A new Bound for an NP-Hard Subclass of 3-SAT using Backdoors*

Stephan Kottler, Michael Kaufmann, and Carsten Sinz

Eberhard Karls Universität Tübingen, Wilhelm–Schickard–Institute, Tübingen, Germany

Abstract. Knowing a *Backdoor Sets* B for a given SAT instance, satisfiability can be decided by only examining each of the $2^{|B|}$ truth assignments of the variables in B. However, one problem is to efficiently find a small backdoor up to a particular size and, furthermore, if no backdoor of the desired size could be found, there is in general no chance to conclude anything about satisfiability.
In this paper we introduce a complete deterministic algorithm for an NP-hard subclass of 3-SAT, that is also a subclass of Mixed Horn Formulas. For an instance of the described class the absence of two particular kinds of backdoor sets can be used to prove unsatisfiability. The upper bound of this algorithm is $O(p(n) * 1.427^n)$ which is less than the currently best upper bound for deterministic algorithms for 3-SAT and MHF.

1 Introduction and Definitions

The boolean satisfiability problem (SAT) is one of the well known hard problems in theoretical computer science. Even when restricting the number of literals in each clause to a maximum of three (3-SAT), deciding satisfiability of a given instance is known to still be NP-complete. From the theoretical point of view the upper bound to solve 3-SAT could be improved steadily (see [13]). From the practical point of view we know by experience that many SAT instances evolving from real-world applications can be solved within nearly linear time. This is often due to some hidden structure that facilitates the solving process enormously. One possibility to measure this structure, namely *Backdoor Sets*, was introduced in 2003 by Williams, Gomes and Selman [15]. On the one hand it was shown that small backdoor sets are often related to real-world instances [15, 12], on the other hand minimal backdoors of randomized, hence unstructured 3-SAT instances contain from 30% to 65% of all variables [6].
We use backdoors not as a measure of structure but rather to guide an algorithm for a NP-hard subclass of 3-SAT and Mixed Horn Formulas (MHF). MHF denotes the set of all SAT instances in conjunctive normal form where each clause is either Horn or binary [10, 11].

* This work was partly supported by DFG-SPP 1307, project "Structure-based Algorithm Engineering for SAT-Solving"

Strong Backdoor Sets We use the definition of strong backdoor sets that is given in [8]. Note that there are also *weak backdoor sets* [15, 8], however, they are not relevant for this paper. A backdoor is defined with respect to a class $\mathcal{C}$ of formulas that can be recognized and solved in polynomial time. A set $\mathcal{B}$ of variables $\mathcal{V}$ of a boolean formula F is a *strong backdoor set* of F with respect to $\mathcal{C}$ (strong $\mathcal{C}$-backdoor) if $F[\tau] \in \mathcal{C}$ for every truth assignment $\tau : \mathcal{B} \mapsto \{0, 1\}$. $F[\tau]$ denotes the result of removing all clauses that contain a literal x with $\tau(x) = true$ and removing all literals y with $\tau(y) = false$ from F.
We particularly use a variant of strong backdoors, so called *deletion backdoors* [9, 14]: $\mathcal{B}$ is a deletion backdoor if the formula $F - \mathcal{B}$ belongs to $\mathcal{C}$, where $F - \mathcal{B}$ denotes the result of removing all occurrences (both positive and negative) of the variables in B from the clauses of formula F. Every deletion backdoor is a strong backdoor, if class $\mathcal{C}$ is *clause-induced* ($F \in \mathcal{C} \Rightarrow F' \in \mathcal{C}$ for all $F' \subseteq F$) [9]. In this paper we solely deal with the two clause-induced classes Horn and 2-Sat as base classes of backdoors.

Parameterized Algorithms constitute one possible approach to cope with computational intractability [7]. One basic idea of parameterized algorithms is to ask whether a given NP-hard problem has a solution that can be bounded by some non-negative integer parameter k. If a problem is *fixed-parameter tractable* this question can be solved in time that is only exponential in k but not in the size of the original problem. Since our approach rather applies than creates parameterized algorithms we refer the reader to [7] for a formal definition and more detailed information on parameterized complexity.

