

On Minimum Representations of Matched Formulas (Extended Abstract) *

Ondřej Čepek and Štefan Gurský and Petr Kučera

Department of Theoretical Computer Science and Mathematical Logic,
Charles University in Prague, Faculty of Mathematics and Physics, Czech Republic
ondrej.cepek@mff.cuni.cz, stevko@mail.ru, kucerap@ktiml.mff.cuni.cz

Abstract

A Boolean formula in conjunctive normal form (CNF) is called matched if the system of sets of variables which appear in individual clauses has a system of distinct representatives. We present here two results for matched CNFs: The first result is a shorter and simpler proof of the fact that Boolean minimization remains complete for the second level of polynomial hierarchy even if the input is restricted to matched CNFs. The second result is structural — we show that if a Boolean function f admits a representation by a matched CNF then every clause minimum CNF representation of f is matched.

1 Introduction

In this paper we study the class of matched formulas introduced in [Franco and Van Gelder, 2003]. Given a CNF formula φ we consider its *incidence graph* $I(\varphi)$ defined as follows. $I(\varphi)$ is a bipartite graph with one part consisting of clauses of φ and the other part containing the variables of φ . An edge $\{C, x\}$ for a clause C and variable x is in $I(\varphi)$ if x appears in C . It was observed in [Aharoni and Linial, 1986; Tovey, 1984] that if $I(\varphi)$ admits a matching (i.e. a set of pairwise disjoint edges) of size m (where m is the number of clauses in φ), then φ is satisfiable. Later in [Franco and Van Gelder, 2003] the formulas satisfying this condition were called *matched formulas*. Since a matching of maximum size in a given graph can be found in polynomial time (see e.g. [Lovász and Plummer, 1986]), we can check whether a given formula is matched. Given an arbitrary CNF φ we can measure how far it is from being matched by considering its maximum deficiency $\delta^*(\varphi)$, the number of clauses which remain unmatched in a maximum matching of $I(\varphi)$. A formula φ is thus matched iff $\delta^*(\varphi) = 0$. A weaker notion of deficiency $\delta(\varphi) = m - n$ (where m is the number of clauses and n the number of variables in φ) is also often being considered.

Since their introduction, matched formulas were considered as a base class in parameterized algorithms for satisfiability (see e.g. [Flum and Grohe, 2006] for an overview

of parameterized algorithms theory). In particular, the authors of [Fleischner *et al.*, 2002] show that satisfiability of the formulas whose maximum deficiency is bounded by a constant can be decided in polynomial time. This result was later improved in [Szeider, 2003] to an algorithm for satisfiability parameterized with maximum deficiency of a formula. Parameterization based on backdoor sets with respect to matched formulas were considered in [Szeider, 2007].

Several generalizations of matched formulas were considered in the literature, too. In [Kullmann, 2000], matched formulas were generalized into the class of linearly satisfiable formulas. Autarkies based on matchings were studied in [Kullmann, 2003]. Another generalization was considered in [Szeider, 2005] as classes of bi-clique satisfiable and var-satisfiable formulas. Unfortunately, for both bi-clique and var-satisfiable formulas it is hard to check if a formula falls into one of these classes [Szeider, 2005].

The results listed in the previous paragraphs show that matched formulas play a significant role in the theory of satisfiability solving which is, without any doubt, one of the most studied problems in theoretical computer science that has many practical applications. Despite this fact, little is known about the structure of matched CNFs. It is not hard to come up with examples of matched CNFs such that logically equivalent prime CNFs are not matched. This is quite a surprising phenomenon which does not occur in most classes with polynomial time satisfiability testing such as quadratic CNFs, Horn CNFs, and their various generalizations, for which once a CNF is in the class, all logically equivalent prime CNFs are guaranteed to be in the class as well. This brings an interesting question: given a (nonprime) matched CNF does there exist at least one equivalent prime CNF which is also matched? In this paper we give an affirmative answer to this question, and then use this fact to prove a stronger result: if a Boolean function f admits a representation by a matched CNF then every clause minimum CNF representation of f is matched. This is the main result of the current paper.

