From Crossing-Free Resolution to Max-SAT Resolution

Mohamed Sami Cherif
Aix-Marseille Univ, Université de Toulon, CNRS, LIS, France

Djamal Habet
Aix-Marseille Univ, Université de Toulon, CNRS, LIS, France

Matthieu Py
Aix-Marseille Univ, Université de Toulon, CNRS, LIS, France

Abstract

Adapting a SAT resolution proof into a Max-SAT resolution proof without considerably increasing its size is an open problem. Read-once resolution, where each clause is used at most once in the proof, represents the only fragment of resolution for which an adaptation using exclusively Max-SAT resolution is known and trivial. Proofs containing non read-once clauses are difficult to adapt because the Max-SAT resolution rule replaces the premises by the conclusions. This paper contributes to this open problem by defining, for the first time since the introduction of Max-SAT resolution, a new fragment of resolution whose proofs can be adapted to Max-SAT resolution proofs without substantially increasing their size. In this fragment, called crossing-free resolution, non read-once clauses are used independently to infer new information thus enabling to bring along each non read-once clause while unfolding the proof until a substitute is required.

2012 ACM Subject Classification Theory of computation → Proof theory

Keywords and phrases Satisfiability, Proof, Max-SAT Resolution

Digital Object Identifier 10.4230/LIPIcs.CP.2022.12

1 Introduction

The maximum satisfiability (Max-SAT) problem is an optimization extension of the satisfiability (SAT) problem and consists, given a formula in Conjunctive Normal Form (CNF), in determining the maximum number of clauses that it is possible to satisfy by an assignment of the variables. This well known formalism is used to represent and solve many real-world and crafted problems making it of great academic and industrial interest [3, 4]. SAT and Max-SAT are strongly related and share many aspects. In fact, SAT solving techniques are often used in the context of Max-SAT solving, particularly in SAT-based and Branch and Bound (BnB) algorithms for Max-SAT [1, 2, 21]. Yet, in theory, bridging the gap between SAT and Max-SAT inference remains one of the main challenges in the last decade.

One of the first proof systems for Max-SAT is based on an inference rule called Max-SAT resolution [6, 7, 16, 17], which is an extension of the resolution rule [28] introduced in the context of SAT. Max-SAT resolution is sound, complete and is the most studied inference rule for Max-SAT, both in theory and practice [1, 5, 18, 19, 23, 24, 27]. However, adapting a resolution proof to get a valid Max-SAT resolution proof of reasonable size remains an open problem. Bonet et al. state that "*it seems difficult to adapt a classical resolution proof to get a Max-SAT resolution proof, and it is an open question if this is possible without increasing substantially*[1] *the size of the proof*" [7]. Indeed, unlike resolution, the Max-SAT resolution rule replaces the premises with the conclusions, which is necessary to maintain

[1] typically when the size of the adapted proof is exponential with respect to the size of the initial one.

28th International Conference on Principles and Practice of Constraint Programming (CP 2022).
Editor: Christine Solnon; Article No. 12; pp. 12:1–12:17
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Max-SAT equivalence after its application. Moreover, aside from the traditional resolvent clause, additional clauses[2] are also added to ensure Max-SAT equivalence. In [17], Larrosa et al. describe Max-SAT resolution as "*a movement of knowledge*". As such, read-once resolution proofs, where each clause is used once, represent the only fragment of resolution for which an immediate and trivial adaptation is possible [6, 7, 12]. Recent works [11, 24] try to circumvent this problem by allowing the use of the split rule, which intuitively allows to duplicate a clause by adding one literal, to linearly adapt tree-like resolution refutations. More specifically, the adaptation takes advantage of the structure of such proofs and applies the split rule to fix the non read-once input clauses. However, the resulting proofs are in the ResS proof system [18] in which Max-SAT resolution is augmented with the split rule. To bridge the gap between SAT and Max-SAT resolution, non read-once clauses need to be inferred using the clauses produced by Max-SAT resolution.

In this paper, we contribute to this open problem by identifying a new fragment of resolution, that we call crossing-free resolution, for which an adaptation using only Max-SAT resolution is possible without substantially increasing the size of the proof. Crossing-free derivations are defined using the ensuing derivations of non read-once clauses. Intuitively, non read-once clauses are used independently to infer new information in crossing-free resolution proofs. The adaptation of such proofs to Max-SAT resolution proofs is shown possible modulo some minor syntactic subtleties. Furthermore, we show that k-stacked diamond patterns, which were shown exponential for the adaptation in [24], fall within the crossing-free resolution fragment and can be adapted into Max-SAT resolution proofs without increasing their size.

This paper is organized as follows. Section 2 gives some necessary definitions and notations and presents the necessary background on resolution for SAT and Max-SAT as well as related work. The crossing-free resolution refinement is introduced in Section 3 and its adaptation to Max-SAT resolution is presented in Section 4. We study (k-stacked) diamond patterns and show that they can be adapted without increasing their size in Section 5. Finally, we conclude in Section 6.

2 Preliminaries

2.1 Definitions and Notations

Let X be the set of propositional variables. A literal l is a variable $x \in X$ or its negation $\overline{x}$. A clause C is a disjunction (or a set) of literals. If $|C| = 1$, C is a unit clause. A formula in Conjunctive Normal Form (CNF) ϕ is a conjunction (or a multiset) of clauses. An assignment $I : X \rightarrow \{true, false\}$ maps each variable to a boolean value and can be represented as a set of literals. A literal l is satisfied (resp. falsified) by an assignment I if $l \in I$ (resp. $\bar{l} \in I$). A clause C is satisfied by an assignment I if at least one of its literals is satisfied by I, otherwise it is falsified by I. The empty clause $\square$ contains zero literals and is always falsified. A clause C is a tautology if it contains both a literal l and its negation $\bar{l}$, i.e., $\exists l \in C$ s.t $\bar{l} \in C$, and in such case it is always satisfied. A clause C opposes a clause C' if C contains a literal whose negation is in C', i.e., $\exists l \in C$ s.t $\bar{l} \in C'$. We denote $var(l)$, $var(C)$ and $var(\phi)$ the variables appearing respectively in the literal l, the clause C and the formula ϕ. The width of a clause C is the number of literals occurring in it. A CNF formula ϕ is satisfied by an assignment I, that we call model of ϕ, if each clause $C \in \phi$ is satisfied by I, otherwise it is falsified by I.

[2] referred to as compensation clauses

Solving the Satisfiability (SAT) problem consists in determining whether there exists an assignment I that satisfies a given CNF formula ϕ. In the case where such an assignment exists, we say that ϕ is satisfiable, otherwise we say that ϕ is unsatisfiable or inconsistent. The cost of an assignment I, denoted $cost_I(\phi)$, is the number of clauses falsified by I. The Maximum Satisfiability (Max-SAT) problem is an optimization extension of SAT which, for a given CNF formula ϕ, consists in determining the maximum number of clauses that can be satisfied by an assignment of the variables. Equivalently, it consists in determining the minimum number of clauses that each assignment must falsify, i.e., $\min_{I} cost_I(\phi)$.

2.2 Resolution for SAT

A well-known proof and refutation system for SAT is based on the resolution rule [28]. Given two opposed clauses, this rule, defined below, deduces a resolvent clause which can be added to the formula. A resolution proof or derivation of a clause C is a finite sequence of resolutions starting from the clauses of ϕ and deducing C usually represented as a finite sequence of clauses. If C is the empty clause $\square$, the proof is referred to as a refutation of ϕ. A resolution proof can also be represented in the form of a Directed Acyclic Graph (DAG) whose nodes are clauses in the proof either having two or zero incoming arcs (resp. if they are resolvents or clauses of the initial formula). The size of a resolution derivation π, denoted $s(\pi)$, is the number of resolvents in it whereas its width, denoted $w(\pi)$, is the maximum width of all its clauses.

▶ **Definition 1** (Resolution [28]). *Given two opposed clauses C_1 and C_2, the resolution rule is defined as follows:*

$$\frac{C_1 = x \vee A \qquad C_2 = \overline{x} \vee B}{C_3 = A \vee B}$$

Many restricted classes of resolution have been studied in the literature, e.g read-once resolution [13], tree (or tree-like) resolution [15] and linear resolution [22] among others. In particular, a resolution proof is read-once if each clause is used at most once in the proof. Similarly, a resolution derivation is tree-like if every intermediate clause, i.e., resolvent, is used at most once in the derivation. Linear resolution, defined below, lies between tree-like and general resolution in terms of proof complexity [8, 9]. In this fragment, the proofs are linear in the sense that each deduced clause is used as premise in the next resolution step. Note that, when the first condition of (c) holds in the definition, the clause D_i is called the input parent clause of C_{i+1}.