2 A NP-Hard Subclass of 3-SAT

Definition 1. *Let* $2^\star$-CNF *be the subclass of 3-*Sat *with the restriction that any clause* C *with* $|C| = 3$ *must only contain negative literals.*

Theorem 1. $2^\star$-CNF *is NP-complete.*

Proof. The definition of $2^\star$-CNF as a subclass of 3-Sat $\in$ NP directly implies $2^\star$-CNF to be in NP. The NP-completeness of $2^\star$-CNF can be shown by the polynomial time reduction 3-Sat $\leq_p$ $2^\star$-CNF.
Let F be a boolean formula represented in 3-Sat. We need to specify a formula $F' \in 2^\star$-CNF such that F' is satisfiable if and only if F is satisfiable.
Let therefore $\mathcal{C}^p$ denote the set of all clauses C of F with $|C| = 3$ and C containing at least one positive literal. Let $\mathcal{C}^n := \mathcal{C} \setminus \mathcal{C}^p$ denote the remaining clauses. Moreover, let $\mathcal{V}^p \subseteq \mathcal{V}$ denote those variables of F which occur positively in at least one clause of $\mathcal{C}^p$.
In order to transform a formula $F \in$ 3-Sat into $F' \in 2^\star$-CNF, all clauses $\mathcal{C}^n$ can be adopted unchanged ($\mathcal{C}'^n := \mathcal{C}^n$) for F'. For every variable $x_i \in \mathcal{V}^p$ we introduce one variable x_i^* and one clause $(x_i^* \vee x_i)$ in F'. We refer to the set of these added clauses as $\mathcal{C}'^a$. Furthermore, all clauses in $\mathcal{C}^p$ are modified to clauses $\mathcal{C}'^p$ by replacing each occurrence of a positive literal x_i by the (negative) literal $\overline{x_i^*}$. Hence, F' belongs to class $2^\star$-CNF and it is: F is satisfiable $\Leftrightarrow$ F' is satisfiable

'⇒': If F is satisfiable there exists a model M_F (set of satisfying literals). We create an according model for F' by initializing $M_{F'} := M_F$. Moreover, for all variables $x_i \in \mathcal{V}^p$ we apply the following rule:

$$M_{F'} = \begin{cases} M_{F'} \cup \{\overline{x_i^*}\} & \text{if (positive) literal } x_i \in M_F \\ M_{F'} \cup \{x_i^*\} & \text{else} \end{cases}$$

With this, all clauses ${\mathcal{C}'}^n \cup {\mathcal{C}'}^a$ are satisfied. Let now C' be any arbitrary clause in the remaining set of clauses ${\mathcal{C}'}^p$. There exists at least one literal $l_j \in M_F$ (of variable x_j) which satisfies the according clause $C \in \mathcal{C}^p$. Due to the initialization it is $l_j \in M_{F'}$. In case l_j is a negative literal, C' also contains l_j and hence is satisfied. In case l_j is a positive literal, C' contains the literal $\overline{x_j^*}$ that was chosen for $M_{F'}$ and hence satisfies C'. Consequently all clauses in F' are satisfied.

'⇐': F' is satisfiable by the assignment of the literals in $M_{F'}$. Initializing $M_F := \{l \in M_{F'} : l \text{ belongs to } F\}$ satisfies at least all clauses in $\mathcal{C}^n$. Now consider any clause $C \in \mathcal{C}^p$: Since the according clause $C' \in {\mathcal{C}'}^p$ is satisfied there exists at least one literal $l \in M_{F'}$ satisfying C'. In case l belongs to F then $l \in M_F$ and thus, $C \in F$ is satisfied. If, on the other hand, l does not belong to F then l must be an added and negated variable of the form $\overline{x_i^*}$, whereas clause C contains literal $x_i \in F$. Since $M_{F'}$ is a model for F', in particular the added clause $(x_i^* \vee x_i) \in F'$ is also satisfied by $M_{F'}$. Hence, literal x_i has to be contained in $M_{F'}$ and so it is also contained in M_F. With this M_F is a model for F.
Since for any positive literal in F we added at most one new variable and one new clause to F', the reduction is polynomial. Thus, it is NP-complete to decide whether a given formula $F \in 2^\star$-CNF is satisfiable. □

Note that an alternative proof could adapt the idea to prove NP-hardness for MHF [10]. It turns out that $2^\star$-CNF $\subset$ MHF encodes the problem to decide whether the vertices of a graph can be colored with at most three different colors such that no vertices with the same color are connected by an edge.

