Propositional Abduction is Almost Always Hard

Gustav Nordh*
Dept. of Computer & Information Science
Linköpings Universitet
S-581 83 Linköping, Sweden
gusno@ida.liu.se

Bruno Zanuttini
GREYC, UMR CNRS 6072
Université de Caen, Bd. du Maréchal Juin
F-14 032 Caen Cedex, France
zanutti@info.unicaen.fr

Abstract

Abduction is a fundamental form of nonmonotonic reasoning that aims at finding explanations for observed manifestations. Applications of this process range from car configuration to medical diagnosis. We study here its computational complexity in the case where the application domain is described by a propositional theory built upon a fixed constraint language and the hypotheses and manifestations are described by sets of literals. We show that depending on the language the problem is either polynomial-time solvable, NP-complete, or Σ_2^P-complete. In particular, we show that under the assumption P≠NP, only languages that are affine of width 2 have a polynomial algorithm, and we exhibit very weak conditions for NP-hardness.

1 Introduction

In this paper we investigate the computational complexity of abduction, a method of reasoning extensively studied by Peirce [1955]. Abductive reasoning is used to search for explanations of observed manifestations. The importance of this problem to Artificial Intelligence was first emphasized by Morgan [1971] and Pople [1973].

The abductive process has demonstrated its practical importance in many domains. In particular, it has been used to formalize processes in medical diagnosis [Bylander *et al.*, 1991], text interpretation [Hobbs *et al.*, 1993], system diagnosis [Stumptner and Wotawa, 2001] and configuration problems [Amilhastre *et al.*, 2002].

In this paper we are interested in propositional logic-based abduction, i.e., the background knowledge is represented by a propositional theory. Even in this framework several formalizations of the problem have been studied in the literature, depending on the syntactic restrictions imposed to the manifestations and on the hypotheses over which explanations must be formed. We use the following formalization: Given a propositional theory T formalizing a particular application domain, a set M of literals describing a set of manifestations, and a set H of literals containing possible hypotheses, find an explanation for M, that is, a set $E \subseteq H$ such that $T \cup E$ is consistent and logically entails M. This framework is more general than most frameworks studied in the literature; an exception is the definition by Marquis [Marquis, 2000], which allows the manifestation to be encoded by any propositional formula.

*Supported by the *National Graduate School in Computer Science* (CUGS), Sweden.

Example 1 *Consider the following example from the domain of motor vehicles, inspired by [Console* et al.*, 1991].*

$$\begin{array}{rcl} T & = & (\neg \textit{rich\_mixture} \vee \neg \textit{lean\_mixture}) \wedge \\ & & (\textit{rich\_mixture} \rightarrow \neg \textit{low\_fuel\_consumption}) \wedge \\ & & (\textit{lean\_mixture} \rightarrow \textit{overheating}) \wedge \\ & & (\textit{low\_water} \rightarrow \textit{overheating}), \\ H & = & \{\textit{rich\_mixture, lean\_mixture, low\_water}\}, \\ M & = & \{\neg \textit{low\_fuel\_consumption, overheating}\}. \end{array}$$

Then $E = \{$rich_mixture, low_water$\}$ is an explanation of M, while, e.g., $\{$lean_mixture,low_water$\}$ is not because it does not explain low_fuel_consumption, and $\{$rich_mixture, lean_mixture$\}$ is not because it is inconsistent with T.

Formalizations of abduction also differ in the literature in the notions of preferred explanations: In this setting, the aim is to compute an explanation that is minimal among all explanations according to some criteria (e.g., inclusion or cardinality). A good overview is given by Eiter and Gottlob [1995]. However we are not concerned here with these different notions; we indeed focus on the decision problem asking whether there exists an explanation at all for a given instance. This problem is from now on denoted ABDUCTION.

This problem is well studied from the computational complexity perspective [Eshghi, 1993; Selman and Levesque, 1990; Eiter and Gottlob, 1995; del Val, 2000b; 2000a; Zanuttini, 2003], although with different formalizations of the problem, as mentioned above. It has been proved that it is Σ_2^P-complete in the general case [Eiter and Gottlob, 1995], while propositional deduction is known to be "only" coNP-complete [Cook, 1971]. This negative result raises the problem of identifying restricted cases in which ABDUCTION has computational complexity lower than the general case.

The most natural way to study such restrictions is to study restrictions on the theories representing knowledge bases. This is also the approach followed in most of the previous research in the area. For example, it is known that when the knowledge base T is a conjunction of Horn clauses, then ABDUCTION is NP-complete [Selman and Levesque, 1990].

