Do Hard SAT-Related Reasoning Tasks Become Easier in the Krom Fragment?

Nadia Creignou, Reinhard Pichler, Stefan Woltran

Do Hard SAT-Related Reasoning Tasks Become Easier in the Krom Fragment?

Nadia Creignou
Aix Marseille Univ, Université de Toulon, CNRS, LIS
13 288 Marseille, France
nadia.creignou@univ-amu.fr

Reinhard Pichler
TU Wien
Favoritenstraße 9–11
A-1040 Vienna, Austria
pichler@dbai.tuwien.ac.at

Stefan Woltran
TU Wien
Favoritenstraße 9–11
A-1040 Vienna, Austria
woltran@dbai.tuwien.ac.at

March 23, 2019

Abstract

Many reasoning problems are based on the problem of satisfiability (SAT). While SAT itself becomes easy when restricting the structure of the formulas in a certain way, the situation is more opaque for more involved decision problems. For instance, the CardMinSat problem which asks, given a propositional formula φ and an atom x, whether x is true in some cardinality-minimal model of φ, is easy for the Horn fragment, but, as we will show in this paper, remains Θ_2P-complete (and thus NP-hard) for the Krom fragment (which is given by formulas in CNF where clauses have at most two literals). We will make use of this fact to study the complexity of reasoning tasks in belief revision and logic-based abduction and show that, while in some cases the restriction to Krom formulas leads to a decrease of complexity, in others it does not. We thus also consider the CardMinSat problem with respect to additional restrictions to Krom formulas towards a better understanding of the tractability frontier of such problems.

Keywords: Complexity, Satisfiability, Belief Revision, Abduction, Krom Formulas.

1 Introduction

By Schaefer's famous theorem [23], we know that the SAT problem becomes tractable under certain syntactic restrictions such as the restriction to Horn formulas (i.e., formulas in CNF where clauses have at most one positive literal) or to Krom formulas (i.e., clauses have at most two literals). Propositional formulas play an important role in reasoning problems in a great variety of areas such as belief change, logic-based abduction, closed-world reasoning, etc. Most of the relevant problems in these areas are intractable. It is therefore a natural question whether restrictions on the formulas involved help to decrease the high complexity. While the restriction to Horn formulas is usually well studied, other restrictions in Schaefer's framework have received much less attention. In this work, we have a closer look at the restriction to Krom formulas and its effect on the complexity of several hard reasoning problems from the AI domain.

One particular source of complexity of such problems is the involvement of some notion of *minimality*. In closed world reasoning, we add negative literals or more general formulas to a theory only if they are true in every minimal model of the theory in order to circumvent inconsistency. In belief revision we want to revise a given belief set by some new belief. To this end, we retain only those models of the new belief which have *minimal* distance to the models of the given belief set. In abduction, we search for a subset of the hypotheses (i.e., an "explanation") that is consistent with the given theory and which – together with the theory – explains (i.e., logically entails) all manifestations. Again, one is usually not contented with any subset of the hypotheses but with a minimal one. However, different notions of minimality might be considered, in particular, minimality w.r.t. set inclusion or w.r.t. cardinality. In abduction, these two notions are directly applied to explanations [9]. In belief revision, one is interested in the models of the new belief which have minimal distance from the models of the given belief set. Distance between models is defined here via the symmetric set difference Δ of the atoms assigned to true in the compared models. Dalal's revision operator [7] seeks to minimize the cardinality of Δ while Satoh's operator [22] defines the minimality of Δ in terms of set inclusion.

The chosen notion of minimality usually has a significant impact on the complexity of the resulting reasoning tasks. If minimality is defined in terms of cardinality, we often get problems that are complete for some class $\Delta_k\mathrm{P}[O(\log n)]$ with $k \geq 2$ (which is also referred to as $\Theta_k\mathrm{P}$, [26]). For instance, two of the most common reasoning tasks in belief revision (i.e., model checking and implication) are $\Theta_2\mathrm{P}$-complete for Dalal's revision operator [8, 17]. Abduction aiming at cardinality-minimal explanations is $\Theta_3\mathrm{P}$-complete [9]. If minimality is defined in terms of set inclusion, completeness results for one of the classes $\Sigma_k\mathrm{P}$ or $\Pi_k\mathrm{P}$ for some $k \geq 2$ are more common. For instance, belief revision with Satoh's revision operator becomes $\Sigma_2\mathrm{P}$-complete (for model checking) [17] respectively $\Pi_2\mathrm{P}$-complete (for implication) [8]. Similarly, abduction is $\Sigma_2\mathrm{P}$-complete if we test whether some hypothesis is contained in a subset-minimal explanation [9].

For the above mentioned problems, various ways to decrease the complexity have been studied. Indeed, in almost all of these cases, a restriction of the involved formulas to Horn makes the complexity drop by one level in the polynomial hierarchy. Only belief revision with Dalal's revision operator remains $\Theta_2\mathrm{P}$-complete in the Horn case, see [8, 17, 9]. The restriction to Krom has not been considered yet for these problems. In this paper we show that the picture is very similar to the Horn case. Indeed, for the considered problems in belief revision and abduction, we get for the Krom case exactly the same complexity classifications. Actually, we choose the problem reductions for our hardness proofs in such a way that we thus also strengthen the previous hardness results by showing that they even hold if formulas are restricted to Horn *and* Krom at the same time.

To get a deeper understanding why certain problems remain $\Theta_2\mathrm{P}$-hard in the Krom case, we have a closer look at a related variant of the SAT problem, where also cardinality minimality is involved, namely the CardMinSat problem: given a propositional formula φ and an atom x, is x true in some *cardinality-minimal* model of φ? It is easy to show that this problem is $\Theta_2\mathrm{P}$-complete. If φ is restricted to Horn, then this problem becomes trivial, since Horn formulas have a unique minimal model which can be efficiently computed. But what happens if we restrict φ to Krom? We show that $\Theta_2\mathrm{P}$-completeness holds also in this case. This hardness result will then also be very convenient for proving our $\Theta_2\mathrm{P}$-hardness results for the considered problems in belief revision and abduction. Since CardMinSat seems to be central in evaluating the complexity of reasoning problems in which some kind of cardinality-minimality is involved we investigate its complexity in a deeper way, in particular by characterizing the tractable cases.

The **main contributions of the paper** are as follows:

- *Prototypical problems for* Δ_2P *and* Θ_2P. In Section 3, we first review the prototypical Δ_2P-problem LexMaxSat. Hardness for this problem is due to [15] but only follows implicitly from that work. We reformulate the proofs in terms of standard terminology in order to explicitly prove Θ_2P-completeness for the related problem LogLexMaxSat in an analogous way. Note that LogLexMaxSat, which will be the basis for our forthcoming results, is not mentioned explicitly in [15].

- *SAT variants.* In Section 4, we investigate the complexity of the CardMinSat problem and the analogously defined CardMaxSat problem. Our central result is that these problems remain Θ_2P-complete for formulas in Krom form.

- *Applications.* We investigate several reasoning problems in the areas of belief revision and abduction in Section 5. For the restriction to Krom form, we establish the same complexity classifications as for the previously known restriction to Horn form. In fact, we thus also strengthen the previously known hardness results by showing that hardness even holds for the simultaneous restriction to Horn *and* Krom.

- *Classification of* CardMinSat *inside the Krom fragment.* In Section 6 we investigate the complexity of CardMinSat *within* the Krom fragment in order to identify the necessary additional syntactic restrictions towards tractability. To this aim we use the well-known framework by Schaefer and obtain a complete complexity classification of the problem.

2 Preliminaries

Propositional logic. We assume familiarity with the basics of propositional logic [1]. We only want to fix some notation and conventions here. We say that a formula is in k-CNF (resp. k-DNF) for $k \geq 2$ if all its clauses (resp. terms) have at most k literals. Formulas in 2-CNF are also called *Krom*. A formula is called *Horn* (resp. *dual Horn*) if it is in CNF and in each clause at most one literal is positive (resp. negative). It is convenient to identify truth assignments with the set of variables which are true in an assignment. It is thus possible to consider the cardinality and the subset-relation on the models of a formula. If in addition an order is defined on the variables, we may alternatively identify truth assignments with bit vectors, where we encode true (resp. false) by 1 (resp. 0) and arrange the propositional variables in decreasing order. We thus naturally get a lexicographical order on truth assignments.

Complexity Classes. All complexity results in this paper refer to classes in the Polynomial Hierarchy (PH) [20]. The building blocks of PH are the classes P and NP of decision problems solvable in deterministic resp. non-deterministic polynomial time. The classes Δ_kP, Σ_kP, and Π_kP of PH are inductively defined as $\Delta_0\mathrm{P} = \Sigma_0\mathrm{P} = \Pi_0\mathrm{P} = \mathrm{P}$ and $\Delta_{k+1}\mathrm{P} = \mathrm{P}^{\Sigma_k\mathrm{P}}$, $\Sigma_{k+1}\mathrm{P} = \mathrm{NP}^{\Sigma_k\mathrm{P}}$, $\Pi_{k+1}\mathrm{P} = \text{co-}\Sigma_{k+1}\mathrm{P}$, where we write $\mathrm{P}^{\mathcal{C}}$ (resp. $\mathrm{NP}^{\mathcal{C}}$) for the class of decision problems that can be decided by a deterministic (resp. non-deterministic) Turing machine in polynomial time using an oracle for the class $\mathcal{C}$. The prototypical Σ_kP-complete problem is $\exists$-QSAT$_k$, (i.e., quantified satisfiability with k alternating blocks of quantifiers, starting with $\exists$), where we have to decide the satisfiability of a formula $\varphi = \exists X_1 \forall X_2 \exists X_3 \dots Q X_k \psi$ (with $Q = \exists$ for odd k and $Q = \forall$ for even k) with no free propositional variables. W.l.o.g., one may assume here that ψ is in 3-DNF (if $Q = \forall$), resp. in 3-CNF (if

$Q = \exists$). In [26], several restrictions on the oracle calls in a Δ_kP computation have been studied. If on input x with $|x| = n$ at most $O(\log n)$ calls to the Σ_{k-1}P oracles are allowed, then we get the class $\mathrm{P}^{\Sigma_{k-1}\mathrm{P}[O(\log n)]}$ which is also referred to as Θ_kP. A large collection of Θ_2P-complete problems is given in [15, 11, 10]. Sections 3 and 4 in this work will also be devoted to Θ_2P-completeness results. Several problems complete for Θ_kP with $k \geq 3$ can be found in [16]. Hardness results in all these classes are obtained via log-space reductions, and we write $A \leq B$ when a problem A is log-space reducible to a problem B.

3 Δ_2P- and Θ_2P-completeness

The following problems are considered as prototypical for the classes Δ_2P and Θ_2P. In particular, Θ_2P-hardness of LogLexMaxSat will be used as starting point for our reductions towards Θ_2P-hardness of CardMinSat for Krom formulas in Section 4.

Problem:	LexMaxSat
Input:	Propositional formula φ and an order $x_1 > \cdots > x_\ell$ on the variables in φ.
Question:	Is x_ℓ true in the lexicographically maximal model of φ?

Problem:	LogLexMaxSat
Input:	Propositional formula φ and an order $x_1 > \cdots > x_\ell$ on some of the variables in φ with $\ell \leq \log |\varphi|$.
Question:	Is x_ℓ true in the lexicographically maximal bit vector $(b_1, \ldots, b_\ell)$ that can be extended to a model of φ?

The LexMaxSat problem will serve as our prototypical Δ_2P-complete problem while the LogLexMaxSat problem will be the basis of our Θ_2P-completeness proofs. The Δ_2P-completeness of LexMaxSat is stated in [15] – without proof though (see Theorem 3.4 in [15]). The Δ_2P-membership is easy. For the Δ_2P-hardness, the proof is implicit in a sequence of lemmas and theorems in [15]. However, the main goal in [15] is to advocate a new machine model (so-called NP *metric Turing machines*) for defining new complexity classes of optimization problems (the so-called OptP and OptP[$z(n)$] classes). The LogLexMaxSat problem is not mentioned explicitly in [15] but, of course, it is analogous to the LexMaxSat problem.