Another problem we study in this paper is Boolean minimization of matched CNFs. Boolean minimization problem (BM) can be stated as follows: given a CNF find a logically equivalent CNF with a minimum possible number of clauses. This natural optimization version can be turned into a decision version by adding a number k and asking whether there exists a logically equivalent CNF with at most

*This paper is an updated extended abstract of an article in the Journal of Artificial Intelligence Research [Čepek *et al.*, 2014].

k clauses. The decision version of BM was shown to be Σ_2^P -complete [Umans, 2001] for general CNFs (where Σ_2^P denotes the second level of polynomial hierarchy). It is also long known that for some classes of CNFs where SAT is solvable in polynomial time, BM is Σ_1^P -complete (where Σ_1^P denotes the first level of polynomial hierarchy which is better known as class NP), i.e. easier than in the general case. Maybe the best known example is the class of Horn CNFs (a CNF is Horn if every clause in it contains at most one positive literal). NP-completeness of BM for Horn CNFs was proved independently in several papers [Ausiello *et al.*, 1986; Āeppek, 1995; Hammer and Kogan, 1993; Maier, 1980; Boros *et al.*, 2013]. For some classes of CNFs BM is even easier: there exists a hierarchy of tractable subclasses of Horn CNFs for which there are polynomial time minimization algorithms, namely acyclic and quasi-acyclic Horn CNFs [Hammer and Kogan, 1995], and CQ Horn CNFs [Boros *et al.*, 2009]. There are also few heuristic minimization algorithms for Horn CNFs [Boros *et al.*, 1998].

The complexity of BM for matched CNFs does not fit the above picture. Despite the fact that SAT is trivial for matched CNFs, BM for this class is Σ_2^P -complete, i.e. as hard as for the general case. This fact was proved in [Gurský, 2011], where the proof modifies the proof for the general case from [Umans, 2001]. In this paper we give a much simpler proof of the same fact which is based on an observation, that equivalence testing is co-NP-complete for matched CNFs.

2 Preliminaries

In this section we shall give the necessary definitions and recall several results we shall use in this paper.

2.1 Boolean Functions

An n -variable *Boolean function* is a mapping $f : \{0, 1\}^n \rightarrow \{0, 1\}$. A *literal* is either a variable or its negation. A *clause* is a disjunction of literals. Formula φ is in *conjunctive normal form (CNF)* if it is a conjunction of clauses (we also say that φ is a CNF formula). We shall often treat a clause as a set of its literals and a CNF formula as a set of its clauses. Thus $|\varphi|$ will denote the number of clauses in φ . If two CNF formulas φ_1 and φ_2 define the same function, we say that they are *equivalent* and we denote this fact with $\varphi_1 \equiv \varphi_2$. A CNF ψ is called *clause minimum* if for every CNF φ such that $\psi \equiv \varphi$ we have $|\psi| \leq |\varphi|$.

Clause C is called an *implicate* of f if every assignment $\vec{x} \in \{0, 1\}^n$ satisfying f (i.e. $f(\vec{x}) = 1$) also satisfies C (i.e. $C(\vec{x}) = 1$). We say that a clause C_1 *subsumes* a clause C_2 , if every literal from C_1 occurs also in C_2 (i.e. $C_1 \subseteq C_2$). C is a *prime implicate* of a function f if it is an implicate of f and there is no other implicate C' of f subsuming C (i.e. C is a set-minimal implicate of f). We say that CNF formula φ is *prime* if it contains only prime implicates. A CNF formula φ is *irredundant* if there is no sub-CNF $\varphi' \subset \varphi$ which represents the same function as φ .

An assignment t which assigns values to only a subset of (possibly to all) variables of a function f on n is called a *partial assignment*. Formally, partial assignment can be viewed as a mapping $t : Y \mapsto \{0, 1\}$ where Y is a subset of variables

of f . Given a CNF φ , $\varphi(t)$ denotes the CNF after applying a partial assignment t .