3 A Backdoor–driven Approach

Based on the concept of backdoor sets we can specify a simple deterministic algorithm to decide satisfiability for arbitrary formulas of the class $2^\star$-CNF. The main algorithm is listed in Alg. 1 and is explained in detail in this section.

In the second line we define a constant c whose value solely depends on the runtime of two parameterized algorithms we use as subroutines further below. The particular value will become more clear when analyzing the complexity of the algorithm. In line 3 we first consider all clauses C^+ of F that consist of exactly two positive literals. Note that with F being an instance of class $2^\star$-CNF, any clause within the set $\{F \setminus C^+\}$ contains at most one positive literal and thus these clauses are all Horn clauses. In the next line we aim to find the smallest possible set of variables $\mathcal{B}^+$ such that every clause in C^+ contains at least one variable of the set $\mathcal{B}^+$. Since by definition all clauses within C^+ are binary clauses the problem to find the smallest possible set $\mathcal{B}^+$ can be seen as a Vertex-Cover-problem:

Algorithm 1: A Backdoor-driven $2^\star$-CNF Solver

1 **Function** `bd_solve`(F)
2   $c \leftarrow \log_{4.151}(2.0755) \approx 0.513$
3   $C^+ \leftarrow \{(x_i \vee x_j) \in F :\ x_i, x_j \text{ positive}\}$
4   Choose minimum $\mathcal{B}^+ \subseteq \mathcal{V}$, such that $\forall\ C \in C^+\ \exists\ x_i \in \mathcal{B}^+ : x_i \in C$
5   **if** $|\mathcal{B}^+| \leq c * |\mathcal{V}|$ **then**
6   **return** Solve F by using the Horn-Backdoor $\mathcal{B}^+$
7   $C^- \leftarrow \{(\overline{x_h} \vee \overline{x_i} \vee \overline{x_j}) \in F :\ \overline{x_h}, \overline{x_i}, \overline{x_j} \text{ negative}\}$
8   Choose minimum $\mathcal{B}^- \subseteq \mathcal{V}$, such that $\forall\ C \in C^-\ \exists\ x_i \in \mathcal{B}^- : \overline{x_i} \in C$
9   **if** $|\mathcal{B}^-| \leq (1-c) * |\mathcal{V}|$ **then**
10   **return** Solve F by using the Binary-Backdoor $\mathcal{B}^-$
11   **return** F Unsatisfiable

Understanding binary clauses as edges and the variables of the two literals of each clause as the endpoints of an edge, our task responds to find the smallest set of endpoints to cover each edge. It is easy to verify that the set of variables $\mathcal{B}^+$ constitutes a deletion backdoor with the base class $\mathcal{C}$ = Horn: Each clause of the instance $F - \mathcal{B}^+$ contains at most one positive literal. For complexity reasons we target to determine a set $\mathcal{B}^+$ of the size not greater than $c * |\mathcal{V}|$.
If the instance F does not contain a Horn-backdoor of the desired size, we then consider the set of all clauses (C^-) consisting of three literals. Recall that these literals are all negative. In line 8 we aim to find a smallest possible set of variables $\mathcal{B}^-$ such that each clause within the set C^- contains at least one (negative) literal of the variables within the set $\mathcal{B}^-$. This task corresponds to a 3-Hitting-Set problem (see [6]): Clauses of the set C^- can be seen as subsets of three items (variables) each. For $\mathcal{B}^-$ we search for the smallest set of items to hit each subset in C^-. Note that any clause in $F \setminus C^-$ consists of at most two literals. Hence it is clear that the set of variables $\mathcal{B}^-$ constitutes a deletion backdoor with base class $\mathcal{C}$ = 2-Sat: The instance $F - \mathcal{B}^-$ belongs to class 2-Sat, since from each clause with three literals (C^-) at least one is removed. Again for complexity reasons we focus on finding a Binary-backdoor with size not greater than $(1-c)*|\mathcal{V}|$.

