Discrete Applied Mathematics 200 (2016) 1–4

Contents lists available at ScienceDirect

Discrete Applied Mathematics

journal homepage: www.elsevier.com/locate/dam

On the structure and the number of prime implicants of 2-CNFs

Navid Talebanfard

Department of Mathematical and Computing Sciences, Tokyo Institute of Technology, Meguro-ku Ookayama 2-12-1, 152-8552, Japan

ARTICLE INFO

Article history:
Received 5 February 2014
Received in revised form 16 June 2015
Accepted 18 June 2015
Available online 18 July 2015

Keywords:
Computational complexity
Combinatorial problems
Conjunctive normal form
Prime implicant

ABSTRACT

Let $m(n, k)$ be the maximum number of prime implicants that any k-CNF on n variables can have. We show that $3^{\frac{n}{3}} \leq m(n, 2) \leq (1 + o(1))3^{\frac{n}{3}}$.

1. Introduction

A *prime implicant* of a Boolean function f is a maximal subcube contained in $f^{-1}(1)$. It is known that among Boolean functions on n variables the maximum number of prime implicants is between $\Omega(\frac{3^n}{n})$ and $O(\frac{3^n}{\sqrt{n}})$ (see [2]). It is interesting to give finer bounds for restricted classes of functions. This problem has indeed been studied for DNFs with a bounded number of terms. It is known that a DNF with k terms has at most $2^k - 1$ prime implicants (see e.g. [7]).

In this note we consider the same problem for the class of k-CNF functions. Understanding the structure of the set of satisfying assignments of k-CNF formulas has been a crucial subject in computational complexity, in particular in developing k-SAT algorithms and bounded depth circuit lower bounds. Notable examples are the characterization of satisfying assignments of a 2-CNF which yields a polynomial time algorithm for 2-SAT (see e.g. [1]), and the Satisfiability Coding Lemma of [6] which bounds the number of isolated satisfying assignments of k-CNFs (these are the assignments such that if we flip the value of any single variable, the formula is not satisfied anymore). This was later improved in [5] to obtain the best known k-SAT algorithm and depth-3 circuit lower bounds for an explicit function.

To describe our main results we need a few definitions which follow shortly.

A *restriction* on a set X of variables is a mapping $\rho : X \to \{*, 0, 1\}$. We call a variable *free* if it is assigned $*$, and we call it *fixed* otherwise. We say that a restriction is *partial* if it leaves at least one variable free. For a function f over a set X of variables and a restriction ρ, we define f_ρ to be the subfunction obtained after setting values to the fixed variables according to ρ. An *implicant* of f is a restriction ρ such that f_ρ is the constant 1 function. We define a *prime implicant* of f to be an implicant ρ of f such that unspecifying any fixed variable does not yield the constant 1 function. A *partial prime implicant* is a prime implicant that leaves at least one variable free. To see that the concept of prime implicant generalizes that of isolated satisfying assignments, note that any isolated solution is in fact a prime implicant. Furthermore if ρ is a partial prime implicant, it is easy to see that if we remove all the variables in $\rho^{-1}(*)$ from the formula, then ρ restricted to $X \setminus \rho^{-1}(*)$ is in fact an isolated solution of this derived formula.

E-mail address: navid@is.titech.ac.jp.

http://dx.doi.org/10.1016/j.dam.2015.06.036
0166-218X/

The following lemma due to Paturi, Pudlák and Zane gives a bound on the number of isolated solutions of a k-CNF.

Lemma 1 (*The Satisfiability Coding Lemma [6]*). *Any k-CNF on n variables has at most $2^{(1-\frac{1}{k})n}$ isolated satisfying assignments.*

In an attempt to extend this result we define $m(n, k)$ to be the maximum number of prime implicants over k-CNF formulas on n variables. It is a natural question to give sharp bounds for $m(n, k)$. A similar problem was studied by Miltersen, Radhakrishnan and Wegener [3] which asks for the smallest size of a DNF equivalent to a given k-CNF. We first give a lower bound for $m(n, k)$.

