Algorithms for Variable-Weighted 2-SAT and Dual Problems

Stefan Porschen and Ewald Speckenmeyer

Institut für Informatik, Universität zu Köln, D-50969 Köln, Germany.
{porschen,esp}@informatik.uni-koeln.de

Abstract. In this paper we study NP-hard weighted satisfiability optimization problems for the class 2-CNF providing worst-case upper time bounds. Moreover we consider the monotone dual class consisting of clause sets where all variables occur at most twice. We show that weighted SAT, XSAT and NAESAT optimization problems for this class are polynomial time solvable using appropriate reductions to specific polynomial time solvable graph problems.

Key Words: weighted satisfiability, optimization problem, NP-hardness, edge cover, graph factor, perfect matching

1 Introduction

Weighted variants of search or decision problems are of certain importance for computational complexity theory as they can provide a gap from easy to hard. Consider, e.g., the satisfiability problem for propositional 2-CNF formulas (2-SAT). As is well known 2-SAT can be decided in linear time and in positive case even a model for an input formula C can be found in linear time [2]. But asking for a minimum cardinality model of C, i.e., a model of least number of variables assigned to true is an NP-hard optimization problem: 2-SAT can be regarded as a generalization of the minimum vertex cover problem in undirected graphs (see below). An immediate generalization of minimum cardinality 2-SAT is minimum weight 2-SAT, where the variables of the input formula are equipped with real-valued weights. Clearly, if each variable has weight 1, we obtain the minimum cardinality problem. As shall be seen below, also the problem of maximum weight 2-SAT is NP-hard.

Weighted versions of satisfiability problems have applications e.g. in code generation where certain problems can be encoded in weighted satisfiability [1].

In this paper we address the NP-hard optimization problems minimum and maximum weight SAT for arbitrarily variable-weighted 2-CNF formulas. We provide worst-case upper bounds of $O(2^{0.5284n})$ for these problems extending results presented in [17, 19] only holding for minimum weight 2-SAT for non-negative variable weights, respectively, maximum weight 2-SAT for non-positive weights. The latter results are based on techniques provided for *mixed Horn formulas* introduced and studied in [18]. The main variants of satisfiability, namely exact satisfiability and not-all-equal satisfiability for weighted 2-CNF have been shown to be linear time solvable in [12]. Moreover, the optimization cases of XSAT for variable weighted formulas has been shown to be solvable in time $O(2^{0.2441n})$ for arbitrary CNF [13, 15], respectively, in time $O(2^{0.16254n})$ restricted to 3-CNF [10, 11]. Exactly solving NAESAT for arbitrary CNF formulas (or restricted to 3-CNF) exactly in less than the trivial 2^n steps remains an open problem for the decision problem as well as for its NP-hard variable-weighted optimization variants.

Recently also counting versions of weighted SAT have been considered: #XSAT for variable weighted CNF formulas can be solved in time $O(n^2 \cdot \|C\| + 2^{0.40567 \cdot n})$ as shown in [14, 15]. Fürer et al. [5] provided an algorithm for counting all maximum weight solutions of SAT for variable weighted 2-CNF formulas. Clearly, only counting models cannot provide a solution of the underlying optimization problem, as no solutions are generated explicitly.

We also consider the class $\mathrm{CNF}_+(\leq 2)$, containing clauses of arbitrary length, but each variable occurs, regarding its polarity in at most two distinct clauses. We show that the variable-weighted optimization versions of satisfiability, exact satisfiability, and not-all-equal-satisfiability restricted to $\mathrm{CNF}_+(\leq 2)$ all are solvable in polynomial time via reductions to specific graph problems.

Organisation of the paper: Section 2 describes basic definitions and terminology used throughout, followed by explaining useful monotonization tools in Section 3. Section 4 discusses optimum weight 2-Satisfiability. In Section 5 we provide polynomial time algorithms for the weighted optimizations of satisfiability and its variants for the dual class of 2-CNF. In Section 6, we finish with some open problems and concluding remarks.

2 Preliminaries

To fix notation, a *literal* is a propositional variable $x \in \{0, 1\}$ or its negation $\overline{x} := \neg x$ (negated variable). The *complement* of a literal l is $\bar{l}$. A *clause* c is the disjunction of different literals and is represented as a literal set. A CNF formula C is a conjunction of different clauses and is represented as a clause set. Throughout we use the term *formula* meaning a clause set as defined. For a given formula C, clause c, by $V(C), V(c)$ we denote the set of variables contained in C, c, respectively. Similarly, given a literal l, $V(l)$ denotes the underlying variable. $V_+(C)$ (resp. $V_-(C)$) denotes the set of all variables occuring unnegated (resp. negated) in C. We distinguish between the length $\|C\|$ of a formula C and the number $|C|$ of its clauses. Let CNF denote the set of all formulas and let CNF_+ denote the set of *positive monotone* formulas, i.e., each clause contains only variables, no negated variables. Recall that the *intersection graph* of a monotone formula C has a vertex for each clause and an edge for each two intersecting clauses. For each $x \in V(C)$ let $C(x) := \{c \in C : x \in V(c)\}$. Let $\mathrm{CNF}(\leq k)$ be the set of formulas C such that each $x \in V(C)$ occurs in at most k clauses of C regardless whether negated or unnegated. Let k-CNF, k-CNF_+ denote the set of formulas C such that $|c| \leq k$ for all $c \in C$. A *Horn* formula is a member of CNF such that each clause has at most one positive literal.

For $X \subseteq V(C)$, we denote by C^X the formula obtained from C by complementing exactly those literals l in C with $V(l) \in X$, and we write $C^x := C^{\{x\}}$, respectively, $C^\gamma := C^{V(C)}$. Similarly, for a truth assignment t of C let t^X be obtained from t by complementing exactly the values $t(x)$ for all $x \in X$. Again we write $t^x := t^{\{x\}}$, and $t^\gamma := t^{V(C)}$. For $C \in$ 2-CNF, we denote by $P(C)$ its positive monotone part.