To free the reader from the burden of tracing the line of argumentation in [15] via several lemmas and theorems on the OptP and OptP[$O(\log n)$] classes, we give direct proofs of the Δ_2P- and Θ_2P-completeness of LexMaxSat and LogLexMaxSat, respectively, in the standard terminology of oracle Turing machines (cf. [20]). In the first place, we thus have to establish the connection between oracle calls and optimization. To this end, we introduce the following problems:

Problem:	NP-Max
Input:	Turing machine M running in non-deterministic polynomial time and producing a binary string as output; string x as input to M.
Question:	Let w denote the lexicographically maximal output string over all computation paths of M on input x; does the last bit of w have value 1?

Problem: LogNP-Max

Input:	Turing machine M running in non-deterministic polynomial time and producing a binary string, whose length is logarithmically bounded in the size of the Turing machine and the input; string x as input to M.
Question:	Let w denote the lexicographically maximal output string over all computation paths of M on input x; does the last bit of w have value 1?

Theorem 1. *The* NP-MAX *problem is* Δ_2P*-complete and the* LOGNP-MAX *problem is* Θ_2P*-complete.*

Proof. The Δ_2P-*membership* of NP-MAX and the Θ_2P-*membership* of LOGNP-MAX is seen by the following algorithm, which runs in deterministic polynomial time and has access to an NP-oracle. The algorithm maintains a bit vector $(v_1, v_2, \dots)$ of the lexicographically maximal prefix of possible outputs of TM M on input x. To this end, we initialize i to 0 and ask the following kind of questions to an NP-oracle: Does there exist a computation path of TM M on input x, such that the first i output bits are $(v_1, \dots, v_i)$ and M outputs yet another bit? If the answer to this oracle call is "no" then the algorithm stops with acceptance if ($i \geq 1$ and $v_i = 1$) holds and it stops with rejection if ($i = 0$ or $v_i = 0$) holds.

If the oracle call yields a "yes" answer, then our algorithm calls another NP-oracle with the question: Does there exist a computation path of TM M on input x, such that the first $i+1$ output bits are $(v_1, \dots, v_i, 1)$? If the answer to this oracle call is "yes", then we set $v_{i+1} = 1$; otherwise we set $v_{i+1} = 0$. In either case, we then increment i by 1 and continue with the first oracle question (i.e., does there exist a computation path of TM M on input x, such that the first i output bits are $(v_1, \dots, v_i)$ and M outputs yet another bit?).

Suppose that the lexicographically maximal output produced by M on input x has m bits. Then our algorithm needs in total $2m + 1$ oracle calls. Clearly, the oracles work in non-deterministic polynomial time. Moreover, if the size of the output string of M is logarithmically bounded, then the number of oracle calls is logarithmically bounded as well.

For the *hardness* part, we first concentrate on the NP-MAX problem and discuss afterwards the modifications required to prove the corresponding complexity result also for the LOGNP-MAX problem. Consider an arbitrary problem $\mathcal{P}$ in Δ_2P, i.e., $\mathcal{P}$ is decided in deterministic polynomial time by a Turing machine N with access to an oracle N_{SAT} for the SAT-problem.

Now let x be an arbitrary instance of problem $\mathcal{P}$. From this we construct an instance $M; x$ of NP-MAX, where we leave x unchanged and we define M as follows: In principle, M simulates the execution of N on input x. However, whenever N reaches a call to the SAT-oracle, with some input φ_i say, then M non-deterministically executes N_{SAT} on φ_i. In other words, in the computation tree of M, the subtree corresponding to this non-deterministic execution of N_{SAT} on φ_i is precisely the computation tree of N_{SAT} on φ_i. On every computation path ending in acceptance (resp. rejection) of N_{SAT}, the TM M writes 1 (resp. 0) to the output. After that, M continues with the execution of N as if it had received a "yes" (resp. a "no") answer from the oracle call. After the last oracle call, M executes N to the end. If N ends in acceptance, then M outputs 1; otherwise it outputs 0 as the final bit.

It remains to prove the following claims:

1. *Correctness.* Let w denote the lexicographically maximal output string over all computation paths of M on input x. Then the last bit of w has value 1 if and only if x is a positive instance of $\mathcal{P}$.

2. *Polynomial time.* The total length of each computation path of M on input x is polynomially bounded in $|x|$.

3. *Logarithmically bounded output.* If the number of oracle calls of TM N is logarithmically bounded in its input (i.e., problem $\mathcal{P}$ is in Θ_2P), then the size of the output produced by M on input x is also logarithmically bounded.

Correctness. We first have to prove the following claim: for every $i \geq 1$ and for every bit vector w_i of length i: w_i is a prefix of the lexicographically maximal output string over all computation paths of M on input x if and only if w_i encodes the correct answers of the first i oracle calls of TM N on input x, i.e., for $j \in \{1, \dots, i\}$, $w_i[j] = 1$ (resp. $w_i[j] = 0$) if the j-th oracle call yields a "yes" (resp. a "no") answer. This claim can be easily verified by induction on i. For instance, consider the induction begin with $i = 1$. We have to show that the first bit of the output string is 1 if and only if the first oracle call yields a "yes" answer. Indeed, suppose that the first oracle does yield a "yes" answer. This means that *at least one* of the computation paths of the non-deterministic oracle machine N ends with acceptance. By our construction of M, there exists at least one computation path of the non-deterministic TM M on which value 1 is written as first bit to the output. Hence, the string w_1 = '1' is indeed the prefix of length 1 of the lexicographically maximal output string over all computation paths of M on input x. Conversely, suppose that the first oracle call yields a "no" answer. This means that *all* of the computation paths of the non-deterministic oracle machine N end with rejection. By our construction of M, all computation paths of the non-deterministic TM M write 0 as the first bit to the output. Hence, the string w_1 = '0' is indeed the prefix of length 1 of the lexicographically maximal output string. The induction step is shown by the same kind of reasoning.

Now let m denote the number of oracle calls carried out by TM N on input x and let w denote the lexicographically maximal output of TM M over all its computation paths. Then w has length $m + 1$ such that the first m bits encode the correct answers of the oracle calls of TM N on input x. Moreover, by the construction of M, we indeed have that the last bit of w is 1 (resp. 0), if and only if x is a positive (resp. negative) instance of $\mathcal{P}$.

Polynomial time. Suppose that N on input x with $|x| = n$ is guaranteed to hold after at most $p(n)$ steps and that N_{SAT} on input φ with $|\varphi| = m$ is guaranteed to hold after at most $q(m)$ steps for polynomials $p()$ and $q()$. Then the total length of the computation of N counting also the computation steps of the oracle machine N_{SAT} is bounded by a polynomial $r(n)$ with $r(n) = O(p(n) * q(p(n)))$. To carry this upper bound over to every branch of the computation tree of M on input x we have to solve a subtle problem: The upper bound $r(n)$ on the execution length of M on input x clearly applies to every computation path where for every oracle call φ_i, a correct computation path of N_{SAT} on input φ_i is simulated. However, in our simulation of N with oracle N_{SAT} by a non-deterministic computation of M on input x, we possibly produce computation paths which N on input x can never reach, e.g.: if the correct answer of N_{SAT} on oracle input φ_1 is "yes", then the continuation of the simulation of N on input x on all computation paths where answer "no" on oracle input φ_1 is assumed, can never be reached by a the computation of N (with oracle N_{SAT}) on input x. To make sure that the polynomial upper bound applies to *every* computation path of M, we have to extend TM M by a counter such that M outputs 0 and halts if more than $r(n)$ steps have been executed.

Logarithmically bounded output. Suppose that $\mathcal{P}$ is an arbitrary problem in Θ_2P and that, therefore, TM N only has logarithmically many oracle calls. By similar considerations as for the polynomial time bound of the computation of M on x, we can make sure that the size of the output on every computation path of M on x is logarithmically bounded. To this end, we add a counter also for the number of oracle calls. Clearly, the logarithmic bound (say $c \log n + d$ for constants c, d and $n = |x|$) on the size of the output applies to every computation path, where for every oracle call φ_i, a correct computation path of N_{SAT} on input φ_i is simulated by M. By adding to M a counter for the size of the output, we can modify M in such a way that M outputs 0 and stops if the number $c \log n + d$ of output bits of M (which corresponds to the number of oracle calls of N) is exceeded. $\square$

The above problems will allow us now to prove the Δ_2P- and Θ_2P-completeness of the problems LexMaxSat and LogLexMaxSat, respectively. As mentioned above, credit for these results (in particular, the Δ_2P-completeness of LexMaxSat) goes to Mark W. Krentel [15]. However, we hope that sticking to the standard terminology of oracle Turing machines (and avoiding the "detour" via the OptP and OptP$[O(\log n)]$ classes based on a new machine model) will help to better convey the intuition of these results.

Theorem 2. *The* LexMaxSat *problem is* Δ_2P*-complete and the* LogLexMaxSat *problem is* Θ_2P*-complete. The hardness results hold even if* φ *is in 3-CNF.*

Proof. The Δ_2P-*membership* of LexMaxSat and the Θ_2P-*membership* of LogLexMaxSat is seen by the following algorithm, which runs in deterministic polynomial time and has access to an NP-oracle. Let m denote the number of variables in φ (in case of the LexMaxSat problem) or the number of variables for which an order is given (in case of the LogLexMaxSat problem). The algorithm maintains a bit vector $(v_1, \dots, v_m)$ of the lexicographically maximal (prefix of a possible) model of φ. To this end, we initialize i to 0 and ask the following kind of questions to an NP-oracle: Does there exist a model of φ such that the truth values of the first i variables $(x_1, \dots, x_i)$ are $(v_1, \dots, v_i)$ and variable x_{i+1} is set to 1? If the answer to this oracle call is "yes", then we set $v_{i+1} = 1$; otherwise we set $v_{i+1} = 0$. In either case, we then increment i by 1 and continue with the next oracle call.

When $i = m + 1$ is reached, then the algorithm checks the value of v_m and stops with acceptance if $v_m = 1$ and it stops with rejection if $v_m = 0$ holds. Clearly, if the number m of variables of interest (i.e., those for which an order is given) is logarithmically bounded in the size of φ, then also the number of oracle calls of our algorithm is logarithmically bounded in the size of φ.

The *hardness* proof is by reduction from NP-Max (resp. LogNP-Max) to the LexMaxSat (resp. LogLexMaxSat) problem. Let $M; x$ be an arbitrary instance of NP-Max (resp. LogNP-Max). We make the following assumptions on the Turing machine M: let M have two tapes, where tape 1 serves as input and worktape while tape 2 is the dedicated output tape. This means that tape 2 initially has the starting symbol $\triangleright$ in cell 0 and blanks $\sqcup$ in all further tape cells. At time instant 0, the cursor on tape 2 makes a move to the right and leaves $\triangleright$ unchanged. At all later time instants, either the current symbol (i.e., blank $\sqcup$) and the cursor on tape 2 are both left unchanged or the current symbol is overwritten by 0 or 1 and the cursor is moved to the right. Recall that as output alphabet we have $\{0, 1\}$.

Following the standard proof of the Cook-Levin Theorem (see e.g. [20]), one can construct in polynomial time a propositional formula φ such that there is a one-to-one correspondence between the computation paths of the non-deterministic TM M on input x and the satisfying truth assignments of φ. Let N denote the maximum length of a computation path of M on any input of length $|x|$. Then formula φ is built from the following collection of variables:

$\boldsymbol{tape}_i[\tau, \pi, \sigma]$ for $1 \leq i \leq 2$, $0 \leq \tau \leq N$, $0 \leq \pi \leq N$, and $\sigma \in \Sigma$, to express that at time instant τ of the computation, cell number π of tape i contains symbol σ, where Σ denotes the set of all tape symbols of TM M.