The ultimate goal of this line of research is to determine the complexity of every restricted special case of the problem. The first result of this type was proved by Schaefer [1978], who proved that the satisfiability problem of conjunctions of Boolean constraints over a fixed language is either in P or NP-complete, depending on the language. Recall the result due to Ladner [1975] stating that if P $\neq$ NP, then there exist decision problems in NP that are neither in P nor NP-complete. Hence the existence of dichotomy theorems like Schaefer's cannot be taken for granted. Creignou *et al.*'s book [2001] surveys such results.

In this paper we completely classify the complexity of ABDUCTION in Schaefer's framework and exhibit a trichotomy. More precisely, we prove that ABDUCTION is:

- In P if the language is affine of width 2,
- Otherwise, NP-complete if the language is Horn, dual Horn, bijunctive or affine,
- Otherwise, Σ_2^{P}-complete.

As far as we know, the (only) polynomial case and the minimal NP-hard languages that we exhibit are all new results. Note that the only property that guarantees a polynomial algorithm is that of being affine of width 2, i.e., the case where the theory representing the domain is only a conjunction of constants and (in)equalities between two variables. Thus the problem is very hard as soon as sets of literals are allowed as hypotheses and manifestations (instead of atoms or sets of atoms as in, e.g., [Zanuttini, 2003]).

We do not consider directly the complexity of the search problem consisting of computing an explanation or asserting there is none. It is however easily seen that this problem is hard as soon as ABDUCTION is; as for languages that are affine of width 2, our proof that ABDUCTION is in P exhibits an efficient algorithm.

2 Preliminaries

The set of all n-tuples of elements from $\{0,1\}$ is denoted by $\{0,1\}^n$. Any subset of $\{0,1\}^n$ is called an *n-ary relation* on $\{0,1\}$. The set of all finitary relations over $\{0,1\}$ is denoted by BR. A *constraint language* over $\{0,1\}$ is an arbitrary finite set $\Gamma \subseteq BR$. Constraint languages are the way in which we impose restrictions on the knowledge bases for the ABDUCTION problem.

A *constraint* over the constraint language Γ is an application of a relation R in Γ to a tuple of variables, written $R(x_1,\ldots,x_n)$ (possibly with repeated variables). An assignment m to the variables *satisfies* the constraint $R(x_1,\ldots,x_n)$ if $(m(x_1),\ldots,m(x_n))$ is a tuple in R. Such a satisfying assignment m is called a *model* of $R(x_1,\ldots,x_n)$. A *theory* T over Γ is a conjunction of constraints over Γ; a *model* m of T is an assignment that satisfies all its constraints simultaneously, denoted $m \models T$. If there is such a model, T is said to be *satisfiable*. Finally, a theory T *entails* a theory T', written $T \models T'$, if every model of T is a model of T'.

The unary relations $F = \{(0)\}$ and $T = \{(1)\}$, which force the value of a variable to 0 and 1 respectively, have a special role for the ABDUCTION problem. We will often use the shorthand notation $\neg x$ and x to denote the constraints $F(x)$ and $T(x)$ respectively. Constraint languages containing F and T will be of particular importance to us. Given a constraint language Γ, the *idempotent constraint language* corresponding to Γ is $\Gamma \cup \{F,T\}$ which is denoted by Γ^{id}.

Given a theory T, $Vars(T)$ denotes the set of all variables that occurs in T. Given a set of variables V, $Lits(V)$ denotes the set of constraints $\bigcup_{x\in V}\{F(x) \cup T(x)\}$ or, equivalently, the set of all literals formed upon the variables in V. The opposite of a literal ℓ is written $\overline{\ell}$. Given a set of literals L, their conjunction is denoted by $\bigwedge L$.

The ABDUCTION problem restricted to the finite constraint language Γ is denoted by ABDUCTION(Γ) and is defined as follows; the problem is said to be *parameterized* by Γ.

Problem 2 (ABDUCTION(Γ)) *An instance $\mathscr{P}$ of* ABDUCTION(Γ) *consists of a tuple (V,H,M,T), where:*

- *V is a set of variables,*
- *$H \subseteq Lits(V)$ is the set of hypotheses,*
- *$M \subseteq Lits(V)$ is the set of manifestations, and*
- *T is a theory over Γ with $Vars(T) = V$.*

The question is whether there exists an explanation for $\mathscr{P}$, i.e., a set $E \subseteq H$ such that $T \wedge \bigwedge E$ is satisfiable and $T \wedge \bigwedge E \models \bigwedge M$.

The size of $\mathscr{P} = (V,H,M,T)$ is the total number of occurrences of variables in it.

Recall the following standard restrictions on the constraint languages (see, e.g. [Creignou *et al.*, 2001]).