The satisfiability problem (SAT) asks in its *decision* version, whether there is a truth assignment $t : V(C) \to \{0, 1\}$ assigning one literal in each clause of C to 1, such a truth assignment is called a *model of* C. SAT is known to be NP-complete [3]. In the *search* version one has to decide whether $C \in$ SAT and in the positive case one has to find a model t of C.

*Exact satisfiability (*XSAT*)* means to find a truth assignment that assigns exactly one literal in each clause of a formula to 1, called *x-model* or XSAT-*model*. *Not-all-equal satisfiability* (NAESAT) searches for a truth assignment assigning at least one literal in each clause of C to 1 and at least one literal to 0, called *nae-model* or NAESAT-*model*. The decision versions of XSAT and NAESAT are defined analogously, and are known to be NP-complete [20].

An optimization variant of SAT is obtained when weights are assigned to the variables: Given $C \in$ CNF and $w : V(C) \to \mathbb{R}$, MINW-SAT asks whether $C \in$ SAT and in the positive case one has to find a *minimum model* of C, i.e., a model t of the least weight among all models of C. The weight of a model t is defined by $w(t) = \sum_{x \in t^{-1}(1)} w(x) = \sum_{x \in V(C)} w(x)t(x)$. Analogously, the optimization problems MINW-XSAT and MINW-NAESAT are defined, which all are NP-hard for the class CNF. Similarly we obtain the maximization versions MAXW-Π, when searching for a maximum Π-model, for $\Pi \in \{\text{SAT}, \text{XSAT}, \text{NAESAT}\}$, correspondingly.

Given $M \subseteq \mathbb{R}$, let an *M-weighted formula* be a pair (C, w) where $C \in$ CNF and $w : V(C) \to M$, and let $-M := \{-m | m \in M\}$. For weight function w, let $-w$ denote the weight function obtained from w by pointwise multiplying its values by -1. For $\Pi \in \{\text{SAT}, \text{XSAT}, \text{NAESAT}\}$, let $T^\Pi(C)$ denote the set of all Π-models of C and similarly let $T^\Pi_\mu(C, w) \subseteq T^\Pi(C)$ denote the set of all μ-weight Π-models of (C, w), with $\mu \in \{\min, \max\}$.

3 Reduction Tools for Variable-Weighted Formulas

Let us collect some useful tools for later considerations. The first assertion slightly generalizes Lemma 7 in [15] restricted to weighted XSAT, and enabling us to reduce maximum weight problems to minimum weight problems in specific cases:

Lemma 1. *Let $\Pi \in \{\text{SAT}, \text{XSAT}, \text{NAESAT}\}$. If there exists an algorithm $\mathcal{A}$ solving* MINW-Π *for M-weighted members of a formula class $\mathcal{C} \subseteq$* CNF *in $O(f(\|C\|))$ time then $\mathcal{A}$ also solves* MAXW-Π *for $(-M)$-weighted members of $\mathcal{C}$*

and vice versa. Moreover, in case $M = -M$, algorithm $\mathcal{A}$ can easily be modified to $\mathcal{A}'$ solving MAXW-Π *for M-weighted members of $\mathcal{C}$ in $O(f(\|C\|))$ time.*

Proof. Let $\mathcal{C} \subseteq$ CNF be a fixed formula class for which $\mathcal{A}$ is an algorithm solving MINW-Π for M-weighted input instances $C \in \mathcal{C}$. We claim that $T^{\Pi}_{\min}(C, w) = T^{\Pi}_{\max}(C, -w)$. From that claim the first assertion obviously follows. Moreover, if $M = -M$, given an M-weighted formula (C, w), let $\mathcal{A}'$ first compute $(C, -w)$ in linear time which then also is M-weighted, and then $\mathcal{A}'$ performs $\mathcal{A}$ on $(C, -w)$ finding an element $t \in T^{\Pi}_{\min}(C, -w)$, if existing, therefore $t \in T^{\Pi}_{\max}(C, w)$ as required. Since $\mathcal{A}$ at least must have linear running time, $\mathcal{A}'$ also has time bound $O(f(\|C\|))$. To verify the claim let $t \in T^{\Pi}_{\min}(C, w)$, and assume $t \notin T^{\Pi}_{\max}(C, w')$ where $w' := -w$. Then there exists $t_0 \in T^{\Pi}(C)$ with $w'(t_0) > w'(t)$ which is equivalent to $-w'(t_0) < -w'(t)$ meaning $w(t_0) < w(t)$ contradicting $t \in T^{\Pi}_{\min}(C, w)$. Therefore $T^{\Pi}_{\min}(C, w) \subseteq T^{\Pi}_{\max}(C, -w)$. Analogously, we obtain $T^{\Pi}_{\max}(C, -w) \subseteq T^{\Pi}_{\min}(C, w)$.

The vice versa assertion stating that an algorithm solving MAXW-Π for $(-M)$-weighted members in $\mathcal{C}$ also solves MINW-Π for M-weighted formulas analogously follows from the claim that $T^{\Pi}_{\max}(C, w) = T^{\Pi}_{\min}(C, -w)$ holds, for an arbitrary $(-M)$-weighted formula (C, w), $C \in \mathcal{C}$. This claim is shown as the previous one. □

Next, we state a basic proposition relating bijections between Π-model spaces to bijections between weighted Π-model spaces, where $\Pi \in \{\mathrm{SAT}, \mathrm{XSAT}, \mathrm{NAESAT}\}$. This result is a slight generalization of the same result restricted to weighted XSAT shown in [13, 15]:

Proposition 1. *For arbitrary M-weighted formulas $(C, w), (C', w')$ ($M \subseteq \mathbb{R}$), assume that there exists a bijection*

$$F : T^{\Pi}(C) \ni t \mapsto t' := F(t) \in T^{\Pi}(C')$$

such that $(*)$: $w(t) = w'(t') + \alpha$, where $\alpha \in \mathbb{R}$ is a constant independent of t and t'. Then the restricted mapping $F_\mu := F|T^{\Pi}_{\mu}(C, w)$ is a bijection between $T^{\Pi}_{\mu}(C, w)$ and $T^{\Pi}_{\mu}(C', w')$, so we have $|T^{\Pi}_{\mu}(C, w)| = |T^{\Pi}_{\mu}(C', w')|$, for $\mu \in \{\min, \max\}$.