▶ **Definition 2** (Linear resolution [22]). *Let ϕ be a CNF formula and C be a clause. A linear resolution derivation of C from ϕ is a sequence of clauses $C_1, ..., C_m$ such that:*

(a) *C_1 is a clause in ϕ*
(b) *C_m is the clause C*
(c) *For every $i < m$, C_{i+1} is the resolvent of C_i either with a clause D_i from ϕ or with a clause C_k for some $k < i$.*

2.3 Resolution for Max-SAT

One of the first and most studied proof systems for Max-SAT is the Max-SAT resolution calculus (MaxRes) which relies on an inference rule extending resolution for Max-SAT [6, 7, 16, 17]. Other than the resolvent clause, this rule, called Max-SAT resolution and defined below, introduces new clauses referred to as compensation clauses and essential to preserve Max-SAT equivalence. As a sound and complete rule for Max-SAT [6, 7], Max-SAT resolution plays an important role in the context of Max-SAT theory and solving [5, 18, 24, 27]. In particular, for a given CNF formula, it is possible to generate a Max-SAT resolution proof of its optimum by applying the saturation algorithm [7]. Furthermore, it is extensively used and studied in the context of Branch and Bound algorithms for Max-SAT [1, 10, 14, 19] and more marginally in the context of SAT-based ones [12, 23].

▶ **Definition 3** (Max-SAT equivalence). *Let ϕ and ϕ' be two CNF formulas. ϕ and ϕ' are Max-SAT equivalent iff for any assignment $I : var(\phi) \cup var(\phi') \rightarrow \{true, false\}$, we have $cost_I(\phi) = cost_I(\phi')$.*

▶ **Definition 4** (Max-SAT resolution [6, 7, 16, 17]). *Given two opposed clauses C_1 and C_2, the Max-SAT resolution rule is defined as follows:*

$$\begin{array}{c} C_1 = x \vee A \qquad C_2 = \overline{x} \vee B \\ \hline C_r = A \vee B \\ CC_1 = x \vee A \vee \overline{B} \\ CC_2 = \overline{x} \vee \overline{A} \vee B \end{array}$$

where C_r is the resolvent clause and CC_1, CC_2 are compensation clauses.

Note that the following rewriting is used to represent the compensation clauses in compacted form: $C \vee \overline{a_1 \vee a_2 \vee ... \vee a_n} = (C \vee \overline{a_1}) \wedge (C \vee a_1 \vee \overline{a_2}) \wedge ... \wedge (C \vee a_1 \vee a_2 \vee ... \vee \vee \overline{a_n})$. This rewriting was introduced in [17] as a recursive rule to transform the compensation clauses into CNF form. This also entails that the Max-SAT resolution rule depends on the ordering of the literals, as reported in [7, 17]. For the sake of simplification, we will allow the use of this rewriting as two full-fledged rules to manipulate clauses in compacted form. We will refer to the left-right rewriting as expansion and right-left one as compaction. This may entail abusing some notations but it is useful to further simplify the proofs. Furthermore, given three sets of literals A, B and C, the equality $C \vee A \vee \overline{B} \stackrel{*}{=} C \vee A \vee \overline{A \vee B}$ is sound for Max-SAT as reported in [17] (c.f. Remark 13) and may be as such used in the proofs. We discuss these subtleties following Theorem 14 in Section 4.

A Max-SAT resolution proof or derivation of a formula ϕ' from ϕ is a finite sequence of Max-SAT resolutions starting from the clauses of ϕ and deducing ϕ' and is usually represented as a finite sequence of formulas. Note that we may allow the addition of tautological clauses to any formula in the proof. We discuss this syntactic subtlety at the end of Section 4. A Max-SAT resolution proof can also be represented as a bipartite DAG whose nodes are either clauses or inference steps (in which case they will be omitted for more simplicity). A sequence of Max-SAT resolution steps deducing one empty clause is referred to as Max-SAT resolution refutation. For a given CNF formula, it is possible to generate a Max-SAT resolution proof of its optimum by applying the saturation algorithm [7]. Note that other inference rules and proof systems were also studied in the context of Max-SAT [5, 11, 18, 20, 27].

Unlike resolution, the Max-SAT resolution rule replaces the premises by the conclusions. Larrosa et al. describe Max-SAT resolution as "*a movement of knowledge*" [17]. Because of this specificity, it is not easy to adapt a resolution proof to obtain a Max-SAT resolution proof. Indeed, in resolution proofs, several resolution steps can share the same premise, because the premises are not consumed after the application of a resolution step. On the other hand, the premises of a Max-SAT resolution step are consumed after its application. Consequently, the immediate adaptation of a resolution proof for Max-SAT is only possible if it is read-once [6, 7, 12]. In this fragment, it is simply sufficient to replace every resolution step in the proof by a Max-SAT resolution step to produce a Max-SAT resolution proof of similar size. However, adapting any resolution proof to a Max-SAT proof without substantially increasing its size remains an open problem.

Recent works [11, 24] augment the Max-SAT resolution rule by the split rule defined below, forming a new system stronger than MaxRes and called ResS [18], to linearly adapt tree-like resolution refutations into ResS refutations. More specifically, the adaptation takes advantage of the structure of such proofs and applies the split rule, which intuitively allows to duplicate a clause by adding one literal, to fix the non read-once input clauses. Furthermore, the substitution algorithm introduced in [26] also enables to generate substitutes for non read-once clauses using SAT oracles but no guarantee is provided for the size of the computed ResS refutations. To the best of our knowledge, read-once resolution remains the only fragment of resolution for which an adaptation using exclusively Max-SAT resolution is possible without substantially increasing the proof size. In the next section, we define a new refinement of resolution for which this is possible.

▶ **Definition 5** (Split). *Given a clause C and variable x, the split rule is defined as follows:*

$$\frac{C}{x \vee C \qquad \overline{x} \vee C}$$

3 Crossing-Free Resolution

The main difficulty in adapting resolution proofs to Max-SAT resolution ones lies in inferring a substitute for non read-once clauses. Indeed, such clauses must be naturally inferred using Max-SAT resolution while unfolding (i.e., reading and applying) the initial resolution proof, contrary to previous works [11, 24] where non read-once clauses are artificially fixed using the split rule before the actual unfolding of the proof. In this section, we define a new fragment of resolution, referred to as crossing-free resolution. The idea behind this refinement is to ensure enough manoeuvrability of proofs in terms of structure in order to infer substitutes for non read-once clauses when necessary. To this end, we define below the notion of ensuing derivation of a non read-once clause. Intuitively, this particular derivation is ensued from a non read-once clause in the sense that it is sufficient to delimit the impact of its multiple use. Note that a node where a set of given paths in a resolution proof intersect will be referred to as their junction node.

▶ **Definition 6** (Ensuing derivation). *Let ϕ be a CNF formula and π a resolution derivation of clause C from ϕ. The ensuing derivation of a non read-once clause C' in π, denoted $ED(C')$, is the sub-derivation of π formed by all the resolution steps in the paths starting from C' in π until their first junction node. We call the clause derived in the junction node, the ensued clause of C', denoted $EC(C')$.*

▶ **Example 7.** We consider the resolution derivation π represented in Figure 1 of clause $C = x_6$ from the formula $\phi = \{\overline{x_1} \vee x_3 \vee \overline{x_4},\ x_4 \vee x_5,\ x_4 \vee \overline{x_5},\ x_1 \vee \overline{x_4},\ x_2 \vee \overline{x_4} \vee x_6,\ x_5 \vee \overline{x_7},\ \overline{x_2} \vee \overline{x_3} \vee x_7,\ \overline{x_5} \vee \overline{x_7}\}$. The non read-once clauses x_4 and $\overline{x_2} \vee \overline{x_3} \vee x_7$ and their ensuing derivations are respectively represented in red and blue. Furthermore, we have $EC(x_4) = x_6$ and $EC(\overline{x_2} \vee \overline{x_3} \vee x_7) = \overline{x_2} \vee \overline{x_3}$.