Definition 3 (restrictions on Γ)

- Γ *is* Horn *if every relation R in Γ is the set of models of a CNF formula having at most one unnegated variable in each clause,*
- Γ *is* dual Horn *if every relation R in Γ is the set of models of a CNF formula having at most one negated variable in each clause,*
- Γ *is* bijunctive *if every relation R in Γ is the set of models of a CNF formula having at most two literals in each clause,*
- Γ *is* affine *if every relation R in Γ is the set of models of a system of linear equations over $GF(2)$, the field with two elements,*
- Γ *is* affine of width 2 *if every relation R in Γ is the set of models of a system of linear equations over $GF(2)$ in which each equation has at most 2 variables.*

We emphasize that Γ is assumed to be finite in the definition of ABDUCTION(Γ). Moreover, it parameterizes the problem, and is not part of its input. Thus we can assume any convenient presentation of the relations in Γ is stored in a catalog; thus we will assume, e.g., that if Γ is Horn then a theory T over Γ is given as a CNF formula with at most one unnegated variable per clause.

We now recall Schaefer's result, which will be of great importance throughout the paper. Call SAT(Γ) the problem of deciding whether a given theory over Γ is satisfiable. Schaefer completely classified the complexity of SAT(Γ); we only report the result for idempotent constraint languages.

Theorem 4 ([Schaefer, 1978]) $\text{SAT}(\Gamma^{id})$ *is in P if* Γ *is Horn, dual Horn, bijunctive or affine. Otherwise it is NP-complete.*

Finally, we assume that the reader is familiar with the basic notions of complexity theory, but we briefly recall the following. P is the class of decision problems solvable in deterministic polynomial time. NP is the class of decision problems solvable in nondeterministic polynomial time. $\Sigma_2^{\text{P}} = \text{NP}^{\text{NP}}$ is the class solvable in nondeterministic polynomial time with access to an NP-oracle. A problem is NP-complete (Σ_2^{P}-complete) if every problem in NP (Σ_2^{P}) is polynomial-time reducible to it. Throughout the paper we assume P≠NP≠Σ_2^{P}.

3 Polynomial case

The following proposition gives the only (maximal) polynomial case of $\text{ABDUCTION}(\Gamma)$. Note that, as mentioned in the introduction, its proof gives a polynomial algorithm that *computes* an explanation if there exists one.

Throughout the proof we write $(\ell = a)$ $(a \in \{0,1\})$ for the linear equation $(x = a)$ if $\ell = x$, and for the linear equation $x = a \oplus 1$ if $\ell = \neg x$. The shorthand $(\ell = \ell')$ is used in the same manner and is equivalent to $(\ell \oplus \ell' = 0)$.

Proposition 5 *If* Γ *is affine of width* 2*, then* $\text{ABDUCTION}(\Gamma)$ *is polynomial.*

Proof Let $\mathscr{P} = (V, H, M, T)$ be an instance of $\text{ABDUCTION}(\Gamma)$, where Γ is affine of width 2. If M' is the set of all manifestations m such that $T \not\models m$, then obviously the explanations of $\mathscr{P}$ are exactly those of (V, H, M', T). Since $T \models m$ can be decided efficiently with Gaussian elimination on $T \wedge (m = 0)$, we assume $M' = M$.

For every manifestation $m \in M$ write E_m for the set of literals $\{h \in H \mid T \models (h = m)\}$; once again every E_m can be computed efficiently with Gaussian elimination on $T \wedge (h \oplus m = 1)$ for every $h \in H$. We show that $\mathscr{P}$ has an explanation if and only if $T \wedge \bigwedge_{m \in M} \bigwedge E_m$ is satisfiable and no E_m is empty. Since the satisfiability of $T \wedge \bigwedge_{m \in M} \bigwedge E_m$ can be decided efficiently with again Gaussian elimination (on $T \wedge \bigwedge_{m \in M} \bigwedge_{h \in E_m} (h = 1)$), this will conclude the proof.

Assume first $\mathscr{P}$ has an explanation E. Then $T \wedge \bigwedge M$ is consistent; since for every $m \in M$ and $h \in E_m$ we have $T \models (h = m)$, we also have $T \wedge \bigwedge M \models T \wedge \bigwedge_{m \in M} \bigwedge E_m$, and thus $T \wedge \bigwedge_{m \in M} \bigwedge E_m$ is satisfiable. Now we also have $\forall m \in M, T \models (\bigwedge E \rightarrow m)$. Since T is affine of width 2 it is bijunctive, thus every clause entailed by T can be minimized into a bijunctive one; since $T \wedge \bigwedge E$ is satisfiable and $T \not\models m$, the only possibility is a minimal clause of the form $h \rightarrow m$ with $h \in E$. But since T is affine this implies that $m \rightarrow h$ also is an implicate of it, and finally we have $T \models (h \oplus m = 0)$, which shows that E_m is nonempty. For more details we refer the reader to [Zanuttini and Hébrard, 2002]. Conversely, assume $T \wedge \bigwedge_{m \in M} \bigwedge E_m$ is satisfiable and no E_m is empty. Then since $(h \oplus m = 0) \models (h \rightarrow m)$ it is easily seen that $\bigwedge_{m \in M} \bigwedge E_m$ is an explanation for $\mathscr{P}$. □

4 NP-complete cases

We now exhibit the NP-complete cases of $\text{ABDUCTION}(\Gamma)$. Since $T \wedge \bigwedge E \models \bigwedge M$ holds if and only if $T \wedge \bigwedge E \wedge \overline{\ell}$ is unsatisfiable for every $\ell \in M$, the following result is obvious.