2.2 Resolution

We say that two clauses have a *conflict in variable x* if there is a positive occurrence of x in one clause and a negative occurrence in the other. Two clauses $C_1 = (D_1 \vee x)$ and $C_2 = (D_2 \vee \bar{x})$ are *resolvable over x* if D_1 and D_2 do not have a conflict in any variable. We write $R(C_1, C_2) = D_1 \vee D_2$ and this disjunction is called a *resolvent* of the *parent clauses* C_1 and C_2 .

2.3 Exclusive Sets of Implicates of a Boolean Function

In this section we follow [Boros *et al.*, 2010]. By $\mathcal{I}^p(f)$ we shall denote the set of all prime implicates of a function f and by $\mathcal{I}(f)$ the resolution closure of $\mathcal{I}^p(f)$.

Definition 2.1 ([Boros *et al.*, 2010]) Let f be a Boolean function and let $\mathcal{X} \subseteq \mathcal{I}(f)$ be a set of clauses. We shall say, that \mathcal{X} is an *exclusive set of clauses of f* if for every pair of resolvable clauses $C_1, C_2 \in \mathcal{I}(f)$ the following implication holds:

$$R(C_1, C_2) \in \mathcal{X} \implies C_1 \in \mathcal{X} \text{ and } C_2 \in \mathcal{X},$$

i.e. the resolvent is in \mathcal{X} only if both parent clauses are in \mathcal{X} .

Theorem 2.2 ([Boros *et al.*, 2010]) Let f be an arbitrary Boolean function, let $\mathcal{C}_1, \mathcal{C}_2 \subseteq \mathcal{I}(f)$ be two distinct sets of clauses which both represent f , and let $\mathcal{X} \subseteq \mathcal{I}(f)$ be an exclusive set of clauses. Then $\mathcal{C}_1 \cap \mathcal{X} \equiv \mathcal{C}_2 \cap \mathcal{X}$, i.e. both represent the same function.

Based on this proposition we define an exclusive component of a Boolean function.

Definition 2.3 ([Boros *et al.*, 2010]) Let f be an arbitrary Boolean function, $\mathcal{X} \subseteq \mathcal{I}(f)$ be an exclusive set of clauses of f , and $\mathcal{C} \subseteq \mathcal{I}(f)$ be a set of clauses which represents f . The Boolean function $f_{\mathcal{X}}$ represented by the set $\mathcal{C} \cap \mathcal{X}$ is called the *\mathcal{X} -component* of the function f . We shall simply call a function g an *exclusive component* of f , if $g = f_{\mathcal{X}}$ for some exclusive subset $\mathcal{X} \subseteq \mathcal{I}(f)$.

Theorem 2.2 guarantees that the \mathcal{X} -component $f_{\mathcal{X}}$ is well defined for every exclusive set $\mathcal{X} \subseteq \mathcal{I}(f)$. Theorem 2.2 has the following corollary.

Corollary 2.4 ([Boros *et al.*, 2010]) Let $\mathcal{C}_1, \mathcal{C}_2 \subseteq \mathcal{I}(f)$ be two distinct sets of clauses such that $\mathcal{C}_1 \equiv \mathcal{C}_2 \equiv f$, i.e. such that both sets represent f , and let $\mathcal{X} \subseteq \mathcal{I}(f)$ be an exclusive set of clauses. Then $(\mathcal{C}_1 \setminus \mathcal{X}) \cup (\mathcal{C}_2 \cap \mathcal{X})$ also represents f .

2.4 Autarkies

Autarky is a special type of partial assignment which satisfies each clause in which it substitutes a value for some literal.

Definition 2.5 Let ψ be a CNF on the set V of variables, let $Y \subseteq V$ be a subset of variables, let $L = \{x \mid x \in Y\} \cup \{\bar{x} \mid x \in Y\}$ be the corresponding set of literals, and let $t : Y \mapsto \{0, 1\}$ be a partial assignment on ψ . Then t is an *autarky* on ψ if for every clause $C \in \psi$ either $C \cap L = \emptyset$ or C is satisfied by t .