Proof. First consider the minimization case. Let $t \in T^{\Pi}_{\min}(C, w)$ and assume that $t' := F_{\min}(t) \notin T^{\Pi}_{\min}(C', w')$. Then there is a Π-model $t'_0 \in T^{\Pi}(C')$ with $w'(t'_0) < w'(t')$. Let $t_0 := F^{-1}(t'_0)$ be the corresponding Π-model of C. Applying $(*)$ twice we obtain $w(t_0) = w'(t'_0) + \alpha < w'(t') + \alpha = w(t)$, contradicting the assumption that t is minimum. Hence $F_{\min}(t) \in T^{\Pi}_{\min}(C', w')$ holds for each $t \in T^{\Pi}_{\min}(C, w)$. Conversely, let $t' \in T^{\Pi}_{\min}(C', w')$ and assume $t := F^{-1}(t') \notin T^{\Pi}_{\min}(C, w)$. Then there is a Π-model $t_0 \in T^{\Pi}(C)$ with $w(t_0) < w(t)$. Let $t'_0 := F(t_0)$ be the corresponding Π-model of C'. As above, by $(*)$, we derive $w'(t'_0) = w(t_0) - \alpha < w(t) - \alpha = w'(t')$, contradicting the assumption that t' is minimum. Hence $F^{-1}(t') \in T^{\Pi}_{\min}(C, w)$ holds for each $t' \in T^{\Pi}_{\min}(C', w')$. Thus, F^{-1} restricted to $T^{\Pi}_{\min}(C', w')$ equals $F^{-1}_{\min}$ from which the assertion follows. Proving the assertion for maximum model spaces proceeds analogously. □

4 Optimum Weight 2-SAT

As is well known, 2-SAT, i.e., SAT restricted to 2-CNF can be decided and solved in linear time in the length of the formula [2]. However, a straightforward reduction from the minimum weight vertex cover problem (MINW-VC) in graphs tells us that the weighted version MINW-2SAT is NP-hard. To that end simply observe that the edges of the graph represent clauses of a monotone formula whose variables correspond to the graph vertices. Obviously a minimum weight model as defined above is equivalent to a minimum vertex cover, i.e., a smallest weight subset of vertices covering all graph edges. Therefore, MINW-VC and MINW-2SAT for monotone formulas are identical. In this section we provide an algorithm for optimum weight SAT restricted to arbitrarily weighted 2-CNF (also called *quadratic*) formulas.

For Horn formulas, SAT can be decided and solved in linear time which is a well-known result [8, 9]. The minimization problem for weighted Horn formulas can also be solved in linear time:

Lemma 2 ([17]). *Minimum weight satisfiability for a Horn formula H and weight function $w : V(H) \to \mathbb{R}_+$, can be solved in linear time.*

Observe that the maximization problem cannot be reduced to the minimization case as only non-negative weights are allowed.

If $P(C)$ is empty then C is a 2-CNF Horn formula, we thus obtain immediately:

Corollary 1. *For $C \in$ 2-*CNF *with $P(C) = \varnothing$,* MINW-XSAT *can be solved in linear time, where $w : V(C) \to \mathbb{R}_+$.* □

For non-negatively weighted 2-CNF formulas with n variables, MINW-SAT has been shown to be solvable in time $O(2^{0.5284 \cdot n})$ [17, 19] regarding it as a specific *mixed Horn formula* introduced in [18], which can be represented as the union of an arbitrary Horn formula, and an arbitrary quadratic formula:

Lemma 3 ([19]). *Minimum (resp. maximum) weight satisfiability can be solved in $O(2^{0.5284|V(C)|})$ time, for formulas $C \in$* 2-CNF *and $w : V(C) \to \mathbb{R}_+$ (resp. $w : V(C) \to \mathbb{R}_-$).*

The proof uses the Johnson-Papadimitriou-Yannakakis algorithm [7] for generating all maximal independent sets in the graph with polynomial delay yielding time bound $O(2^{0.5284n})$, for n vertices.

The last result can be generalized to arbitrarily weighted 2-CNF formulas resting on the next assertion stating that optimum weight Π-models are preserved in a certain sense when variables are complemented.

Lemma 4. *Let (C, w) with $C \in$* CNF*, $w : V(C) \to \mathbb{R}$, and $X \subseteq V(C)$ be arbitrary. Then, for (C^X, w^X) with $w^X(x) := w(x)$, $\forall x \in V(C) \setminus X$, and $w^X(x) := -w(x), \forall x \in X$, we have for each fixed $\Pi \in \{\mathrm{SAT}, \mathrm{XSAT}, \mathrm{NAESAT}\}$:*
(i) $|T^{\Pi}(C)| = |T^{\Pi}(C^X)|$, given by $t \mapsto t^X$,
(ii) $|T^{\Pi}_{\mu}(C, w)| = |T^{\Pi}_{\mu}(C_x, w_x)|$, $\mu \in \{\min, \max\}$.

Proof. Let $\mathit{\Pi} \in \{\mathrm{SAT}, \mathrm{XSAT}, \mathrm{NAESAT}\}$ then t obviously is a $\mathit{\Pi}$-model of C iff t^X is a $\mathit{\Pi}$-model of C^X, for each $X \subseteq V(C)$. Indeed, let ℓ be a literal at any fixed position p in C regarded as a vector of length $\|C\|$, then the truth value at position p is $t(\ell)$. Now either holds $V(\ell) \in X$ or $V(\ell) \notin X$. In the first case $\bar{\ell}$ is the corresponding literal at position p in C^X having truth value $t^X(\bar{\ell})) = 1 - t(\bar{\ell}) = t(\ell)$. In the remaining case the truth value of ℓ is not affected. Therefore the truth value vectors of C, t, resp. C^X, t^X, are identical completing the argumentation and implying that $F : T^{\mathit{\Pi}}(C, w) \ni t \mapsto t^X \in T^{\mathit{\Pi}}(C^X, w^X)$ is a bijection of $\mathit{\Pi}$-model spaces, hence (i) is true. Since $t^X(x) = 1 - t(x)$, for each $x \in X$ and $t^X(y) = t(y)$, for each $y \in V(C) - X$, we have

$$w^X(t^X) = \sum_{x \in X} (-w(x))(1 - t(x)) + \sum_{x \in V(C) - X} w(x)t(x) = -w(X) + w(t)$$

