The Complexity of Closed World Reasoning and Circumscription

Marco Cadoli, Maurizio Lenzerini
Dipartimento di Informatica e Sistemistica
Università di Roma "La Sapienza"
via Buonarroti 12, 00185 Roma, Italia

Abstract

Closed world reasoning is a common nonmonotonic technique that allows for dealing with negative information in knowledge and data bases. We present a detailed analysis of the computational complexity of the different forms of closed world reasoning for various fragments of propositional logic. The analysis allows us to draw a complete picture of the tractability/intractability frontier for such a form of nonmonotonic reasoning. We also discuss how to use our results in order to characterize the computational complexity of other problems related to nonmonotonic inheritance, diagnosis, and default reasoning.

1 Introduction

Closed World Reasoning (CWR) is a common nonmonotonic technique that allows for dealing with negative information in knowledge and data bases.

The simplest form of closed world reasoning is the *(simple) closed world assumption* (CWA), introduced by Reiter [13], which states that a negative ground fact of the form $\neg p$ is inferred from a knowledge base T just in case the corresponding positive fact p cannot be deduced from T. This represents the idea that every positive fact that is not known to be true, should be considered false. A semantic characterization of the CWA can be given in terms of minimal Herbrand models: reasoning with the CWA is equivalent to assuming that every ground fact which is not included in the intersection of all the Herbrand models of T is false.

Starting from the consideration that the simple CWA may lead to inconsistency when the knowledge base contains disjunctive positive assertions, such as $(a \vee b)$, Minker [10] proposed a new form of closed world reasoning, called *generalized closed world assumption* (GCWA), which states that the negative facts to be inferred from a knowledge base T, should be those which do not appear in *any* of the minimal models of T. This principle takes into account that a (non-Horn) theory may have, in general, more than one minimal model, and every such model should be considered in closed world inference.

Gelfond and Przymusinska [4] extend the work of Minker, by allowing the generalized closed world assumption to be applied to a specified set P of (not necessarily all) predicates of the knowledge base. The resulting form of closed world reasoning, called *careful closed world assumption* (CCWA), is shown to be more expressive than the original generalized closed world assumption. In particular, it allows the derivation of new positive facts, which is impossible in both the CWA and the GCWA.

Gelfond, Przymusinska, and Przymusinski [5] deal with an extension of the CCWA, proposing the so-called *extended closed world assumption* (ECWA), which is shown to be the most powerful formalization of closed world reasoning. In particular, they prove the equivalence (at least for propositional theories) between ECWA and circumscription [9].

Although the importance of the above forms of CWR has often been stressed both from a theoretical and a practical point of view, a complete analysis of the computational complexity of CWR is missing. Notice that, on the contrary, such an analysis has been provided for other forms of nonmonotonic reasoning, such as default reasoning [6], abduction [2], and path-based inheritance [17].

The aim of this paper is to present a detailed analysis of the computational complexity of closed world reasoning. In particular, we are interested in exploring the tradeoff between the tractability of the inference problem and the expressive power of the representation language. To this purpose, we concentrate our attention on propositional logic (although our results can be generalized in several ways, as mentioned in Section 4). which provides a representation language which is decidable, and, at least in some case, tractable, and we consider different subclasses of propositional formulae, each one characterized by some syntactic restriction.

In the analysis, we shall refer to the results concerning the computational complexity of CWR which have appeared in the literature since now, namely [1,7,8,16].

The paper is organized as follows. In Section 2 we provide some definitions and results which will be used in the subsequent sections. In Section 3 we present the complexity analysis: the analysis is performed along two directions, one concerning the expressive power of the language, and one concerning the form of CWR. Finally, in Section 4 we discuss possible applications of our results to other reasoning problems.

2 Preliminaries

We always refer to propositional formulae in conjunctive normal form, called CNF formulae or simply formulae. The set of propositional letters appearing in a formula is called its alphabet. A CNF formula is a conjunction of clauses, where a clause is a disjunction of literals, and a literal is either a propositional letter or its negation.