We shall prove two simple lemmas about autarkies which will be needed later in this paper. The first lemma is from [Kullmann, 2000] and the second one is proved in [Čepek *et al.*, 2014].

Lemma 2.6 *Let ψ be a CNF on the set V of variables, let $Y \subseteq V$ be a subset of variables, and let $t : Y \mapsto \{0, 1\}$ be an autarky on ψ . Then t is an autarky on $\mathcal{I}(f)$.*

Corollary 2.7 *Let ψ be a CNF on the set V of variables, let $Y \subseteq V$ be a subset of variables, and let $t : Y \mapsto \{0, 1\}$ be an autarky on ψ . Let $\varphi \subseteq \mathcal{I}(f)$ be an arbitrary representation of f . Then t is an autarky on φ .*

Let us mention that Corollary 2.7 does not hold without the assumption $\varphi \subseteq \mathcal{I}(f)$. Consider e.g. a CNF $\psi = (x \vee y) \wedge z$ and a CNF $\varphi = (x \vee y) \wedge (z \vee y) \wedge (z \vee \bar{y})$. It should be obvious that $\psi \equiv \varphi$, i.e. both CNFs represent the same function f , but $\varphi \not\subseteq \mathcal{I}(f)$. Now if t is an assignment which sets y to 1, t is autarky on ψ . On the other hand t is not autarky on φ because $(z \vee \bar{y})$ is not satisfied by t .

Lemma 2.8 *Let ψ be a CNF on the set V of variables, let $Y \subseteq V$ be a subset of variables, let $L = \{x \mid x \in Y\} \cup \{\bar{x} \mid x \in Y\}$ be the corresponding set of literals, and let $t : Y \mapsto \{0, 1\}$ be an autarky on ψ . Then $\psi(t)$ represents an exclusive component $f_{\mathcal{X}}$ of f defined by the exclusive set of clauses*

$$\mathcal{X} = \{C \in \mathcal{I}(f) \mid C \cap L = \emptyset\}.$$

2.5 Matched Formulas

Now we shall define the key concept of this paper. Given an undirected graph $G = (V, E)$, a subset of edges $M \subseteq E$ is a *matching* in G if the edges in M are pairwise disjoint. A *bipartite graph* $G = (A, B, E)$ is an undirected graph with disjoint sets of vertices A and B , and the set of edges E satisfying $E \subseteq A \times B$. For a set W of vertices of G , let $\Gamma(W)$ denote the *neighbourhood* of W in G , i.e. the set of all vertices adjacent to some element of W . Before giving the definition of matched CNFs let us state the following well-known result on matchings in bipartite graphs:

Theorem 2.9 (Hall’s Theorem [Hall, 1935; Lovász and Plummer, 1986]) *Let $G = (A, B, E)$ be a bipartite graph. A matching M of size $|M| = |A|$ exists if and only if for every subset S of A we have that $|S| \leq |\Gamma(S)|$.*

Definition 2.10 Let $\varphi = C_1 \wedge \dots \wedge C_m$ be a CNF on n variables $X = \{x_1, \dots, x_n\}$. We shall associate a bipartite graph $I(\varphi) = (\varphi, X, E)$ with φ , where the vertices correspond to clauses in φ and the variables in X . A clause C_i is connected to a variable x_j (i.e. $\{C_i, x_j\} \in E$) if C_i contains x_j or \bar{x}_j . A CNF φ is *matched* if $I(\varphi)$ has a matching of size m .