Hence $w^X(t^X) = w(t) + \alpha$, where $\alpha := -w(X)$ is a constant as X is fixed. So, assertion (ii) can immediately be derived from Prop. 1. □

The last observation helps us to solve minimum weight 2-SAT for arbitrarily weighted formulas:

Theorem 1. *Minimum (resp. maximum) weight satisfiability can be solved in* $O(2^{0.5284|V(C)|})$ *time, for formulas* $C \in$ 2-CNF *and* $w : V(C) \to \mathbb{R}$.

Proof. Let (C, w) with $C \in$ 2-CNF and $w : V(C) \to \mathbb{R}$ be an arbitrary input for MINW-2SAT. Define $X := \{x \in V(C) : w(x) < 0\} \subseteq V(C)$. Then compute a minimum weight model t of (C^X, w^X) due to Lemma 3 which is possible because $w^X : V(C^X) \to \mathbb{R}_+$. According to Lemma 4, t^X then is a minimum model of (C, w). So the assertion for the minimization case follows because C^X, w^X, and t^X can be computed in linear time $O(|V(C)|)$ using appropriate data structures. The maximization case follows due to Lemma 1. □

If (C, w) is an $\mathbb{R}$-weighted formula possessing a set $X \subseteq V(C)$ such that C^X is a Horn formula, and only the variables in X are negatively weighted, then by Lemma 1 we can solve minimum weight SAT for (C^X, w^X) in linear time: By the last theorem, we then also obtain a solution for the original input C with $w : V(C) \to \mathbb{R}$, because we only have to complement the model values for the variables in X.

It should be noted that, via the above approach, we cannot improve on Theorem 9 in [19], i.e., we cannot show that minimum (resp. maximum) weight satisfiability for each mixed Horn formula C with arbitrary weights can be solved in $O(2^{0.5284|V(C)|})$ time. The reason is that an arbitrary mixed Horn formula remains not mixed Horn if a certain subset of variables is complemented. But fortunately 2-CNF formulas remain stable under complementations of arbitrary variable sets.

XSAT resp. NAESAT are the same for 2-CNF formulas containing no unit clauses: A formula containing unit clauses obviously does not belong to NAESAT. However unit clauses are not critical as the corresponding literals must be set to true regardless of variable weights. The optimization versions minimum and maximum weight XSAT resp. NAESAT are solvable in linear time for 2-CNF, cf. Thm. 1, [12].

5 The Weighted Dual Class $\mathrm{CNF}_+(\leq 2)$

In this section MINW-Π, resp., MAXW-Π for the case of monotone weighted input formulas (C, w), i.e. $C \in \mathrm{CNF}_+(\leq 2)$, $w : V(C) \to \mathbb{R}$, is treated, for each fixed $\Pi = \{\mathrm{XSAT}, \mathrm{SAT}, \mathrm{NAESAT}\}$. Recall that, by definition, each variable occurs in at most two distinct clauses of C, hence $|C(x)| \leq 2$, for each $x \in V(C)$. Observe that $\mathrm{CNF}(\leq 2)$ can be regarded as dual to 2-CNF in the sense that assigning a set S_x to each variable $x \in V(C)$ defined by $S_x := \{c \in C : x \in V(c)\}$ yields "variable-clauses" of length at most 2: $|S_x| \leq 2$.
The algorithmic strategy is as follows focusing first on MINW-Π, for each fixed $\Pi \in \{\mathrm{SAT}, \mathrm{XSAT}, \mathrm{NAESAT}\}$: We reduce MINW-$\Pi$ in polynomial time to a corresponding equivalent problem on an edge weighted graph that is closely related to the (edge-weighted) intersection graph G_C of (C, w). Then we construct a polynomial time algorithm solving this graph problem, thereby yielding a minimum Π-model of (C, w) or responding that none such exists. An algorithm for the maximization version then is provided by Lemma 1.
Recall that the intersection graph of a monotone formula $C \in \mathrm{CNF}_+$ has a vertex for each clause, and two vertices are joined by an edge iff the corresponding clauses have non-empty intersection. We will make use of a modification of the intersection graph that we call the *clause graph* G_C associated to a variable-weighted $C \in \mathrm{CNF}_+(\leq 2)$ also incorporating variables that uniquely occur in C, i.e., in only one clause: Thus, if C admits no unique variables the clause graph simply is the intersection graph, such that each edge is labeled by a variable of least weight in the intersection and is weighted by that weight. In case C admits unique variables, make a copy of its intersection graph, pose to each edge in either copy its label and its weight as mentioned above. Finally, join each two vertices in either copy that correspond to the same clause by an edge iff the clause contains a unique variable; label that edge with a unique variable of least weight, and assign to it that weight value yielding G_C. The formula graph at most has $2|C|$ vertices and $|V(C)|$ edges, and can obviously be built in $O(|C^2| \cdot |V(C)|)$ time. Since each variable occurs in at most two clauses, i.e. $\forall x \in V(C) : |C(x)| \leq 2$, we have $|C| \leq \|C\| = \sum_{x \in V(C)} |C(x)| \leq 2|V(C)|$. If G_C is not connected we first compute its components in linear time then treat all components independently decreasing running times obviously.
For the case $\Pi = \mathrm{XSAT}$, we have:

Lemma 5 ([12]). *A minimum, resp. maximum,* XSAT*-model of* (C, w) *with* $C \in \mathrm{CNF}_+(\leq 2)$*,* $w : V(C) \to \mathbb{R}$*, can be computed, respectively, it can be reported that none exists in time* $O(|V(C)|^3)$*.*