$\boldsymbol{cursor}_i[\tau, \pi]$ for $1 \leq i \leq 2$, $0 \leq \tau \leq N$, and $0 \leq \pi \leq N$, to express that at time instant τ, the cursor on tape i points to cell number π.

state$[\tau, s]$ for $0 \leq \tau \leq N$ and $s \in K$, to express that at time instant τ, the NTM T is in state s, where K denotes the set of all states of TM M.

Suppose that m denotes the maximum length of output strings of M on any input of length $|x|$. Then we introduce additional propositional variables $\vec{x} = (x_1, x_1', x_2, x_2', \dots, x_m, x_m')$ and z and we construct the propositional formula $\psi = \varphi \wedge \chi$, where χ is defined as follows:

$$
\begin{aligned}
\chi \quad = & \quad \bigwedge_{\pi=1}^{m} x_\pi \leftrightarrow \mathit{tape}_2[N, \pi, 1] \;\; \wedge \\
\wedge & \quad \bigwedge_{\pi=1}^{m} x'_\pi \leftrightarrow \neg \mathit{tape}_2[N, \pi, \sqcup] \;\; \wedge \\
\wedge & \quad z \leftrightarrow \bigvee_{\pi=1}^{N} \Big(x_\pi \wedge \bigwedge_{\pi'=\pi+1}^{N} \neg x'_{\pi} \Big)
\end{aligned}
$$

In other words, χ makes sure that the variables in $\vec{x}$ encode the output string and z encodes the last bit of the output string, i.e.: for every i, variable x_i is true in a model J of ψ if the i-th bit of the output string (along the computation path of M corresponding to J when considered as a model of φ) is 1. Consequently, truth value false of x_i can mean that, on termination of M, the symbol in the i-th cell of tape 2 is either 0 or $\sqcup$. The latter two cases are distinguished by the truth value of variable x_i', which is false if and only if the symbol in the i-th cell of tape 2 is $\sqcup$. Variable z is true in a model J of ψ if the last bit of the output string is 1. The last bit (in the third line of the above definition of formula χ, this is position π) is recognized by the fact that after it, all cells on tape 2 contain the blank symbol. Note that the truth value of z in any model of ψ is uniquely determined by the truth values of the $\vec{x}$ variables.

Finally, we transform ψ into ψ^* in 3-CNF by some standard transformation (e.g., by the Tseytin transformation [25]). We thus introduce additional propositional variables such that every model of ψ can be extended to a model of ψ^* and every model of ψ^* is also a model of ψ. Now let $\vec{y}$ denote the vector of variables $\mathit{tape}_i[\tau, \pi, \sigma]$, $\mathit{cursor}_i[\tau, \pi]$, and $\mathit{state}[\tau, s]$ plus the additional variables introduced by the transformation into 3-CNF. Let the variables in $\vec{y}$ be arranged in arbitrary order and let ℓ denote the number of variables in $\vec{y}$.

In the reduction from NP-Max to LexMaxSat, we define the following order on all variables in ψ^*: $x_1 > x_1' > \cdots > x_m > x_m' > y_1 > \cdots > y_\ell > z$. In the reduction from LogNP-Max to LogLexMaxSat, we define an order only on part of the variables in ψ^*, namely: $x_1 > x_1' > \cdots > x_m > x_m' > z$ (i.e., we ignore the variables in $\vec{y}$).

We now construct a particular model J of ψ^* for which we will then show that it is in fact the lexicographically maximal model of ψ^*. Let w denote the lexicographically maximal output string produced by M on input x. Every computation path of M corresponds to one or more models of ψ^*. Consider the truth assignment J on the variables $\vec{x}$, $\vec{y}$, and z obtained as follows: J restricted to $\vec{y}$ is chosen such that it is lexicographically maximal among all truth assignments on $\vec{y}$ corresponding to a computation path of M on input x and with output w. Note that the truth assignments of J on $\vec{x}$ and z are uniquely defined by the output w of M (i.e., no matter which concrete truth assignment on $\vec{y}$ to encode a computation path of M with this output we choose).

We claim that J is the lexicographically maximal model of ψ^*. To prove this claim, suppose to the contrary that there exists a lexicographically bigger model J'

of ψ. Then we distinguish 3 cases according to the group of variables where J' is bigger than J:

(1) If J' is bigger than J on $\vec{x}$, then (since the truth values of $\vec{x}$ encode the output string of some computation of M on x) there exists a bigger output than w. This contradicts our assumption that w is maximal. (2) If J' coincides with J on $\vec{x}$ and J' is bigger than J on $\vec{y}$, then J' restricted to $\vec{y}$ corresponds to a computation path producing the same output string w as the computation path encoded by J on $\vec{x}$. This contradicts our choice of truth assignment J on $\vec{y}$. (3) The truth value of z in any model of ψ^* is uniquely determined by the truth value of $\vec{x}$. Hence, it cannot happen that J' and J coincide on $\vec{x}$ but differ on z.

From the correspondence between the lexicographically maximal output string w of M on input x and the lexicographically maximal model J of ψ^*, the correctness of our problem reductions (both, from NP-Max to LexMaxSat and from LogNP-Max to LogLexMaxSat) follows immediately, namely: For the LexMaxSat problem: the last bit in the lexicographically maximal output string w of M on input x is 1 if and only if the truth value of variable z in the lexicographically maximal model J of ψ^* is 1 (i.e., true). Likewise, for the LogLexMaxSat problem: the last bit in the lexicographically maximal output string w of M on input x is 1 if and only if the truth value of variable z in the lexicographically maximal truth assignment on $(x_1, x'_1, \dots, x_m, x'_m, z)$ that can be extended to a model J of ψ^* is 1 (i.e., true). Note that in case of the LogLexMaxSat problem, we may indeed simply ignore the $\vec{y}$ variables because the truth values of $\vec{x}$ and z are uniquely determined by the output w of M – independently of the concrete choice of truth assignments to the $\vec{y}$ variables. □

4 Θ_2P-complete variants of SAT

In this section, we study two natural variants of Sat:

Problem:	CardMinSat
Input:	Propositional formula φ and an atom x_i.
Question:	Is x_i true in a cardinality-minimal model of φ?

Problem:	CardMaxSat
Input:	Propositional formula φ and an atom x_i.
Question:	Is x_i true in a cardinality-maximal model of φ?

Both problems will be shown Θ_2P-complete. We start with hardness for CardMaxSat.

Lemma 3. CardMaxSat *is Θ_2P-hard, even for formulas in 3-CNF.*

Proof. The Θ_2P-hardness of CardMaxSat is shown by reduction from LogLexMaxSat. Consider an arbitrary instance of LogLexMaxSat, which is given by a propositional formula φ and an order $x_1 > \cdots > x_\ell$ over logarithmically many variables from φ. From this we construct an instance of CardMaxSat, which will be defined by a propositional formula ψ and a dedicated variable in ψ, namely x_ℓ. We simulate the lexicographical order over the variables $x_1 > \cdots > x_\ell$ by adding "copies" of each variable x_i, i.e., for every $i \in \{1, \dots, \ell\}$, we introduce $2^{\ell-i} - 1$ new variables $x_i^{(1)}, x_i^{(2)}, \dots, x_i^{(r_i)}$ with $r_i = 2^{\ell-i} - 1$. The formula ψ is now obtained from φ by adding the following subformulas. We add to φ the conjuncts $(x_i \leftrightarrow x_i^{(1)}) \wedge \cdots \wedge (x_i \leftrightarrow x_i^{(r_i)})$. Hence, setting x_i to true in a model of the resulting formula forces us to set all its $2^{\ell-i} - 1$ "copies" to true. Finally, for the remaining variables $x_{\ell+1}, \dots, x_n$ in φ, we add "copies" $x'_{\ell+1}, \dots, x'_n$ and further extend φ by the conjuncts $(x_{\ell+1} \leftrightarrow \neg x'_{\ell+1}) \wedge \cdots \wedge (x_n \leftrightarrow \neg x'_n)$ to make the cardinality of models indistinguishable on these variables.

Since $\ell \leq \log|\varphi|$, this transformation of φ into ψ is feasible in polynomial time. Also note that ψ is in 3-CNF, whenever φ is in 3-CNF. We claim that our problem reduction is correct, i.e., let $\vec{b} = (b_1, \ldots, b_\ell)$ denote the lexicographically maximal vector that can be extended to a model of φ. We claim that x_ℓ is true in a model I of φ, s.t. I is an extension of $\vec{b}$ iff x_ℓ is true in a cardinality-maximal model of ψ. In fact, even a slightly stronger result can be shown, namely: if a model of φ is an extension of $\vec{b}$, then I can be further extended to a cardinality maximal model J of ψ. Conversely, if J is a cardinality maximal model of ψ, then J is also a model of φ and J extends $\vec{b}$. $\square$

Our ultimate goal in this section is to show that the Θ_2P-completeness of CardMaxSat and also of CardMinSat hold even if restricted to the Krom case. In a first step, we reduce the CardMaxSat problem to a variant of the Independent Set problem, which we define next. Note that the standard reduction from 3SAT to Independent Set [20] is not sufficient: suppose we are starting off with a propositional formula φ with K clauses. Then, if φ is satisfiable, every maximum independent set selects precisely K vertices. Hence, additional work is needed to preserve the information on the cardinality of the models of φ. The variant of the Independent Set problem we require here, is as follows: For the sake of readability, we first explicitly introduce a lower bound on the independent sets of interest and then we show how to encode this lower bound implicitly.

Problem:	MaxIndependentSet
Input:	Undirected graph $G = (V, E)$, vertex $v \in V$, and positive integer K.
Question:	Is v in a maximum independent set I of G with $|I| \geq K$?

Lemma 4. *The* MaxIndependentSet *problem is* Θ_2P*-hard.*

Proof. We extend the standard reduction from 3SAT to Independent Set [20] to a reduction from CardMaxSat to MaxIndependentSet. The result then follows from Lemma 3.

Let (φ, x) denote an arbitrary instance of CardMaxSat with $\varphi = c_1 \wedge \cdots \wedge c_m$, s.t. each clause c_i is of the form $c_i = l_{i1} \vee l_{i2} \vee l_{i3}$ where each l_{ij} is a literal. Let $X = \{x_1, \ldots, x_n\}$ denote the set of variables in φ. We construct an instance (G, v, K) of MaxIndependentSet where $K := L * m$ with L "sufficiently large", e.g., $L := n+1$; G consists of $3*K+n$ vertices $V = \{l^{(j)}_{i1}, l^{(j)}_{i2}, l^{(j)}_{i3}\} \mid 1 \leq i \leq m$ and $1 \leq j \leq L\} \cup \{u_1, \ldots, u_n\}$; and $v := u_i$, for $x = x_i$. It remains to specify the edges E of G:

(1) For every $i \in \{1, \ldots, m\}$ and every $j \in \{1, \ldots, L\}$, E contains edges $[l^{(j)}_{i1}, l^{(j)}_{i2}]$, $[l^{(j)}_{i1}, l^{(j)}_{i3}]$, and $[l^{(j)}_{i2}, l^{(j)}_{i3}]$, i.e., G contains L triangles $\{l^{(j)}_{i1}, l^{(j)}_{i2}, l^{(j)}_{i3}\}$ with $j \in \{1, \ldots, L\}$.

(2) For every $\alpha, \beta, \gamma, \delta$, s.t. $l_{\alpha\beta}$ and $l_{\gamma\delta}$ are complementary literals, E contains L edges $[l^{(i)}_{\alpha\beta}, l^{(j)}_{\gamma\delta}]$ with $i, j \in \{1, \ldots, L\}$.