Note that a matching of maximum size in a given graph can be found in polynomial time (see e.g. [Lovász and Plummer, 1986]) and thus we can test in polynomial time whether a given CNF is matched. A variable which is matched to some clause in matching M is called *matched* in M , it is *free* in $M otherwise. Note, that a matched CNF is trivially satisfiable. The name “matched” was given to these formulas in [Franco and Van Gelder, 2003], although they appeared already in [Aharoni and Linial, 1986; Tovey, 1984].$

3 Equivalence Testing and Hardness of Clause Minimization of Matched Formulas

Following a definition from [Čepek *et al.*, 2012] a class of CNFs \mathcal{X} is called *tractable* if it satisfies the following four properties.

- *Recognition:* Given an arbitrary CNF φ it is decidable in poly-time with respect to $|\varphi|$ whether $\varphi \in \mathcal{X}$.
- *Satisfiability:* For every CNF $\varphi \in \mathcal{X}$ it is decidable in poly-time with respect to $|\varphi|$ whether φ is satisfiable.
- *Partial assignment:* Given an arbitrary CNF $\varphi \in \mathcal{X}$, if ψ is produced from φ by fixing some variables to 0 or 1 and substituting these values into φ , then $\psi \in \mathcal{X}$.
- *Prime representations:* Given an arbitrary CNF $\varphi \in \mathcal{X}$, if φ represents a function f then all prime CNF representations of f belong to \mathcal{X} .

It was shown in [Čepek *et al.*, 2012] that given two CNFs from a tractable class, it can be tested in polynomial time, whether these two CNFs are logically equivalent or not. The class of matched CNFs satisfies the first two tractability conditions, but fails to satisfy the remaining two. Constructing a counterexample to the third property is easy. The CNF

$$(x \vee y \vee z) \wedge (x \vee \bar{y} \vee z) \wedge (x \vee y \vee \bar{z})$$

is clearly matched, but a partial assignment $x \leftarrow 0$ gives

$$(y \vee z) \wedge (\bar{y} \vee z) \wedge (y \vee \bar{z})$$

that is not matched. We defer the counterexample to the fourth property to the next section. In the light of these findings it is an interesting question what is the complexity of equivalence testing for matched CNFs. Surprisingly, despite the fact that satisfiability is trivial for matched CNFs, equivalence testing is co-NP-complete.

MATCHED-EQ
Instance : Two matched CNFs φ and ψ
Question : Does $\varphi \equiv \psi$ hold?

Theorem 3.1 *MATCHED-EQ is co-NP-complete.*

Proof : A nondeterministic polynomial procedure checking that two CNFs are not equivalent guesses an assignment t and checks $\varphi(t) \neq \psi(t)$. Thus MATCHED-EQ is in co-NP.

To show co-NP-hardness we use reduction from the problem of checking that a given CNF α is unsatisfiable. Let α be an arbitrary CNF on n variables and m clauses, in particular let $\alpha = C_1 \wedge C_2 \wedge \dots \wedge C_m$. Let us define a clause $D = (a_1 \vee a_2 \vee \dots \vee a_m)$ on m new variables not occurring in α and let us define two CNFs:

$$\varphi = (C_1 \vee D) \wedge (C_2 \vee D) \wedge \dots \wedge (C_m \vee D) \text{ and } \psi = D$$

Both φ and ψ are matched, since each clause $C'_i = (C_i \vee D)$ can be matched to a variable a_i . Because $\varphi \equiv \alpha \vee D \equiv \alpha \vee \psi$ we have $\varphi \equiv \psi$ iff $\alpha \equiv \perp$, i.e. iff α is unsatisfiable. ■

The fact, that equivalence testing is co-NP-hard, is likely the main reason behind the fact proved in [Gurský, 2011]

that clause minimization of matched CNFs is Σ_2^p -complete. The proof in [Gurský, 2011] basically follows the proof of Σ_2^p -completeness of general Boolean minimization (BM) from [Umans, 2001] and is quite long and complicated. Here we present a much shorter and simpler proof based on a similar idea as the proof for the hardness of equivalence testing.

MATCHED-MIN

Instance : A matched CNF φ and an integer k

Question : \exists a CNF ψ such that $\psi \equiv \varphi$ and $|\varphi| \leq k$?