The proof is based on the fact that a minimum weight perfect matching in the clause graph is equivalent to a minimum weight XSAT-model of (C, w), if exactly the variables that label the selected matching edges are set to true. The running time is determined by the matching algorithm. We even have:

Theorem 2. *A minimum, respectively, maximum weight* XSAT*-model of* (C, w)*, with* $C \in \mathrm{CNF}(\leq 2)$*,* $w : V(C) \to \mathbb{R}$*, can be computed, respectively, it can be reported that none exists in* $O(|V(C)|^3)$ *time.* □

The proof is based on a polynomial time reduction transforming (C, w) to a weighted monotone formula (C', w') such that the induced mapping $F^{\mathrm{XSAT}}_{\min} : T^{\mathrm{XSAT}}_{\min}(C, w) \to T^{\mathrm{XSAT}}_{\min}(C', w')$ on the minimum XSAT-model spaces is a bijection. Then a minimum XSAT-solution t'_0 of (C', w') can be transformed into a minimum XSAT-solution of (C, w) via $t_0 := F^{\mathrm{XSAT}^{-1}}_{\min}(t'_0)$. Hence, we arrive at a polynomial time algorithm for MINW-XSAT on arbitrarily variable-weighted members in CNF(≤ 2). From the latter algorithm we obtain a polynomial time algorithm solving MAXW-XSAT according to Lemma 1.
The next results provide elementary transformation steps for eliminating pure negative literals, complemented pairs in clauses, and negative literals that have its positive complement in a different clause. Regarding pure literals we are done by Lemma 4, for $X = \{x\}$:

Corollary 2. *For $\Pi \in \{\mathrm{SAT}, \mathrm{XSAT}, \mathrm{NAESAT}\}$, and (C, w), with $C \in \mathrm{CNF}$, $w : V(C) \to \mathbb{R}$, let $x \in V(C)$ be a variable only occuring negated in C. Then for (C^x, w^x), where $w^x : V(C) \to \mathbb{R}$ be defined as w except for $w^x(x) := -w(x)$ we have $|T^{\Pi}_{\mu}(C, w)| = |T^{\Pi}_{\mu}(C_x, w_x)|$, $\mu \in \{\min, \max\}$.*

Lemma 6 ([13, 15]). *For $C \in \mathrm{CNF}$ with $w : V(C) \to \mathbb{R}$, holds $T^{\mathrm{XSAT}} = \varnothing$ if there is $c \in C$ containing more than one complemented pairs. Let $c \in C$ contain exactly one complemented pair $x, \overline{x}$, and let C_c be the formula obtained from C by removing c and assigning all literals to 0 that occur in $c' := c - \{x, \overline{x}\}$ (which can be empty), and by finally removing all duplicate clauses. Let w_c be the restriction of w to $V(C_c) = V(C) - V(c')$, then there is a bijection providing $|T^{\mathrm{XSAT}}_{\mu}(C, w)| = |T^{\mathrm{XSAT}}_{\mu}(C_c, w_c)|$, $\mu \in \{\min, \max\}$. Moreover the transformation (C, w) to (C_c, w_c) as well as the* XSAT*-model space bijection can be computed in polynomial time.* □

Lemma 7 ([13, 15]). *Let $C \in \mathrm{CNF}$, $w : V(C) \to \mathbb{R}$ such that no clause contains a complemented pair. Let $c_i = \{x\} \cup u, c_j = \{\overline{x}\} \cup v \in C$ where $x \in V(C)$ and u, v are literal sets. Let C_{ij} be obtained from C as follows:*
(1) $C_{ij} := C - \{c_i, c_j\} \cup \{u \cup v\}$
(2) set all literals in $u \cap v$ to 0, then remove all duplicate clauses from the resulting formula.
Let $w_{ij} := V(C_{ij}) \to \mathbb{R}$ be the following weight function: For each $y \in V(C_{ij}) - V(u \oplus v)$, set $w_{ij}(y) := w(y)$, and moreover, only in case that $u \oplus v \neq \varnothing$, define:
(1') if $V_+(u \oplus v) \cap V_-(u \oplus v) = \{z\}$, then set

$$w_{ij}(y) := \begin{cases} \quad w(y) \quad , & \textit{if } y \in V(u \oplus v) - \{z\} \\ w(z) + w(x), & \textit{if } z = y \textit{ and } \overline{z} \in u, z \in v \\ w(z) - w(x), & \textit{if } z = y \textit{ and } z \in u, \overline{z} \in v \end{cases}$$

(2') if $V_+(u \oplus v) \cap V_-(u \oplus v) = \varnothing$, then set

$$w_{ij}(y) := \begin{cases} \quad w(y) \quad , & \textit{if } y \in V(v - u) \\ w(y) + w(x), & \textit{if } y \in V_-(u - v) \\ w(y) - w(x), & \textit{if } y \in V_+(u - v) \end{cases}$$

Then we have:
(i) $V(C_{ij}) = V(C) - [\{x\} \cup V(u \cap v)]$, $C_{ij} \in \mathrm{CNF}(\leq 2)$*, and* $|C_{ij}| \leq |C| - 1$,
(ii) $|T_{\mu}^{\mathrm{XSAT}}(C, w)| = |T_{\mu}^{\mathrm{XSAT}}(C_{ij}, w_{ij})|$*, for* $\mu \in \{\min, \max\}$.
Moreover the transformation (C, w) *to* (C_{ij}, w_{ij}) *as well as the* XSAT*-model space bijection can be computed in polynomial time.* □

Next, consider $\Pi = \mathrm{SAT}$: Since C is monotone we can set to true each variable that is weighted non-positively, and remove all clauses that are satisfied thereby. All variables that are removed from the formula and are not yet set have strictly positive weights and appear only in clauses already satisfied, therefore we can set them to false. Moreover if the remaining formula has clauses containing unique variables only, remove these clauses and set to true exactly one variable of least weight and to false all other variables. All steps above can obviously be done in linear time $O(\|C\|)$. It remains a monotone formula (C', w') such that each variable has strictly positive weight and occurs in at most two clauses, no clause exclusively has unique variables. Therefore the clause graph contains no isolated vertices.

