq 2^{n-2s}$.*

¹Duris *et al.* [2004] limit to submatrices built from at least $n/2$ columns rather than $n/3$; however their result can easily be adapted.

Combining Lemmas 4 and 5 with Theorem 1, one gets the following result that was already observed in [Mengel, 2016]:

Theorem 3. *There exists a class of linear codes \mathcal{C} such that, for any code from \mathcal{C} of length n , any \mathbf{d} -DNNF encoding its characteristic function has size $2^{\Omega(n)}$.*

In the following, we will show that not only are characteristic functions hard to represent exactly as \mathbf{d} -DNNF, they are even hard to strongly approximate.

Given the characteristic function f of a length n linear code of check matrix H , f has exactly $2^{n-\text{rk}(H)}$ models. When $\text{rk}(H)$ is at least a constant fraction of n , f satisfies the condition of Lemma 2, so for every $\varepsilon > 0$ and n large enough, f is trivially weakly ε -approximable (w.r.t. the uniform distribution). However we will show that any strong ε -approximation \tilde{f} of f (w.r.t. the uniform distribution) only has \mathbf{d} -DNNF encodings of size exponential in n .

To show this result, we will use the discrepancy method: we are going to find a bound on the discrepancy of f on any rectangle from a balanced disjoint rectangle cover of \tilde{f} . Then we will use the bound in Lemma 3 and combine the result with Theorem 1 to finish the proof.

Note that it is possible that a rectangle from a disjoint rectangle cover of \tilde{f} makes no false positives on f . In fact, if this is the case for all rectangles in the cover, then $\tilde{f} \leq f$. In this case, lower bounds can be shown essentially as in the proof of Theorem 3. The more interesting case is thus that in which rectangles make false positives. In this case, we assume that no rectangle makes more false positives on f than it accepts models of f , because if such a rectangle r exists in a disjoint cover of \tilde{f} , then deleting r leads to a better approximation of f than \tilde{f} . Thus it is sufficient to consider approximations and rectangle covers in which all rectangles verify $|r^{-1}(1) \cap f^{-1}(1)| \geq |r^{-1}(1) \cap f^{-1}(0)|$.

Definition 3. *Let r be a rectangle. A core rectangle (more succinctly a core) of r w.r.t. f is a rectangle r_{core} with the same partition as r such that*

- a) $r_{\text{core}} \leq f$ and $r_{\text{core}} \leq r$,
- b) r_{core} is maximal in the sense that there is no r' satisfying a) such that $|r'^{-1}(1)| > |r_{\text{core}}^{-1}(1)|$.

Note that if $r \leq f$, then the only core rectangle of r is r itself. Otherwise r may have several core rectangles. We next state a crucial lemma on the relation of discrepancy and cores whose proof we defer to later parts of this section.

Lemma 6. *Let f be the characteristic function of some length n linear code, let r be a rectangle with more true positives than false positives on f , and let r_{core} be a core rectangle of r with respect to f , then*

$$\text{Disc}(f, r) \leq \frac{1}{2^n} |r_{\text{core}}^{-1}(1)|.$$

Lemma 6 says the following: consider a rectangle $r_{\text{core}} \leq f$ which is a core of a rectangle r . If r accepts more models of f than r_{core} , then for each additional such model r accepts at least one false positive. With Lemma 6, it is straightforward to show the main result of this section.

Theorem 4. *Let $0 \leq \varepsilon < 1$. There is a class of Boolean functions \mathcal{C} such that any $f \in \mathcal{C}$ on n variables is trivially weakly ε -approximable w.r.t. \mathcal{U} but any \mathbf{d} -DNNF encoding a strong ε -approximation w.r.t. \mathcal{U} has size $2^{\Omega(n)}$.*

Proof. Choose \mathcal{C} to be the class of characteristic functions for length n linear codes characterized by $(m-1)$ -good check matrices with $m = n/100$. Existence of these functions as n increases is guaranteed by Lemma 4. Let \tilde{f} be a strong ε -approximation of $f \in \mathcal{C}$ w.r.t. \mathcal{U} and let $\bigvee_{k=1}^K r_k$ be a rectangle cover of \tilde{f} . Combining Lemma 6 with Lemma 5, we obtain $\text{Disc}(f, r_k) \leq 2^{-n} 2^{n-2(m-1)}$. We then use Lemma 3 to get $K \geq (1-\varepsilon)2^{2m-n} |f^{-1}(1)|/4$. The rank of the check matrix of f is at most m so $|f^{-1}(1)| \geq 2^{n-m}$ and $K \geq (1-\varepsilon)2^m/4 = (1-\varepsilon)2^{\Omega(n)}$. We use Theorem 1 to conclude. \square

Note that Theorem 4 is optimal w.r.t. ε since for $\varepsilon = 1$ there is always the trivial approximation by the constant 0-function.