(3) For every $i \in \{1, \ldots, n\}$ and every α, β, if $l_{\alpha\beta}$ is of the form $\neg x_i$, then E contains L edges $[l^{(j)}_{\alpha\beta}, u_i]$ $(j \in \{1, \ldots, L\})$.

The intuition of this reduction is as follows: The $L * m$ triangles $\{l^{(j)}_{i1}, l^{(j)}_{i2}, l^{(j)}_{i3}\}$ correspond to L copies of the standard reduction from 3SAT to Independent Set [20]. Likewise, the edges between complementary literals are part of this standard reduction. It is easy to verify that for every independent set I with $|I| \geq K$, there also exists an independent set I' with $|I'| \geq |I|$, s.t. I' chooses from every copy of a triangle the "same" endpoint, i.e., for every $i \in \{1, \ldots, m\}$, if $l^{(\beta)}_{i\alpha} \in I'$ and $l^{(\delta)}_{i\gamma} \in I'$, then $\alpha = \gamma$.

Since $L = n+1$, the desired lower bound $K = L * m$ on the independent set can only be achieved if exactly one vertex is chosen from each triangle. We thus get the usual correspondence between models of φ and independent sets of G of size $\geq K$. Note that this correspondence leaves some choice for those variables x where the independent set contains no vertex corresponding to the literal $\neg x$. In such a case, we assume that the variable x is set to true since in CardMaxSat, our goal is to maximize the number of variables set to true. Then a vertex u_i may be added to an independent set I with $|I| \geq K$, only if no vertex $l^{(j)}_{\alpha\beta}$ corresponding to a literal $l_{\alpha\beta}$ of the form $\neg x_i$ has been chosen into the independent set. Hence, $u_i \in I$ if and only if x_i is to true in the corresponding model of φ. □

Theorem 5. CardMaxSat *(resp.* CardMinSat*) is Θ_2P-complete even when formulas are restricted to Krom and moreover the clauses consist of negative (resp. positive) literals only.*

Proof. Membership proceeds by the classical binary search [20] for finding the optimum, asking questions like "Does there exist a model of size at least k?" or "Does there exist a model of size at most k?". With logarithmically many such calls to an NP-oracle, the maximal (resp. minimal) size M of all models of φ can thus be computed, and we check in a final question to an NP-oracle if x_i is true in a model of size M.

For hardness, we start with the case of CardMaxSat. To this end, we first reduce MaxIndependentSet to the following intermediate problem: Given an undirected graph $G = (V, E)$ and vertex $v \in V$, is v in a maximum independent set I of G? The fact that this intermediate problem reduces to CardMaxSat then follows by expressing this problem via a Krom formula with propositional variables V and clauses $\neg v_i \vee \neg v_j$ for every edge $[v_i, v_j]$ in E. Hence, we obtain hardness also for the case of Krom formulas with negative literals only.

Hence, let us turn to the first reduction and consider an arbitrary instance (G, v, K) of MaxIndependentSet with $G = (V, E)$ and $v \in V$. We define the corresponding instance (H, v) of the intermediate problem with $H = (V', E')$ where $V' = V \cup U$ with $U = \{u_1, \dots, u_K\}$ for fresh vertices u_i, and $E' = E \cup \{[u_i, v_j] \mid 1 \leq i \leq K, v_j \in V\}$. The additional edges in E' make sure that an independent set of H contains either only vertices from V or only vertices from U. Clearly, H contains the independent set U with $|U| = K$. This shows Θ_2P-hardness of CardMaxSat.

To show the Θ_2P-hardness result for CardMinSat restricted to Krom and positive literals, we now can give a reduction from CardMaxSat (restricted to Krom and negative literals). Given, such a CardMaxSat instance (φ, x) we construct $(\hat{\varphi}, x)$ where $\hat{\varphi}$ is given as

$$\bigwedge_{x \in \mathrm{var}(\varphi)} \big((x \vee x') \wedge (x \vee x'')\big) \wedge$$
$$\bigwedge_{(\neg x \vee \neg y) \in \varphi} \big((x' \vee y') \wedge (x'' \vee y') \wedge (x' \vee y'') \wedge (x'' \vee y'')\big).$$

The intuition of variables x', x'' is to represent that x is assigned to false. We have the following observations: (1) a cardinality-minimal model of $\hat{\varphi}$ either sets x or jointly, x' and x'', to true (for each variable $x \in \mathrm{var}(\varphi)$); i.e. it is of the form $\tau(I) := I \cup \{x', x'' \mid x \in \mathrm{var}(\varphi) \setminus I\}$ for some $I \subseteq \mathrm{var}(\varphi)$; (2) for each $I \subseteq \mathrm{var}(\varphi)$, it holds that I is a model of φ iff $\tau(I)$ is a model of $\hat{\varphi}$; (3) for each $I, J \subseteq \mathrm{var}(\varphi)$, $|I| \leq |J|$ iff $|\tau(I)| \geq |\tau(J)|$. It follows that $(\hat{\varphi}, x)$ is a yes-instance of CardMinSat iff (φ, x) is a yes-instance of CardMaxSat. □

5 Applications: Belief Revision and Abduction

In this section, we make use of the hardness results for CardMinSat in the previous section, in order to show novel complexity results for problems from the field of knowledge representation when restricted to the Krom fragment and the combined Horn-Krom fragment (i.e. Krom formulas with at most one positive literal per clause).

Belief Revision. Belief revision aims at incorporating a new belief, while changing as little as possible of the original beliefs. We assume that a belief set is given by a propositional formula ψ and that the new belief is given by a propositional formula μ. Revising ψ by μ amounts to restricting the set of models of μ to those models which are "closest" to the models of ψ. Several *revision operators* have been proposed which differ in how they define what "closest" means. Here, we focus on the revision operators due to Dalal [7] and Satoh [22].

Dalal's operator measures the distance between the models of ψ and μ in terms of the *cardinality of model change*, i.e., let M and M' be two interpretations and let $M\Delta M'$ denote the symmetric difference between M and M', i.e., $M\Delta M' = (M\backslash M')\cup(M'\backslash M)$. Further, let $|\Delta|^{min}(\psi,\mu)$ denote the minimum number of propositional variables on which the models of ψ and μ differ. We define $|\Delta|^{min}(\psi,\mu) := \min\{|M\Delta M'| : M \in \mathrm{mod}(\psi), M' \in \mathrm{mod}(\mu)\}$, where $\mathrm{mod}(\cdot)$ denotes the models of a formula. Dalal's operator is now given as: $\mathrm{mod}(\psi \circ_D \mu) = \{M \in \mathrm{mod}(\mu) : \exists M' \in \mathrm{mod}(\psi)\ s.t.\ |M\Delta M'| = |\Delta|^{min}(\psi,\mu)\}$.

Satoh's operator interprets the minimal change in terms of *set inclusion*. Thus let $\Delta^{min}(\psi,\mu) = min_{\subseteq}(\{M\Delta M' : M \in \mathrm{mod}(\psi), M' \in \mathrm{mod}(\mu)\})$, where the operator $min_{\subseteq}(\cdot)$ applied to a set S of sets selects those elements $s \in S$, s.t. S contains no proper subset of s. Then we define Satoh's operator as $\mathrm{mod}(\psi \circ_S \mu) = \{M \in \mathrm{mod}(\mu) : \exists M' \in \mathrm{mod}(\psi)\ s.t.\ M\Delta M' \in \Delta^{min}(\psi,\mu)\}$.

Let $\circ_r$ denote a revision operator with $r \in \{D, S\}$. We analyse two well-studied problems in belief revision.

Problem:	BR-Implication
Input:	Propositional formulas ψ and μ, and an atom x.
Question:	Does $\psi \circ_r \mu \models x$ hold?

Problem:	BR-Model Checking
Input:	Propositional formulas ψ and μ, and model M of μ.
Question:	Does $M \models \psi \circ_r \mu$ hold?

The complexity of these problems has been intensively studied [8, 17]. For arbitrary propositional formulas ψ and μ, both problems are Θ_2P-complete for Dalal's revision operator. For Satoh's revision operator, BR-Model Checking (resp. BR-Implication) is Σ_2P-complete (resp. Π_2P-complete). Both [8] and [17] have also investigated the complexity of some restricted cases (e.g., when the formulas are restricted to Horn). Below, we pinpoint the complexity of these problems in the Krom case. Our hardness results will subsume known hardness results for Horn case as well.

Theorem 6. BR-Implication *and* BR-Model Checking *are* Θ_2P*-complete for Dalal's revision operator even if the formulas* ψ *and* μ *are restricted to Krom and Horn form.*

Proof. The membership even holds without any restriction [8, 17]. The hardness is shown as follows: Given an instance (φ, x_0) of CardMinSat where $X =$

$\{x_0, x_1, \ldots, x_n\}$ is the set of variables in φ. By Theorem 5, this problem is Θ_2P-complete even if φ is in Krom form with positive literals only. We define the following instances of BR-Implication and BR-Model Checking:

Let $\mathcal{C} = \{c_1, \ldots, c_m\}$ denote the set of clauses in φ. Let $X = \{x_0, \ldots, x_n\}$ be a set of variables and let y, z be two further variables. We define ψ and μ as:

$$\psi = \Big(\bigwedge_{(x_i \vee x_j) \in \mathcal{C}} (\neg x_i \vee \neg x_j)\Big) \wedge y$$

$$\mu = \Big(\bigwedge_{i=1}^{n} x_i \Big) \wedge (x_0 \vee \neg y) \wedge (\neg x_0 \vee y)$$

Obviously, both ψ and μ are from the Horn-Krom fragment and can be constructed efficiently from (φ, x_0).

Note that μ has exactly two models $M = M_1 = \{x_1, \ldots, x_n\}$ and $M_2 = \{x_1, \ldots, x_n, x_0, y\}$. Let k denote the size of a minimum model of φ. We claim that x_0 is contained in a minimum model of φ iff $M \models \psi \circ_D \mu$ iff $\psi \circ_D \mu \not\models y$. The second equivalence is obvious. We prove both directions of the first equivalence separately.

First, suppose that φ has a minimum model N with $x_0 \in N$. Then the minimum distance of M_1 from models of ψ is k, which is witnessed by the model $I = X \setminus N \cup \{y\}$ of ψ (note that $x_0 \notin M_1$). It can be easily verified that M_2 does not have smaller distance from any model of ψ. At best, M_2 also has distance k, namely from any model I_2 of ψ of the form $I_2 = (X \setminus N') \cup \{y\}$, s.t. N' is a minimum model of φ.

Now suppose that every model N of φ with $x_0 \in N$ has size $\geq k+1$. Then the distance of M_1 from any model I_1 of ψ is $k+1$, since $M_1 \Delta I_1$ contains y and at least k elements from $\{x_1, \ldots, x_n\}$. On the other hand, the distance of M_2 from models of ψ is k witnessed by any model $I_2 = (X \setminus N') \cup \{y\}$ of ψ where N' is a minimum model of φ.

In summary, we have thus shown that $M_2 \models \psi \circ_D \mu$ is guaranteed to hold. But $M_1 \models \psi \circ_D \mu$ holds if and only if there exists a minimum model N of φ with $x_0 \in N$. □

Compared to the Θ_2P-hardness proof for Dalal's revision operator in the Horn case [8, 17], our construction is much simpler, thanks to the previously established hardness result for CardMinSat. Moreover, it is not clear how the constructions from [8, 17] can be adapted to the Horn-Krom case.

The above theorem states that, for Dalal's revision operator, the complexity of the BR-Implication and BR-Model Checking problem does not decrease even if the formulas are restricted to Krom and Horn form. In contrast, we shall show below that for Satoh's revision, the complexity of BR-Implication and BR-Model Checking drops one level in the polynomial hierarchy if ψ and μ are Krom. Hence, below, both the membership in case of Krom form and the hardness (which even holds for the restriction to Horn and Krom form) need to be proved. Also here, our hardness reductions differ substantially from those in [8, 17].

