Is Intractability of Non-Monotonic Reasoning a Real Drawback?

Marco Cadoli and **Francesco M. Donini** and **Marco Schaerf**
Dipartimento di Informatica e Sistemistica
Università di Roma "La Sapienza"
Via Salaria 113, I-00198 Roma, Italy
email: <lastname>@assi.dis.uniroma1.it

Abstract

Several studies about complexity of NMR showed that inferring in non-monotonic knowledge bases is significantly harder than reasoning in monotonic ones. This contrasts with the general idea that NMR can be used to make knowledge representation and reasoning simpler, not harder. In this paper we show that, to some extent, NMR has fulfilled its goal. In particular we prove that circumscription allows for more compact and natural representation of knowledge. Results about intractability of circumscription can therefore be interpreted as the price one has to pay for having such an extra-compact representation. On the other hand, sometimes NMR really makes reasoning simpler; we give prototypical scenarios where closed-world reasoning accounts for a faster and unsound approximation of classical reasoning.

Introduction

The complexity of non-monotonic reasoning (NMR) has been extensively analyzed in recent years. Several studies showed that inferring in non-monotonic knowledge bases is significantly harder than reasoning in monotonic ones. As an example, while inference in propositional Horn formulae can be done in linear time provided we reason in the classical semantics (Dowling & Gallier 1984), inference under circumscription in such formulae is a co-NP-complete problem (Cadoli & Lenzerini 1990).

Although there are cases in which non-monotonic inference has a complexity which is comparable to classical inference, the general picture shows that tractable problems may become intractable (e.g. the complexity raises from polynomial to NP-complete (Kautz & Selman 1991)), intractable problems may become "more" intractable (e.g. from NP-complete to Σ_2^p-complete (Gottlob 1992)), decidable problems may become undecidable (Baader & Hollunder 1992), and undecidable problems may become "more" undecidable (e.g. from r.e.-complete to Π_2^1-complete (Schlipf 1987)). An up-to-date survey on computational aspects of NMR appears as (Cadoli & Schaerf 1993).

This aspect of NMR is acknowledged in the AI community. Brachman (1990, p. 1090) writes:

> "An irony of work on NMR is that, while the easy adoption and retraction of assumptions is most useful for speeding up natural everyday reasoning, most current NMR proposals drastically compound the already difficult problem of deductive reasoning. We urgently need to determine how NMR can be used to make commonsense inference faster, not slower."

The general idea about NMR is that it can be seen as a fast but *unsound* approximation of ordinary reasoning, as it is not possible to make reasoning more efficient while preserving both soundness and completeness.

Apart from AI, NMR is sometimes used to represent knowledge in a more compact fashion. Noticeably, negation through *cut* is commonly used among PROLOG programmers for writing more compact and efficient programs (Sterling & Shapiro 1986, Chap.11). Moreover, closed-world reasoning allows for effective representation of implicit knowledge in relational as well as deductive databases, and has been widely used among database practitioners for many years now.

Hence, there is a mismatch between the intuition behind NMR and the theoretical results on its computational complexity. The following questions naturally arise:

- is high complexity of NMR a bug or a feature?
- is the intuition that NMR simplifies reasoning wrong?
- are there complexity analyses showing that NMR fits its intuition?

The goal of this paper is to give a preliminary answer to the above questions. In particular we address two topics:

1. It is clear that NMR captures additional – wrt to classical reasoning – information and that such information makes reasoning harder. Now suppose we want to make the same inferences that we do in NMR knowledge bases "without using NMR", i.e. suppose we have a KB K and we want a new KB K' s. t. $K \vdash_{NMR} Q$ iff $K' \models Q$. How would the monotonic KB K' look like? How large would it be?

2. It is advocated that NMR makes reasoning faster and unsound. Theoretical results seem to contradict it. Can we show specific examples and frameworks which concretely support the above idea?

Discussion on the above topics will be carried out by means of examples taken from the academic domain.

The structure of the paper is the following: In the rest of the Introduction we recall some definitions about NMR; then we devote a section to each of the two topics above. In the last section we draw some conclusions.

Preliminaries

Throughout the paper we restrict our attention to propositional knowledge bases. We analyze closed-world reasoning and circumscription, two of the major NMR formalisms. Since they are widely known, in this paper we just mention the main definitions underlying them. Interpretations and models of propositional formulae will be denoted as sets of atoms (those which are mapped into 1).