It remains to show Lemma 6 in the remainder of this section to complete the proof of Theorem 4. To this end, we make another definition.

Definition 4. *Let (X_1, X_2) be a partition of the variables of f . A core extraction operator w.r.t. f is a mapping \mathcal{C}_f that maps every pair (S_1, S_2) of sets of assignments over X_1 and X_2 , respectively, to a pair (S'_1, S'_2) with*

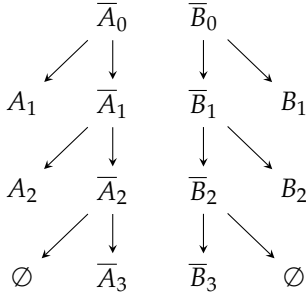
- a) $S'_1 \subseteq S_1$ and $S'_2 \subseteq S_2$,
- b) assignments from $S'_1 \times S'_2$ are models of f ,
- c) if f has no model in $S_1 \times S_2$, then $S'_1 = S'_2 = \emptyset$,
- d) S'_1 and S'_2 are maximal in the sense that for every $S''_1 \subseteq S_1$ and every $S''_2 \subseteq S_2$ respecting the properties a), b) and c), we have $|S'_1||S'_2| \geq |S''_1||S''_2|$.

Intuitively S'_1 and S'_2 are the largest subsets one can extract from S_1 and S_2 such that assignments from $S'_1 \times S'_2$ are models of f . Note that, similarly to rectangle cores, the sets S'_1 and S'_2 are not necessarily uniquely defined. In this case, we assume that \mathcal{C}_f returns an arbitrary pair with the required properties. One can show that core extraction operators yield core rectangles, as their name suggests.

Claim 1. *Let $r := \rho_1 \wedge \rho_2$ be a rectangle w.r.t. the partition (X_1, X_2) and denote $(A, B) := \mathcal{C}_f(\rho_1^{-1}(1), \rho_2^{-1}(1))$. Then the rectangle $\mathbb{1}_A \wedge \mathbb{1}_B$ is a core rectangle of r w.r.t. f .*

The proof of Claim 1 and those of several other claims in this section are deferred to the full version due to space constraints. At this point, recall that f is the characteristic function of a linear code for a $m \times n$ check matrix H .

Claim 2. *Let $r := \rho_1 \wedge \rho_2$ be a rectangle w.r.t. the partition (X_1, X_2) . Let $(A, B) := \mathcal{C}_f(\rho_1^{-1}(1), \rho_2^{-1}(1))$ and consider the core rectangle $r_{\text{core}} := \mathbb{1}_A \wedge \mathbb{1}_B$. Let $\bar{A} = \rho_1^{-1}(1) \setminus A$ and $\bar{B} = \rho_2^{-1}(1) \setminus B$. Then all assignments from $\bar{A} \times \bar{B}$ and $A \times \bar{B}$ are false positives of r on f .*


 Figure 1: An iterative core extraction with $l = 2$

Proof. Index the n columns of H with the variables in X (x_1 for column 1, x_2 for column 2, and so on). Let H_1 (resp. H_2) be the matrix obtained taking only the columns of H whose indices are in X_1 (resp. X_2). Obviously all vectors in $\bar{A} \times B$ and $A \times \bar{B}$ are models of r , but we will prove that they are not models of f . For every $\mathbf{a}' \in \bar{A}$ there is $\mathbf{b} \in B$ such that $H(\mathbf{a}', \mathbf{b}) = H_1 \mathbf{a}' + H_2 \mathbf{b} \neq \mathbf{0}^m$, otherwise the core rectangle would not be maximal. By definition of A and B , given $\mathbf{a} \in A$, for all $\mathbf{b} \in B$ we have $H(\mathbf{a}, \mathbf{b}) = H_1 \mathbf{a} + H_2 \mathbf{b} = \mathbf{0}^m$, so $H_2 \mathbf{b}$ is constant over B . Therefore if $H_1 \mathbf{a}' \neq H_2 \mathbf{b}$ for some $\mathbf{b} \in B$ then $H_1 \mathbf{a}' \neq H_2 \mathbf{b}$ for all $\mathbf{b} \in B$. But then no vector from $\{\mathbf{a}'\} \times B$ can be a model of f and since \mathbf{a}' has been chosen arbitrarily in \bar{A} , all vectors from $\bar{A} \times B$ are false positives. The case for $A \times \bar{B}$ follows analogously. \square