Theorem 7. BR-Implication *is* coNP*-complete and* BR-Model Checking *is* NP*-complete for Satoh's operator if the formulas ψ and μ are restricted to Krom. Hardness holds even if ψ and μ are further restricted to Krom and Horn.*

Proof. For the membership proofs, recall the coNP-membership and NP-membership proof of BR-Implication and BR-Model Checking for the Horn case in [8], Theorem 7.2; resp. [17], Theorem 20. The key idea there is that, given a model I of ψ and a model M of μ the subset-minimality of $I \Delta M$ can be tested in polynomial time by reducing this problem to a SAT problem involving the formulas ψ and μ. The same idea holds for Krom form.

The crucial observation there is that, given a model I of ψ and a model M of μ, one can check efficiently whether there exists a model J of ψ and a model N of μ with $J\Delta N \subset I\Delta M$. Indeed, let $Var(\psi) \cup Var(\mu) = \{x_1, \dots, x_n\}$ and let $\{y_1, \dots, y_n\}$ be fresh, pairwise distinct variables. Then a model J of ψ and a model N of μ with $J\Delta N \subset I\Delta M$ exist iff for some variable $x_j \in I\Delta M$, the following propositional formula is satisfiable:

$$\psi[x/y] \wedge \mu \wedge (y_j \leftrightarrow x_j) \wedge \bigwedge_{x_i \notin I\Delta M} (y_i \leftrightarrow x_i)$$

Here, $\psi[x/y]$ denotes the formula that we obtain from ψ by replacing every x_i by y_i. Clearly, if ψ and μ are Horn (and, likewise, if they are Krom), then this satisfiability check is feasible in polynomial time.

The hardness is shown by the following reduction from 3SAT respectively co-3SAT: Let φ be an arbitrary Boolean formula in 3-CNF over the variables $X = \{x_1, \dots, x_n\}$, i.e., $\varphi = c_1 \wedge \cdots \wedge c_m$, s.t. each clause c_i is of the form $c_i = l_{i1} \vee l_{i2} \vee l_{i3}$, where the l_{ij}'s are literals over X. Let $Y = \{y_1, \dots, y_n\}$, $A = \{a_1, \dots, a_m\}$, $B = \{b_1, \dots, b_m\}$, and $\{d\}$ be sets of fresh, pairwise distinct propositional variables. We define ψ and μ as follows:

$$\begin{aligned}
\psi &= \big(\bigwedge_{i=1}^{n} (\neg x_i \vee \neg y_i)\big) \wedge \big(\bigwedge_{j=1}^{m} \bigwedge_{k=1}^{3} (l^*_{jk} \rightarrow \neg a_j)\big) \wedge \\
&\quad \big(\bigwedge_{j=1}^{m} (a_j \leftrightarrow b_j)\big) \\
\mu &= \big(\bigwedge_{i=1}^{n} (x_i \wedge y_i)\big) \wedge \big(\bigwedge_{j=1}^{m} a_j\big) \wedge \big(\bigwedge_{j=1}^{m} (b_j \rightarrow d)\big)
\end{aligned}$$

where we set $l^*_{jk} = x_\alpha$ if $l_{jk} = x_\alpha$ for some $\alpha \in \{1, \dots, n\}$ and $l^*_{jk} = y_\alpha$ if $l_{jk} = \neg x_\alpha$. Finally, we define $M = X \cup Y \cup A$. Both ψ and μ are from the Horn-Krom fragment and can be constructed efficiently from φ. We claim that φ is satisfiable iff $M \models \psi \circ_S \mu$ iff $\psi \circ_S \mu \not\models d$.

Clearly, every model of μ must set the variables in $X \cup Y \cup A$ to true. The truth value of the variables in B may be chosen arbitrarily. However, as soon as at least one $b_j \in B$ is set to true, we must set d to true due to the last conjunct in μ. Hence, M is the only model of μ where d is set to false. From this, the second equivalence above follows. Below we sketch the proof of the first equivalence.

First, suppose that φ is satisfiable and let G be a model of φ. We set $I := \{x_i \mid x_i \in G\} \cup \{y_i \mid x_i \notin G\}$. Clearly, I is a model of ψ. We claim that I is a witness for $M \models \psi \circ_S \mu$, i.e., $I\Delta M$ is minimal. To prove this claim, let J be a model of ψ and N a model of μ, s.t. $J\Delta N \subseteq I\Delta M$. It suffices to show that then $J\Delta N = I\Delta M$ holds. To this end, we compare I and J first on $X \cup Y$ then on A and finally on $B \cup \{d\}$: By the first group of conjuncts in ψ, we may conclude from $J\Delta N \subseteq I\Delta M$ that I and J coincide on $X \cup Y$. In particular, both I and J are models of φ. But then they also coincide on A since the second group of conjuncts in ψ enforces $I(a_j) = J(a_j) =$ false for every j. Note that $(I\Delta M) \cap (B \cup \{d\}) = \emptyset$. Hence, no matter how we choose J and N on $B \cup \{d\}$, we have $I\Delta M \subseteq J\Delta N$.

Now suppose that φ is unsatisfiable. Let I be a model of ψ. To prove $M \not\models \psi \circ_S \mu$, we show that $I\Delta M$ cannot be minimal. By the unsatisfiability of φ, we know that I (restricted to X) is not a model of φ. Hence, there exists at least one clause c_j which is false in I. The proof goes by constructing J, N with $J\Delta N \subset I\Delta M$. The crucial observation is that the symmetric difference $I\Delta M$ can be decreased by setting $J(a_j) = J(b_j) = N(b_j) =$ true and $N(d) =$ true. □

Abduction. Abduction is used to produce explanations for some observed manifestations. Therefore one of its primary fields of application is diagnosis. A *propositional abduction problem* (PAP) $\mathcal{P}$ consists of a tuple $\langle V, H, M, T\rangle$, where V is a finite set of *variables*, $H \subseteq V$ is the set of *hypotheses*, $M \subseteq V$ is the set of *manifestations*, and T is a consistent *theory* in the form of a propositional formula. A set $\mathcal{S} \subseteq H$ is a *solution* (also called *explanation*) to $\mathcal{P}$ if $T \cup \mathcal{S}$ is consistent and $T \cup \mathcal{S} \models M$ holds. A *system diagnosis problem* can be represented by a PAP $\mathcal{P} = \langle V, H, M, T\rangle$ as follows. The theory T is the system description. The hypotheses $H \subseteq V$ describe the possibly faulty system components. The manifestations $M \subseteq V$ are the observed symptoms, describing the malfunction of the system. The solutions $\mathcal{S}$ of $\mathcal{P}$ are the possible explanations of the malfunction.

Often, one is not interested in *any* solution of a given PAP $\mathcal{P}$ but only in *minimal* solutions, where minimality is defined w.r.t. some preorder $\preceq$ on the powerset 2^H. Two natural preorders are set-inclusion $\subseteq$ and smaller cardinality denoted as $\leq$. Note that allowing *any* solution corresponds to choosing "=" as the preorder. In [6] a trichotomy (of Σ_2P-completeness, NP-completeness, and tractability) has been proved for the Solvability problem of propositional abduction, i.e., deciding if a PAP $\mathcal{P}$ has at least one solution (the preorder "=" has thus been considered). Nordh and Zanuttini [18] have identified many further restrictions which make the Solvability problem tractable. All of the above mentioned preorders have been systematically investigated in [9]. A study of the counting complexity of abduction with these preorders has been carried out by Hermann and Pichler [13]. Of course, if a PAP has *any* solution than it also has a $\preceq$-minimal solution. Hence, the preorder is only of interest for problems like the following one:

Problem:	$\preceq$-Relevance
Input:	PAP $\mathcal{P} = \langle V, H, M, T\rangle$ and hypothesis $h \in H$.
Question:	Is h relevant, i.e., does $\mathcal{P}$ have a $\preceq$-minimal solution $\mathcal{S}$ with $h \in \mathcal{S}$?

Known results [9] are as follows: The $\preceq$-Relevance problem is Σ_2P-complete for preorders = and $\subseteq$ and Θ_3P-complete for preorder $\leq$. Moreover, the complexity drops by one level in the polynomial hierarchy if the theory is restricted to Horn. In [6], the Krom case was considered for the preorder =. For the preorder $\subseteq$, the Krom case was implicitly settled in [9]. Indeed, an inspection of the NP-hardness proof of the $\preceq$-Relevance problem in the Horn case reveals that NP-hardness holds even if the theory is simultaneously restricted to Horn *and* Krom (see the proof of Theorem 5.2 in [9]). Below we show that also for the preorder $\leq$, the complexity in the Krom case matches the Horn case.

Theorem 8. *Suppose that we only consider PAPs $\mathcal{P} = \langle V, H, M, T\rangle$ where the theory T is Krom. Then the $\preceq$-Relevance problem for preorder $\leq$ is Θ_2P-complete. Hardness holds even if the theory is restricted simultaneously to Horn and Krom.*

Proof. The membership proof is analogous to the corresponding one in [9] for the general case. The decrease of complexity compared with arbitrary theories is due to the tractability of satisfiability testing in the Krom case. Θ_2P-hardness is shown by the following problem reduction from CardMinSat. Consider an arbitrary instance (φ, x_i) of CardMinSat. By Theorem 5, we may assume that φ is in positive Krom form. Let $\varphi = (p_1 \vee q_1) \wedge \cdots \wedge (p_m \vee q_m)$ over variables $X = \{x_1, \ldots, x_n\}$ and let $G = \{g_1, \ldots, g_m\}$ be a set of fresh, pairwise distinct variables. We define the PAP

$\mathcal{P} = \langle V, H, M, T \rangle$ as follows:

$$\begin{array}{rcl} V & = & X \cup G \\ H & = & X \\ M & = & G \\ T & = & \{p_i \to g_i \mid 1 \le i \le m\} \cup \{q_i \to g_i \mid 1 \le i \le m\} \end{array}$$

It is easy to verify that the models of φ coincide with the solutions of $\mathcal{P}$. Hence, x_i is in a minimum model of φ iff x_i is in a minimum solution of $\mathcal{P}$. $\square$

We note that our Θ_2P-hardness reduction for $\le$-RELEVANCE is much easier than the Θ_2P-hardness reduction in [9] for the Horn case. (In fact, the latter reduction maps a certain MAXSAT problem to abduction, and it is not immediate how this reduction can be adapted to work in the combined Horn-Krom case.) Again the reason why our reduction is quite simple relies on the fact that we can start from the more closely related problem CARDMINSAT for Krom.

6 Complete Classification of CARDMINSAT

Since neither the full Krom fragment nor even the Krom *and* (dual) Horn fragment makes the problems investigated tractable (especially the ones that are Θ_2P-complete) it is worth making a step further and studying how much we have to restrict the syntactic form of the Krom formulas to decrease the complexity. A key for such tractability results is to go through a more fine-grained complexity study of CARDMINSAT. To this aim we propose to investigate this problem within Schaefer's framework that we introduce next.