When reaching line 11 we know that there exists neither a set of variables $\mathcal{B}^+$ nor a set $\mathcal{B}^-$ with the desired size. In this case we can conclude unsatisfiability of F. Since the considered clauses within C^+ solely consist of positive literals we need to set the values of at least $|\mathcal{B}^+|$ variables to true in order to satisfy all the clauses in C^+. Analogously the size $\mathcal{B}^-$ indicates the number of variables whose values have to be set to $false$ in order to satisfy all clauses within the set C^-. This is impossible with $|\mathcal{B}^+|$ being greater than $c * |\mathcal{V}|$ and $|\mathcal{B}^-|$ being greater than $(1-c) * |\mathcal{V}|$ at the same time.

A similar argument to prove unsatisfiability of big random 3-Sat instances has been used by Franco and Swaminathan in [5]. The authors show that an approximation algorithm for 3-Hitting-Set can determine bounds on how many variables must be set to $true$ and how many must be set to $false$.

Complexity of the Algorithm

It is easy to verify that satisfiability of a boolean formula of class $2^{\star}$-CNF can be decided by the algorithm described above. In this subsection we analyze the complexity of Algorithm 1. In particular we have to focus on the following four computationally intensive tasks of the algorithm:

1. In order to compute the set of variables $\mathcal{B}^+$ a VERTEX COVER problem has to be solved (line 4). There are several good approximation algorithms to deal with VERTEX COVER problems. However, in our case we need to know exactly the minimum set of variables to cover all clauses in C^+ which cannot be achieved by using approximation methods. Considering the fact that we are only interested in a variable set $\mathcal{B}^+$ up to a particular size, we can make use of a parameterized algorithm.
Given a graph $G = (V_G, E_G)$ the parameterized VERTEX COVER problem asks if there is a subset of vertices $C \subseteq V_G$ with k or fewer vertices such that each edge in E_G has at least one of its endpoints in C. According to [7] there are algorithms solving the parameterized VERTEX COVER in time $O(k * |V_G| + 1.29^k)$. Since in our case the parameter k is given by $c * |\mathcal{V}| \approx 0.513 * |\mathcal{V}|$ and $|V_G| = |\mathcal{V}| = n$ the complexity of this task can be limited by $O(n^2 + 1.14^n)$.

2. Solving the instance F by using a Horn-backdoor $\mathcal{B}^+$ with at most $c * n = 0.513 * n$ variables (line 6) may in the worst case imply to examine all possible truth assignments of the variables in $\mathcal{B}^+$. More precisely this might mean that for each of the $2^{0.513n} \approx 1.427^n$ truth assignments a Horn instance has to be solved. The satisfiability of a Horn instance can be decided in linear time by applying for example the algorithm described in [3]. Concluding, the complexity of this part is limited by $O(1.427^n * |F|)$.

3. Analogously, to determine the set $\mathcal{B}^+$, we can use a parameterized algorithm in order to solve the 3-HITTING-SET problem to detect whether there is a set $\mathcal{B}^-$ with at most $(1 - c) * |\mathcal{V}| = (1 - 0.513) * |\mathcal{V}|$ variables (line 8).
Given a collection Q of subsets of size at most three of a finite set S and a non-negative integer k, the parameterized 3-HITTING-SET problem asks if there is a subset $S' \subseteq S$ with $|S'| \leq k$ which allows S' to contain at least one element from each subset in Q [7]. Algorithms to solve this problem have been steadily improved in the last years. In 2004 Fernau published an algorithm for the parameterized 3-HITTING SET problem with an upper bound $O(2.179^k + |S|)$ [4]. Wahlström recently improved this result and gave an algorithm with an upper bound $O(p(n)*2.0755^k)$ with a polynomial $p(n)$. With $k := (1-c)*|\mathcal{V}| \approx 0.487*n$ in our case the complexity can be bounded by $O(1.427^n * p(n))$.