Recall that an *edge cover* in a graph $G = (V, E)$ of no isolated vertices is a subset $F \subseteq E$ such that each vertex $x \in V$ is incident to at least one edge in F. It is not hard to see that a minimum weight edge cover F in the clause graph $G_{C'}$ yields a minimum weight SAT-model of (C', w') by setting exactly those variables to true that label the edges in F.

It is well known that the minimum cardinality of an edge cover in G is closely related to the matching number $\nu(G)$ of G, i.e., the cardinality of a maximum matching according to the relation $|F| = |V| - \nu(G)$. This is easy to see: all $2\nu(G)$ vertices in a maximum matching are covered already, all remaining vertices are independent from each other and only have neighbours in the covered set. So, for each remaining vertex we have to choose exactly one edge not contained in the matching.

There is no immediate connection as above between maximum weight matchings and minimum weight edge covers. However, let (G, w) be a connected edge weighted graph with $w : E \to \mathbb{R}_+ - \{0\}$. Transforming the edge weights $w(e)$ to $\hat{w}(e)$, for each $e = x - y \in E$, according to

$$(*)\ \hat{w}(x-y) := -w(x-y) + \min\{w(x-z) | z \in N(x)\} + \min\{w(y-z) | z \in N(y)\}$$

where $N(v) \subset V$ denotes the set of all neighbours of $v \in V$ in G, yields weight function $\hat{w} : E \to \mathbb{R}$, cf. e.g. [4]. Now, perform a general maximum weight matching algorithm, for arbitrarily edge-weighted graphs, on $(G, \hat{w})$; and let $M \subseteq E$ such that $\hat{w}(M) = \max\{\hat{w}(M') | M' \subset E \text{ matching in } G\}$, be its result. Let $V(M) \subset V$ denote that part of the vertices in G incident to an edge in M. Then we claim (i): for each $x \in V - V(M)$ there exists an edge $x - y$ with $y \in V(M)$, and (ii): selecting one such edge of least weight for each $x \in V - V(M)$ collected in M' provides a minimum weight edge cover $M \cup M'$ in (G, w). Observe that $\hat{w} = w$ in case w is a constant function, and by the procedure described we obtain a minimum cardinality edge cover as explained above. To verify the first claim, let $x \in V - V(M)$ be such that all its neighbours are members in $V - V(M)$

and let $E(x)$ denote the set of corresponding edges. Let $e = x - y \in E(x)$ such that $w(e) = \min\{w(e')|e' \in E(x)\}$. Therefore, with $(*)$

$$\hat{w}(e) := -w(e) + w(e) + \min\{w(y - z)|z \in N(y)\} > 0$$

because at least $x \in N(y)$. Hence, e would enlarge $w(M)$ and must be contained in M yielding a contradiction.
For (ii), clearly an edge $e = x - y$ is included into M only if $\hat{w}(e) \geq 0$, hence in view of $(*)$ e is the edge of least weight covering x and y, as all w-values are strictly positive. Similary, for each $x \in V - V(M)$, by (i) there is an edge $x - y$ of least weight $w(x - y)$ and $y \in V(M)$. It remains to show that such an edge is a cover of least weight for x if there is also $e' = x - z$ with $y \in V - V(M)$ of least w-weight among all such egdes. Indeed, because e' is independent of M we have $\hat{w}(e') \leq 0$ therefore, by $(*)$, follows $w(e') > 0$. Thus, we have verified that $M \cup M'$ provides an edge of least weight for each $x \in V$ and has least possible cardinality, therefore it is a minimum weight edge cover in (G, w).
Observe that transformation $w \to \hat{w}$ can be carried out in time $O(|V| \cdot |E|)$. Moreover, it is well known that a maximum weight matching in G can be computed relying on Edmonds blossom algorithm for perfect weighted matchings in $O(|V^2| \cdot |E|)$ time [4]. Hence, a minimum weight edge cover in (G, w) of strictly positive edge weights can be computed in $O(|V^2| \cdot |E|)$ time. Recalling that the clause graph has $O(|C|)$ vertices, $O(|V(C)|)$ edges, and $|C| \leq 2|V(C)|$, we obtain in summary:

Theorem 3. *A minimum, resp. maximum weight* SAT *model of* (C, w)*,* $C \in \mathrm{CNF}_+(\leq 2)$*,* $w : V(C) \to \mathbb{R}$*, can be computed in time* $O(|V(C)|^3)$*.* □

Finally, we consider NAESAT, which for the unweighted case has been solved in [16] based on Euler tour techniques. Unfortunately, in the weighted case this approach does not apply.
Let $G = (V, E)$ be a connected graph, and for fixed $F \subseteq E$ let $F(x)$ denote the set of edges incident to $x \in V$. Now let $f, g : V \to \mathbb{Z}$ be two functions such that $f \leq g$. Recall that a *f-factor* in G is a set $M \subseteq E$ such that for each $x \in V$ holds $|M(x)| = f(x)$, which in general does not exist. E.g., for $f = 1$, an f-factor is a perfect matching. More generally, an *$[f, g]$-factor* in G is $M \subseteq E$ with $|M(x)| \in [f(x), g(x)]$ for each $x \in V$. For an edge-weighted graph (G, w) with $w : E \to \mathbb{R}$ an *optimum weight* $[f, g]$-factor, is an $[f, g]$-factor M of optimal weight $w(M)$.
Let (C, w) with $C \in \mathrm{CNF}_+(\leq 2)$ containing no unit clauses (otherwise the formula a priori admits no NAESAT-models) and $w : V(C) \to \mathbb{R}$. Each $c \in C$ containing exclusively unique clauses can be minimally NAESAT-satisfied independently: If c has only variables of non-negative weights, then set exactly one of the smallest weight to 1 and all other variables to 0. If c has only variables of non-positive weights, then set exactly one of the greatest weight to 0 and all other variables to 1. Observe that also the case is included where all variables have weight 0, and we have the convention that the variable with the smallest (largest) index then, by definition, is that of smallest (largest) weight. In all