Recall that clauses are consumed after the application of Max-SAT resolution. Therefore, it seems difficult to adapt resolution derivations in which ensuing derivations of non read-once clauses cross. Indeed, in such cases, the formula can significantly evolve as compensation clauses may be used while others may be generated. As such, crossing-free resolution ensures that ensuing derivations are disjoint, i.e., do not cross, as defined below.

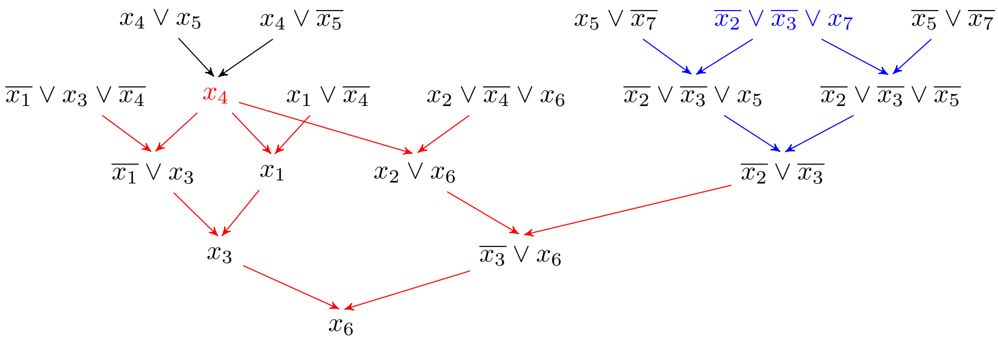

Figure 1 Ensuing derivations in a crossing-free resolution proof.

▶ **Definition 8** (Crossing-free resolution derivation)**.** *Let ϕ be a CNF formula and π a resolution derivation of clause C from ϕ. π is crossing-free iff for every pair of non read-once clauses (C_1, C_2), $ED(C_1)$ and $ED(C_2)$ are disjoint, i.e., they do not contain a shared arc.*

▶ **Example 9.** We consider the same formula ϕ in Example 7. The resolution derivation π of clause $C = x_6$ from ϕ represented in Figure 1 is crossing-free since the ensuing derivations of the non read-once clauses x_4 and $\overline{x_2} \vee \overline{x_3} \vee x_7$ are disjoint.

Note that the crossing-free resolution refinement entails an interesting property established in the following proposition. Intuitively, this property ensures that non read-once clauses are used independently to infer new information in crossing-free resolution proofs. This entails that each ensuing derivation in a crossing-free resolution proof can be adapted independently as described in the next section.

▶ **Proposition 10.** *Let ϕ be a CNF formula, π be a crossing-free resolution derivation of clause C from ϕ and C' a non read-once clause in π. Every clause Cl in $ED(C')$ s.t $Cl \notin \{C', EC(C')\}$ is read-once.*

Proof. Let Cl be a clause in $ED(C')$ s.t $Cl \notin \{C', EC(C')\}$. Clearly, if Cl is not read once, $ED(Cl)$ shares at least one arc with $ED(C')$ which is absurd since π is crossing-free. ◀

4 From Crossing-Free Resolution to Max-SAT Resolution

In this section, we show that crossing-free resolution derivations can be adapted to Max-SAT resolution derivations modulo some minor syntactic subtleties without substantially increasing their size. In the following proposition, we first provide some patterns which will be encountered in the adaptation.

▶ **Proposition 11.** *Let A, B, C and $\{l\}$ be four sets of literals s.t $|C| > 0$. The following deductions can be done in $O(|C|)$ inference steps:*

(a) $(A \vee C) \wedge (B \vee \overline{C}) \vdash_{MaxRes} A \vee B$

(b) $(l \vee A \vee \overline{C}) \wedge (\bar{l} \vee B) \vdash_{MaxRes} A \vee B \vee \overline{C}$

(c) $(l \vee A \vee \overline{C}) \wedge (\bar{l} \vee B \vee \overline{C}) \vdash_{MaxRes} A \vee B \vee \overline{C}$

Proof. We provide the proof for case (a) by induction on $|C| = n$:

- If $n = 1$, then $C = \{l'\}$. Clearly, $(A \vee l') \wedge (B \vee \overline{l'}) \vdash_{MaxRes} A \vee B$ by application of a Max-SAT resolution step on literal $var(l')$.

- Suppose $n > 1$ and let $l' \in C$. By the induction hypothesis, we can deduce $(A \vee C) \wedge (B \vee \overline{C \setminus \{l'\}}) \vdash_{MaxRes} A \vee B \vee l'$ in $n-1$ inference steps. Furthermore, $B \vee \overline{C} = (B \vee \overline{C \setminus \{l'\}}) \wedge (B \vee \overline{l'})$ by expansion and $(A \vee B \vee l') \wedge (B \vee \overline{l'}) \vdash_{MaxRes} A \vee B$ by application of a Max-SAT resolution step on variable $var(l')$. Therefore, we conclude that we can deduce $(A \vee C) \wedge (B \vee \overline{C})\} \vdash_{MaxRes} A \vee B$ in $O(n)$ inference steps.

Proofs for cases (b) and (c) are similar by induction on $|C|$. ◀

Next, we start dealing with the adaptation of crossing-free resolution derivations and particularly ensuing derivations. To generate a substitute for a non read-once clause, note that we can use the literals in the junction nodes (c.f. Lemma 1 in [24]) of an ensuing derivation, i.e., nodes where paths starting from the non read-once clause intersect. To generate such substitutes using Max-SAT resolution, we start by dealing with read-once linear parts in the proof. Informally, we want to drag (i.e., bring along) each non read-once clause while unfolding the proof until they are reused. This is formally established for read-once linear parts of the proof in the following lemma. Note that the implications of equality $\stackrel{*}{=}$ in the proof will be further discussed at the end of the section.

▶ **Lemma 12.** *Let ϕ be a CNF formula, $\pi = C_1, ..., C_{s(\pi)}$ be a read-once linear resolution derivation of clause $C \neq \square$ from ϕ. We can deduce $\phi \vdash_{MaxRes} C \wedge (C_1 \vee \overline{C})$ in $O(s(\pi) \times w(\pi))$ inference steps.*

Proof. Let $m = s(\pi)$. Since π is read-once, it can be trivially adapted into a Max-SAT resolution derivation of C from ϕ of the same size by replacing every resolution step with a Max-SAT resolution step [6, 7, 12]. Next, we prove by induction on $i \in \{1, .., m-1\}$ that we can infer $C'_i = C_1 \vee \overline{C_{i+1}}$ at the i^{th} Max-SAT resolution step:

- For $i = 1$, the first Max-SAT resolution on clauses $C_1 = l_1 \vee A_1$ and $D_1 = \overline{l_1} \vee B_1$ w.r.t $var(l_1)$ generates the following compensation clause:

$$CC_{1|1} = l_1 \vee A_1 \vee \overline{B_1} \stackrel{*}{=} l_1 \vee A_1 \vee \overline{A_1 \vee B_1} = C_1 \vee \overline{C_2} = C'_1$$

  Note that to establish the equality $\stackrel{*}{=}$, we can add the tautological clauses $l_1 \vee A_1 \vee B_1 \vee \overline{A_1}$ (or alternatively $l_1 \vee A_1 \vee \overline{A_1}$) to the formula) in which case $l_1 \vee A_1 \vee \overline{A_1 \vee B_1}$ can be trivially inferred by compaction. Furthermore, if D_1 is a unit clause, $CC_{1|1}$ is not generated. However, we can simply add the tautological clauses $l_1 \vee A_1 \vee \overline{A_1}$ which correspond to $C_1 \vee \overline{C_2}$ since $D_1 = \overline{l_1}$ (i.e., B_1 is empty).

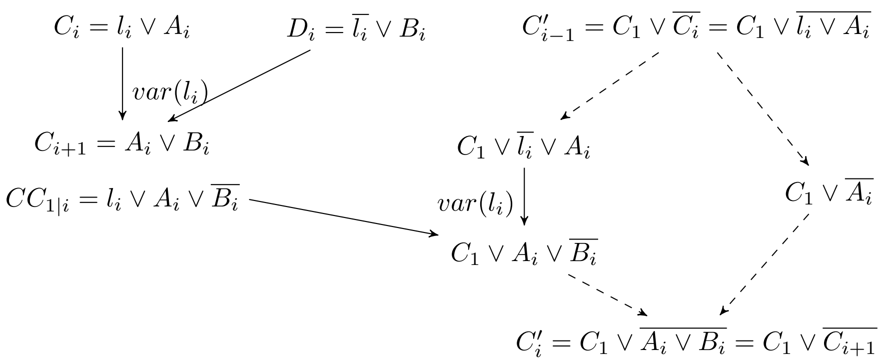

Figure 2 Induction step to infer C'_i at the i^{th} step. Solid lines represent the application of the Max-SAT resolution rule whereas dashed lines represent compaction or expansion. Unused compensation clauses are omitted.

