Complexity results for propositional closed world reasoning and circumscription from tractable knowledge bases

Sylvie Coste-Marquis
CRIL/Université d'Artois et IUT de Lens
rue de l'Université, S.P. 16
62307 Lens cedex - France

Pierre Marquis
CRIL/Université d'Artois
rue de l'Université, S.P. 16
62307 Lens cedex - France

Abstract

This paper presents new complexity results for propositional closed world reasoning (CWR) from tractable knowledge bases (KBs). Both (basic) CWR, generalized CWR, extended generalized CWR, careful CWR and extended CWR (equivalent to circumscription) are considered. The focus is laid on tractable KBs belonging to target classes for exact compilation functions: Blake formulas, DNFs, disjunctions of Horn formulas, and disjunctions of renamable Horn formulas. The complexity of inference is identified for all the forms of CWR listed above. For each of them, new tractable fragments are exhibited. Our results suggest knowledge compilation as a valuable approach to deal with the complexity of CWR in some situations.

1 Introduction

Closed world reasoning (CWR) is a widely used inference technique in Artificial Intelligence, Database Theory and Logic Programming. It relies on the idea that negative information is often not represented in an explicit way; in this situation, every piece of positive information which cannot be deduced from a knowledge base (KB) is assumed false.

In order to define CWR in a formal way, an approach consists in characterizing the formulas which must be assumed false in the KB: CWR is then viewed as deduction from the closure of the KB, i.e., the KB completed with these assumptions. Several policies for characterizing such assumptions have been developed so far, giving rise to several forms of CWR. Let us mention the (basic) closed world assumption (CWA) [Reiter, 1978], the generalized closed world assumption (GCWA) [Minker, 1982], the extended generalized closed world assumption (EGCWA) [Yahya and Henschen, 1985], the careful closed world assumption (CCWA) [Gelfond and Przymusinska, 1986], and the extended closed world assumption (ECWA) [Gelfond *et al.*, 1989]. The most sophisticated closed world assumption is ECWA; in the propositional case, Gelfond, Przymusinska and Przymusinski [1989] show it is equivalent to circumscription, as defined in [McCarthy, 1986].

The complexity of propositional CWR has already been investigated by several researchers. Eiter and Gottlob [1993] show that CWR is hard in the general case: typically at the second level of the polynomial hierarchy. Cadoli and Lenzerini [1994] focus on the complexity of CWR from various fragments of propositional logic for which clause deduction is tractable, especially the Horn CNF class, the reverse Horn CNF class and the Krom one. For such classes of formulas, the complexity of CWR falls one level down in the polynomial hierarchy.

The aim of this paper is to complete the complexity results pointed out in [Cadoli and Lenzerini, 1994] by focusing on some other tractable fragments of propositional logic. Especially, four classes are considered:

- The *Blake* class is the set of formulas given in prime implicates normal form,
- the *DNF* class is the set of formulas given in disjunctive normal form (DNF),
- the *Horn cover* class is the set of disjunctions of Horn CNF formulas,
- the *renamable Horn cover* class is the set of disjunctions of renamable Horn CNF formulas.

As a main contribution, the complexity of clause inference is identified for each form of CWR and each tractable class listed above. Several new tractable subcases are exhibited for each form of CWR; some new intractable subcases are presented as well.

Interestingly, our tractability results for CWR apply to classes of formulas strictly more expressive than some of the most expressive tractable fragments for CWR pointed out up to now. In particular, some of the classes we focus on include classes considered in [Cadoli and Lenzerini, 1994] as strict subcases. Additionally, the fragments we consider are target classes for exact knowledge compilation functions. Thus, every propositional formula can be turned ("compiled") into a formula from the Blake class (the target class for the compilation function given in [Reiter and de Kleer, 1987]), the DNF class (the target class for the function given in [Schrag, 1996]), the Horn cover class and the renamable Horn cover class (which are specific instances of the class of tractable covers considered in [Boufkhad *et al.*, 1997]). Since CWR from such tractable formulas is shown computationally easier than CWR from unconstrained formulas, our study suggests a two step approach to compute CWR: the KB is first compiled off-line, giving rise to a formula from one of the target classes, then queries are addressed on-line w.r.t. the compiled KB. While such a pre-processing is known as not computationally helpful in the general case, our results indicate that it can prove valuable when CWR from the compiled KB can be achieved in polynomial time, as long as the KB does not often change and the size of its compiled form remains "small enough".