A *closed world reasoning rule* (CWR-rule) is a rule specifying a set of clauses to be conjoined to a formula T in order to obtain the *closure* of T according to some closed world resoning criterion. Some CWR-rule requires the alphabet of the formula to be partitioned into three sets, denoted P, Q, Z respectively. P contains the letters to be minimized, Z contains the letters whose truth value can vary when minimizing the letters in P, and Q contains all the remaining letters. Given a set of letters R, we denote by R^+ (resp. R^-) the set of all positive (resp. negative) literals from R.

All the forms of CWR we consider in this paper can be abstractly characterized as follows.

Definition 1 *Let T be a propositional formula, and $\langle P;Q;Z\rangle$ be a partition of the letters of its alphabet. We define the closure of T with respect to the CWR-rule α as follows:*

$$\alpha(T;P;Q;Z) = T \cup \{\neg K \mid K \text{ is free for negation in } T \text{ with respect to } \alpha\}$$

where K is a formula whose form depends on α.

As a notational convenience, we say that the formula K is α-ffn to mean that it is free for negation with respect to the CWR-rule α in $\langle T;P;Q;Z\rangle$. In order to precisely characterize the different forms of CWR, we now consider every CWR-rule α, specifying what it means for a formula to be α-ffn.

• The CWA-rule corresponds to the (simple) closed world assumption [13]. K is CWA-ffn if K is a positive literal and $T \not\models K$.

• The GCWA-rule corresponds to the generalized closed world assumption [10]. K is GCWA-ffn if K is a positive literal and, for each positive clause B such that $T \not\models B$, it holds that $T \not\models B \vee K$.

• The EGCWA-rule corresponds to the extended generalized closed world assumption [18]. K is EGCWA-ffn if K is a conjunction of positive literals and, for each positive clause B such that $T \not\models B$, it holds that $T \not\models B \vee K$.

• The CCWA-rule corresponds to the careful closed world assumption [4]. The letters of T are partitioned into $\langle P;Q;Z\rangle$. K is CCWA-ffn if K is a positive literal from P and, for each positive clause B whose literals belong to $P^+ \cup Q^+ \cup Q^-$ such that $T \not\models B$, it holds that $T \not\models B \vee K$.

• The ECWA-rule corresponds to the extended closed world assumption [5]. The letters of T are partitioned into $\langle P;Q;Z\rangle$. K is ECWA-ffn if K is an arbitrary formula not involving literals from Z and, for each positive clause B whose literals belong to $P^+ \cup Q^+ \cup Q^-$ such that $T \not\models B$, it holds that $T \not\models B \vee K$.

Notice that in the CWA, GCWA, EGCWA rules, there is no need to partition the letters into P, Q, Z, and therefore, we can simplify the notation and write $CWA(T)$ $GCWA(T)$, and $EGCWA(T)$.

The CWR-rules can also be given a semantical characterization, which is based on the notion of *minimal* model. A model of a formula T is a truth assignment that satisfies T. For any two models M, N of T, we write $M \leq N$ if the set of letters of T which are assigned true by M is a subset of the analogous set for N. Moreover, if $\langle P;Q;Z\rangle$ is a partition of the letters of T, we write $M \leq_{(P;Z)} N$ if M and N assign the same truth value to the letters in Q, and the set of letters of P which are assigned true by M is a subset of the analogous set for N. We say that a model M is minimal for T if there exists no model N of T such that $N \leq M$ and $M \not\leq N$. Analogously, we say that a model M is $(P;Z)$-minimal for T if there exists no model N of T such that $N \leq_{(P;Z)} M$ and $M \not\leq_{(P;Z)} N$. Notice that $(P;Z)$-minimality reduces to minimality when $Q = Z = \emptyset$.

The semantical characterization of freeness for negation is now given by the following properties.

• K is CWA-ffn in T iff there exists a model M of T such that $M \not\models K$.

• K is GCWA-ffn in T iff for each minimal model M of T it holds that $M \not\models K$.