For A and B defined as in Claim 2, we know that the assignments from $A \times B$ are models of f , and that those from $\bar{A} \times B$ and $A \times \bar{B}$ are not, but we have yet to discuss the case of $\bar{A} \times \bar{B}$. There may be additional models in this last set. The key to proving Lemma 6 is to iteratively extract core rectangles from $\mathbb{1}_{\bar{A}} \wedge \mathbb{1}_{\bar{B}}$ and control how many false positives are generated at each step of the iteration. To this end we define the collection $((A_i, B_i))_{i=0}^{l+1}$ as follows:

- $A_0 = \rho_1^{-1}(1)$ and $B_0 = \rho_2^{-1}(1)$,
- for $i \geq 1$, $(A_i, B_i) = \mathcal{C}_f(A_0 \setminus \bigcup_{j=1}^{i-1} A_j, B_0 \setminus \bigcup_{j=1}^{i-1} B_j)$,
- A_{l+1} and B_{l+1} are empty, but for any $i < l + 1$, neither A_i nor B_i is empty.

Denoting $\bar{A}_i := A_0 \setminus \bigcup_{j=1}^i A_j$ and $\bar{B}_i := B_0 \setminus \bigcup_{j=1}^i B_j$, we can write $(A_i, B_i) = \mathcal{C}_f(\bar{A}_{i-1}, \bar{B}_{i-1})$ (note that $\bar{A}_0 = A_0$ and $\bar{B}_0 = B_0$). Basically, we extract a core $(\mathbb{1}_{A_1} \wedge \mathbb{1}_{B_1})$ from r , then we extract a core $(\mathbb{1}_{A_2} \wedge \mathbb{1}_{B_2})$ from $(\mathbb{1}_{\bar{A}_1} \wedge \mathbb{1}_{\bar{B}_1})$, and so on until there is no model of f left in $\bar{A}_l \times \bar{B}_l$, in which case no core can be extracted from $(\mathbb{1}_{\bar{A}_l} \wedge \mathbb{1}_{\bar{B}_l})$ and $\mathcal{C}_f(\bar{A}_l, \bar{B}_l)$ returns (\emptyset, \emptyset) . The construction is illustrated in Figure 1.

Claim 3. For any $i > 0$, all assignments from $F_i := (A_i \times \bar{B}_i) \cup (\bar{A}_i \times B_i)$ are false positives of r on f . Furthermore for every $i \neq j$ we have $F_i \cap F_j = \emptyset$.

Claim 4. The function $\bigvee_{i=1}^l (\mathbb{1}_{A_i} \wedge \mathbb{1}_{B_i})$ is a disjoint rectangle cover of $r \wedge f$. Furthermore, if r is balanced, so are the rectangles from $\bigvee_{i=1}^l (\mathbb{1}_{A_i} \wedge \mathbb{1}_{B_i})$.

With Claim 3 and Claim 4, we can now prove Lemma 6.

Proof of Lemma 6. Claims 3 and 4 show that $\bigcup_{i=1}^l (A_i \times B_i) = r^{-1}(1) \cap f^{-1}(1)$ and $\bigcup_{i=1}^l ((A_i \times \bar{B}_i) \cup (\bar{A}_i \times B_i)) \subseteq r^{-1}(1) \cap f^{-1}(0)$ and that these unions are disjoint. First we focus on the models of f covered by r .

$$|r^{-1}(1) \cap f^{-1}(1)| = \sum_{i=1}^l |A_i| |B_i| = |r_{\text{core}}^{-1}(1)| + \sum_{i=2}^l |A_i| |B_i|$$

where $r_{\text{core}} = \mathbb{1}_{A_1} \wedge \mathbb{1}_{B_1}$ is the first (therefore the largest) core rectangle extracted from r w.r.t. f . Now focus on the false positives of r on f

$$\begin{aligned} |r^{-1}(1) \cap f^{-1}(0)| &\geq \sum_{i=1}^l (|A_i| |\bar{B}_i| + |\bar{A}_i| |B_i|) \\ &\geq \sum_{i=1}^l (|A_i| |B_{i+1}| + |A_{i+1}| |B_i|) \end{aligned}$$