Proposition 1. *There exists k_0 such that for all $n \geq k \geq k_0$, $m(n, k) \geq 3^{(1-O(\frac{\log k}{k}))n}$.*

Proof. We follow the construction of Chandra and Markowsky [2]. We divide the set of n variables in n/k parts, each of size k. On each of these parts, we represent the Chandra–Markowsky function as a k-CNF, that is the disjunction of all conjunctions of $2k/3$ variables, exactly $k/3$ of which are negated. We can do this since each such function depends only on k variables. Formula F would then be obtained by conjuncting all these functions together. In [2] it was shown that each block has at least $\Omega(3^k/k)$ prime implicants. It is easy to see that prime implicants of F are obtained by concatenating prime implicants of the blocks. Therefore the total number of prime implicants is at least $\Omega((3^k/k)^{n/k}) = 3^{(1-O(\log k/k))n}$. $\square$

For $k = 2$ we manage to give almost tight bounds. Note that in this case the above bound is not applicable as it is only valid as long as k is large.

Theorem 1. $3^{\frac{n}{3}} \leq m(n, 2) \leq (1 + o(1))3^{\frac{n}{3}}$.

2. Proof of Theorem 1

We first prove the lower bound. Let $n = 3m$ and consider the following formula on variable set $\{x_1, \ldots, x_m, y_1, \ldots, y_m, z_1, \ldots, z_m\}$ suggested to us by Dominik Scheder:

$$T(x, y, z) = \bigwedge_{i=1}^{m} (x_i \vee y_i) \wedge (y_i \vee z_i) \wedge (x_i \vee z_i).$$

It is easy to see that every prime implicant of T and every $1 \leq i \leq m$ must set exactly two variables among x_i, y_i and z_i to 1. Therefore $T(x, y, z)$ has $3^{\frac{n}{3}}$ prime implicants.

We now move on to the upper bound. Let F be any Boolean function on $\{x_1, \ldots, x_n\}$ and let ρ be a prime implicant that fixes all the variables. We claim that ρ is an *isolated* satisfying assignment for F, that is if we change the value of any single one of the variables, the formula evaluates to 0. To see this note that if changing the value of some variable x_i still satisfies F, we can simply unspecify x_i and get a smaller restriction which yields the constant 1 function, contradicting the minimality of ρ. When F is a 2-CNF we can apply Lemma 1 and bound the number of such prime implicants by $2^{\frac{n}{2}}$.

It thus remains only to bound the number of partial prime implicants. Assume without loss of generality that F contains no clauses with only one literal, since the value of such literal is forced. We need some terminology which we borrow from [1]. We define the *implication digraph* of F which we denote by $D(F)$ as follows. The vertex set consists of all literals $x_1, \bar{x}_1, \ldots, x_n, \bar{x}_n$. For every clause $x \vee y$ in F we put two directed edge $\bar{x} \to y$ and $\bar{y} \to x$ in $D(F)$. We say that a literal u *implies* a literal v, if there exists a directed path from u to v in $D(F)$. From here on we will assume without loss of generality that $D(F)$ is loopless. This is because a loop in $D(F)$ corresponds to a clause of the form $u \vee \bar{u}$ which is always true. It is easy to see that one can characterize the set of satisfying assignments of 2-CNFs in terms of their implication digraphs.

Proposition 2 (*[1]*). *An assignment α satisfies a 2-CNF F if and only if there is no edge in $D(F)$ going out of the set of true literals.*

We now give a similar characterization of partial prime implicants. For a restriction ρ we partition the set of literals into three sets A_ρ, B_ρ and C_ρ containing false literals, true literals, and those that are free, respectively. For any set S of vertices let $N^-(S)$ and $N^+(S)$ be the set of literals not in S implying some literal in S and the set of literals not in S implied by some literal in S, respectively. We first make the following observation.

Proposition 3. *Let F be a 2-CNF and let S be the set of all literals that appear in some directed cycle in $D(F)$. Then for any implicant ρ of F, we have $S \cap C_\rho = \emptyset$.*

Proof. Fix any directed cycle T in $D(F)$. In any satisfying assignment of F, all literals in T should be assigned the same value, since otherwise there would be a path connecting a true literal to a false one. Similarly, in an implicant of F, if a literal in T is free then in fact all literals in T must be free, since fixing the value of any such literal forces the value of all other literals in T. But this means that those clauses that contain variables only in T are completely untouched by ρ and hence not satisfied. This contradicts the assumption that ρ is an implicant. $\square$

Proposition 4. *Let F be a 2-CNF. A restriction ρ is an implicant of F if and only if*

1. *there is no edge into A_ρ*
2. *there is no edge out of B_ρ*
3. *C_ρ is an independent set.*