Theorem 3.2 MATCHED-MIN is Σ_2^p -complete.

Proof : Since MATCHED-MIN is a special case of BM which is known to be Σ_2^p -complete, MATCHED-MIN is in Σ_2^p . For the Σ_2^p -hardness we reduce the Σ_2^p -complete BM to it.

Let (α, k) be an instance of BM, $\alpha = C_1 \wedge \dots \wedge C_m$. Now let us repeat the construction from the proof of Theorem 3.1. Let $D = (a_1 \vee \dots \vee a_m)$ be a clause on m new variables not occurring in α , let $\varphi = (C_1 \vee D) \wedge \dots \wedge (C_m \vee D)$, and let (φ, k) be an instance of MATCHED-MIN.

Let (α, k) be a positive instance of BM. Then there exists CNF $\beta = D_1 \wedge D_2 \wedge \dots \wedge D_{k'}$ (with $k' \leq k$) that is equivalent to α . Let ψ be a CNF equivalent to $\beta \vee (a_1 \vee a_2 \vee \dots \vee a_m)$ that is let $\psi = D'_1 \wedge D'_2 \wedge \dots \wedge D'_{k'}$ where $D'_i = D_i \vee a_1 \vee a_2 \vee \dots \vee a_m$ for $1 \leq i \leq k'$. Clearly ψ is equivalent to φ and has at most k clauses. Therefore (φ, k) is a positive instance of MATCHED-MIN.

To see the other direction let (φ, k) be a positive instance of MATCHED-MIN and let ψ be a CNF equivalent to φ with at most k clauses. Let β be a CNF originating from ψ by a partial assignment that sets all a -variables to zero and sets no other variable. Since ψ is equivalent to φ and that is equivalent to $\alpha \vee (a_1 \vee a_2 \vee \dots \vee a_m)$, we have that β is equivalent to α . Clearly $|\beta| \leq |\psi| \leq k$ and since β is equivalent to α we conclude that (α, k) is a positive instance of BM. ■

4 Prime Representations of Matched Formulas

It is not difficult to see that unlike some well-behaved classes of CNFs (such as e.g. Horn CNFs or quadratic CNFs) for which all prime and irredundant CNFs lie inside the class, this is not the case for matched CNFs. Consider the CNF

$$(\bar{a} \vee b) \wedge (\bar{b} \vee c) \wedge (\bar{c} \vee a)$$

which is matched and a logically equivalent CNF

$$(\bar{a} \vee b) \wedge (\bar{b} \vee a) \wedge (\bar{c} \vee b) \wedge (\bar{b} \vee c)$$

which is not matched despite being prime and irredundant. Thus it is a legitimate question, whether given a (nonprime) matched CNF, there exists at least one logically equivalent prime and irredundant CNF which is also matched. The answer to this question is affirmative, however the proof requires a series of very technical lemmas. Due to the page limit we provide only the final statement here and refer the interested reader to [Čepek *et al.*, 2014].

Theorem 4.1 Let ψ be a matched CNF representing function f . Then there exists a matched CNF φ representing f such that φ is prime and irredundant.

5 Minimum Representations of Matched Formulas

In the previous section we have seen that for a matched CNF there may be some logically equivalent prime and irredundant CNFs which are not matched and some which are matched, the latter set being always non-empty. Here we shall show that a stronger statement holds for CNFs which are not only prime and irredundant but also clause minimum, namely that all such CNFs belong to the latter set.

Theorem 5.1 Let φ be a matched CNF representing function f on set of variables V and let ψ be a clause minimum CNF representation of f . Then ψ is a matched CNF.