Lemma 6 *If* $\text{SAT}(\Gamma^{id})$ *is in P, then* $\text{ABDUCTION}(\Gamma)$ *is in NP.*

We first establish NP-completeness for particular languages. In Section 5 we will establish more general results.

Let $R_{\neg x \vee y} = \{(1,1), (0,1), (0,0)\}$, i.e., the set of models of $\neg x \vee y$. Observe that $R_{\neg x \vee y}$ is both Horn and dual Horn.

Proposition 7 $\text{ABDUCTION}(\{R_{\neg x \vee y}\})$ *is NP-complete.*

Proof Membership in NP follows from Theorem 4 and Lemma 6. As regards hardness, we give a reduction from the NP-complete problem MONOTONE-SAT [Garey and Johnson, 1979], i.e., the satisfiability problem for CNF formulas where each clause contains either only positive literals or only negative literals. Let $\psi = \bigwedge_{i=1}^{k} N_i \wedge \bigwedge_{i=1}^{\ell} P_i$ be such a formula, where each N_i is a negative clause, written $N_i = \bigvee_{j=1}^{\nu_i} \neg x_i^j$, and every P_i is a positive clause written $P_i = \bigvee_{j=1}^{\pi_i} y_i^j$. We build the instance $\mathscr{P} = (V, H, M, T)$ of $\text{ABDUCTION}(R_{\neg x \vee y})$ where:

- $V = \{\gamma_i \mid i = 1, \ldots, k\} \cup \{\delta_i \mid i = 1, \ldots, \ell\} \cup Vars(\psi)$; $\neg\gamma_i$ will intuitively represent satisfaction of clause N_i and δ_i, that of clause P_i
- $T = \bigwedge_{i=1}^{k} \bigwedge_{j=1}^{\nu_i} (x_i^j \vee \neg\gamma_i) \wedge \bigwedge_{i=1}^{\ell} \bigwedge_{j=1}^{\pi_i} (\neg y_i^j \vee \delta_i)$; this encodes the implications $\neg x_i^j \rightarrow \neg\gamma_i$ and $y_i^j \rightarrow \delta_i$, i.e., the fact that N_i (P_i) is satisfied if at least one of the $\neg x_i^j$ (y_i^j), $j \in \{1, \ldots, \nu_i\}$ ($j \in \{1, \ldots, \pi_i\}$) is
- $H = Lits(Vars(\varphi))$
- $M = \bigwedge_{i=1}^{k} \neg\gamma_i \wedge \bigwedge_{i=1}^{\ell} \delta_i$.

Obviously enough, the theory T is over the language $\{R_{\neg x \vee y}\}$. Now it is easily seen that if ψ has at least one model, say m, then $E = \{\ell \mid m \models \ell\}$ is an explanation for $\mathscr{P}$, and that if $\mathscr{P}$ has an explanation E, then any assignment m to $Vars(\psi)$ with $\forall \ell \in E, m \models \ell$ is a model of ψ. □

Similarly, we now prove that $\text{ABDUCTION}(\Gamma)$ is NP-complete if Γ is the singleton language containing only $R_{x \vee y} = \{(1,1), (1,0), (0,1)\}$.

Proposition 8 $\text{ABDUCTION}(\{R_{x \vee y}\})$ *is NP-complete.*

Proof Since $R_{x \vee y}$ is dual Horn, membership in NP follows from Theorem 4 and Lemma 6. As for hardness, we give a reduction from MONOTONE-SAT (see the proof of Proposition 7), where positive clauses in an instance of this problem are restricted to contain at most two literals. Thus an instance of this problem is a formula of the form $\psi = \bigwedge_{i=1}^{k} N_i \wedge \bigwedge_{i=1}^{\ell} (y_i^1 \vee y_i^2)$, where the y_i^j's are variables and every N_i is a negative clause written $N_i = \bigvee_{j=1}^{\nu_i} \neg x_i^j$. The NP-completeness of this restricted problem follows directly from Schaefer's result [1978].