remaining cases c contains at least one strictly positive-weighted and at least one strictly negative-weighted variable, so we set all the latter to 1 and the remaining variables to 0. It is obvious that so we have minimal NAESAT-satisfied clauses containing unique variables only which hence can be omitted from the formula. Next consider a clause c in the remaining formula that contains more than one unique variable collected in $U \subset V(c)$. We intend to assign truth values to all except one of these unique variables such that the remaining fragment of the clause is indepent w.r.t. minimum weight NAESAT: If all variables in U have non-negative weights, set to 0 all except for exactly one of the smallest weight, which will not yet be assigned. If all variables in U have non-positive weights, set to 1 all except for exactly one of the greatest weight, which will not yet be assigned. In all remaining cases c has at least one variable of strictly positive weight and at least one of strictly negative weight. Set all the latter variables to 1, and set to 0 all of the remaining (non-negative) variables except for exactly one of the smallest weight, which will not yet be assigned. From each such clause c remains a fragment c' containing only one unique variable.
Observe that the resulting formula (C', w'), where $w := w'|V(C')$, yields an edge-weighted clause graph $G_{C'}$ such that each vertex has at least degree 2. Moreover, each variable in (C', w') labels a unique edge in the clause graph, and vice versa. Finally, the earlier eliminated unique variables had, for each clause containing them, been set appropriately. Now it easily follows that a minimum weight NAESAT-model of (C', w') is provided by setting to 1 exactly the variables labeling the edges in a minimum weight $[1, \deg]$-factor, if existing, in $G_{C'}$. Here $\deg : V(G_{C'}) \to \mathbb{Z}$ denotes the degree function, i.e., $\deg(x) = |N(x)|$, for each $x \in V(G_{C'})$.
With standard linear programming techniques one can solve a related problem in polynomial time, namely the maximum weight $[f, g]$-matching problem [4]: Given an edge-weighted graph $(G = (V, E), w)$, and f, g as above, one searches for $\mu : E \to \mathbb{Z}$ such that for each $x \in V$ holds $\sum_{y \in N(x)} \mu(x - y) \in [f(x), g(x)]$ and $\sum_{e \in E} \mu(e)w(e)$ is maximal. The maximum weight $[f, g]$-factor problem obviously gets the maximum weight $[f, g]$-factor problem if, one poses the further constraint $0 \leq \mu(e) \leq 1$ for each $e \in E$, then $\mu(E)$ is the characteristic vector of a matching. Hence the latter problem also can be solved in polynomial time. The minimization version can be derived easily from the maximization version in the same manner as described in Lemma 1. So we arrive at:

Theorem 4. *A minimum, resp. maximum,* NAESAT*-model of* (C, w) *with* $C \in \mathrm{CNF}_+(\leq 2)$*,* $w : V(C) \to \mathbb{R}$*, can be computed, respectively, it can be reported that none exists in polynomial time.* □

6 Concluding Remarks and Open Problems

We proved that minimum (and also maximum) weight SAT for 2-CNF formulas of n arbitrarily weighted variables can be solved in time $O(2^{0.5284n})$. So an open problem is to construct a faster algorithm for optimum weight 2-SAT. Clearly, for monotone formulas minimum weight 2-SAT is the same as minimum weight vertex cover. Thus the question arise whether one can provide an appropriate polynomial time monotonization scheme also reducing general minimum weight 2-SAT to minimum vertex cover such that the formula does not increase. However, the monotonization methods discussed in Section 5 unfortunately do not apply, because simple resolution fails.

Regarding the second part, we leave open the question whether there can be constructed monotonization schemes, as valid for XSAT [13, 15], solving the optimization versions of SAT and NAESAT even for arbitrary, i.e., not necessarily monotone members of CNF(≤ 2). What is missing, is a weighted version of the simple resolution rule for the SAT and NAESAT cases (holding in the unweighted case [16]).

Indeed only simple resolution is missing, because pure literal elimination already is provided by Lemma 2. And regarding complemented pairs in clauses we have:

Proposition 2. *For $\Pi \in \{\mathrm{SAT}, \mathrm{NAESAT}\}$ and $C \in \mathrm{CNF}$, $w : V(C) \to \mathbb{R}$, assume there is $c \in C$ containing complemented pairs, where $W := V(C) - V(C - \{c\})$, i.e., the set of variables only occuring in c which may be empty. Let $\hat{C}$ be the relevant part of C obtained from C by setting to 0 all $x \in W$ with $w(x) = 0$, let $\hat{w}$ be the restriction of w to $V(\hat{C})$. Let $\hat{C}_c := \hat{C} - \{c\}$ and let $\hat{w}_c$ be the restriction of $\hat{w}$ to $V(\hat{C}_c)$, then we have $T^{\Pi}_{\mu}(\hat{C}, \hat{w}) \neq \varnothing$ iff $T^{\Pi}_{\mu}(C, w) \neq \varnothing$, and $T^{\Pi}_{\mu}(\hat{C}, \hat{w}) \subseteq T^{\Pi}_{\mu}(C, w)$; moreover, there is a bijection providing $|\hat{T}^{\Pi}_{\mu}(\hat{C}, \hat{w})| = |T^{\Pi}_{\mu}(\hat{C}_c, \hat{w}_c)|$, $\mu \in \{\min, \max\}$. Finally, the transformation (C, w) to $(\hat{C}_c, \hat{w}_c)$ as well as the Π-model space bijection can be computed in polynomial time.*

Proof. We distinguish two cases. (1): $W = \varnothing$, i.e., each variable in $V(c)$ also occurs in C_c, then $C = \hat{C}$, $w = \hat{w}$, and obviously each $t \in T^{\Pi}(C)$ also is a member of $T^{\Pi}(C_c)$ and vice versa, for $\Pi \in \{\mathrm{SAT}, \mathrm{NAESAT}\}$ providing a bijection (namely the identity) between the Π model spaces. As $w = w_c$, in that case, we also have $(*)$ of Prop. 1; so we are done.