• K is EGCWA-ffn in T iff for each minimal model M of T it holds that $M \not\models K$.

• K is CCWA-ffn in $\langle T;P;Q;Z\rangle$ iff for each $(P;Z)$-minimal model M of T it holds that $M \not\models K$.

• K is ECWA-ffn in $\langle T;P;Q;Z\rangle$ iff for each $(P;Z)$-minimal model M of T it holds that $M \not\models K$.

From the above properties, one can easily show that for any formula F, $EGCWA(T) \models F$ iff for each minimal model M of T it holds that $M \models F$, and $ECWA(T;P;Q;Z) \models F$ iff for each $(P;Z)$-minimal model M of T it holds that $M \models F$. Moreover, it is shown in [5] that for any formula F,

$ECWA(T;P;Q;Z) \models F$ iff $CIRC(T;P;Q;Z) \models F$, where $CIRC(T;P;Q;Z)$ denotes the *circumscription* of P in T with variables Z. In other words, the ECWA is equivalent to circumscription, at least for propositional formulae.

The different forms of CWR are not independent from each other. In the rest of this paper we shall make use of the following facts relating the various CWR-rules.

Fact 1: for each formula F, if T is Horn, i.e. is constituted by clauses with at most one positive literal, then $CWA(T) \models F$ iff $GCWA(T) \models F$ iff $EGCWA(T) \models F$ iff $ECWA(T;P;\emptyset;\emptyset) \models F$ iff $CCWA(T;P;\emptyset;\emptyset) \models F$.

Fact 2: for each formula F, $EGCWA(T) \models F$ iff $ECWA(T;P;\emptyset;\emptyset) \models F$.

Fact 3: for each formula F, $GCWA(T) \models F$ iff $CCWA(T;P;\emptyset;\emptyset) \models F$.

Fact 4: for each literal L belonging to $P^+ \cup P^-$, $CCWA(T;P;Q;Z) \models L$ iff $ECWA(T;P;Q;Z) \models F$.

As we said in the introduction, in this paper we are concerned with CWR in the context of propositional logic (although our results can be generalized in several ways, see Section 4). Since we are interested in the tractability frontier for CWR, we focus our attention on those classes of formulae for which monotonic inference is a polynomial task. The results reported in [15] show that this holds for three classes of formulae, namely, Horn, dual-Horn, and Krom.

In the rest of the paper, we shall refer to various subsets of such classes. A complete classification of all the classes of propositional formulae that we consider is as follows:

- Horn: at most one positive literal per clause
- dual-Horn: at most one negative literal per clause
- Krom: at most two literals per clause – either positive or negative
- Definite: exactly one positive literal per clause
- HornKrom: Horn and Krom
- dual-HornKrom: dual-Horn and Krom
- HornKrom$^-$: HornKrom with no negative clauses having 2 literals
- 2-positive-Krom: exactly two positive literals, and no negative literal, per clause.

The complexity analysis presented in the next section, will be performed along two directions, concerning the expressiveness of the language, and the form of CWR, respectively. For those CWR-rules requiring the letters of a formula T to be partitioned into the three sets P, Q, Z (e.g. CCWA and ECWA), we will consider two special cases, namely $Q = \emptyset$ or $Z = \emptyset$.

3 Complexity Analysis

In this section we consider the different forms of closed world reasoning, and, for each form, we analyze the various classes of formulae, providing lower complexity bounds for the deduction problem (for the sake of brevity, most of the proofs are omitted).

With regard to the upper complexity bounds, we notice that, from the analysis of Schlipf [16] it follows that performing deduction under the CCWA in propositional logic is both an NP-hard and a coNP-hard problem. Hence it is unlikely that the problem is either in NP or in coNP. Obviously, by Fact 4, this applies to the ECWA as well. In fact, we can go one step further, and prove that the deduction problem under the ECWA is in PSPACE, and in particular in the class Π_2^P of the polynomial hierarchy, although we do not know whether the problem is complete for such a class. A simple intuition supporting this fact is that, determining if $ECWA(T;P;Q;Z) \models \gamma$ can be done by checking if γ is satisfied in *all* the truth assignments of T which are $(P;Z)$-minimal models of T, and the problem of checking if a truth assignment M of T is a $(P;Z)$-minimal model is in the class of coNP (it suffices to compare M with *all* the other truth assignments of T). A more detailed analysis of the upper complexity bounds for several classes of propositional formulae is described in [3].