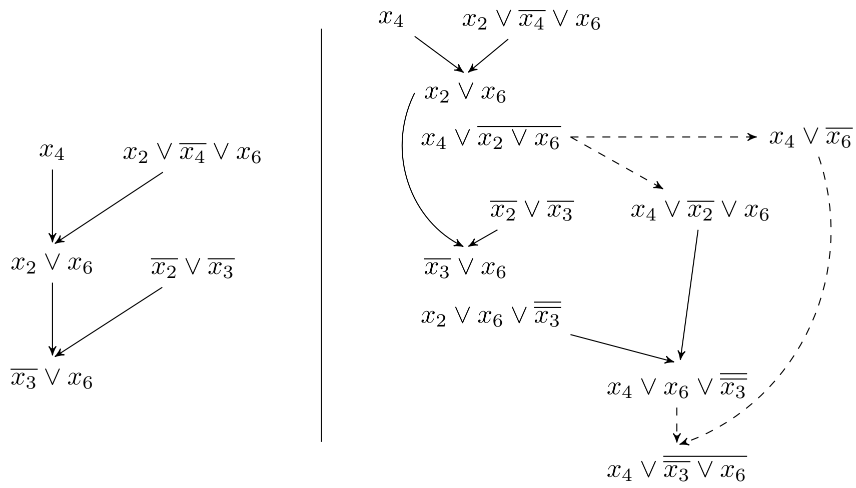

Figure 3 Dragging the non read-once clause while unfolding a read-once linear section of the proof. Solid lines represent the application of the Max- SAT resolution rule whereas dashed lines represent compaction or expansion. Unused compensation clauses are omitted.

- Suppose that we can generate $C'_{i-1} = C_1 \vee \overline{C_i}$ at the $i^{th} - 1$ Max-SAT resolution step. The i^{th} step on $C_i = l_i \vee A_i$ and $D_i = \overline{l_i} \vee B_i$ w.r.t $var(l_i)$ generates the resolvent $C_{i+1} = A_i \vee B_i$ and the compensation clauses $CC_{1|i} = l_i \vee A_i \vee \overline{B_i}$ and $CC_{2|i} = \overline{l_i} \vee \overline{A_i} \vee B$. The induction step to infer C'_i is represented in Figure 2. Note that similarly to the base case, if $D_i = \overline{l_i} (i > 1)$ is a unit clause, i.e., the i^{th} step corresponds to a deletion of literal l_i from $C_i = l_i \vee A_i$ deducing the resolvent $C_{i+1} = A_i$, the tautological clauses $l_i \vee A_i \vee \overline{A_i}$ can be added to the formula thus replacing $CC_{1|i}$ in Figure 2. However, as showcased in the same figure, the addition of such clauses in case D_1 is unit can be avoided since the initial expansion step on C'_{i-1} suffices to generate $C_1 \vee \overline{A_i} = C_1 \vee \overline{C_{i+1}} = C'_i$.

Finally, by Proposition 11 (case b.), the inference of $C_1 \vee A_i \vee \overline{B_i}$ in Figure 2 requires $O(|B_i|)$ Max-SAT resolution steps and, thus, every step in π is clearly adapted in $O(w(\pi))$ inference steps to generate C and $C_1 \vee \overline{C_m}$. Therefore, we conclude that we can deduce $\phi \vdash_{MaxRes} C \wedge (C_1 \vee \overline{C_m})$ in $O(s(\pi).w(\pi))$ inference steps. ◀

▶ **Example 13.** We consider the read-once linear derivation of clause $\overline{x_3} \vee x_6$ from $\phi = \{x_4,\ x_2 \vee \overline{x_4} \vee x_6,\ \overline{x_2} \vee \overline{x_3}\}$ represented on the left of Figure 3. The Max-SAT resolution proof deducing $\overline{x_3} \vee x_6$ and $x_4 \vee \overline{\overline{x_3} \vee x_6}$ is represented on the right of Figure 3.

Next, we establish our main result on the adaptation of crossing-free resolution derivations. The proof in the following theorem particularly deals with the junction nodes in ensuing derivations, i.e., nodes where the paths starting from the non read-once clauses intersect. More specifically, we want to drag or bring along the non read-once clause through these particular nodes. We provide an illustration of a full adaptation in Example 15.

▶ **Theorem 14.** *Let ϕ be a CNF formula and π be a crossing-free resolution derivation of clause C from ϕ. We can deduce $\phi \vdash_{MaxRes} C$ in $O(s(\pi) \times (s(\pi) + w(\pi))^2)$ inference steps.*

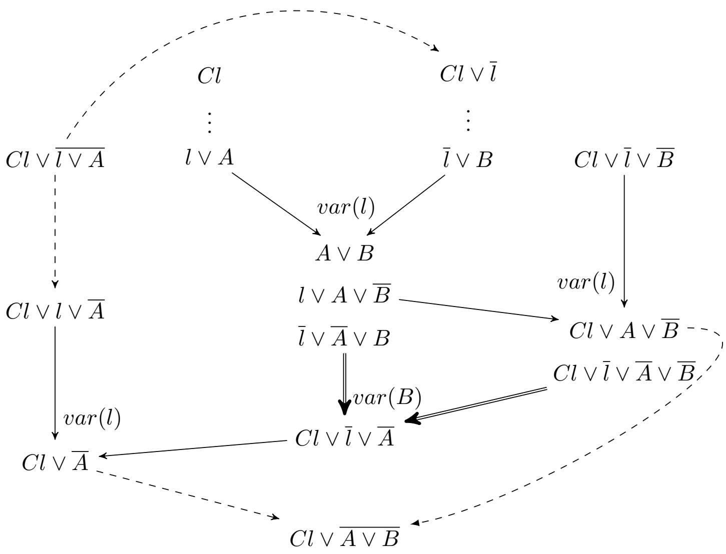

Figure 4 Inferring $Cl \lor \overline{A \lor B}$ in a junction node of $ED(Cl)$. Solid lines represent the application of the Max-SAT resolution rule, bold double arcs represent the application of Max-SAT resolution to delete opposed sets of literals and dashed lines represent compaction or expansion. Unused compensation clauses are omitted.

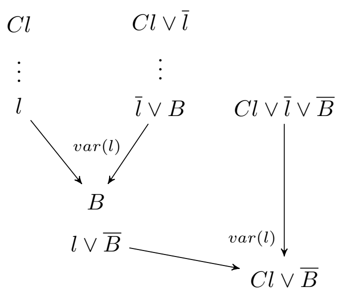

Figure 5 Inferring $Cl \lor \overline{C'}$ in case A is empty in a junction node of $ED(Cl)$. Solid lines represent the application of the Max-SAT resolution rule whereas dashed lines represent compaction and expansion. Unused compensation clauses are omitted.

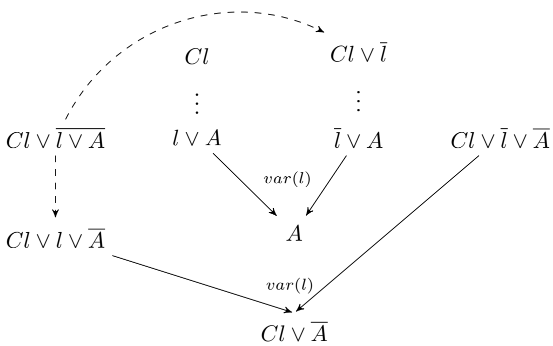

Figure 6 Inferring $Cl \lor \overline{C'}$ in case $A = B$ in a junction node of $ED(Cl)$. Solid lines represent the application of the Max-SAT resolution rule whereas dashed lines represent compaction and expansion. Unused compensation clauses are omitted.