Given an instance ψ of MONOTONE-SAT as above we build the instance $\mathscr{P} = (V, H, M, T)$ of $\text{ABDUCTION}(\{R_{x \vee y}\})$ where:

- $V = \{\gamma_i \mid i = 1, \ldots, k\} \cup Vars(\psi)$; γ_i will intuitively represent satisfaction of clause N_i

- $T = \bigwedge_{i=1}^{k} \bigwedge_{j=1}^{\nu_i} (x_i^j \vee \gamma_i) \wedge \bigwedge_{i=1}^{\ell} (y_i^1 \vee y_i^2)$; clauses $(x_i^j \vee \gamma_i)$ encode the implications $\neg x_i^j \rightarrow \gamma_i$
- $H = Lits(Vars(\varphi))$
- $M = \bigwedge_{i=1}^{k} \gamma_i$.

We show that ψ has a model if and only if $\mathscr{P}$ has an explanation. Assume first that ψ has a model m; then it is easily seen that $E = \{\ell \mid m \models \ell\}$ is an explanation for $\mathscr{P}$. Now assume $\mathscr{P}$ has an explanation E. Then from $\forall i = 1, \ldots, k, T \wedge \bigwedge E \models \gamma_i$ it follows that for every $i = 1, \ldots, k$ E contains at least one $\neg x_i^j$, and thus any assignment satisfying E satisfies every negative clause of ψ; on the other hand, since $T \wedge \bigwedge E$ is satisfiable there is a model m of $T \wedge \bigwedge E$ that satisfies every positive clause of ψ, and this m thus satisfies ψ. □

The following proposition can be shown with the same proof as Proposition 8 with all variables renamed.

Proposition 9 *Let* $R_{\neg x \vee \neg y} = \{(1,0),(0,1),(0,0)\}$. ABDUCTION($\{R_{\neg x \vee \neg y}\}$) *is NP-complete.*

We finally prove that ABDUCTION(Γ) is NP-complete for a particular affine language. This will be achieved by reducing to it another important problem in nonmonotonic reasoning, namely the inference problem for propositional circumscription. A model $m = (m_1, \ldots, m_n)$ of a formula φ is said to be a *minimal model* of φ if there is no model $m' = (m'_1, \ldots, m'_n)$ of φ such that $m \neq m'$ and $\forall i = 1, \ldots, n, m'_i \leq m_i$.

Durand and Hermann proved that the inference problem for propositional circumscription of affine formulas is coNP-complete. In the process, they proved the following theorem.

Theorem 10 ([Durand and Hermann, 2003]) *The problem of deciding whether there is a minimal model of a given affine formula φ that does not satisfy a given negative clause $(\neg q_1 \vee \ldots \vee \neg q_n)$ ($\forall i = 1, \ldots, n, q_i \in Vars(\varphi)$) is NP-complete.*

A careful reading of their proof shows that the theorem remains true even if the linear equations in the input affine formulas are all restricted to contain at most 6 variables. We thus define the language Γ_{6aff} to be the set of all k-ary affine relations with $k \leq 6$. Obviously, Γ_{6aff} is finite, which is necessary for problem ABDUCTION(Γ_{6aff}) to be well-defined.

Proposition 11 ABDUCTION(Γ_{6aff}) *is NP-complete.*

Proof Membership in NP follows from Theorem 4 and Lemma 6. As for hardness, let φ be a formula and $q_1, \ldots, q_n \in Vars(\varphi)$. We show that the clause $(\neg q_1 \vee \ldots \vee \neg q_n)$ is false in some minimal model of φ if and only if the abduction problem with $T = \varphi$, $M = \{q_1, \ldots, q_n\}$, and $H = \{\neg x \mid x \in Vars(\varphi) \backslash \{q_1, \ldots, q_n\}\}$ has an explanation, which will conclude by Theorem 10 and the above remark.

Assume first that $(\neg q_1 \vee \ldots \vee \neg q_n)$ is false in a minimal model m of φ. Define E to be $\{\neg x_i \mid m \models \neg x_i\}$. Since $m \models \varphi$ by assumption and $m \models \bigwedge E$ by construction, $\varphi \wedge \bigwedge E$ is satisfiable. Now assume for sake of contradiction that there is m' satisfying $\varphi \wedge \bigwedge E \wedge (\neg q_1 \vee \ldots \vee \neg q_n)$. Then since $m' \models \bigwedge E$ and E is negative we get $\forall x \in E, m'(x) \leq m(x)$; now for $x \in Vars(\varphi) \setminus Vars(E)$ we have by assumption $m(x) = 1$ and thus $m'(x) \leq m(x)$ again. Finally, we have $\exists q_i, m'(q_i) = 0 < 1 = m(q_i)$, which contradicts the minimality of m. Thus $\varphi \wedge \bigwedge E \models (q_1 \wedge \ldots \wedge q_n)$ and E is an explanation.