3.1 CWA

The results reported in [13] show that the application of the CWA to Krom and dual-Horn formulae may lead to inconsistency. Moreover, Minker [10] shows that, when the CWA is consistent, it is equivalent to GCWA (see Subsection 3.2).

In [1], it is shown that the deduction problem under CWA for Definite formulae is solvable in polynomial time. Since negative clauses do not contribute to the deduction (see [13]), this property extends to Horn formulae. By Fact 1, one can conclude that, for Horn formulae, the deduction problem under both the GCWA and the EGCWA is polynomial too.

3.2 GCWA

In [7], it is shown that the problem of determining whether $GCWA(T) \models L$, where T is a dual-Horn formula and L is a literal, is coNP-hard. By Facts 1,2, and 3, this holds also for EGCWA, CCWA and ECWA.

The deduction problem under the GCWA for Krom formulae is polynomial; this derives from Fact 3 and from a stronger result which will be presented in Subsection 3.4.

Recently, Rajasekar, Lobo and Minker [12] have proposed a weak form of GCWA, called Weak GCWA (WGCWA), which applies to disjunctive logic programs (i.e. logic programs whose rules have any number of positive literals in the head) and is defined in such a way that every deduction under such a form reduces polinomially to deduction under the CWA on definite logic programs. It is easy to see that the WGCWA yields polynomial time deduction algorithms in the propositional case.

3.3 EGCWA

We already mentioned that the deduction problem under the EGCWA is polynomial for Horn formulae. The following theorem shows that this is the only tractable case.

Theorem 1 *Let T be a 2-positive-Krom formula, and γ any clause. Then determining if $EGCWA(T) \models \gamma$ is a coNP-hard problem.*

PROOF (sketch): We reduce the unsatisfiability problem to our problem by exhibiting a mapping from any CNF formula π to a 2-positive-Krom formula π'' and a clause C such that π is unsatisfiable iff $EGCWA(\pi'') \models C$.

Let π be a CNF formula on the alphabet L. Let L' be the alphabet $L \cup \{\overline{a} \mid a \in L\}$. We define π' on L' as follows: 1. for each letter a of L, there is a clause $\neg a \vee \neg \overline{a}$ in π'; 2. for each clause $\neg w_1 \vee \cdots \vee \neg w_n \vee w_{n+1} \vee \cdots \vee w_{n+m}$ in π, there is a clause $\overline{w_1} \vee \cdots \vee \overline{w_n} \vee w_{n+1} \vee \cdots \vee w_{n+m}$ in π'. Obviously, π' is satisfiable iff π is satisfiable. Let L'' be the alphabet obtained from L' by adding a new letter g_i for each clause γ_i in π. We define π'' on L'' as follows: 1. for each letter a of L, there is a clause $a \vee \overline{a}$ in π''; 2. for each clause $\gamma_i = \neg w_1 \vee \cdots \vee \neg w_n \vee w_{n+1} \vee \cdots \vee w_{n+m}$ in π, there are $n+m$ clauses $g_i \vee \overline{w_1}, \ldots, g_i \vee \overline{w_n}, g_i \vee w_{n+1}, \ldots, g_i \vee w_{n+m}$ in π''. Notice that the above mapping from π to π'' is clearly polynomial, and π'' is a 2-positive-Krom formula. Moreover, given a model M of π', we can build a disjunction B such that $\pi'' \models B \vee K$ and $\pi'' \not\models B$, and, conversely, given a disjunction B such that $\pi'' \models B \vee K$ and $\pi'' \not\models B$, we can build a model M of π'. Therefore, the conjunction $K = g_1 \wedge \cdots \wedge g_h$, where $g_1, \ldots, g_h$ are all the letters of L'' corresponding to the clauses in π, is not EGCWA-ffn in π'' iff π' is satifiable. Taking into account that $\neg K$ is a disjunction, and $EGCWA(\pi'') \models \neg K$ iff K is EGCWA-ffn in π'', it follows that $EGCWA(\pi'') \models \neg K$ iff π is unsatisfiable. □