Proof. Property 10 ensures that each ensuing derivation can be adapted independently. Let Cl be a non read-once clause in π and w.l.o.g we only consider it ensuing derivation $ED(Cl)$. We prove that at each step of $ED(Cl)$ deriving clause C', we can infer C' and $SC \vee \overline{C'}$ where SC is either Cl or its substitute in the path leading to C'. The proof is by induction on the size of the derivation. The base case where the derivation is empty is trivial. Next, using Lemma 12, we can suppose w.l.o.g that C' is derived in a junction node (of paths starting from Cl). Let $l \vee A$ and $\bar{l} \vee B$ be the premises of the resolution step deriving $C' = A \vee B$.

The induction hypothesis ensures that there exists a Max-SAT resolution derivation of $l \vee A$ and $C \vee \overline{l \vee A}$. As showcased in Figure 4, $Cl \vee \bar{l}$ can be used to replace the occurrences of Cl in the derivation of $\bar{l} \vee B$. Note that to avoid using tautological substitutes, we can suppose w.l.o.g that $l \notin Cl$ by interchanging the proofs of $l \vee A$ and $l \vee B$ when necessary thus entailing a different unfolding order of the original proof and the generation of the exact same clause as a substitute in such nodes. Again, similarly to the left side, the induction hypothesis ensures the existence of a Max-SAT resolution derivation of $\bar{l} \vee B$ and $Cl \vee \bar{l} \vee \overline{\bar{l} \vee B}$ and, therefore, $Cl \vee \bar{l} \vee \overline{B}$ by expansion. Clearly, $C' = A \vee B$ can be derived by Max-SAT resolution and we showcase in Figure 4 how $Cl \vee \overline{C'} = C \vee \overline{A \vee B}$ can be inferred using the compensation clauses as well as $Cl \vee \overline{l \vee A}$ and $Cl \vee \bar{l} \vee \overline{B}$.
Note that the following particular cases can occur:

- A or B is empty, in which case a unit clause is used to derive $C' = A \vee B$. We represent in Figure 5 how to derive $Cl \vee \overline{C'}$ in case A is empty. The derivation in case B is empty is symmetric and thus omitted. Notice that in the case both A and B are empty, π is a refutation and there is no need to derive $Cl \vee \overline{C'}$ in the last Max-SAT resolution step. In fact, more generally, this is also not necessary for the last junction node in an ensuing derivation in π.
- $A = B$ in which case the generated compensation clauses are tautological and are not necessary to derive $Cl \vee \overline{C'} = Cl \vee \overline{A}$ as showcased in Figure 6.

Finally, in each junction node we need $O(|B|)$ inference steps to deduce $Cl \vee A \vee \overline{B}$ using case (c) in Proposition 11. Similarly, using expansion on A and pattern (b) in Proposition 11, we need $O(|A| \times |B|)$ inference steps to deduce $Cl \vee \bar{l} \vee \overline{A}$. It is important to note that the width of the proof may evolve while generating substitutes for non read-once clauses as literals may be added in junction nodes. However, the width remains bounded by $w(\pi) + s(\pi)$ and thus each junction node can be adapted in $O((w(\pi) + s(\pi))^2)$ inference steps. Therefore, we conclude that we can deduce $\phi \vdash_{MaxRes} C$ in $O(s(\pi) \times (s(\pi) + w(\pi))^2)$ inference steps. ◀

▶ **Example 15.** We consider the formula $\phi = \{\overline{x_1} \vee x_3 \vee \overline{x_4},\ x_4 \vee x_5,\ x_4 \vee \overline{x_5},\ x_1 \vee \overline{x_4},\ x_2 \vee \overline{x_4} \vee x_6,\ \overline{x_2} \vee \overline{x_3}\}$ and the derivation π of clause x_6 from ϕ represented in Figure 1. We omit the section of the proof (in blue) deriving clause $\overline{x_2} \vee \overline{x_3}$ for simplicity. Note that this omitted part, i.e., the ensuing derivation of the non read-once clause $\overline{x_2} \vee \overline{x_3} \vee \overline{x_7}$ corresponds to a diamond pattern [24]. Such patterns will be studied in Section 5 (refer to Example 23 for the adaptation). The adaptation of proof π is reported in Figure 7. We reuse the adaptation of the linear read-once section in Example 3. The non read-once clause and its substitutes are colored in red and added tautological clauses are represented in green. Note that this is one of the possible adaptations depending on the order chosen for adapting the branches of $ED(x_4)$. Finally, we stress the fact that we could have generated the clause $C = x_4 \vee \overline{x_6}$ after the last Max-SAT resolution step on clauses $\overline{x_3} \vee x_3$ (but we omit this inference since $x_6 = EC(x_4)$ as mentioned in the proof of Theorem 14). Indeed, C can be inferred by an additional Max-SAT resolution step on the compensation clauses obtained in the last step, i.e., clauses $x_3 \vee \overline{x_6}$ and $x_4 \vee \overline{x_3} \vee \overline{x_6}$. In the proof of Theorem 14, this corresponds to the case where B is empty in a junction node of an ensuing derivation.

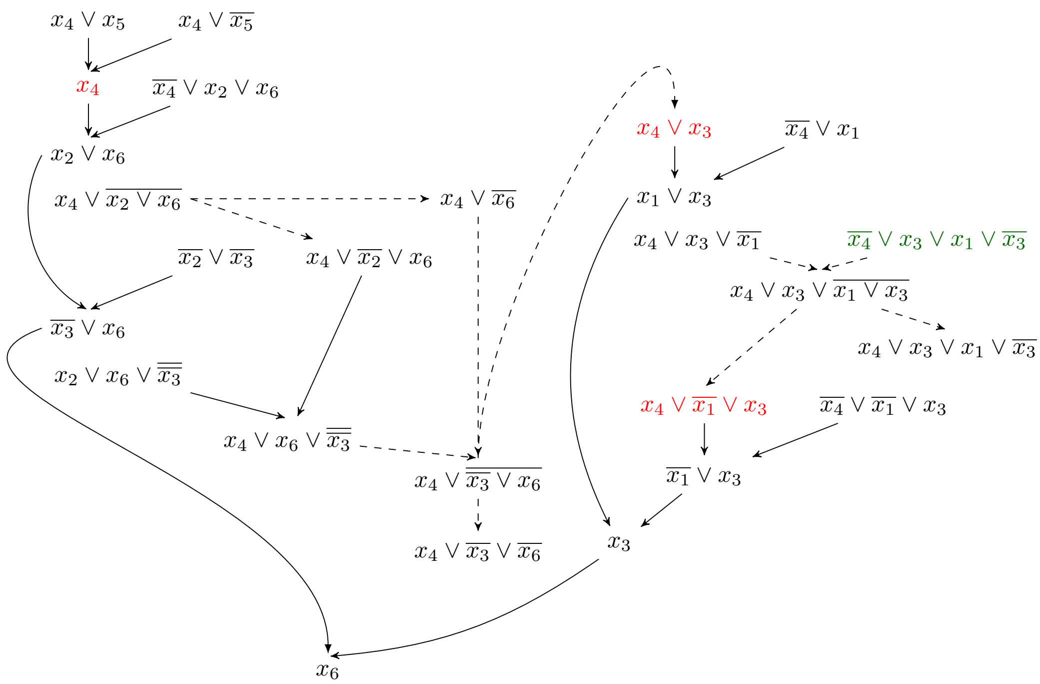

Figure 7 Adaptation of a crossing-free resolution derivation. Solid lines represent the application of the Max-SAT resolution rule whereas dashed lines represent compaction and expansion. Unused compensation clauses are omitted.

Next, we discuss some minor syntactic subtleties that occur in the adaptation. First, it is important to note that the use of the expansion and compaction rewritings as full fledged rules is relevant for simplification but not necessary. Recall that these two rules are mainly used in order to switch between the different equivalent forms of $\overline{C}$ when it is written in CNF form. Each form corresponds to a different ordering of the literals in C. When applying Max-SAT resolution, a relevant order may be chosen when necessary. However, an application of a compaction followed by an expansion may correspond to a certain rearrangement of the variables in CNF form. This may occur when adapting the read-once linear part of the proof. Indeed, as showcased in Figure 2, a compaction may be followed by an expansion to isolate the clause $C_1 \vee \overline{l_i} \vee A_i$ from the compact form $C_1 \vee \overline{A_i}$. Similarly, as shown in Figure 4, it may be necessary to isolate the clause $Cl \vee \overline{l}$ from the compact form $Cl \vee l \vee \overline{A}$ when dealing with junction nodes.