Conversely, assume that E is an explanation. Then $\varphi \wedge \bigwedge E$ is satisfiable; write m for one of its minimal models. By assumption the formula $\varphi \wedge \bigwedge E \wedge (\neg q_1 \vee \ldots \vee \neg q_n)$ is unsatisfiable, thus $m \not\models (\neg q_1 \vee \ldots \vee \neg q_k)$. We also have $m \models \varphi$ by assumption. Finally, assume for sake of contradiction that m is not a minimal model of φ, and let m' be such that $m' \models \varphi$, $m' \leq m$ and $m' \neq m$. Then since E is negative (because H is) and $m \models \bigwedge E$ we have $m' \models \bigwedge E$, thus $m' \models \varphi \wedge \bigwedge E$, which contradicts the minimality of m among the models of $\varphi \wedge \bigwedge E$. □

5 Classification

We finally put together the results in the previous sections for obtaining our complete classification. The concept of a *relational clone* is central to our approach.

Definition 12 (relational clone) *Let $\Gamma \subseteq BR$. The relational clone of Γ is written $\langle \Gamma \rangle$ and is the set of all relations that can be expressed using relations from $\Gamma \cup \{=\}$ ($=$ is the equality relation on $\{0,1\}$), conjunction, and existential quantification.*

Intuitively, the constraints over $\langle \Gamma \rangle$ are those which can be simulated by constraints over Γ.

The following result states that when studying the complexity of ABDUCTION(Γ) it is enough to consider constraint languages that are relational clones.

Lemma 13 *Let Γ be a finite constraint language and $\Gamma' \subseteq \langle \Gamma \rangle$ finite. Then* ABDUCTION(Γ') *is polynomial-time reducible to* ABDUCTION(Γ).

Proof Let (V', H', M', T') be an instance of ABDUCTION(Γ'). By the definition of a relational clone there is a set of variables W disjoint from V' and a theory $T_=$ over $\Gamma \cup \{=\}$ with $Vars(T_=) = V_= = V' \cup W$ and such that T' is logically equivalent to the formula $\exists W, T_=$. Since W is disjoint from V' there is no variable occurring in H' or M' and in W at the same time, and it is then easily seen that the abduction problem $(V_=, H', M', T_=)$ has an explanation if and only if (V', H', M', T') has one. Now for every constraint $(x_i = x_j)$ $(i < j)$ it is enough to replace x_j with x_i everywhere in $V_=, T_=, M'$ and H' and to remove the constraint from $T_=$ for obtaining a still equivalent instance (V, H, M, T) of ABDUCTION(Γ), which concludes. □

We can reduce even further the set of constraints languages to be considered, namely to idempotent ones.

Lemma 14 *Let Γ be a finite constraint language.* ABDUCTION(Γ^{id}) *is polynomial time reducible to* ABDUCTION(Γ).

Proof Let $\mathscr{P} = (V, H, M, T)$ be an instance of ABDUCTION(Γ^{id}). We build an instance $\mathscr{P}' = (V, H', M', T')$ of ABDUCTION(Γ) by removing every constraint $F(x)$ or $T(x)$ from T and adding it to H and M. It is then easy to see that (V, H', M', T') has an explanation if and only if (V, H, M, T) has one. □

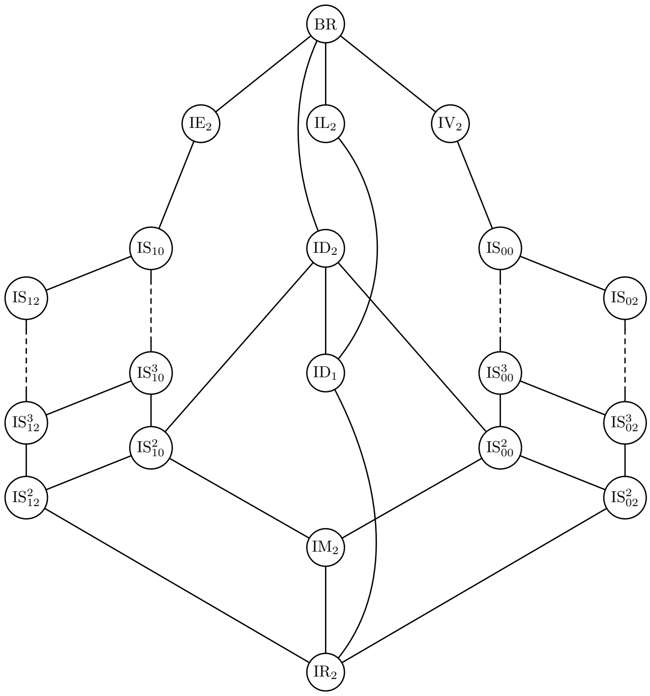

Figure 1: Lattice of all idempotent Boolean relational clones.

Given these two lemmas, our classification of the complexity of ABDUCTION(Γ) heavily relies on Post's remarkable classification of all Boolean relational clones [Post, 1941]. Post proved in particular that the relational clones form a lattice under set inclusion. An excellent introduction to Post's lattice can be found in the recent survey articles [Böhler *et al.*, 2003; 2004].