The rest of this paper is organized as follows. Some formal preliminaries are given in Section 2. Basic definitions and complexity results about CWR are recalled in Section 3. Section 4 presents the complexity of CWR for each of the fragments we focus on. Section 5 concludes the paper. Proofs are given in [Coste-Marquis and Marquis, 1999] available from the authors.

2 Formal Preliminaries

$PROP_{PS}$ denotes the propositional language built up from a denumerable set PS of symbols and the connectives in the standard way. The size of a formula Σ from $PROP_{PS}$, noted $|\Sigma|$, is the number of signs (symbols and connectives) used to write it. Every propositional symbol of PS is also called a positive literal and a negated one a negative literal. For every subset V of PS, L_V (resp. L_V^+, L_V^-) is the set of literals (resp. positive literals, negative literals) built up from the propositional symbols of V. $Var(\Sigma)$ denotes the set of propositional symbols occurring in Σ.

Formulas are interpreted in the classical way. Every set of formulas is interpreted conjunctively. A formula is Horn CNF (resp. reverse Horn CNF) iff it is a CNF formula s.t. every clause in it contains at most one positive (resp. negative) literal. A Krom formula is a CNF formula in which every clause contains at most two literals. A renamable Horn CNF formula Σ is a CNF formula which can be turned into a Horn CNF formula by substituting in a uniform way in Σ some literals of $L_{Var(\Sigma)}$ by their negation.

We assume that the reader is familiar with some basic notions of computational complexity, especially the complexity classes P, NP, and coNP, and the classes Δ_k^p, Σ_k^p and Π_k^p of the polynomial hierarchy PH (see [Papadimitriou, 1994] for details). The class $\mathsf{P}^{X[O(log n)]}$ contains the decision problems which can be solved in polynomial time with no more than $O(log\ n)$ calls to an oracle for deciding a problem $Q \in X$ for "free" (i.e., within a constant time), n representing the size of the problem instance. Let us recall that a decision problem is said at the k^{th} level of PH iff it belongs to Δ_{k+1}^p, and is either Σ_k^p-hard or Π_k^p-hard. It is strongly believed that PH does not collapse (at any level), i.e., is a truly infinite hierarchy (for every integer k, $\mathsf{PH} \neq \Sigma_k^p$).

3 Closed World Reasoning

All the forms of CWR pointed out so far can be characterized through the notions of *closure* and *free for negation formula*. A clause γ is then considered as a (non monotonic) consequence of a KB Σ interpreted under some closed world assumption policy $*CWA$ (where the generic character $*$ can be replaced by G, EG, CC, E or the empty string) iff it is a logical consequence of the closure of Σ w.r.t. the policy. Both the CCWA and ECWA policies require $Var(\Sigma)$ to be partitioned into three sets, P, Q, and Z. P contains the symbols preferred false, Z contains the symbols the truth value of which can vary when trying to falsify the symbols from P, and Q contains the symbols the truth value of which cannot vary.

Definition 3.1 (closure of a KB)
Let $*CWA$ be any closed world assumption policy among (basic) CWA, GCWA, EGCWA, CCWA and ECWA. Let Σ be a formula from $PROP_{PS}$ and $\langle P, Q, Z\rangle$ a partition of $Var(\Sigma)$. The closure $*CWA(\Sigma, \langle P, Q, Z\rangle)$[1] of Σ given $\langle P, Q, Z\rangle$ w.r.t. $*CWA$ is the formula
$\Sigma \cup \{\neg\alpha$; α is a $*CWA$-free for negation formula w.r.t. Σ and $\langle P, Q, Z\rangle\}$.