Proof : Due to Theorem 4.1 we may assume that φ is prime and irredundant and thus $\varphi \subseteq \mathcal{I}(f)$. Let us assume by contradiction that ψ is not matched and that $\psi_X \subseteq \psi$ is a maximal (under inclusion) sub-CNF violating Hall's condition (such a subset must exist due to Theorem 2.9). Let us denote X the set of variables in the sub-CNF ψ_X , $Y = V \setminus X$ the set of remaining variables in ψ , and $\psi_Y = \psi \setminus \psi_X$ the remaining clauses in ψ (note that clauses in ψ_Y may contain variables not only from Y but also from X). Now the following holds:

- By the violation of Hall's condition $|\psi_X| > |X|$ holds.
- By the maximality of ψ_X there exists a matching M of all clauses in ψ_Y to variables in Y , i.e. ψ_Y is a matched CNF even if we drop all variables in X from its clauses. This follows from the fact that every subset of ψ_Y must satisfy Hall's condition even with respect to the variables in Y , since otherwise any such violating subset could be added to ψ_X contradicting its maximality.

The existence of matching M implies that ψ_Y can be satisfied using only variables from Y (each clause can be satisfied by its matched variable). So let $t : Y \mapsto \{0, 1\}$ be some partial assignment satisfying all clauses in ψ_Y (t is not necessarily unique). Clearly, t is an autarky on ψ as it satisfies every clause containing an assigned literal.

It follows from Lemma 2.8 that $\psi(t) = \psi_X$ represents an exclusive component f_X of f defined by the exclusive set $\mathcal{X} \subseteq \mathcal{I}(f)$, which contains all clauses consisting only of variables from X . Since φ also represents f and we assumed $\varphi \subseteq \mathcal{I}(f)$, it follows from Corollary 2.7 that t is an autarky also on φ . Thus, similarly as for $\psi(t)$ above, we can conclude that $\varphi(t)$ is a sub-CNF of φ which represents the exclusive component f_X of f , i.e. $\psi(t) \equiv \varphi(t)$. However, φ is matched, so every its sub-CNF (and in particular $\varphi(t)$) is matched, and thus $|\varphi(t)| \leq |X|$ while $|\psi(t)| = |\psi_X| > |X|$. But now, since both $\varphi(t)$ and $\psi(t) = \psi_X$ represent an exclusive component of f , also CNF $\psi' = (\psi \setminus \psi(t)) \cup \varphi(t)$ represents f by Corollary 2.4. However, we get $|\psi'| < |\psi|$ contradicting the assumed minimality of ψ . ■

Acknowledgments

The authors gratefully acknowledge a support by the Czech Science Foundation (grant P202/15-15511S).