The maximality property of \mathcal{C}_f implies $|A_i| |B_i| \geq |A_{i+1}| |B_{i+1}|$, and it follows that $|A_i| |B_{i+1}| + |A_{i+1}| |B_i| \geq |A_{i+1}| |B_{i+1}|$. Thus

$$|r^{-1}(1) \cap f^{-1}(0)| \geq |r^{-1}(1) \cap f^{-1}(1)| - |r_{\text{core}}^{-1}(1)|.$$

By assumption, r accepts more models of f than false positives so $\text{Disc}(f, r) = (|r^{-1}(1) \cap f^{-1}(1)| - |r^{-1}(1) \cap f^{-1}(0)|) / 2^n$ and the lemma follows directly. \square

6 Conclusion

In this paper, we have formalized and studied weak and strong approximation in knowledge compilation and shown functions that are hard to approximate by d-DNNFs with respect to these two notions. In particular, we have shown that strong approximations by d-DNNFs generally require exponentially bigger d-DNNF representations than weak approximations.

Let us sketch some directions for future research. One obvious question is to find for which classes of functions there are efficient algorithms computing approximations by d-DNNFs. In [Chubarian and Turán, 2020], it is shown that this is the case for certain Bayesian networks. It would be interesting to extend this to other settings to make approximation more applicable in knowledge compilation. Of particular interest are in our opinion settings in which models are typically learned from data and thus inherently inexact, e.g. other forms of graphical models and neural networks.

Another question is defining and analyzing more approximation notions beyond weak and strong approximation. In fact, the latter was designed to allow approximate (weighted) counting as needed in probabilistic reasoning. Are there ways of defining notions of approximation that are useful for other problems, say optimization or entailment queries?

A more technical question is if one can show lower bounds for non-deterministic DNNFs. In that setting, different rectangles may share the same false positives in which case our lower bound techniques break down. Are there approaches to avoid this problem?

Acknowledgments

This work has been partly supported by the PING/ACK project of the French National Agency for Research (ANR-18-CE40-0011).