4. To determine satisfiability of F by using the Binary-backdoor $\mathcal{B}^-$ with at most $(1 - c) * n = 0.487 * n$ variables (line 10) may in the worst case imply to solve a 2-SAT instance for each possible truth assignment of the variables in $\mathcal{B}^-$. Since 2-SAT can be solved in linear time [1, 3] the complexity of this part can be limited by $O(1.402^n * |F|)$.

With this, the complexity of Algorithm 1 is bounded by $O(1.427^n * p(n))$. Hence, the idea of considering two different types of backdoors yields a good upper bound for the special class $2^\star$-CNF $\subset$ 3-Sat. This bound is slightly better than the bound $O(2^{0.5284n}) = O(1.4423^n)$ to solve the more general class MHF [10]. Just for comparison, the currently best deterministic algorithm for 3-Sat has an upper bound of $O(1.473^n)$ [2].

4 Conclusion

Based on the concept of backdoor sets we have bounded the complexity to decide satisfiability for $2^\star$-CNF $\subset$ 3-Sat. The complexity for our algorithm mainly depends on the runtime to solve parameterized 3-Hitting Set problems.
It would be interesting to study whether the idea to compute different minimum backdoors of a Sat instance can be used to generate algorithms for further NP-hard subclasses of Sat or MHF.

References

1. B. Aspvall, M. F. Plass, and R. E. Tarjan. A linear-time algorithm for testing the truth of certain quantified boolean formulas. *Inf. Proc. Lett.*, 8:121–123, 1979.
2. T. Brüggemann and W. Kern. An improved deterministic local search algorithm for 3-sat. *Theor. Comput. Sci.*, 329(1-3):303–313, 2004.
3. A. del Val. On 2-SAT and renamable horn. In *AAAI: 17th National Conference on Artificial Intelligence*. AAAI / MIT Press, 2000.
4. H. Fernau. A top-down approach to search-trees: Improved algorithmics for 3-hitting set. *Electronic Colloquium on Computational Complexity*, TR04-073, 2004.
5. J. Franco and R. Swaminathan. On good algorithms for determining unsatisfiability of propositional formulas. *Discrete Appl. Math.*, 130(2):129–138, 2003.
6. Y. Interian. Backdoor sets for random 3-sat. In *Theory and Applications of Satisfiability Testing - SAT 2003*, 2003.
7. R. Niedermeier. *Invitation to Fixed-Parameter Algorithms*. Universität Tübingen, October 2002.
8. N. Nishimura, P. Ragde, and S. Szeider. Detecting backdoor sets with respect to horn and binary clauses. In *Theory and Applications of Satisfiability Testing - SAT 2004*, 2004.
9. N. Nishimura, P. Ragde, and S. Szeider. Solving #SAT using vertex covers. In *Theory and Applications of Satisfiability Testing - SAT 2006*, 2006.
10. S. Porschen and E. Speckenmeyer. Worst case bounds for some NP-complete modified Horn-SAT problems. In *Theory and Applications of Satisfiability Testing - SAT 2004*, 2004.
11. S. Porschen and E. Speckenmeyer. Satisfiability of mixed Horn formulas. *Discrete Applied Mathematics*, 155(11):1408–1419, 2007.
12. Y. Ruan, H. A. Kautz, and E. Horvitz. The backdoor key: A path to understanding problem hardness. In *AAAI*, pages 124–130, 2004.
13. U. Schöning. A probabilistic algorithm for k-sat and constraint satisfaction problems. In *Symposium on Foundations of Computer Science*, 1999.
14. S. Szeider. Matched formulas and backdoor sets. In *Theory and Applications of Satisfiability Testing - SAT 2007*, 2007.
15. R. Williams, C. Gomes, and B. Selman. Backdoors to typical case complexity. In *IJCAI*, 2003.