Taking into account Fact 2, we can conclude that the above result applies to the ECWA too.

3.4 CCWA

In [8] a detailed analysis of CCWA is presented. In particular it is shown that determining if $CCWA(T; P; Q; \emptyset) \models L$, where L is a literal, is coNP-hard for Horn formulae. Moreover, two different problems are shown to be polynomial: $CCWA(T; P; Q; Z) \models \gamma$, where γ is an arbitrary clause and T is HornKrom, and $CCWA(T; P; \emptyset; Z) \models \gamma$, where γ is an arbitrary clause and T is Krom. We complete the analysis of [8] by providing the following result:

Theorem 2 *Let T be a Krom formula, and γ be a clause. Then the problem of determining if $CCWA(T; P; Q; Z) \models \gamma$ is polynomial (see [3] for an $O(|T|^2)$ algorithm).*

Notice that, by Fact 3, the above result on Krom formulae can be extended to the GCWA.

3.5 ECWA

We first present two intractability results concerning HornKrom and Definite formulae, respectively.

Theorem 3 *Let T be a HornKrom formula, γ be a clause and L be a literal. Then both determining if $ECWA(T; P; Q; \emptyset) \models \gamma$ and if $ECWA(T; P; Q; Z) \models L$ are coNP-hard problems.*

Theorem 4 *Let T be a Definite formula, and γ be a clause. Then the problem of determining if $ECWA(T; P; Q; \emptyset) \models \gamma$ is coNP-hard.*

The above theorems can be proven similarly to theorem 1. Moreover, we can strengthen the result of theorem 1, by showing that, as far as 2-positive-Krom formulae are concerned, the problem of deducing literals under the ECWA is a coNP-hard problem, even if $Q = \emptyset$.

Theorem 5 *Let T be a 2-positive-Krom formula, and L be a literal. Then the problem of determining if $ECWA(T; P; \emptyset; Z) \models L$ is coNP-hard.*

The above theorem, together with theorem 2 and Fact 4, allows us to conclude that determining whether a given literal L logically follows from $ECWA(T; P; \emptyset; Z)$ can be characterized as follows: the problem is polynomial for $L \in P^+ \cup P^-$, whereas is coNP-hard for $L \in Z^+ \cup Z^-$. Notice that the same holds for the case when $Q \neq \emptyset$ (see [3]).

In the rest of this subsection, we discuss two tractable cases, concerning Horn and HornKrom$^-$, respectively.

Theorem 6 *Let T be a Horn formula, and γ be a clause. Then $ECWA(T; P; \emptyset; Z) \models \gamma$ iff $CCWA(T; P; \emptyset; Z) \models \gamma$. Hence determining if $ECWA(T; P; \emptyset; Z) \models \gamma$ is a polynomial problem (see [8] for a polynomial algorithm).*

Theorem 7 *Let T be a HornKrom$^-$ formula, and γ be a clause. Then the problem of determining if $ECWA(T; P; Q; Z) \models \gamma$ is polynomial.*

In [3], we present an algorithm that, given a HornKrom$^-$ formula T, and a clause γ, determines if $ECWA(T; P; Q; Z) \models \gamma$. The algorithm uses a graph representation for T, and runs in $O(T^2)$ time.