References

- [Beame and Liew, 2015] Paul Beame and Vincent Liew. New Limits for Knowledge Compilation and Applications to Exact Model Counting. In *Conference on Uncertainty in Artificial Intelligence, UAI*, pages 131–140, 2015.
- [Beame *et al.*, 2017] Paul Beame, Jerry Li, Sudeepa Roy, and Dan Suci. Exact Model Counting of Query Expressions: Limitations of Propositional Methods. *ACM Trans. Database Syst.*, 42(1):1:1–1:46, 2017.
- [Biere *et al.*, 2009] Armin Biere, Marijn Heule, Hans van Maaren, and Toby Walsh, editors. *Handbook of Satisfiability*, volume 185 of *Frontiers in Artificial Intelligence and Applications*. IOS Press, 2009.
- [Bollig *et al.*, 2002] Beate Bollig, Martin Sauerhoff, and Ingo Wegener. On the Nonapproximability of Boolean Functions by OBDDs and Read-k-Times Branching Programs. *Inf. Comput.*, 178(1):263–278, 2002.
- [Bova *et al.*, 2016] Simone Bova, Florent Capelli, Stefan Mengel, and Friedrich Slivovsky. Knowledge Compilation Meets Communication Complexity. In *International Joint Conference on Artificial Intelligence, IJCAI*, pages 1008–1014, 2016.
- [Bryant, 1986] Randal E. Bryant. Graph-Based Algorithms for Boolean Function Manipulation. *IEEE Trans. Computers*, 35(8):677–691, 1986.
- [Capelli, 2017] Florent Capelli. Understanding the complexity of #SAT using knowledge compilation. In *ACM/IEEE Symposium on Logic in Computer Science, LICS*, pages 1–10, 2017.
- [Chan and Darwiche, 2003] Hei Chan and Adnan Darwiche. Reasoning about Bayesian Network Classifiers. In *Conference in Uncertainty in Artificial Intelligence, UAI*, pages 107–115, 2003.
- [Choi *et al.*, 2013] Arthur Choi, Doga Kisa, and Adnan Darwiche. Compiling Probabilistic Graphical Models Using Sentential Decision Diagrams. In *Symbolic and Quantitative Approaches to Reasoning with Uncertainty, ECSQARU*, volume 7958, pages 121–132, 2013.
- [Chubarian and Turán, 2020] Karine Chubarian and György Turán. Interpretability of Bayesian Network Classifiers: OBDD Approximation and Polynomial Threshold Functions. In *International Symposium on Artificial Intelligence and Mathematics, ISAIM*, 2020.
- [Darwiche and Marquis, 2002] Adnan Darwiche and Pierre Marquis. A Knowledge Compilation Map. *J. Artif. Intell. Res.*, 17:229–264, 2002.
- [Darwiche, 2001] Adnan Darwiche. On the Tractable Counting of Theory Models and its Application to Truth Maintenance and Belief Revision. *Journal of Applied Non-Classical Logics*, 11(1-2):11–34, 2001.
- [Darwiche, 2004] Adnan Darwiche. New Advances in Compiling CNF into Decomposable Negation Normal Form. In *European Conference on Artificial Intelligence, ECAI*, pages 328–332, 2004.
- [Darwiche, 2011] Adnan Darwiche. SDD: A New Canonical Representation of Propositional Knowledge Bases. In *International Joint Conference on Artificial Intelligence, IJCAI*, pages 819–826, 2011.
- [Duris *et al.*, 2004] Pavol Duris, Juraj Hromkovic, Stasys Jukna, Martin Sauerhoff, and Georg Schnitger. On multi-partition communication complexity. *Inf. Comput.*, 194(1):49–75, 2004.
- [Gopalan *et al.*, 2011] Parikshit Gopalan, Adam R. Klivans, Raghu Meka, Daniel Stefankovic, Santosh S. Vempala, and Eric Vigoda. An FPTAS for #Knapsack and Related Counting Problems. In *IEEE Symposium on Foundations of Computer Science, FOCS*, pages 817–826, 2011.
- [Krause *et al.*, 1999] Matthias Krause, Petr Savický, and Ingo Wegener. Approximations by OBDDs and the Variable Ordering Problem. In *International Colloquium Automata, Languages and Programming, ICALP*, pages 493–502, 1999.
- [Kushilevitz and Nisan, 1997] Eyal Kushilevitz and Noam Nisan. *Communication complexity*. Cambridge University Press, 1997.
- [Lagniez and Marquis, 2017] Jean-Marie Lagniez and Pierre Marquis. An Improved Decision-DNNF Compiler. In *International Joint Conference on Artificial Intelligence, IJCAI*, pages 667–673, 2017.
- [Mengel, 2016] Stefan Mengel. Parameterized Compilation Lower Bounds for Restricted CNF-Formulas. In *International Conference Theory and Applications of Satisfiability Testing, SAT*, pages 3–12, 2016.
- [Muise *et al.*, 2012] Christian J. Muise, Sheila A. McIlraith, J. Christopher Beck, and Eric I. Hsu. Dsharp: Fast d-DNNF Compilation with sharpSAT. In *Canadian Conference on Artificial Intelligence*, pages 356–361, 2012.
- [Oztok and Darwiche, 2015] Umut Oztok and Adnan Darwiche. A Top-Down Compiler for Sentential Decision Diagrams. In *International Joint Conference on Artificial Intelligence, IJCAI*, pages 3141–3148, 2015.
- [Pipatsrisawat and Darwiche, 2010] Thammanit Pipatsrisawat and Adnan Darwiche. A Lower Bound on the Size of Decomposable Negation Normal Form. In *AAAI Conference on Artificial Intelligence, AAAI*, pages 345–350, 2010.
- [Selman and Kautz, 1996] Bart Selman and Henry A. Kautz. Knowledge Compilation and Theory Approximation. *J. ACM*, 43(2):193–224, 1996.
- [Shih *et al.*, 2019] Andy Shih, Arthur Choi, and Adnan Darwiche. Compiling Bayesian Network Classifiers into Decision Graphs. In *AAAI Conference on Artificial Intelligence, AAAI*, pages 7966–7974, 2019.
- [Takenaga *et al.*, 1997] Yasuhiko Takenaga, Mitsushi Nouzoe, and Shuzo Yajima. Size and Variable Ordering of OBDDs Representing Threshold Functions. In *International Conference on Computing and Combinatorics, COCOON*, pages 91–100, 1997.