Lemmas 13 and 14 say that for any finite $\Gamma' \subseteq \langle \Gamma^{id} \rangle$, ABDUCTION($\Gamma'$) is polynomial-time reducible to ABDUCTION(Γ). In other words, when studying the complexity of ABDUCTION(Γ) it is enough to consider constraint languages that are idempotent relational clones. The lattice of all idempotent Boolean relational clones is given on Figure 1. Those that are most relevant to our classification are the following:

- BR, the set of all Boolean relations,
- IE_2, the set of all Horn relations,
- IV_2, the set of all dual Horn relations,
- ID_2, the set of all bijunctive relations,
- IL_2, the set of all affine relations,
- ID_1, the set of all affine relations of width 2,
- IM_2, the set of all relations that are Horn and dual Horn.

Thus, according to Post's lattice there is only one idempotent relational clone that is not Horn, not dual Horn, not affine, and not bijunctive, namely the relational clone consisting of all Boolean relations BR. Hence the following result follows intuitively from the result due to Eiter and Gottlob [1995] stating that ABDUCTION is Σ_2^{P}-complete for the general case of theories given by CNF formulas.

Proposition 15 *If Γ is not Horn, not dual Horn, not affine and not bijunctive then* ABDUCTION(Γ) *is Σ_2^{P}-complete.*

Proof It is well-known that for any CNF formula ψ there is a set of variables W disjoint from $Vars(\psi)$ and a CNF formula ψ' over $Vars(\psi) \cup W$ with at most 3 variables per clause such that the formulas ψ and $\exists W \psi'$ are logically equivalent. That fact together with a proof similar to that of Lemma 13 show that the abduction problem for general CNF theories reduces to ABDUCTION(Γ_3), where Γ_3 is the (finite) set of all ternary relations. Since Γ_3 is not Horn, not dual Horn, not affine and not bijunctive, we have $\langle \Gamma_3^{id} \rangle = BR$, and Lemma 13 and Lemma 14 concludes. □

We are finally able to completely classify the complexity of ABDUCTION(Γ).

Theorem 16 (classification) *Let Γ be a constraint language.* ABDUCTION(Γ) *is:*

- *In P if Γ is affine of width 2,*
- *Otherwise, NP-complete if Γ is Horn, dual Horn, bijunctive or affine,*
- *Otherwise, Σ_2^{P}-complete.*

Proof Proposition 5 shows the result for languages that are affine of width 2. Now it can be seen that the relations $R_{\neg x \vee y}$, $R_{x \vee y}$ and $R_{\neg x \vee \neg y}$ of Propositions 7, 8 and 9 are in the relational clones IM_2, IS_{02}^2 and IS_{12}^2, respectively; this can be verified by checking that they are invariant under the operations defining the corresponding clones (for more details see [Böhler *et al.*, 2003; 2004]). Moreover, the language Γ_{6aff} of Proposition 11 is affine, thus it is in IL_2. Consequently, Figure 1 shows that the minimal idempotent relational clones that are not affine of width 2, namely IM_2, IS_{02}^2, IS_{12}^2 and IL_2 are NP-complete. On the other hand, we know from Theorem 4 and Lemma 6 that the relational clones IL_2 (affine), ID_2 (bijunctive), IE_2 (Horn) and IV_2 (dual Horn) are in NP. Thus ABDUCTION(Γ) is NP-complete when $\langle \Gamma^{id} \rangle$ contains IM_2, IS_{02}^2, IS_{12}^2 or IL_2 and is contained in IL_2, ID_2, IE_2, or IV_2. This covers exactly the languages that are Horn, dual Horn, bijunctive, or affine and that are not affine of width 2.

Finally, Proposition 15 concludes the proof. □

6 Discussion and future work

We have completely classified the complexity of propositional abduction in Schaefer's framework when manifestations and hypotheses are described by sets of literals. This result can prove useful in helping the designers of knowledge based systems to deal with the expressivity/tractability trade-off when choosing a language for their system. Our result indeed completes the picture of the complexity of reasoning for propositional constraint languages. In particular, we have shown that this problem is very hard, in the sense that only languages that are affine of width 2 allow for polynomial abduction. Also note that in many cases NP-hardness remains even when restricting further the problem; e.g., to $H = Lits(V \setminus Vars(M))$ (see the proofs of Propositions 7–9).

It is important to note that the complexity of abduction for a constraint language given in extension (i.e., by the set of all tuples in every relation) can be determined efficiently; the case of Theorem 16 in which a language falls can indeed be determined efficiently by using the closure properties of the concerned co-clones (see, e.g., [Böhler *et al.*, 2004]).