References

- [Aharoni and Linial, 1986] Ron Aharoni and Nathan Linial. Minimal non-two-colorable hypergraphs and minimal unsatisfiable formulas. *Journal of Combinatorial Theory, Series A*, 43(2):196 – 204, 1986.
- [Ausiello *et al.*, 1986] G. Ausiello, A. D’Atri, and D. Sacca. Minimal representation of directed hypergraphs. *SIAM Journal on Computing*, 15(2):418–431, May 1986.
- [Boros *et al.*, 1998] E. Boros, O. Čepek, and A. Kogan. Horn minimization by iterative decomposition. *Annals of Mathematics and Artificial Intelligence*, 23:321 – 343, 1998.
- [Boros *et al.*, 2009] Endre Boros, Ondřej Čepek, Alexander Kogan, and Petr Kučera. A subclass of horn cnfs optimally compressible in polynomial time. *Annals of Mathematics and Artificial Intelligence*, 57:249–291, 2009.
- [Boros *et al.*, 2010] Endre Boros, Ondřej Čepek, Alexander Kogan, and Petr Kučera. Exclusive and essential sets of implicates of boolean functions. *Discrete Applied Mathematics*, 158(2):81 – 96, 2010.
- [Boros *et al.*, 2013] Endre Boros, Ondřej Čepek, and Petr Kučera. A decomposition method for cnf minimality proofs. *Theoretical Computer Science*, 510(0):111 – 126, 2013.
- [Čepek *et al.*, 2012] Ondřej Čepek, Petr Kučera, and Petr Savický. Boolean functions with a simple certificate for cnf complexity. *Discrete Applied Mathematics*, 160(4-5):365 – 382, March 2012.
- [Čepek *et al.*, 2014] Ondřej Čepek, Štefan Gurský, and Petr Kučera. On minimum representations of matched formulas. *Journal of Artificial Intelligence Research*, 51(0):707 – 723, 2014.
- [Čepek, 1995] Ondřej Čepek. *Structural Properties and Minimization of Horn Boolean Functions*. Ph.D. dissertation, Rutgers University, New Brunswick, NJ, October 1995, 1995.
- [Fleischner *et al.*, 2002] Herbert Fleischner, Oliver Kullmann, and Stefan Szeider. Polynomial-time recognition of minimal unsatisfiable formulas with fixed clause-variable difference. *Theoretical Computer Science*, 289(1):503 – 516, 2002.
- [Flum and Grohe, 2006] Jörg Flum and Martin Grohe. *Parameterized complexity theory*, volume 3. Springer, 2006.
- [Franco and Van Gelder, 2003] John Franco and Allen Van Gelder. A perspective on certain polynomial-time solvable classes of satisfiability. *Discrete Appl. Math.*, 125(2-3):177–214, 2003.
- [Gurský, 2011] Štefan Gurský. Minimization of matched formulas. In J. Šafránková and J. Pavlů, editors, *WDS’11 Proceedings of Contributed Papers: Part I – Mathematics and Computer Science*, pages 101–105, Prague, June 2011. Matfyzpress.
- [Hall, 1935] Philip Hall. On representatives of subsets. *Journal of The London Mathematical Society-second Series*, s1-10:26–30, 1935.
- [Hammer and Kogan, 1993] P.L. Hammer and A. Kogan. Optimal compression of propositional horn knowledge bases: Complexity and approximation. *Artificial Intelligence*, 64:131 – 145, 1993.
- [Hammer and Kogan, 1995] P.L. Hammer and A. Kogan. Quasi-acyclic propositional horn knowledge bases: Optimal compression. *IEEE Transactions on Knowledge and Data Engineering*, 7(5):751 – 762, 1995.
- [Kullmann, 2000] Oliver Kullmann. Investigations on autark assignments. *Discrete Applied Mathematics*, 107(1–3):99 – 137, 2000.
- [Kullmann, 2003] Oliver Kullmann. Lean clause-sets: generalizations of minimally unsatisfiable clause-sets. *Discrete Applied Mathematics*, 130(2):209 – 249, 2003.
- [Lovász and Plummer, 1986] László Lovász and Michael D. Plummer. *Matching Theory*. North-Holland, 1986.
- [Maier, 1980] D. Maier. Minimal covers in the relational database model. *Journal of the ACM*, 27:664 – 674, 1980.
- [Szeider, 2003] Stefan Szeider. Minimal unsatisfiable formulas with bounded clause-variable difference are fixed-parameter tractable. In Tandy Warnow and Binhai Zhu, editors, *Computing and Combinatorics*, volume 2697 of *Lecture Notes in Computer Science*, pages 548–558. Springer Berlin Heidelberg, 2003.
- [Szeider, 2005] Stefan Szeider. Generalizations of matched cnf formulas. *Annals of Mathematics and Artificial Intelligence*, 43(1-4):223–238, 2005.
- [Szeider, 2007] Stefan Szeider. Matched formulas and backdoor sets. In João Marques-Silva and Kareem A. Sakallah, editors, *Theory and Applications of Satisfiability Testing – SAT 2007*, volume 4501 of *Lecture Notes in Computer Science*, pages 94–99. Springer Berlin Heidelberg, 2007.
- [Tovey, 1984] Craig A. Tovey. A simplified np-complete satisfiability problem. *Discrete Applied Mathematics*, 8(1):85 – 89, 1984.
- [Umans, 2001] Christopher Umans. The minimum equivalent dnf problem and shortest implicants. *J. Comput. Syst. Sci.*, 63(4):597–611, 2001.