A *logical relation* (or a *Boolean relation*) of arity k is a relation $R \subseteq \{0,1\}^k$. We will refer below to the following binary relation, $\mathrm{Or}_2 = \{(0,1),(1,0),(1,1)\}$. By abuse of notation we do not make a difference between a relation and its predicate symbol. A *constraint* (or *constraint application*), C, is a formula $R(x_1,\dots,x_k)$, where R is a logical relation of arity k and $x_1,\dots,x_k$ are (not necessarily distinct) variables. If u and v are two variables, then $C[u/v]$ denotes the constraint obtained from C in replacing each occurrence of v by u. If V is a set of variables, then $C[u/V]$ denotes the result of substituting u to every occurrence of every variable of V in C. An assignment I of truth values to the variables *satisfies* the constraint if $\big(I(x_1),\dots,I(x_k)\big) \in R$. A *constraint language* Γ is a finite set of logical relations. A Γ-*formula*, φ, is a conjunction of constraint applications using only logical relations from Γ, and hence is a quantifier-free first-order formula. For single-element constraint languages $\{R\}$, we often omit parenthesis and speak about R-formulas instead of $\{R\}$-formulas. With $\mathrm{var}(\varphi)$ we denote the set of variables appearing in φ. A Γ-formula φ is satisfied by a truth assignment I if I satisfies all the constraints in φ, such an I is then a *model* of φ, its *cardinality* refers to the number of variables assigned 1. We say that two quantifier-free first-order formulas φ and ψ are equivalent ($\varphi \equiv \psi$) if they have the same sets of variables and of satisfying assignments. Assuming a canonical order on the variables, we can regard assignments as tuples in the obvious way, and say that a quantifier-free first-order formula *defines* or *implements* the logical relation of its models. For instance the binary clause $(x_1 \vee x_2)$ defines the relation Or_2. This notion can be naturally extended to existentially-quantified formulas in considering their free variables. For instance the formula $\exists y (x_1 \vee y) \wedge (x_2 \vee \neg y)$ defines (or implements) the relation Or_2 as well.

Throughout the text we refer to different types of Boolean relations following Schaefer's terminology [23]. We say that a Boolean relation R is *Horn* (resp. *dual Horn*) if R can be defined by a CNF formula that is Horn (resp. dual Horn). A relation R is *Krom* if it can be defined by a 2-CNF formula. A relation R is *affine* if it can be defined by an *affine* formula, i.e., conjunctions of XOR-clauses (consisting of an XOR of some variables plus maybe the constant 1) — such a formula may also be seen as a system of linear equations over GF[2]. A relation is *width*-2 *affine* if it is definable by a conjunction of clauses, each of them being either a unary clause or a 2-XOR-clause (consisting of an XOR of two variables plus maybe the constant 1) — such a conjunctive formula may also be seen as a set of a conjunction of equalities and disequalities between pairs of variables. A relation R of arity k is *0-valid* (resp., *1-valid*) if $0^k \in R$ (resp., $1^k \in R$). Finally, a constraint language Γ is Horn (resp. dual Horn, Krom, affine, width-2 affine, 0-valid, 1-valid) if every relation in Γ is Horn (resp. dual Horn, Krom, affine, width-2 affine). We say that a constraint language is *Schaefer* if Γ is either Horn, dual Horn, Krom, or affine.

The complexity study of the satisfiability of Γ-formulas, $\textsc{Sat}(\Gamma)$, started in 1978 in the seminal work of Schaefer. He proved a famous dichotomy theorem: $\textsc{Sat}(\Gamma)$ is in P if Γ is either Schaefer, 0-valid or 1-valid, and NP-complete otherwise. We study here the following problem.

Problem:	$\textsc{CardMinSat}(\Gamma)$
Input:	Γ-formula φ and atom x.
Question:	Is x true in a cardinality-minimal model of φ?

Our main result in this section is the following complete classification within the Krom fragment.

Theorem 9. *Let Γ be a Krom constraint language. If Γ is width-2 affine or Horn, then $\textsc{CardMinSat}(\Gamma)$ is decidable in polynomial time, otherwise it is Θ_2P-complete.*

The following proposition covers the polynomial cases of Theorem 9.

Proposition 10. *Let Γ be a set of logical relations. If Γ is width-2 affine or Horn, then $\textsc{CardMinSat}(\Gamma)$ is decidable in polynomial time.*

Proof. Let φ be a Γ-formula. Suppose first that Γ is Horn. In this case φ can be written as a Horn formula. The unique minimal model of φ can be found in polynomial time by unit propagation.

Suppose now that Γ is width-2 affine. Without loss of generality we can suppose that φ does not contain unitary clauses. Then each clause of φ expresses either the equality or the disequality between two variables. Using the transitivity of the equality relation and the fact that in the Boolean case $a \neq b \neq c$ implies $a = c$, we can identify equivalence classes of variables such that each two classes are either independent or they must have contrary truth values. We call a pair (A, B) of classes with contrary truth values *cluster*, B may be empty. It follows easily that any two clusters are independent and thus to obtain a model of φ, we choose for each cluster (A, B) either $A = 1, B = 0$ or $A = 0, B = 1$. We suppose in the following that φ is satisfiable (otherwise, we will detect a contradiction while constructing the clusters). The weight contribution of each cluster to a model is either $|A|$ or $|B|$. It is then enough to consider the cluster (A, B) that contains the atom x_i. If $x_i \in A$ and $|A| \leq |B|$, then x_i belongs to a cardinality-minimal model of φ, else it does not. $\square$

To complete the proof of Theorem 9 it remains to prove Θ_2P-hardness for the remaining cases. These hardness results rely on the application of tools from universal algebra (see, e.g. [12, 14, 24, 19]), which we now introduce.

Let us first recall a well-known closure operator on sets of logical relations.

Definition 11. *Let Γ be a set of logical relations. Then, $\langle\Gamma\rangle$ is the set of relations that can be defined by a formula of the form $\exists X\varphi$, such that φ is a $(\Gamma\cup\{=\})$-formula and $X \subseteq \mathrm{var}(\varphi)$.*

This closure operator is relevant in order to obtain complexity results for the satisfiability problem. Indeed, Assume that $\Gamma_1 \subseteq \langle\Gamma_2\rangle$, then a Γ_1-formula can be transformed into a satisfiability-equivalent Γ_2-formula, thus showing that $\textsc{Sat}(\Gamma_1)$ can be reduced in polynomial time to $\textsc{Sat}(\Gamma_2)$ (see [14]). Hence the complexity of $\textsc{Sat}(\Gamma)$ depends only on $\langle\Gamma\rangle$.

The set $\langle\Gamma\rangle$ is a relational clone (or *co-clone*). Accordingly, in order to obtain a full complexity classification for the satisfiability problem we only have to study the co-clones.

Interestingly, there exists a Galois correspondence between the lattice of Boolean relations (co-clones) and the lattice of Boolean functions (clones) (see [12, 2]). This correspondence is established through the operators Pol(.) and Inv(.) defined below.

Definition 12. Let $f\colon \{0,1\}^m \to \{0,1\}$ and $R \subseteq \{0,1\}^n$. We say that f is a *polymorphism* of R, if for all $x_1, \ldots, x_m \in R$, where $x_i = (x_i[1], x_i[2], \ldots, x_i[n])$, we have $\big(f\big(x_1[1], \cdots, x_m[1]\big), f\big(x_1[2], \cdots, x_m[2]\big), \ldots, f\big(x_1[n], \cdots, x_m[n]\big)\big) \in R$.

For a set of relations Γ we write $\mathrm{Pol}(\Gamma)$ to denote the set of all polymorphisms of Γ, i.e., the set of all Boolean functions that preserve every relation in Γ. For every Γ, $\mathrm{Pol}(\Gamma)$ is a *clone*, i.e., a set of Boolean functions that contains all projections and is closed under composition. As shown first in [12, 2] the operators Pol(.) – Inv(.) constitute a Galois correspondence between the lattice of sets of Boolean relations and the lattice of sets of Boolean functions. In particular for every set Γ of Boolean relations $\mathrm{Inv}(\mathrm{Pol}(\Gamma)) = \langle\Gamma\rangle$, and there is a one-to-one correspondence between clones and co-clones. Hence we may compile a full list of co-clones from the list of clones obtained by Emil Post in [21]. The list of all Boolean clones with finite bases can be found e.g. in [3]. A compilation of all co-clones with finite bases is given in [4]. In the following, when discussing about bases for clones or co-clones we implicitly refer to these two lists. Figure 1 provides a representation of the inclusion structure of the clones, and hence also of the co-clones. For two clones $\mathcal{C}_1$ and $\mathcal{C}_2$, it holds that $\mathrm{Inv}(\mathcal{C}_1) \subseteq \mathrm{Inv}(\mathcal{C}_2)$ if and only if $\mathcal{C}_2 \subseteq \mathcal{C}_1$.

There exist easy criteria to determine if a given relation is Horn, dual Horn, Krom, affine or width-2 affine. Indeed all these classes can be characterized by their polymorphisms (see e.g. [5] for a detailed description). We recall some of these properties here briefly for completeness. The operations of conjunction, disjunction, addition and majority applied on k-ary Boolean vectors are applied coordinate-wise.

- R is Horn if and only if $m, m' \in R$ implies $m \wedge m' \in R$.
- R is dual Horn if and only if $m, m' \in R$ implies $m \vee m' \in R$.
- R is affine if and only if $m, m', m'' \in R$ implies $m \oplus m' \oplus m'' \in R$.
- R is Krom if and only if $m, m', m'' \in R$ implies $Majority(m, m', m'') \in R$.
- R is width-2 affine if and only if $m, m', m'' \in R$ implies both $m \oplus m' \oplus m'' \in R$ and $majority(m, m', m'') \in R$.

In terms of clones, given a set Γ of logical relations, this corresponds to the following:

- Γ is Horn if and only if $\Gamma \subseteq \mathrm{Inv}(\mathrm{E}_2)$.
- Γ is dual Horn if and only if $\Gamma \subseteq \mathrm{Inv}(\mathrm{V}_2)$.

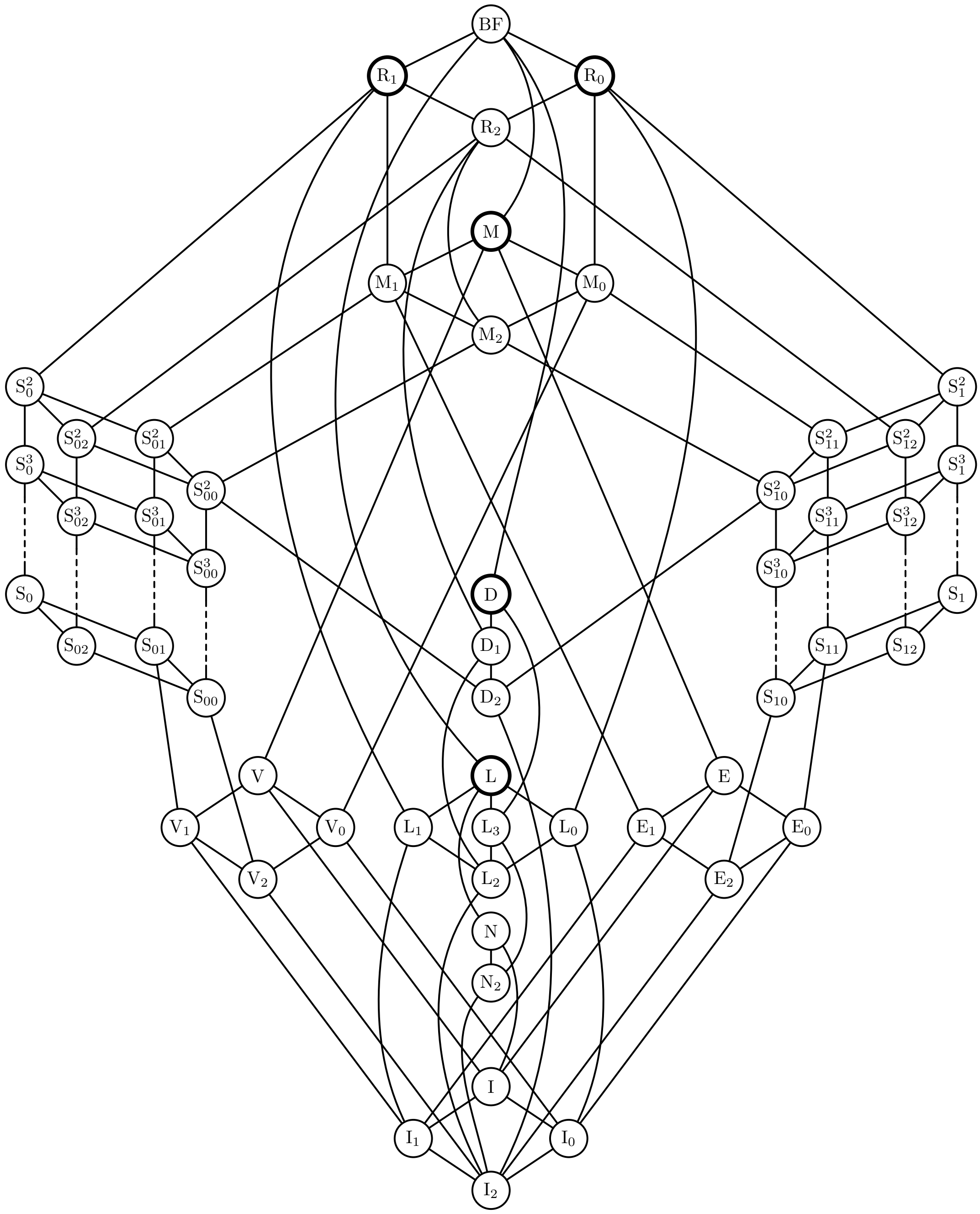

Figure 1: Lattice of all Boolean clones

- Γ is affine if and only if $\Gamma \subseteq \mathrm{Inv}(\mathrm{L}_2)$.
- Γ is Krom if and only if $\Gamma \subseteq \mathrm{Inv}(\mathrm{D}_2)$.
- Γ is width-2 affine if and only if $\Gamma \subseteq \mathrm{Inv}(\mathrm{D}_1)$.