The $*CWA$-free for negation formulas w.r.t. Σ are the negations of the formulas which are assumed false when they are not deducible from Σ. They vary according to the closed world assumption policy under consideration:

Definition 3.2 (*CWA-free for negation formula)
Let Σ and α be two formulas from $PROP_{PS}$ and let $\langle P, Q, Z\rangle$ be a partition of $Var(\Sigma)$.

- *α is CWA-free for negation iff α is a positive literal s.t. $\Sigma \not\models \alpha$ holds.*
- *α is GCWA-free for negation iff α is a positive literal and for each positive clause γ s.t. $\Sigma \not\models \gamma$ holds, $\Sigma \not\models \alpha \vee \gamma$ holds.*
- *α is EGCWA-free for negation iff α is a conjunction of positive literals and for each positive clause γ s.t. $\Sigma \not\models \gamma$ holds, $\Sigma \not\models \alpha \vee \gamma$ holds.*
- *α is CCWA-free for negation iff α is a literal from L_P^+ and for each clause γ containing only literals from $L_P^+ \cup L_Q$ and s.t. $\Sigma \not\models \gamma$ holds, $\Sigma \not\models \alpha \vee \gamma$ holds.*
- *α is ECWA-free for negation iff $Var(\alpha) \cap Z = \emptyset$ and for each clause γ containing only literals from $L_P^+ \cup L_Q$ and s.t. $\Sigma \not\models \gamma$ holds, $\Sigma \not\models \alpha \vee \gamma$ holds.*

Let us note that every symbol not belonging to $Var(\Sigma)$ is CWA, $GCWA$ and $EGCWA$-free for negation. As to $CCWA$ and $ECWA$, every symbol from $PS \setminus Var(\Sigma)$ is assumed to belong to P.

In the rest of this paper, the following decision problems are considered:

Definition 3.3 (*CWA clause inference)
Let $*CWA$ be any closed world assumption policy among

[1] The partition of $Var(\Sigma)$ is not significant for the CWA, $GCWA$ and $EGCWA$ policies.

(basic) CWA, GCWA, EGCWA, CCWA and ECWA. $*CWA$ CLAUSE INFERENCE *is the following decision problem:*

- **Input:** *A formula* Σ *and a clause* γ *from* $PROP_{PS}$*, a partition* $\langle P, Q, Z \rangle$ *of* $Var(\Sigma)$ *and a CWA policy* $*CWA$*.*
- **Query:** *Does* $*CWA(\Sigma, \langle P, Q, Z \rangle) \models \gamma$ *hold?*

$*CWA$ LITERAL INFERENCE is the restriction of the corresponding $*CWA$ CLAUSE INFERENCE problem where γ is restricted to be a literal.

The complexity of propositional CWR has been investigated by several researchers, especially Eiter and Gottlob [1993] and Cadoli and Lenzerini [1994]. CWR has been shown hard: all the forms of CWR except basic CWA are at the second level of the polynomial hierarchy. Thus, CWR is computationally harder than deduction in the general case (CLAUSE DEDUCTION is "only" coNP-complete), unless PH collapses.

In order to circumvent this complexity, several approaches can be considered. One of them is centered around the idea of exact *knowledge compilation*. Knowledge compilation can be viewed as a form of preprocessing (see [Cadoli and Donini, 1997] for a survey): the original KB is turned into a compiled one during an off-line compilation phase and this compiled KB is used to answer the queries on-line. Assuming that the KB does not often change and that answering queries from the compiled KB is computationally easier than answering them from the original KB, the compilation time can be balanced over a sufficient number of queries.