The above theorem is probably the first tractability result concerning parallel circumscription. For example, it

literal	Horn	Krom	Definite	HornKrom	HornKrom$^-$	2posKrom	dual-Horn	dual-HornKrom
pos	≤ 1		1	≤ 1	≤ 1	2		
neg					≤ 1	0	≤ 1	≤ 1
tot		≤ 2		≤ 2	≤ 2	2		≤ 2
CWA	P [1]	–	P [1]	P [1]	P [1]	–	–	–
GCWA	P [1]	P	P [1]	P [1]	P [1]	P	coNP [7]	P
WGCWA	–	–	P [12]	–	–	P [12]	P [12]	P [12]
EGCWA	P [1]	coNP	P [1]	P [1]	P [1]	coNP	coNP [7]	coNP
CCWA $Q = \emptyset$	P [8]	P	P [8]	P [8]	P [8]	P	coNP [7]	P
CCWA $Z = \emptyset$	coNP [8]	P	P [8]	P [8]	P [8]	P	coNP [7]	P
CCWA	coNP [8]	P	P [8]	P [8]	P [8]	P	coNP [7]	P
ECWA $Q = \emptyset$	P	coNP	P	P	P	coNP	coNP [7]	coNP
ECWA $Z = \emptyset$	coNP	coNP	coNP	coNP	P	coNP	coNP [7]	coNP
ECWA	coNP	coNP	coNP	coNP	P	coNP	coNP [7]	coNP

Table 1: Summary of the complexity analysis

can be easily shown that the algorithm in [11] for computing circumscription is exponential even for HornKrom$^-$ formulae. This fact shows that in order to develop efficient methods for computing circumscription, it is necessary to look for structural properties of the ECWA-ffn formulae which are relevant to the deduction.

All the complexity results about closed world reasoning for the different classes of propositional formulae are summarized in Table 1, where P means "polynomial time", coNP means coNP-hardness, and symbol "-" means non-applicable. Each entry referring to a result previously known, is marked with the appropriate reference. Entries without references refer to results presented in this paper.

4 Applications

In this section we briefly discuss some possible applications of the complexity analysis developed in Section 3. In general, our results can be used to characterize the complexity of any reasoning problem which can be formalized in terms of CWR on propositional formulae. We concentrate our attention to three of such problems, namely nonmonotonic inheritance, default reasoning, and diagnosis.

As a first observation, notice that we can directly apply all the above results to the problem of performing CWR on a particular class of first order monadic theories, called Inheritance Networks [8]. Moreover, it can be seen that the problem of reasoning under circumscription on an Inheritance Network N can be reduced to the one of performing deduction under the ECWA on a suitable propositional formula obtained from N by means of a polynomial transformation. By virtue of this fact, some of the results presented in Section 3 can be used to give lower bounds to the complexity of performing deduction on circumscriptive Inheritance Networks, i.e. Inheritance Networks with defeasible rules, where the meaning of defeasible rules is formalized in terms of minimization of abnormalities.

With regard to diagnosis, in [14] a general methodology for solving the problem of diagnosing a malfunctioning system is given. The system is described by means of a logical formula SD in which the distinguished unary predicate symbol $ABNORMAL$ is used to model the malfunctioning of a component. Observations about the system behaviour are represented by a further formula OBS. A *diagnosis* is then defined as a minimal set $\Delta \subseteq COMPONENTS$ such that $SD \cup OBS \cup \{\neg ABNORMAL(c) \mid c \in COMPONENTS \setminus \Delta\}$ is consistent, where $COMPONENTS$ is the set of constant symbols of SD representing the system components.

Reiter shows that there is a natural correspondence between the diagnoses of $SD \cup OBS$ and the extensions of the default theory whose first-order part is $SD \cup OBS$, and whose default rules are $\{: \neg ABNORMAL(c)/\neg ABNORMAL(c) \mid c \in COMPONENTS\}$. It is easy to prove that, when $SD \cup OBS$ can be expressed as a propositional formula, for example when the system is a digital circuit, the diagnosis problem can also be formulated as a deduction problem under ECWA. This allows us to apply some of our results to the diagnosis problem. For example, consider a very simple digital circuit made up by only two kinds of components: the wire and the inverter. A way to model (unidirectionally) the behaviour of these components is by means of the clauses $oninput(wire_i) \wedge \neg ab(wire_i) \supset onoutput(wire_i)$ and $oninput(inverter_i) \wedge \neg ab(inverter_i) \supset \neg onoutput(inverter_i)$. Now, based on the result stated in theorem 1, it can be shown that determining whether a given fact is true in all the diagnoses of a system of the above form is coNP-hard.