Unfortunately, since we are here interested in cardinality-minimal models, it seems difficult to use the Galois connection explained above. Indeed, existential variables and equality constraints that may occur when transforming a Γ_1-formula into a satisfiability-equivalent Γ_2-formula are problematic, as they can change the set of models and the cardinality of each model. Therefore we will use two other notions of closure, which are more restricted.

First we need to introduce the notion of frozen variable in a formula (resp. in a relation).

Definition 13. *Let φ be a formula and $x \in \mathrm{var}(\varphi)$, then x is said to be* frozen *in φ if it is assigned the same truth value in all its models.*

Since a relation can be defined by a formula, by abuse of language we also speak about a frozen variable in a relation.

Let us now define two closure operators on sets of logical relations.

Definition 14. *Let Γ be a set of logical relations.*

- *$\langle\Gamma\rangle_{fr}$ is the set of relations that can be defined by a formula of the form $\exists X \varphi$, such that φ is a $(\Gamma \cup \{=\})$-formula, $X \subseteq \mathrm{var}(\varphi)$ and every variable in X is frozen in φ.*
- *$\langle\Gamma\rangle_{fr,\neq}$ is the set of relations that can be defined by a formula of the form $\exists X \varphi$, such that φ is a Γ-formula, $X \subseteq \mathrm{var}(\varphi)$ and every variable in X is frozen in φ.*

These two notions differ in the sense that the first one allows equality constraints, and the other does not. The symbol $\neq$ stands for "no equality constraint is allowed".

On the one hand, the first notion of closure is well known from an algebraic point of view. Indeed, $\langle\Gamma\rangle_{\mathrm{fr}}$ is a partial frozen co-clone. The lattice of the partial frozen co-clones is partially known, especially within the Krom fragment (see[19]). On the other hand, the more restricted notion of closure, $\langle.\rangle_{\mathrm{fr},\neq}$, is useful for complexity issues. Indeed, this closure operator induces reductions between problems we are interested in, as shown by the following proposition.

Proposition 15. *Let Γ_1 and Γ_2 be constraint languages with $\Gamma_1 \subseteq \langle\Gamma_2\rangle_{fr,\neq}$. Then* $\textsc{CardMinSat}(\Gamma_1) \leq \textsc{CardMinSat}(\Gamma_2)$.

Proof. Let φ_1 be a Γ_1-formula. We construct a formula φ_2 by performing the following steps:

- Replace in φ_1 every constraint from Γ_1 by its defining existentially quantified Γ_2-formula, in which all existential variables are frozen. Use fresh existential variables for each constraint.
- Delete existential quantifiers.

Then, obviously, φ_2 is a Γ_2-formula and φ_1 is satisfiable if and only if φ_2 is satisfiable. Moreover, since all existentially quantified variables are frozen, removing the quantifiers preserves the cardinality-minimal models, i.e., there is a one-to-one correspondence between the cardinality-minimal models of φ_1 and the ones of φ_2, which preserves the truth values of the variables in $\mathrm{var}(\varphi_1)$. Therefore a given atom x is true in a cardinality-minimal model of φ_1 if and only if it is true in a cardinality-minimal model of φ_2. Moreover the complexity of the above transformation is polynomial, thus concluding the proof. $\square$

As a consequence to get hardness results in the Krom fragment we will use both notions of closure. Roughly speaking the main strategy is as follows: Theorem 5 shows that CardMinSat(Or_2) is Θ_2P-hard. Hence, in most of the cases, in order to prove that for some class of constraint languages the problem CardMinSat is Θ_2P-hard, according to Proposition 15 we will prove that for every language Γ in the studied class, $\langle\Gamma\rangle_{\mathrm{fr},\neq}$ contains Or_2. But showing that $\mathrm{Or}_2 \in \langle\Gamma\rangle_{\mathrm{fr},\neq}$ will be done in using information on the corresponding partial co-clone $\langle\Gamma\rangle_{\mathrm{fr}}$. For this we need an additional technical notion. An n-ary relation R is *irredundant* if there is no pair (i,j), $1 \leq i < j \leq n$, such that for all $(\alpha_1,\ldots,\alpha_n) \in R$, $\alpha_i = \alpha_j$, and if there is no i, $1 \leq i \leq n$, such that for all $(\alpha_1,\ldots,\alpha_i,\ldots,\alpha_n) \in R$, the tuple $(\alpha_1,\ldots,1-\alpha_i,\ldots,\alpha_n) \in R$ as well (which means that the ith variable in the formula representing R is unconstrained)

The motivation for defining irredundant relations is that one can easily represent these relations with formulas in which no equality constraints appear. Hence, as stated in the following lemma, if we can define an irredundant relation using the $\langle.\rangle_{\mathrm{fr}}$-operator, we can also obtain an implementation using the $\langle.\rangle_{\mathrm{fr},\neq}$-operator.

Lemma 16. *Let Γ be a set of logical relations and R be an irredundant relation that has no frozen variable. If $R \in \langle\Gamma\rangle_{fr}$, then $R \in \langle\Gamma\rangle_{fr,\neq}$.*

Proof. Let R be such an irredundant relation, and suppose that it is defined by the formula $\exists X\varphi$ where φ is a $(\Gamma \cup \{=\})$-formula in which all variables from X are frozen. Two variables from $\mathrm{var}(\varphi)\backslash X$ (this set corresponds to variables of R) cannot appear in an equality constraint in φ since R is irredundant. A variable from X and a variable from $\mathrm{var}(\varphi)\backslash X$ cannot occur together in an equality constraint in φ either since all variables from X are frozen in φ and R has no frozen variable. Therefore equality constraints can only involve variables from X. In using the transitivity of the equality relation we can identify equivalence classes. In the formula $\exists X\varphi$, all variables from X that are in the same equivalence class can be replaced by a single variable, thus obtaining a frozen implementation of R with no equality. □

We are now in a position to prove the following proposition, which covers the hard cases of Theorem 9.

Proposition 17. *Let Γ be a Krom constraint language. If Γ is neither width-2 affine nor Horn, then* CardMinSat(Γ) *is Θ_2P-hard.*

Proof. Observe that while the Galois connection explained above does not seem to be useful to study the complexity of CardMinSat(Γ), nevertheless the obtained classification obeys the borders among co-clones and the classification will be obtained through a case study based on Post's lattice. Since Γ is Krom, $\Gamma \subseteq \mathrm{Inv}(\mathrm{D}_2)$. Since Γ is neither width-2 affine nor Horn, $\Gamma \subseteq \mathrm{Inv}(\mathrm{D}_1)$ and $\Gamma \subseteq \mathrm{Inv}(\mathrm{E}_2)$. Therefore, according to Post's lattice (see Figure 1) there are only five cases to consider, either $\langle\Gamma\rangle = \mathrm{Inv}(\mathrm{D}_2)$, $\mathrm{Inv}(\mathrm{S}^2_{00})$, $\mathrm{Inv}(\mathrm{S}^2_{01})$, $\mathrm{Inv}(\mathrm{S}^2_{02})$ or $\mathrm{Inv}(\mathrm{S}^2_{0})$.

If $\langle\Gamma\rangle = \mathrm{Inv}(\mathrm{S}^2_{00})$ or $\langle\Gamma\rangle = \mathrm{Inv}(\mathrm{S}^2_{02})$ or $\langle\Gamma\rangle = \mathrm{Inv}(\mathrm{S}^2_{0})$. According to [19, Theorem 15], the three co-clones $\mathrm{Inv}(\mathrm{S}^2_{00})$, $\mathrm{Inv}(\mathrm{S}^2_{02})$ and $\mathrm{Inv}(\mathrm{S}^2_{0})$ are covered by a single partial frozen co-clone. Therefore, if $\langle\Gamma\rangle = \mathrm{Inv}(\mathrm{S}^2_{00})$ (respectively, $\langle\Gamma\rangle = \mathrm{Inv}(\mathrm{S}^2_{02})$, $\langle\Gamma\rangle = \mathrm{Inv}(\mathrm{S}^2_{0})$), then $\langle\Gamma\rangle_{\mathrm{fr}} = \langle\Gamma\rangle = \mathrm{Inv}(\mathrm{S}^2_{00})$ (respectively, $\langle\Gamma\rangle_{\mathrm{fr}} = \langle\Gamma\rangle = \mathrm{Inv}(\mathrm{S}^2_{02})$, $\langle\Gamma\rangle_{\mathrm{fr}} = \langle\Gamma\rangle = \mathrm{Inv}(\mathrm{S}^2_{0})$). Since these three co-clones contain the relation Or_2 (see e.g. [4]), we obtain that $\langle\Gamma\rangle_{\mathrm{fr}}$ contains Or_2. Since the relation Or_2 is irredundant and has no frozen variable, it follows from Lemma 16 that $\mathrm{Or}_2 \in \langle\Gamma\rangle_{\mathrm{fr},\neq}$. We conclude in using Theorem 5 and Proposition 15.

If $\langle\Gamma\rangle = \mathrm{Inv}(\mathrm{D}_2)$. The co-clone $\mathrm{Inv}(\mathrm{D}_2)$ is not covered by a single frozen co-clone, but the structure of partial frozen co-clones covering this co-clone is known [19, Theorem 19]. In particular it holds that $\langle\Gamma\rangle_{fr} \supseteq \Gamma_4^p$, where Γ_4^p is a partial frozen co-clone that contains the relation R_4^p defined by $R_4^p(x_1, x_2, x_3, x_4) = (x_1 \vee x_2) \wedge (x_1 \neq x_3) \wedge (x_2 \neq x_4)$. Since the relation R_4^p is irredundant and has no frozen variable, by Lemma 16 it follows that $R_4^p \in \langle\Gamma\rangle_{\mathrm{fr},\neq}$. We now prove that $\textsc{CardMinSat}(\mathrm{Or}_2) \leq \textsc{CardMinSat}(R_4^p)$, thus concluding the proof according to Theorem 5 and Proposition 15.