Existing researches about knowledge compilation can be split into two categories. The first category gathers theoretical works about *compilability*, which indicates whether or not the objective can be expected to be reached in the general case by focusing on the size of the compiled form (see e.g., [Cadoli *et al.*, 1997a; Liberatore, 1998]). Indeed, if the size of the compiled form is exponentially larger than the size of the original KB, significant computational improvements are hard to be expected. Some decision problems are compilable, while others are (probably) not compilable. Thus, CLAUSE DEDUCTION (from a fixed KB) is (probably) not compilable: the existence of an equivalence-preserving compilation function $COMP$ s.t. it is guaranteed that for every propositional CNF formula Σ, CLAUSE DEDUCTION from $COMP(\Sigma)$ is in P and $|COMP(\Sigma)|$ is polynomially bounded in $|\Sigma|$ would make PH to collapse at the second level (see [Kautz and Selman, 1992; Cadoli *et al.*, 1997a] for more details). The second category contains works that are much more oriented towards the design of compilation algorithms, and their empirical evaluations (see e.g. [Reiter and de Kleer, 1987; Schrag, 1996; Boufkhad *et al.*, 1997; Boufkhad, 1998]).

The compilability of CWR has been analyzed in depth in [Cadoli *et al.*, 1996; 1997b]. The results are typically negative. In a nutshell, inference under ECWA from a fixed KB is shown (probably) not compilable. Contrastingly, it is shown that CWA CLAUSE INFERENCE and CCWA CLAUSE INFERENCE from Horn CNF, reverse Horn CNF or Krom formulas are compilable (the fixed part of the problem is the KB plus the partition in the CCWA case).

From the practical side, Nerode, Ng and Subrahmanian [1995] present an algorithm for compiling a KB interpreted under circumscription into the set of its minimal models. The algorithm is based on a mixed integer linear programming. The circumscription policy (i.e., the partition of the symbols) is fixed. Several other approaches show how CWR can be reduced to deduction through the computation of the closure of the KB (or the computation of any equivalent formula), see e.g. [Raiman and de Kleer, 1992; Castell *et al.*, 1996]. All these approaches confirm the (probable) non compilability of the most sophisticated forms of CWR in the sense that in the worst case, the compilation phase outputs a formula that is not polynomially bounded in the size of the original KB.

4 Complexity Results

In the following, the complexity of CWR is investigated for some KBs for which CLAUSE DEDUCTION is tractable (tractable KBs for short):

Definition 4.1 (some tractable classes)
Let Σ *be a formula from* $PROP_{PS}$*.*

- Σ *is a* Blake formula *iff* Σ *is a CNF formula and for every implicate* γ *of* Σ*, there exists a clause* π *in* Σ *s.t.* $\pi \models \gamma$ *holds.*
- Σ *is a* DNF formula *iff* Σ *is a (finite) disjunction of terms.*
- Σ *is a* Horn cover formula *iff* Σ *is a (finite) disjunction of Horn CNF formulas.*
- Σ *is a* renamable Horn cover formula *iff* Σ *is a (finite) disjunction of renamable Horn CNF formulas.*

Blake formulas [Blake, 1937] are CNF formulas given by their prime implicates. Clearly enough, they are tractable since for every clause γ, $\Sigma \models \gamma$ holds iff there exists π in Σ s.t. $\pi \models \gamma$ holds and this test can be easily achieved in time polynomial in $|\pi| + |\gamma|$. Indeed, many algorithms for computing prime implicates have been proposed so far (see [Marquis, 1999] for a survey).

DNF formulas are tractable since for every clause γ, $\Sigma \models \gamma$ holds iff for every term δ in Σ, $\delta \models \gamma$ holds and this test can be easily achieved in time polynomial in $|\delta| + |\gamma|$. Several algorithms for turning a formula into DNF can be found in the literature, e.g. [Schrag, 1996].

Horn (resp. renamable Horn) cover formulas are tractable since for every clause γ, $\Sigma \models \gamma$ holds iff for every Φ in Σ, $\Phi \models \gamma$ holds and this test can be easily achieved in time polynomial in $|\Phi| + |\gamma|$ when Φ is a Horn CNF formula [Dowling and Gallier, 1984] (resp. a renamable Horn CNF formula [Lewis, 1978]). Note that Horn CNF formulas and renamable Horn CNF formulas can be recognized (and checked for satisfiability) in polynomial time. Note also that every term is a Horn CNF formula, every Horn CNF formula and every satisfiable Krom formula is a renamable Horn CNF formula, and every term (resp. Horn CNF formula, renamable Horn CNF formula) is a DNF formula (resp. a Horn cover formula, a renamable Horn cover formula). Hence every DNF formula is a Horn cover formula and every Horn cover formula is a renamable Horn cover formula. Algorithms for computing Horn cover formulas and renamable Horn cover formulas can be found in [Boufkhad *et al.*, 1997; Boufkhad, 1998].