Finally, we observe that our analysis can be used to obtain complexity results in the context of default logic. The basic observation is that *skeptical reasoning* in default theories (i.e. validity in all the extensions of a default theory–see [6]) can be reduced to ECWA, at least for some propositional default theories where defaults are of the form $: \neg a/\neg a$. In [6] it is shown that for a superclass of this class (namely Normal Unary default theories) skeptical reasoning is polynomial if the first-order part of the theory is a conjunction of literals. By exploiting the results of this paper, it is possible to prove that the problem is coNP-hard for a slight enhancement of the expressive power of the language, namely if the first-order part of the theory is a 2-positive-Krom propositional formula.

References

[1] Apt K.R., "Introduction to Logic Programming", Technical Report TR-87-35, University of Texas, Department of Computer Science, 1987.

[2] Bylander T., Allemang D., Tanner M.C., and Josephson J.R., "Some Results Concerning the Computational Complexity of Abduction", *Proceedings of the 1st KR Conference*, 1989.

[3] Cadoli M., and Lenzerini M., "The Complexity of Closed World Reasoning", Technical Report, Università di Roma "La Sapienza", Dipartimento di Informatica e Sistemistica, 1990.

[4] Gelfond M., and Przymusinska H., "Negation as Failure: Careful Closure Procedure", *Artificial Intelligence*, Vol. 30, 1986.

[5] Gelfond M., Przymusinska H., and Przymusinsky T., "On the Relationship between Circumscription and Negation as Failure", *Artificial Intelligence*, Vol. 38, 1989.

[6] Kautz H.A., and Selman B., "Hard Problems for Simple Default Logics", *Proceedings of the 1st KR Conference*, 1989.

[7] Lenzerini M., "Covering and Disjointness Relationships in Hierarchy Networks", Technical Report RAP.17.88, Università di Roma "La Sapienza", Dipartimento di Informatica e Sistemistica, 1988.

[8] Lenzerini M., "Reasoning about Inheritance Networks", Technical Report RAP.22.88, Università di Roma "La Sapienza", Dipartimento di Informatica e Sistemistica, 1988.

[9] McCarthy J., "Circumscription - A form of Non-Monotonic Reasoning", *Artificial Intelligence*, Vol. 13, 1980.

[10] Minker J., "On Indefinite Databases and the Closed World Assumption", *Proceedings of the 6th CADE Conference*, 1982.

[11] Przymusinsky T., "An Algorithm to Compute Circumscription", *Artificial Intelligence*, Vol. 38, 1989.

[12] Rajasekar A., Lobo J., and Minker J., "Weak Generalized Closed World Assumption", *Journal of Automated Reasoning*, Vol. 5, 1989.

[13] Reiter R., "On Closed World Data Bases", in Gallaire H., and Minker J. (eds), *Logic and Data Bases*, Plenum, New York, 1978.

[14] Reiter R., "A Theory of Diagnosis from First Principles", *Artificial Intelligence*, Vol. 32, 1987.

[15] Schaefer T.J., "The Complexity of Satisfiability Problems", *Proceedings of the 10th ACM STOC*, 1978.

[16] Schlipf J.S., "When is Closed World Reasoning Tractable?", *Proceedings of the 3rd ISMIS*, 1988.

[17] Selman B., and Levesque H.J., "The Tractability of Path-Based Inheritance", *Proceedings of the 11th IJCAI*, 1989.

[18] Yahya A., and Henschen L.J., "Deduction in Non-Horn Databases", *Journal of Automated Reasoning*, Vol. 1, 1985.