It would be interesting to try to extend this work into at least three directions. First of all, besides the problem of deciding the existence of an explanation for a given instance, of great importance are the problems of relevance and necessity, which ask whether a given hypothesis is part of at least one (resp. of all) preferred explanation(s). These problems involve a preference criterion which can have a great impact on their complexity; for more details we refer the reader to [Eiter and Gottlob, 1995]. Hence, it would be interesting to investigate the complexity of these problems. In the same vein, Eiter and Makino recently studied the problem of enumerating all the explanations of a Horn abduction problem [Eiter and Makino, 2002]; it would be interesting to try to extend their work to other classes of formulas.

Secondly, although Schaefer's framework is quite general, there are restrictions on propositional formulas that it cannot express. For instance, Eshghi [1993] and del Val [2000a] study such restrictions that yield polynomial cases of the abduction problem. It would thus be of great interest to try to identify still more such tractable classes.

Finally, it would be interesting to study the case where the domains of variables are more general, e.g., for conjunctions of constraints over finite domains.

References

[Amilhastre *et al.*, 2002] J. Amilhastre, H. Fargier, and P. Marquis. Consistency restoration and explanations in dynamic CSPs - application to configuration. *Artificial Intelligence*, 135(1-2):199–234, 2002.

[Böhler *et al.*, 2003] 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.

[Böhler *et al.*, 2004] E. Böhler, N. Creignou, S. Reith, and H. Vollmer. Playing with boolean blocks, part II: Constraint satisfaction problems. *ACM SIGACT-Newsletter*, 35(1):22–35, 2004.

[Buchler, 1955] J. Buchler. *Philosophical Writings of Peirce*. Dover, New York, 1955.

[Bylander *et al.*, 1991] T. Bylander, D. Allemang, M. Tanner, and J. Josephson. The computational complexity of abduction. *Artificial Intelligence*, 49:25–60, 1991.

[Console *et al.*, 1991] L. Console, D. Theseider Dupre, and P. Torasso. On the relationship between abduction and deduction. *J. Logic and Computation*, 1(5):661–690, 1991.

[Cook, 1971] S. Cook. The complexity of theorem proving procedures. In *Proc. STOC'71*, pages 151–158, 1971.

[Creignou *et al.*, 2001] N. Creignou, S. Khanna, and M. Sudan. *Complexity classifications of Boolean constraint satisfaction problems*. SIAM Monographs on Discrete Mathematics and Applications, 2001.

[del Val, 2000a] A. del Val. The complexity of restricted consequence finding and abduction. In *Proc. AAAI'00*, pages 337–342, 2000.

[del Val, 2000b] A. del Val. On some tractable classes in deduction and abduction. *Artificial Intelligence*, 116(1-2):297–313, 2000.

[Durand and Hermann, 2003] A. Durand and M. Hermann. The inference problem for propositional circumscription of affine formulas is coNP-complete. In *Proc. STACS'03*, pages 451–462, 2003.

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

[Eiter and Makino, 2002] T. Eiter and K. Makino. On computing all abductive explanations. In *Proc. AAAI'02*, pages 62–67, 2002.

[Eshghi, 1993] K. Eshghi. A tractable class of abduction problems. In *Proc. IJCAI'93*, pages 3–8, 1993.

[Garey and Johnson, 1979] M.R. Garey and D.S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Freeman, San Francisco, 1979.

[Hobbs *et al.*, 1993] J. Hobbs, M. Stickel, D. Appelt, and P. Martin. Interpretation as abduction. *Artificial Intelligence*, 63:69–142, 1993.

[Ladner, 1975] R. Ladner. On the structure of polynomial time reducibility. *J. of the ACM*, 22:155–171, 1975.

[Marquis, 2000] P. Marquis. *Handbook of Defeasible Reasoning and Uncertainty Management Systems (DRUMS)*, volume 5, chapter Consequence finding algorithms, pages 41–145. Kluwer Academic, 2000.

[Morgan, 1971] C. Morgan. Hypothesis generation by machine. *Artificial Intelligence*, 2:179–187, 1971.

[Pople, 1973] H. Pople. On the mechanization of abductive logic. In *Proc. IJCAI'73*, pages 147–152, 1973.

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

[Schaefer, 1978] T.J. Schaefer. The complexity of satisfiability problems. In *Proc. STOC'78*, pages 216–226, 1978.

[Selman and Levesque, 1990] B. Selman and H. Levesque. Abductive and default reasoning: A computational core. In *Proc. AAAI'90*, pages 343–348, 1990.

[Stumptner and Wotawa, 2001] M. Stumptner and F. Wotawa. Diagnosing tree-structured systems. *Artificial Intelligence*, 127:1–29, 2001.

[Zanuttini and Hébrard, 2002] B. Zanuttini and J.-J. Hébrard. A unified framework for structure identification. *Information Processing Letters*, 81(6):335–339, 2002.

[Zanuttini, 2003] B. Zanuttini. New polynomial classes for logic-based abduction. *J. Artificial Intelligence Research*, 19:1–10, 2003.