More specifically, we may need to rearrange a certain literal at the beginning or at the end of the ordering. In Proposition 16, we prove that it is possible to switch the first and last literals in the CNF form of $\overline{C}$ in $O(|C|)$ inference steps. This entails that in the proof of Theorem 14, the compaction and expansion rules can be omitted and replaced with $O(s(\pi) \times (s(\pi) + w(\pi)))$ Max-SAT resolutions. Clearly, this does not impact our result in terms of the size of the resulting adaptation. In Example 17, we provide the full simplified adaptation of the proof in Example 15 without the use of rewriting rules.

▶ **Proposition 16.** *Let n be a natural number and $l_1, ..., l_n$ be n literals. We can deduce $(\overline{l_1}) \wedge (l_1 \vee \overline{l_2}) \wedge ... \wedge (l_1 \vee ... \vee l_{n-1} \vee \overline{l_n}) \vdash_{MaxRes} (\overline{l_n}) \wedge (l_n \vee \overline{l_2}) \wedge ... \wedge (l_n \vee l_2 \vee ... \vee l_{n-1} \vee \overline{l_1})$ in $O(n)$ inference steps.*

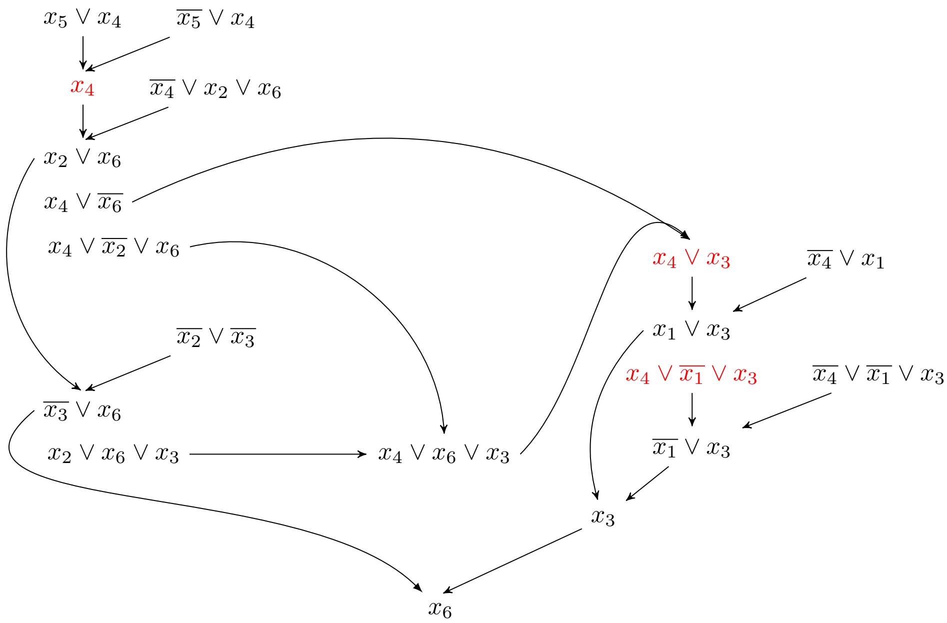

Figure 8 Adaptation of a crossing-free resolution proof to a Max-SAT resolution proof. Unused compensation clauses are omitted.

Proof. By induction on n we have:

- If $n = 1$ the result is trivial.
- For $n > 1$, the application of Max-SAT resolution on clauses $l_1 \vee .... \vee l_{n-2} \vee \overline{l_{n-1}}$ and $l_1 \vee .... \vee l_{n-1} \vee \overline{l_n}$ w.r.t $var(l_{n-1})$ generates the resolvent clause $C = l_1 \vee .... \vee l_{n-2} \vee \overline{l_n}$ and the compensation clause $CC = l_1 \vee .... \vee l_{n-2} \vee l_n \vee \overline{l_{n-1}}$. Furthermore, by induction, we can deduce $(\overline{l_1}) \wedge (l_1 \vee \overline{l_2}) \wedge ... \wedge (l_1 \vee ... \vee l_{n-2} \vee \overline{l_n}) \vdash_{MaxRes} (\overline{l_n}) \wedge (l_n \vee \overline{l_2}) \wedge ... \wedge (l_n \vee l_2 \vee ... \vee l_{n-2} \vee \overline{l_1})$ in $O(n-1)$ inference steps. A single additional Max-SAT resolution step on clauses CC and $l_n \vee l_2 \vee ... \vee l_{n-2} \vee \overline{l_1}$ w.r.t $var(l_1)$ is sufficient to generate the resolvent clause $l_n \vee l_2 \vee ... \vee l_{n-2} \vee \overline{l_{n-1}}$ and the compensation clause $l_n \vee l_2 \vee ... \vee l_{n-1} \vee \overline{l_1}$. Therefore, we deduce the wanted result in $O(n)$ inference steps. ◀

▶ **Example 17.** We consider the same formula ϕ in Example 15. We represent in Figure 8 a Max-SAT resolution proof (without rewriting) of clause x_6 from ϕ. Notice how we use the following rearrangement $\overline{x_6} \wedge (x_6 \vee x_3) \vdash_{MaxRes} x_3 \wedge (x_3 \vee x_6)$ to generate the substitute $x_4 \vee x_3$. Furthermore, the tautological clause $\overline{x_4} \vee x_3 \vee x_1 \vee \overline{x_3}$ colored in green in Figure 7 and the rearrangement in which it is involved are not necessary since the last required substitute for x_1, i.e $x_4 \vee \overline{x_1} \vee x_3$ is naturally generated by the preceding Max-SAT resolution step. Therefore, they can be deleted as is the case for the full adaptation without rewriting in Figure 8.

Next, we discuss the implications of the equality $\stackrel{*}{=}$ used in the proof of Lemma 12, i.e., $l \vee A \vee \overline{B} \stackrel{*}{=} l \vee A \vee \overline{A \vee B}$. Recall that this equality is sound for Max-SAT (c.f. Remark 13 in [17]). However, to avoid adding it as a standalone rule and as explained in the proof of Lemma 12, we can consider the addition of tautological clauses. This may also be required in case of unit clauses. It is important to note that the number of tautological clauses added to the formula in an adaptation of a crossing-free resolution derivation π is in $O(s(\pi) \times (w(\pi) + s(\pi)))$. A similar phenomenon was also noted in [8]. In addition, notice how the adaptation may also rely on tautological compensation clauses which are generated by Max-SAT resolution. Such clauses are usually deleted or omitted in the literature [6, 7, 17] but they may carry important information which is necessary to infer substitutes for non read-once clauses.

Finally, we establish our result on crossing-free refutations in the following corollary. We also illustrate in Example 19 an adaptation of a crossing free resolution refutation to a Max-SAT resolution refutation.

▶ **Corollary 18.** *Let ϕ be an unsatisfiable CNF formula and π be a crossing-free resolution refutation of ϕ. We can deduce $\phi \vdash_{MaxRes} \square$ from ϕ in $O(s(\pi)^3)$ inference steps.*

Proof. Trivially entailed from Theorem 14 since $w(\pi) = O(s(\pi))$ for refutations. ◀

▶ **Example 19.** We consider the unsatisfiable CNF formula $\phi = \{x_1, \overline{x_1} \vee x_3, \overline{x_1} \vee x_2, \overline{x_2} \vee \overline{x_3}\}$ and the refutation π of ϕ represented in Figure 9. Clearly, π is crossing-free since there is only one non read-once clause, i.e., x_1. In fact, π also corresponds to the ensuing derivation of x_1 and $\square$ is its ensued clause, i.e., $ED(x_1) = \pi$ and $EC(x_1) = \square$. Two possible adaptations of π are illustrated in Figure 10. The non read-once clause and its substitutes are colored in red. The possible adaptations correspond to different possible orderings of the proof. In the adaptation on the left, we consider that the resolution step on clauses $\overline{x_1} \vee x_1$ and x_1 precedes the one on clauses x_1 and $\overline{x_1} \vee x_2$, and inversely for the adaptation on the right. Note that the adaptation on the left corresponds to the handmade example provided by Bonet et al. in [6, 7] (c.f. Example 1 in [6] or Example 3 in [7]).