The next proposition is the central result of this paper:

Proposition 4.1 (complexity of CWR)
The complexity of $*CWA$ CLAUSE INFERENCE *and* $*CWA$ LITERAL INFERENCE *from a Blake formula, a DNF formula, a Horn cover formula and a renamable Horn cover formula is reported in Tables 1 and 2*[2].

Intuitively, the fact that CWR typically is at the second level of PH can be explained by the presence of two independent sources of complexity. One of them lies in deduction and the other one in model minimization. As Proposition 4.1 illustrates it, focusing on tractable KBs enables ruling out one source of difficulty in the general case, and both sources in some specific cases.

This work can be related to several studies in which the complexity of non monotonic inference from tractable fragments of propositional logic is analyzed, for instance [Kautz and Selman, 1991; Cadoli and Lenzerini, 1994; Ben-Eliyahu and Dechter, 1996]. Cadoli and Lenzerini [1994] investigate the complexity of CWR from Horn CNF, reverse Horn CNF and Krom formulas. Interestingly, the tractable classes for CWR we consider are strictly more expressive than some of the most expressive tractable classes pointed out up to now. Especially, they include some of the tractable classes given in [Cadoli and Lenzerini, 1994] as subcases. Thus, every Krom formula can be turned into its prime implicates normal form in polynomial time (just because a set of binary clauses has a polynomially bounded number of resolvents). Additionally, every Horn CNF formula belongs to the Horn cover class. However, the converse does not hold: the fragments studied here are strictly more expressive that those given in [Cadoli and Lenzerini, 1994]. For instance, every monotonic CNF formula (i.e., a CNF formula in which every symbol occurs only positively or negatively) can be put into prime implicates normal form in polynomial time but is not equivalent to a Horn CNF or a Krom formula in the general case.

Another significant difference w.r.t. [Cadoli and Lenzerini, 1994] is that our study is directly related to exact knowledge compilation in the sense that the fragments of propositional logic we consider are target classes for exact compilation functions. Thus, every propositional formula can be turned into a Blake formula, a DNF formula, a Horn cover formula and a renamable Horn cover formula without modifying the set of its models. This contrasts with the fragments considered in [Cadoli and Lenzerini, 1994] since some formulas cannot be represented either as a Horn CNF, a reverse Horn CNF or a Krom formula while preserving logical equivalence.

Accordingly, the tractability results given in Proposition 4.1 can be exploited to draw some new conclusions about the usefulness of knowledge compilation for CWR. From the theoretical side, the compilability results for CWR given in [Cadoli *et al.*, 1996] can be completed at the light of our results. Indeed, every polynomially solvable problem is compilable, even if its fixed part is empty since the compilation phase can be achieved in polynomial time. More interestingly, from the practical side, the connection with knowledge compilation suggests a two step process for computing CWR: the KB is first made tractable through knowledge compilation, then CWR is achieved from the compiled KB. The first step is performed off-line and done once at all (unless the KB is modified). Using such an approach, the closed world assumption policy adopted can vary with the queries that are considered; for instance, the set of symbols to be minimized can change with time, without requiring the KB to be re-compiled each time such a modification occurs. This flexibility is particularly interesting when the KB, representing "hard" constraints, is shared by several agents, and different agents may have different preferences, encoded as a CWA policy. Each time an agent asks the KB, the corresponding preferences can be taken into account without requiring any re-compilation of the KB.