(2): $W \neq \varnothing$. Since c contains at least one complemented pair $x, \bar{x}$, c is Π-satisfied by any truth assignment. Therefore, and because the variables in W do not occur in $C - \{c\}$, we can set to 0 all $x \in W$ with $w(x) = 0$ implying $T^{\Pi}_{\mu}(\hat{C}, \hat{w}) \neq \varnothing$ iff $T^{\Pi}_{\mu}(C, w) \neq \varnothing$, and $T^{\Pi}_{\mu}(\hat{C}, \hat{w}) \subseteq T^{\Pi}_{\mu}(C, w)$. Thus it suffices to consider $(\hat{C}, \hat{w})$ in the following. If $\hat{W}$ denotes the set of all remaining variables in W we have $V(\hat{C}_c) = V(\hat{C}) - \hat{W}$, and claim that

$$F^{\Pi} : T^{\Pi}_{\min}(\hat{C}, \hat{w}) \ni t \mapsto F^{\Pi}(t) := t|V(\hat{C}_c) \in T^{\Pi}_{\min}(\hat{C}_c, \hat{w}_c)$$

is a bijection if the reverse is defined as the extension of $t' \in T^{\Pi}_{\mu}(C_c, w_c)$ to $V(\hat{C})$ by assigning to 1 exactly all variables in $\hat{W}$ with $w(x) < 0$ and the others to 1, hence $\hat{w}(\hat{W}) = w(W)$ is minimal. It is easy to see that F^{Π} is one-to-one, and indeed is a bijection of minimum Π-model spaces. The maximization case proceeds analogously. The running time assertions are obvious. □

References

1. A. V. Aho, M. Ganapathi, and S. W. Tjiang, Code Generation Using Tree Matching and Dynamic Programming, ACM Trans. Programming Languages and Systems, 11 (1989) 491-516.
2. B. Aspvall, M. R. Plass, and R. E. Tarjan, A linear-time algorithm for testing the truth of certain quantified Boolean formulas, Inform. Process. Lett. 8 (1979) 121-123.
3. S. Cook, The Complexity of Theorem Proving Procedures, in: "Proceedings of the Third ACM Symposium on Theory of Computing," pp. 151-158, 1971.
4. W. J. Cook, W. H. Cunningham, W. R. Pulleyblank, and A. Schrijver, Combinatorial Optimization, Wiley, New York, 1998. pp. 151-158, 1971.
5. M. Fürer, and S. P. Kasiviswanathan, Algorithms for Counting 2-SAT Solutions and Colorings with Applications, ECCC Report No. 33, 2005.
6. M. R. Garey, and D. S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness, W. H. Freeman and Company, San Francisco, 1979.
7. D. S. Johnson, M. Yannakakis, and C. H. Papadimitriou, On Generating All Maximal Independent Sets, Inform. Process. Lett. 27 (1988) 119-123.
8. H. R. Lewis, Renaming a Set of Clauses as a Horn Set, J. ACM 25 (1978) 134-135.
9. M. Minoux, LTUR: A Simplified Linear-Time Unit Resolution Algorithm for Horn Formulae and Computer Implementation, Inform. Process. Lett. 29 (1988) 1-12.
10. G. Plagge, Über Variablen-Gewichtete X3SAT Optimierungs-Probleme, Diploma Thesis, Univ. Köln, 2006.
11. G. Plagge, and S. Porschen, Solving optimum variable-weight Exact 3-Satisfiability in $O(2^{0.16254n})$ time, Techn. Report, Univ. Köln, 2006, in preparation.
12. S. Porschen, On Some Weighted Satisfiability and Graph Problems, in: "P. Vojtas, et al. (Eds.), Proceedings of the 31st Conference on Current Trends in Theory and Practice of Informatics", Lecture Notes in Computer Science, Vol. 3381, pp. 278-287, Springer-Verlag, Berlin, 2005.
13. S. Porschen, Solving Minimum Weight Exact Satisfiability in Time $O(2^{0.2441n})$, in: X. Deng, D. Du (Eds.), Proceedings ISAAC 2005, Lecture Notes in Computer Science, Vol. 3827, pp. 654-664, Springer-Verlag, Berlin, 2005.
14. S. Porschen, Counting All Solutions of Minimum Weight Exact Satisfiability, in: "T. Calamoneri, I. Finocchi, G. F. Italiano (Eds.), Proceedings of the 6th International Conference on Algorithms and Complexity (CIAC 2006), Rome, Italy", Lect. Notes in Comp. Science, Vol. 3998, pp. 50-59, Springer-Verlag, Berlin, 2006.
15. S. Porschen, On variable-weighted exact satisfiability problems, Techn. Report zaik2006-526, Univ. Köln, 2006.
16. S. Porschen, B. Randerath, and E. Speckenmeyer, Linear Time Algorithms for some Not-All-Equal Satisfiability Problems, in: "E. Giunchiglia, A. Tacchella (Eds.), Proceedings of the 6th International Conference on Theory and Applications of Satisfiability Testing (SAT'03), Santa Margherita Ligure, Italy", Lecture Notes in Computer Science, Vol. 2919, pp. 135-149, Springer-Verlag, Berlin, 2004.
17. S. Porschen, and E. Speckenmeyer, Satisfiability Problems for Mixed Horn Formulas, in: H. Kleine Büning, X. Zhao, (Eds.), Proceedings of the Guangzhou Symposium on Satisfiability and its Applications, pp. 106-113, 2004.
18. S. Porschen, and E. Speckenmeyer, Worst case bounds for some NP-complete modified Horn-SAT problems, in: "Proceedings of the 7th International Conference on Theory and Applications of Satisfiability Testing (SAT'04), Vancouver,

British Columbia, Canada", Lect. Notes in Comp. Science, Vol. 3542, pp. 251-262, 2005.

19. S. Porschen, and E. Speckenmeyer, Satisfiability of Mixed Horn Formulas, Techn. Report zaik499-2005, Univ. Köln, 2005.
20. T. J. Schaefer, The complexity of satisfiability problems, in: Conference Record of the Tenth Annual ACM Symposium on Theory of Computing, San Diego, California, 1-3 May 1978, 216-226.