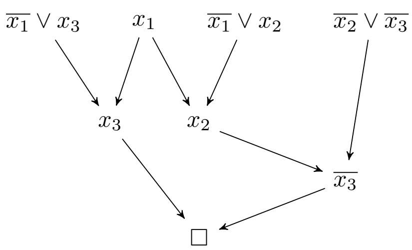

Figure 9 Crossing-free resolution refutation.

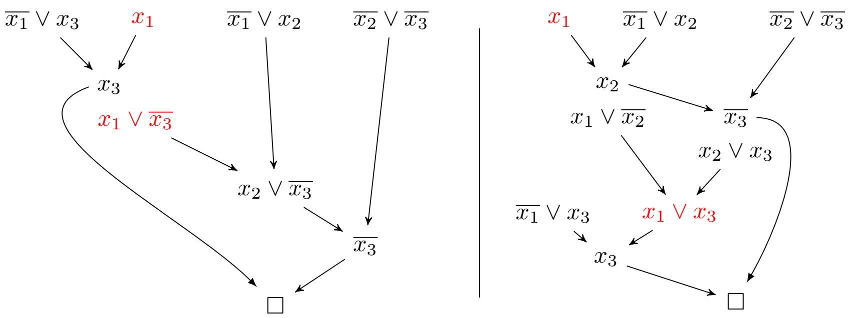

Figure 10 Two possible adaptations of the crossing-free resolution refutation represented in Figure 9 depending on the ordering of the resolution steps involving the non read-once clause x_1. Unused compensation clauses are omitted.

5 On (k-stacked) Diamond Patterns

In this section, we study particular resolution refutations, called k-stacked diamond patterns, which were introduced and shown exponential for the adaptation (to ResS) in [24]. A diamond pattern (x, y, A) where $x, y \notin A$ is the sequence of resolutions represented in Figure 11. Note that the particular diamond pattern $(x, y, \Box)$ is a resolution refutation. Now, imagine that the topmost clause of $(x, y, \Box)$ is derived through another diamond pattern. We iterate the same reasoning to define a k-stacked diamonds pattern as in Definition 20.

▶ **Definition 20** (k-stacked diamond)**.** *Let $k \geq 1$ be a natural number and let x_i and y_i where $1 \leq i \leq k$ be distinct variables. A k-stacked diamond pattern is formed by k diamond patterns (x_i, y_i, A_i) where $1 \leq i \leq k$ such that $A_1 = \Box$ and $A_i = (x_1 \vee \cdots \vee x_{i-1})$ for $1 < i \leq k$. Each diamond (x_i, y_i, A_i) is stacked on top of $(x_{i-1}, y_{i-1}, A_{i-1})$ such that the last conclusion of the former is the topmost central premise of the latter.*

When $k > 2$, the size of a k-stacked diamond P is $s(P) = 3k$ while the size of the computed refutation in ResS [18], i.e., Max-SAT resolution augmented with the split rule, by the adaptation in [24] is at least 2^{k-1} which is exponential in the size of P. First, notice that k-stacked diamond patterns fall within the crossing-free resolution. Furthermore, these patterns can be adapted to Max-SAT resolution refutations without increasing their size as established in 22. Such an adaptation is illustrated in Example 23.

▶ **Proposition 21.** *Let $k \geq 1$ be a natural number. A k-stacked diamond resolution refutation is crossing-free.*

▶ **Proposition 22.** *Let ϕ be a CNF formula, $k \geq 1$ be a natural number and π be a k-stacked diamond resolution refutation. There exists a Max-SAT resolution refutation π' of ϕ s.t $s(\pi') \leq s(\pi)$.*

Proof. We show how to adapt every diamond pattern without increasing its size In Figure 12. This is entailed by the fact that each diamond is clearly a crossing-free derivation and more specifically an ensuing derivation of a non read-once clause. As such, a k-stacked diamond P can be adapted in at most $s(P)$ Max-SAT resolution steps. ◀

▶ **Example 23.** We consider the ensuing derivation of clause $\overline{x_2} \vee \overline{x_3} \vee x_7$ represented in Figure 1. As mentioned in Example 15, this part of the proof corresponds to a diamond pattern. Its adaptation is illustrated in Figure 13 (the non read-once clause and its substitute are represented in red). The adaptation can be added on top of clause $\overline{x_2} \vee \overline{x_3}$ in Figure 8 to obtain the full adaptation of the initial crossing-free proof represented in Figure 1.

6 Conclusion

In this paper, we introduced a new fragment of resolution, called crossing-free resolution, in which ensuing derivations of non read-once clauses are disjoint. We showed that crossing-free

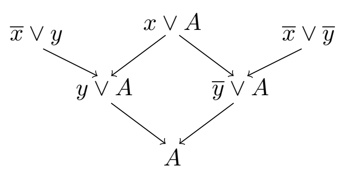

Figure 11 Diamond pattern (x, y, A).

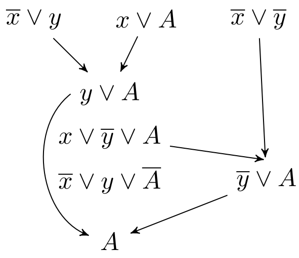

Figure 12 Adaptation of a diamond pattern (x, y, A).

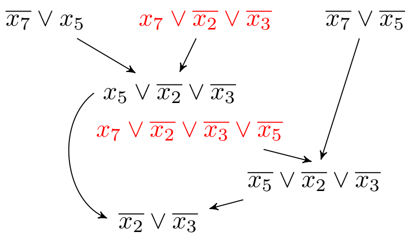

Figure 13 Adaptation of the diamond pattern in the resolution proof represented in Figure 1 (colored in blue) corresponding to the ensuing derivation of clause $x_7 \vee \overline{x_2} \vee \overline{x_3}$. Unused compensation clauses are omitted.

resolution derivations and in particular crossing-free refutations can be adapted to Max-SAT resolution proofs without substantially increasing their size. To the best of our knowledge, this is the first non trivial fragment, i.e., different from read-once resolution, whose adaptation is shown possible using only Max-SAT resolution with a reasonable guarantee on the size of the adapted proofs. The idea behind the adaptation is to naturally infer substitutes for non read-once clauses by dragging them along while unfolding the initial resolution proof and by relying on compensation clauses produced by Max-SAT resolution. Furthermore, we show that diamond patterns, which were shown exponential for the adaptation in [24], fall within the crossing-free resolution fragment and can be adapted into Max-SAT resolution proofs without increasing their size.

Our results contribute to the difficult open problem of adapting resolution proofs to Max-SAT resolution proofs without increasing their size [6, 7] and, therefore, helps to bridge the gap between resolution for SAT and Max-SAT. Furthermore, unlike SAT solvers, Max-SAT solvers are still not able to output certificates in the form of Max-SAT equivalent proofs mainly due to the variety of solving paradigms and due to the theoretical gap between SAT and Max-SAT resolution. Our work can be useful in this regard and particularly in improving the efficiency of independent proof builders for the Max-SAT problem [25]. Finally, as future work, it would be interesting to characterize a larger intersection between SAT and Max-SAT resolution by proving that an adaptation of an extended refinement of resolution (ideally unrestricted resolution) without a substantial increase in the size of the proofs is possible, even through augmenting Max-SAT resolution by other inference rules.

References

1 André Abramé and Djamal Habet. Ahmaxsat: Description and Evaluation of a Branch and Bound Max-SAT Solver. *J. Satisf. Boolean Model. Comput.*, 9(1):89–128, 2014. doi:10.3233/sat190104.

2 Carlos Ansótegui, Maria Luisa Bonet, and Jordi Levy. SAT-based MaxSAT algorithms. *Artificial Intelligence*, 196:77–105, 2013. doi:10.1016/j.artint.2013.01.002.

3 Fahiem Bacchus, Jeremias Berg, Matti Järvisalo, and Ruben Martins, editors. *MaxSAT Evaluation 2021: Solver and Benchmark Descriptions.* Department of Computer Science Report Series B. Department of Computer Science, University of Helsinki, Finland, 2021. URL: https://maxsat-evaluations.github.io/2021/.

4 Fahiem Bacchus, Matti Järvisalo, and Ruben Martins. *Maximum Satisfiability*, pages 929–991. Frontiers in Artificial Intelligence and Applications. IOS PRESS, Netherlands, 2 edition, 2021. doi:10.3233/FAIA201008.