At the light of the complexity results given in Proposition 4.1, our claim is that *some* approaches to knowledge compilation can prove helpful in practice for *some* forms of CWR, *provided that the compiled KB remains "small enough"*. For instance, compiling a KB into a Horn cover formula so as to interpret it under EGCWA reduces the complexity of inference from Π_2^p-complete to P. Thus, if the compilation phase does not result in an exponential blow up in the size of the KB, the time needed for inference from the compiled KB can be much lower than the corresponding time from the original KB; subsequently, the compilation time can be balanced. Let us stress that our claim concerns some specific cases only, not the general one: since CWR is (probably) not compilable [Cadoli *et al.*, 1996; 1997b], knowledge compilation cannot be expected as valuable for improving CWR in the general case. Nevertheless, it is worth noting that, while compilability results for CWR are mainly negative, they do not prevent from exhibiting some instances for which knowledge compilation proves helpful. Accordingly, many experiments with various compilation functions show the practical utility of such an approach for improving CLAUSE DEDUCTION (see e.g., [Schrag, 1996; Boufkhad *et al.*, 1997]), though the problem is known as

[2] Some tractability results reported in the two tables can be generalized to any tractable KB. To be more precise, CWA CLAUSE INFERENCE and both CCWA CLAUSE INFERENCE and ECWA CLAUSE INFERENCE under the restriction $P = \emptyset$ are in P whenever Σ belongs to a class for which CLAUSE DEDUCTION is tractable.

Tractable KB	***CWA**	CLAUSE INFERENCE	LITERAL INFERENCE
Blake	CWA	in P	in P
Blake	GCWA	in P	in P
Blake	EGCWA	coNP-complete	in P
Blake	CCWA	in P	in P
Blake	ECWA	coNP-complete	coNP-complete
DNF	CWA	in P	in P
DNF	GCWA	in P	in P
DNF	EGCWA	in P	in P
DNF	CCWA	coNP-hard and in $\mathsf{P}^{\mathsf{NP}[\mathsf{O}(\log n)]}$	coNP-hard and in $\mathsf{P}^{\mathsf{NP}[\mathsf{O}(\log n)]}$
DNF	ECWA	coNP-complete	coNP-complete
Horn cover	CWA	in P	in P
Horn cover	GCWA	in P	in P
Horn cover	EGCWA	in P	in P
Horn cover	CCWA	coNP-hard and in $\mathsf{P}^{\mathsf{NP}[\mathsf{O}(\log n)]}$	coNP-hard and in $\mathsf{P}^{\mathsf{NP}[\mathsf{O}(\log n)]}$
Horn cover	ECWA	coNP-complete	coNP-complete
renamable Horn cover	CWA	in P	in P
renamable Horn cover	GCWA	coNP-hard and in $\mathsf{P}^{\mathsf{NP}[\mathsf{O}(\log n)]}$	coNP-hard and in $\mathsf{P}^{\mathsf{NP}[\mathsf{O}(\log n)]}$
renamable Horn cover	EGCWA	coNP-complete	coNP-complete
renamable Horn cover	CCWA	coNP-hard and in $\mathsf{P}^{\mathsf{NP}[\mathsf{O}(\log n)]}$	coNP-hard and in $\mathsf{P}^{\mathsf{NP}[\mathsf{O}(\log n)]}$
renamable Horn cover	ECWA	coNP-complete	coNP-complete

Table 1: The complexity of CWR from some tractable KBs.

(probably) not compilable. Our results show that, in the many situations in which some compilation techniques are computationally profitable for CLAUSE DEDUCTION, they can be profitable as well for some forms of CWR, with only a polynomial extra cost.

5 Conclusion

In this paper, the complexity of CWR has been investigated for some tractable fragments of propositional logic. Several new tractable cases have been identified, and some new intractability results have been provided as well. Interestingly, the tractable fragments we have focused on are target classes for some compilation functions; such a connection with exact knowledge compilation can prove valuable from a computational point of view in the situations in which the size of the compiled KB remains "small enough".

Due to close connections between CWR and abduction and between CWR and some simple forms of default reasoning, complexity results for these additional forms of non monotonic inference from tractable KBs can be derived from the results presented in this paper. This is an issue for further research.