Let $\varphi = \bigwedge_{(i,j)\in E} \mathrm{Or}_2(x_i, x_j)$ be an Or_2-formula with $\mathrm{var}(\varphi) = \{x_1, \ldots, x_n\}$ and $E \subseteq \{1, \ldots, n\}^2$. Let us consider the formula φ' built over the set of variables $\{x_1, \ldots, x_n, x'_1, \ldots, x'_n, x''_1, \ldots, x''_n\}$, where the x'_i's and the x''_i's are fresh variables, and defined by $\varphi' = \bigwedge_{(i,j)\in E} R_4^p(x_i, x_j, x'_i, x'_j) \wedge R_4^p(x''_i, x''_j, x'_i, x'_j)$. Observe that in every model of φ', for all i we have $x_i = x''_i$ and $x_i \neq x'_i$. Therefore, it is easily seen that there is a one-to-one correspondence between the set of models of φ and the set of models of φ', preserving the truth values of $x_1, \ldots, x_n$, and transforming every model of weight k of φ into a model of weight $n + k$ of φ'. Therefore a given atom x_i is true in a cardinality-minimal model of φ if and only if it is true in a cardinality-minimal model of φ', thus concluding the proof.

If $\langle\Gamma\rangle = \mathrm{Inv}(\mathrm{S}_{01}^2)$. The co-clone $\mathrm{Inv}(\mathrm{S}_{01}^2)$ is not covered by a single co-clone either and the structure of the partial frozen co-clones covering this co-clone is not known. Therefore we have to use another technique. We prove directly that $\textsc{CardMinSat}(\mathrm{Or}_2) \leq \textsc{CardMinSat}(\Gamma)$, thus concluding the proof according to Theorem 5.

Observe that all relations in Γ are 1-valid and dual-Horn, and there exist at least one relation S in Γ which is not 0-valid and one relation R that is not Horn. Since S is 1-valid but not 0-valid, $\vec{1} \in S$ and $\vec{0} \notin S$. Therefore, the Γ-constraint $C_1(x) = S(x, \ldots, x)$ defines the constant 1.

Consider the constraint $C = R(x_1, \ldots, x_k)$. Since R is non-Horn there exist m_1 and m_2 in R such that $m_1 \wedge m_2 \notin R$. Since R is 1-valid and dual Horn, we have $\vec{1} \in R$ and $m_1 \vee m_2 \in R$. For $i, j \in \{0, 1\}$, set $V_{i,j} = \{x \mid x \in V, m_1(x) = i \wedge m_2(x) = j\}$. Observe that $V_{0,1} \neq \emptyset$ (respectively, $V_{1,0} \neq \emptyset$), otherwise $m_1 \wedge m_2 = m_2$ (resp., $m_1 \wedge m_2 = m_1$), contradicting the fact that $m_1 \wedge m_2 \notin R$. Consider the $\{R\}$-constraint: $M(w, x, y, t) = C[w/V_{0,0},\ x/V_{0,1},\ y/V_{1,0},\ t/V_{1,1}]$. According to the above remark the two variables x and y effectively occur in this constraint. Let us examine the set of models of M: it contains 0011 (since $m_1 \in R$), 0101 (since $m_2 \in R$), 0111 (since $m_1 \vee m_2 \in R$) and 1111 (since R is 1-valid), but it does not contain 0001 (since by assumption $m_1 \wedge m_2 \notin R$). We make a case distinction according to whether $1001 \in M$ or not.

Suppose $1001 \notin M$. To every Or_2-formula $\varphi = \bigwedge_{i=1}^m \mathrm{Or}_2(x_i, y_i)$ we associate the Γ-formula $\varphi' = \bigwedge_{i=1}^m M(\alpha_i, x_i, y_i, t) \wedge C_1(t)$ where α_i for $i = 1, \ldots m$ and t are fresh variables. Observe that cardinality-minimal models of φ and cardinality-minimal models of φ' coincide with $t = 1$ and $\alpha_i = 0$ for $i = 1, \ldots m$, thus concluding the proof.

Suppose now that $1001 \in M$. To every Or_2-formula $\varphi = \bigwedge_{i=1}^m \mathrm{Or}_2(x_i, y_i)$ with $|\mathrm{var}(\varphi)| = n$, we associate the Γ-formula $\varphi' = \bigwedge_{i=1}^m \bigwedge_{j=1}^{n+1} M(\alpha_i^j, x_i, y_i, t) \wedge C_1(t)$ where α_i^j for $i = 1, \ldots m, j = 1, \ldots, n+1$ and t are fresh variables. Observe that minimal models of φ can be extended to minimal models of φ' in setting t to 1 and all α_i^j to 0. Conversely, note that φ' has a model of cardinality $n + 1$, namely m such that $m(t) = 1$, $m(x) = 1$ for all $x \in \mathrm{var}(\varphi)$ and $m(\alpha_i^j) = 0$ for $i = 1, \ldots m, j = 1, \ldots, n + 1$. Therefore every cardinality-minimal model of φ' has cardinality less than or equal to $n + 1$. So, given a cardinality-minimal model m' of φ', $m' \models (x_i \vee y_i)$ for all i, otherwise this would imply $m'(\alpha_i^j) = 1$ for all j, contradicting the minimality of m'. Therefore m' restricted to $\mathrm{var}(\varphi)$ is a model of φ, and even a cardinality-minimal model of φ. This concludes the proof. □

Remark 18. *The complexity classification of* CardMinSat *in the Krom fragment has been obtained by means of partial frozen co-clones. The proof could also have been obtained in using an even more restricted notion of closure, namely* $\langle . \rangle_{\nexists,\neq}$*, which allows neither existential variables nor equality constraints, together with the notion of weak bases introduced in [24]. Nevertheless, since the lattice of partial frozen co-clones is rather well described within the Krom fragment (see [19]), this was an opportunity to popularize these co-clones, which are of independent interest.*

7 Conclusion

Table 1: Complexity of the Reasoning Problems (Completeness Results).

	general case	Horn	Krom	Horn∩Krom
BR-Implication Satoh	Π_2P	coNP	coNP	coNP
BR-Model Checking Satoh	Σ_2P	NP	NP	NP
BR-Implication Dalal	Θ_2P	Θ_2P	Θ_2P	Θ_2P
BR-Model Checking Dalal	Θ_2P	Θ_2P	Θ_2P	Θ_2P
$\leq$-Relevance	Θ_3P	Θ_2P	Θ_2P	Θ_2P

In this work we have investigated how the restriction of propositional formulas to Krom affects the complexity of reasoning problems in the AI domain. Our results on belief revision and abduction are summarized in Table 1, where the complexity classifications for Krom and the combined Horn ∩ Krom case refer to new results we have provided in the paper. Having shown that the complexity of problems involving cardinality minimality, like Dalal's revision operator or $\leq$-Relevance in abduction, is often robust to such a restriction (even for formulas being Horn and Krom at the same time), suggests that further classes within the Krom fragment should be considered to identify the exact tractability/intractability frontier.

The problem CardMinSat seems to be the key for such investigations, thus we initiated a deeper study by giving a complete classification of that problem restricted to Krom within Schaefer's framework. As a next step, we want to study the Krom form (and the yet more restricted fragments thereof) in the context of further hard reasoning problems and analyse which of these fragments suffice to yield tractability.

A complexity classification of CardMinSat in the full propositional logic, which opens the door to many complexity results for reasoning problems from the AI domain, is under investigation. The lattice of partial frozen co-clones being insufficiently well-known, such a classification will require the use of weak bases introduced in [24].

Acknowledgments

This research has been supported by the Austrian Science Fund (FWF) through the projects P25518, P25521and Y698, by the OeAD project FR 12/2013 and by the Campus France Amadeus project 29144UC.

References

[1] M. Ben-Ari. *Mathematical Logic for Computer Science*. Prentice-Hall International, 2003.

[2] V. G. Bodnarchuk, L. A. Kalužnin, V. N. Kotov, and B. A. Romov. Galois theory for Post algebras. I, II. *Cybernetics*, 5:243–252, 531–539, 1969.

[3] E. Böhler, N. Creignou, S. Reith, and H. Vollmer. Playing with Boolean blocks, part I: Post's lattice with applications to complexity theory. *ACM-SIGACT Newsletter*, 34(4):38–52, 2003.

[4] E. Böhler, S. Reith, H. Schnoor, and H. Vollmer. Bases for Boolean co-clones. *Information Processing Letters*, 96:59–66, 2005.

[5] N. Creignou and H. Vollmer. Boolean constraint satisfaction problems: When does Post's lattice help? In *Complexity of Constraints*, volume 5250 of *Lecture Notes in Computer Science*, pages 3–37. Springer, 2008.

[6] N. Creignou and B. Zanuttini. A complete classification of the complexity of propositional abduction. *SIAM J. Comput.*, 36(1):207–229, 2006.

[7] M. Dalal. Investigations into theory of knowledge base revision. In *Proc. AAAI*, pages 475–479. AAAI Press / The MIT Press, 1988.

[8] T. Eiter and G. Gottlob. On the complexity of propositional knowledge base revision, updates, and counterfactuals. *Artif. Intell.*, 57(2-3):227–270, 1992.

[9] T. Eiter and G. Gottlob. The complexity of logic-based abduction. *J. ACM*, 42(1):3–42, 1995.

[10] T. Eiter and G. Gottlob. The complexity class ϑ_2^{P}: Recent results and applications in AI and modal logic. In *Proc. FCT*, volume 1279 of *LNCS*, pages 1–18. Springer, 1997. doi: 10.1007/BFb0036168.

[11] W. I. Gasarch, M. W. Krentel, and K. J. Rappoport. OptP as the normal behavior of NP-complete problems. *Mathematical Systems Theory*, 28(6):487–514, 1995.

[12] D. Geiger. Closed systems of functions and predicates. *Pac. J. Math*, 27(1): 95–100, 1968.

[13] M. Hermann and R. Pichler. Counting complexity of propositional abduction. *J. Comput. Syst. Sci.*, 76(7):634–649, 2010.

[14] P. G. Jeavons. On the algebraic structure of combinatorial problems. *Theor. Comput. Sci.*, 200:185–204, 1998.

[15] M. W. Krentel. The complexity of optimization problems. *J. Comput. Syst. Sci.*, 36(3):490–509, 1988.

[16] M. W. Krentel. Generalizations of OptP to the polynomial hierarchy. *Theor. Comput. Sci.*, 97(2):183–198, 1992.

[17] P. Liberatore and M. Schaerf. Belief revision and update: Complexity of model checking. *J. Comput. Syst. Sci.*, 62(1):43–72, 2001.

[18] G. Nordh and B. Zanuttini. What makes propositional abduction tractable. *Artif. Intell.*, 172(10):1245–1284, 2008.

[19] G. Nordh and B. Zanuttini. Frozen boolean partial co-clones. In *Proc. ISMVL*, pages 120–125, 2009.

[20] C. H. Papadimitriou. *Computational Complexity.* Addison-Wesley, 1994.

[21] E. L. Post. The two-valued iterative systems of mathematical logic. *Annals of Mathematical Studies*, 5:1–122, 1941.

[22] K. Satoh. Nonmonotonic reasoning by minimal belief revision. In *Proc. FGCS*, pages 455–462, 1988.

[23] T. J. Schaefer. The complexity of satisfiability problems. In *Proc. STOC*, pages 216–226. ACM, 1978.

[24] H. Schnoor and I. Schnoor. Partial polymorphisms and constraint satisfaction problems. In N. Creignou, P. G. Kolaitis, and H. Vollmer, editors, *Complexity of Constraints*, volume 5250, pages 229–254. Springer Verlag, Berlin Heidelberg, 2008.

[25] G. Tseytin. On the complexity of derivation in propositional calculus. In A. Slisenko, editor, *Studies in Constructive Mathematics and Mathematical Logic, Part II, Seminars in Mathematics*, page 115–125. Steklov Mathematical Institute, 1970.

[26] K. Wagner. On restricting the access to an NP-oracle. In *Proc. ICALP*, volume 317 of *LNCS*, pages 682–696. Springer, 1988.