5 Maria Luisa Bonet and Jordi Levy. Equivalence Between Systems Stronger Than Resolution. In Luca Pulina and Martina Seidl, editors, *Theory and Applications of Satisfiability Testing - SAT 2020 - 23rd International Conference, Alghero, Italy, July 3-10, 2020, Proceedings*, volume 12178 of *Lecture Notes in Computer Science*, pages 166–181. Springer, 2020. doi:10.1007/978-3-030-51825-7_13.

6 Maria Luisa Bonet, Jordi Levy, and Felip Manyà. A Complete Calculus for Max-SAT. In Armin Biere and Carla P. Gomes, editors, *Theory and Applications of Satisfiability Testing - SAT 2006, 9th International Conference, Seattle, WA, USA, August 12-15, 2006, Proceedings*, volume 4121 of *Lecture Notes in Computer Science*, pages 240–251. Springer, 2006. doi:10.1007/11814948_24.

7 Maria Luisa Bonet, Jordi Levy, and Felip Manyà. Resolution for Max-SAT. *Artif. Intell.*, 171(8-9):606–618, 2007. doi:10.1016/j.artint.2007.03.001.

8 Joshua Buresh-Oppenheim and Toniann Pitassi. The Complexity of Resolution Refinements. *J. Symb. Log.*, 72(4):1336–1352, 2007. doi:10.2178/jsl/1203350790.

9 Sam Buss and Jan Johannsen. On Linear Resolution. *J. Satisf. Boolean Model. Comput.*, 10(1):23–35, 2016. doi:10.3233/sat190112.

10 Mohamed Sami Cherif, Djamal Habet, and André Abramé. Understanding the power of Max-SAT resolution through UP-resilience. *Artif. Intell.*, 289:103397, 2020. doi:10.1016/j.artint.2020.103397.

11 Yuval Filmus, Meena Mahajan, Gaurav Sood, and Marc Vinyals. MaxSAT Resolution and Subcube Sums. In Luca Pulina and Martina Seidl, editors, *Theory and Applications of Satisfiability Testing - SAT 2020 - 23rd International Conference, Alghero, Italy, July 3-10, 2020, Proceedings*, volume 12178 of *Lecture Notes in Computer Science*, pages 295–311. Springer, 2020. doi:10.1007/978-3-030-51825-7_21.

12 Federico Heras and João Marques-Silva. Read-Once Resolution for Unsatisfiability-Based Max-SAT Algorithms. In Toby Walsh, editor, *IJCAI 2011, Proceedings of the 22nd International Joint Conference on Artificial Intelligence, Barcelona, Catalonia, Spain, July 16-22, 2011*, pages 572–577. IJCAI/AAAI, 2011. doi:10.5591/978-1-57735-516-8/IJCAI11-103.

13 Kazuo Iwama and Eiji Miyano. Intractability of Read-Once Resolution. In *Proceedings of the Tenth Annual Structure in Complexity Theory Conference, Minneapolis, Minnesota, USA, June 19-22, 1995*, pages 29–36. IEEE Computer Society, 1995. doi:10.1109/SCT.1995.514725.

14 Adrian Kügel. Improved Exact Solver for the Weighted MAX-SAT Problem. In Daniel Le Berre, editor, *POS-10. Pragmatics of SAT, Edinburgh, UK, July 10, 2010*, volume 8 of *EPiC Series in Computing*, pages 15–27. EasyChair, 2010. doi:10.29007/38lm.

15 Sukhamay Kundu. Tree resolution and generalized semantic tree. In Zbigniew W. Ras and Maria Zemankova, editors, *Proceedings of the ACM SIGART International Symposium on Methodologies for Intelligent Systems, ISMIS 1986, Knoxville, Tennessee, USA, October 22-24, 1986*, pages 270–278. ACM, 1986. doi:10.1145/12808.12838.

16 Javier Larrosa and Federico Heras. Resolution in Max-SAT and its relation to local consistency in weighted CSPs. In Leslie Pack Kaelbling and Alessandro Saffiotti, editors, *IJCAI-05, Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence, Edinburgh, Scotland, UK, July 30 - August 5, 2005*, pages 193–198. Professional Book Center, 2005. URL: `http://ijcai.org/Proceedings/05/Papers/0360.pdf`.

17 Javier Larrosa, Federico Heras, and Simon de Givry. A logical approach to efficient Max-SAT solving. *Artif. Intell.*, 172(2-3):204–233, 2008. `doi:10.1016/j.artint.2007.05.006`.

18 Javier Larrosa and Emma Rollon. Towards a Better Understanding of (Partial Weighted) MaxSAT Proof Systems. In Luca Pulina and Martina Seidl, editors, *Theory and Applications of Satisfiability Testing - SAT 2020 - 23rd International Conference, Alghero, Italy, July 3-10, 2020, Proceedings*, volume 12178 of *Lecture Notes in Computer Science*, pages 218–232. Springer, 2020. `doi:10.1007/978-3-030-51825-7_16`.

19 Chu Min Li, Felip Manyà, and Jordi Planes. New Inference Rules for Max-SAT. *J. Artif. Intell. Res.*, 30:321–359, 2007. `doi:10.1613/jair.2215`.

20 Chu Min Li, Felip Manyà, and Joan Ramon Soler. A Clause Tableau Calculus for MaxSAT. In Subbarao Kambhampati, editor, *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*, pages 766–772. IJCAI/AAAI Press, 2016. URL: `http://www.ijcai.org/Abstract/16/114`.

21 Chu-Min Li, Zhenxing Xu, Jordi Coll, Felip Manyà, Djamal Habet, and Kun He. Combining Clause Learning and Branch and Bound for MaxSAT. In Laurent D. Michel, editor, *27th International Conference on Principles and Practice of Constraint Programming, CP 2021, Montpellier, France (Virtual Conference), October 25-29, 2021*, volume 210 of *LIPIcs*, pages 38:1–38:18. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021. `doi:10.4230/LIPIcs.CP.2021.38`.

22 D. W. Loveland. A linear format for resolution. In M. Laudet, D. Lacombe, L. Nolin, and M. Schützenberger, editors, *Symposium on Automatic Demonstration*, pages 147–162, Berlin, Heidelberg, 1970. Springer Berlin Heidelberg. `doi:10.1007/BFb0060630`.

23 Nina Narodytska and Fahiem Bacchus. Maximum Satisfiability Using Core-Guided MaxSAT Resolution. In Carla E. Brodley and Peter Stone, editors, *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, July 27 -31, 2014, Québec City, Québec, Canada*, pages 2717–2723. AAAI Press, 2014. URL: `https://ojs.aaai.org/index.php/AAAI/article/view/9124`.

24 Matthieu Py, Mohamed Sami Cherif, and Djamal Habet. Towards Bridging the Gap Between SAT and Max-SAT Refutations. In *32nd IEEE International Conference on Tools with Artificial Intelligence, ICTAI*, pages 137–144. IEEE, 2020. `doi:10.1109/ICTAI50040.2020.00032`.

25 Matthieu Py, Mohamed Sami Cherif, and Djamal Habet. A Proof Builder for Max-SAT. In Chu-Min Li and Felip Manyà, editors, *Theory and Applications of Satisfiability Testing - SAT 2021 - 24th International Conference, Barcelona, Spain, July 5-9, 2021, Proceedings*, volume 12831 of *Lecture Notes in Computer Science*, pages 488–498. Springer, 2021. `doi:10.1007/978-3-030-80223-3_33`.

26 Matthieu Py, Mohamed Sami Cherif, and Djamal Habet. Computing Max-SAT Refutations using SAT Oracles. In *33rd IEEE International Conference on Tools with Artificial Intelligence, ICTAI 2021, Washington, DC, USA, November 1-3, 2021*, pages 404–411. IEEE, 2021. `doi:10.1109/ICTAI52525.2021.00066`.

27 Matthieu Py, Mohamed Sami Cherif, and Djamal Habet. Inferring Clauses and Formulas in Max-SAT. In *33rd IEEE International Conference on Tools with Artificial Intelligence, ICTAI 2021, Washington, DC, USA, November 1-3, 2021*, pages 632–639. IEEE, 2021. `doi:10.1109/ICTAI52525.2021.00101`.

28 John Alan Robinson. A Machine-Oriented Logic Based on the Resolution Principle. *J. ACM*, 12(1):23–41, 1965. `doi:10.1145/321250.321253`.