Acknowledgements

This work has been supported by a "Contrat d'objectifs de la Région Nord Pas-de-Calais".

References

[Ben-Eliyahu and Dechter, 1996] R. Ben-Eliyahu and R. Dechter. Default reasoning using classical logic. *Artificial Intelligence*, 84:113–150, 1996.

[Blake, 1937] A. Blake. *Canonical expressions in Boolean algebra*. PhD thesis, University of Chicago, Chicago (IL), 1937.

[Boufkhad *et al.*, 1997] Y. Boufkhad, E. Grégoire, P. Marquis, B. Mazure, and L. Saïs. Tractable cover compilations. In *Proc. of IJCAI'97*, pages 122–127, Nagoya, 1997.

[Boufkhad, 1998] Y. Boufkhad. Algorithms for propositional KB approximation. In *Proc. of AAAI'98*, pages 280–285, Madison (WI), 1998.

[Cadoli and Donini, 1997] M. Cadoli and F.M. Donini. A survey on knowledge compilation. *AI Communications*, 10:137–150, 1997. printed in 1998.

[Cadoli and Lenzerini, 1994] M. Cadoli and M. Lenzerini. The complexity of propositional closed world reasoning and circumscription. *Journal of Computer and System Sciences*, 48:255–310, 1994.

[Cadoli *et al.*, 1996] M. Cadoli, F.M. Donini, and M. Schaerf. Is intractability of non-monotonic reasoning a real drawback? *Artificial Intelligence*, 88:215–251, 1996.

[Cadoli *et al.*, 1997a] M. Cadoli, F.M. Donini, P. Liberatore, and M. Schaerf. Preprocessing of intractable problems. Technical Report DIS 24-97, Dipartimento di Informatica e Sistemistica, Università di Roma "La Sapienza", 1997.

[Cadoli *et al.*, 1997b] M. Cadoli, F.M. Donini, M. Schaerf, and R. Silvestri. On compact representations of propositional circumscription. *Theoretical Computer Science*, 182:183–202, 1997.

Tractable KB	***CWA**	CLAUSE INFERENCE	LITERAL INFERENCE
Blake	ECWA	in P when $P = \emptyset$	in P when $P = \emptyset$ or $Z = \emptyset$
Blake	ECWA	coNP-hard when $Q = Z = \emptyset$	coNP-hard when $Q = \emptyset$
DNF	CCWA	in P when $P = \emptyset$ or $Q = \emptyset$	in P when $P = \emptyset$ or $Q = \emptyset$
DNF	CCWA	coNP-hard when $Z = \emptyset$	coNP-hard when $Z = \emptyset$
DNF	ECWA	in P when $P = \emptyset$ or $Q = \emptyset$	in P when $P = \emptyset$ or $Q = \emptyset$
DNF	ECWA	coNP-hard when $Z = \emptyset$	coNP-hard when $Z = \emptyset$
Horn cover	CCWA	in P when $P = \emptyset$ or $Q = \emptyset$	in P when $P = \emptyset$ or $Q = \emptyset$
Horn cover	CCWA	coNP-hard when $Z = \emptyset$	coNP-hard when $Z = \emptyset$
Horn cover	ECWA	in P when $P = \emptyset$ or $Q = \emptyset$	in P when $P = \emptyset$ or $Q = \emptyset$
Horn cover	ECWA	coNP-hard when $Z = \emptyset$	coNP-hard when $Z = \emptyset$
renamable Horn cover	CCWA	in P when $P = \emptyset$	in P when $P = \emptyset$
renamable Horn cover	CCWA	coNP-hard when $Q = Z = \emptyset$	coNP-hard when $Q = Z = \emptyset$
renamable Horn cover	ECWA	in P when $P = \emptyset$	in P when $P = \emptyset$
renamable Horn cover	ECWA	coNP-hard when $Q = Z = \emptyset$	coNP-hard when $Q = Z = \emptyset$

Table 2: Complexity bounds for CCWA and ECWA inference from tractable KBs in some restricted cases.

[Castell *et al.*, 1996] Th. Castell, C. Cayrol, M. Cayrol, and D. Le Berre. Using the Davis and Putnam procedure for an efficient computation of preferred models. In *Proc. of ECAI'96*, pages 350–354, Budapest, 1996.

[Coste-Marquis and Marquis, 1999] S. Coste-Marquis and P. Marquis. Complexity results for propositional closed world reasoning and circumscription from tractable knowledge bases. Technical Report CRIL/99-SC/PM-01, CRIL, Université d'Artois, 1999.

[Dowling and Gallier, 1984] W. Dowling and J.H. Gallier. Linear time algorithms for testing the satisfiability of propositional Horn formulae. *Journal of Logic Programming*, 1(3):267–284, 1984.

[Eiter and Gottlob, 1993] Th. Eiter and G. Gottlob. Propositional circumscription and extended closed-world reasoning are $\Pi_2{}^P$-complete. *Theoretical Computer Science*, 114:231–245, 1993.

[Gelfond and Przymusinska, 1986] M. Gelfond and H. Przymusinska. Negation as failure: careful closure procedure. *Artificial Intelligence*, 30:273–287, 1986.

[Gelfond *et al.*, 1989] M. Gelfond, H. Przymusinska, and T. Przymusinski. On the relationship between circumscription and negation as failure. *Artificial Intelligence*, 38:49–73, 1989.

[Kautz and Selman, 1991] H.A. Kautz and B. Selman. Hard problems for simple default logic. *Artificial Intelligence*, 49:243–279, 1991.

[Kautz and Selman, 1992] H.A. Kautz and B. Selman. Forming concepts for fast inference. In *Proc. of AAAI'92*, pages 786–793, San Jose (CA), 1992.

[Lewis, 1978] H.R. Lewis. Renaming a set of clauses as a Horn set. *Journal of the Association for Computing Machinery*, 25:134–135, 1978.

[Liberatore, 1998] P. Liberatore. On the compilability of diagnosis, planning, reasoning about actions, belief revision, etc. In *Proc. of KR'98*, pages 144–155, Trento, 1998.

[Marquis, 1999] P. Marquis. *Consequence finding algorithms*, volume 5 of *The Handbook for Uncertain and Defeasible Reasoning*, chapter Algorithms for Uncertain and Defeasible Reasoning. Kluwer Academic Publishers, 1999. (to appear).

[McCarthy, 1986] J. McCarthy. Applications of circumscription to formalizing common-sense knowledge. *Artificial Intelligence*, 28:89–116, 1986.

[Minker, 1982] J. Minker. On indefinite databases and the closed world assumption. In *Proc. of CADE'82*, pages 292–308, 1982.

[Nerode *et al.*, 1995] A. Nerode, R.T. Ng, and V.S. Subrahmanian. Computing circumscriptive databases. I: Theory and algorithms. *Information and Computation*, 116:58–80, 1995.

[Papadimitriou, 1994] Ch. H. Papadimitriou. *Computational complexity.* Addison–Wesley, 1994.

[Raiman and de Kleer, 1992] O. Raiman and J. de Kleer. A minimality maintenance system. In *Proc. of KR'92*, pages 532–538, Cambridge (MA), 1992.

[Reiter and de Kleer, 1987] R. Reiter and J. de Kleer. Foundations of assumption-based truth maintenance systems: preliminary report. In *Proc. of AAAI'87*, pages 183–188, Seattle (WA), 1987.

[Reiter, 1978] R. Reiter. *Logic and Data Bases*, chapter On closed world data bases, pages 119–140. Plenum Press, 1978.

[Schrag, 1996] R. Schrag. Compilation for critically constrained knowledge bases. In *Proc. of AAAI'96*, pages 510–515, Portland (OR), 1996.

[Yahya and Henschen, 1985] A. Yahya and L.J. Henschen. Deduction in non-Horn databases. *Journal of Automated Reasoning*, 1:141–160, 1985.