Definition 1 (Lifschitz 1985) *Let M, N be two models of a propositional formula T and $\langle P; Z\rangle$ a partition of the atoms of T. We write $M \leq_{(P;Z)} N$ if $M \cap P \subseteq N \cap P$.*

A model M is called $(P;Z)$-minimal for a formula T if there is no model N of T such that $N \leq_{(P;Z)} M$ and $M \not\leq_{(P;Z)} N$.

The circumscription $CIRC(T; P; Z)$ of T minimizing the atoms in P and varying the atoms in Z denotes the set of $(P;Z)$-minimal models of T.

Intuitively a propositional atom is placed in P if we don't like it to be in models. In common-sense reasoning typically such atoms denote abnormality (McCarthy 1986). When an atom is in Z we accept that it occurs in models, provided this helps in excluding some of the atoms in P.

When $Z = \emptyset$ the $(P;Z)$-minimal models of a formula are called just minimal. The closed-world assumption $CWA(T)$ of a propositional formula T is defined as follows in (Reiter 1978):

$$CWA(T) = T \cup \{\neg p \mid T \not\models p\}.$$

$CWA(T)$ is consistent iff T has a unique minimal model; in such a case that model is the unique model of $CWA(T)$.

NMR for compact representation of knowledge

This section deals with using NMR for representing information in a compact way, where compactness is measured wrt classical representation of the same knowledge.

We introduce an example dealing with a student who needs to plan his/her *curriculum*. We show that the natural way the student can do this is to reason using circumscription. Given that the problem is computationally intractable, we address the following question: if the student was to reason in a classical fashion – e.g. using a classical theorem prover like OTTER (McCune 1990) – how much implicit information would he/she need to represent explicitly in the new KB? In other words, is it feasible to transform the original KB – dealt with NMR – into a new one – dealt with a classical inference engine – such that the same inferences are possible? The results we prove show that this is not feasible, since NMR allows one to save a huge amount of space.

Example 1: (The lazy student) The set of admissible *curricula* is represented by means of the models of a propositional formula T which might look like the following:

```
DataBases ∨ Algebra
  Algebra ∨ NonMonotonicReasoning
DataBases ∨ ReasoningAboutKnowledge
  Algebra ∨ ReasoningAboutKnowledge
  Algebra ∨ ComputationalComplexity
```

Throughout this section we assume that such formulae are always in 2-CNF, i.e. they have at most two literals per clause. A *curriculum* is just a model of T, e.g. {`DataBases`, `Algebra`, `NonMonotonicReasoning`}. Such a definition does not capture the preferences that a student might have. As an example the student might be better off by doing both courses {`ReasoningAboutKnowledge`, `ComputationalComplexity`} than doing just one of the courses {`DataBases`, `Algebra`, `NonMonotonicReasoning`}. This suggests to use the idea of $(P;Z)$-minimal models – where all exams that the student dislikes are in P; let's say that P = {`DataBases`, `Algebra`, `NonMonotonicReasoning`} and Z = { `ReasoningAboutKnowledge`, `ComputationalComplexity`}. A *minimal curriculum* is a $(P;Z)$-minimal model of T. In the above situation there are three minimal *curricula*: {`Algebra`,`ReasoningAboutKnowledge`}, {`Algebra`,`ReasoningAboutKnowledge`,`ComputationalComplexity`}, {`DataBases`,`NonMonotonicReasoning`,`ReasoningAboutKnowledge`,`ComputationalComplexity`}.

A course c is *mandatory* if it has to be done in all *curricula*, i.e. $T \models c$. A course c is *preferred* if it has to be done in all minimal *curricula*, i.e. $CIRC(T; P; Z) \models c$.

In the above situation there are no mandatory courses, although `ReasoningAboutKnowledge` is preferred. ◇

The above example shows that there are at least four possible computational services that a student might ask:

Problem 1. Find a *curriculum*. The student has no specific preferences, a *curriculum* is just as good as any other one;

Problem 2. Find a minimal *curriculum*. This is an improvement wrt to the previous service, as the student can express some preferences. Nevertheless, there is no explanation why a *curriculum* is provided. In Example 1, {`Algebra`,`ReasoningAboutKnowledge`,`ComputationalComplexity`} is as good as {`Algebra`,`ReasoningAboutKnowledge`}. Even if the second one is provided as an answer, the student does not know whether `ReasoningAboutKnowledge` is preferred or not.

Problem 3. Decide whether a course is mandatory. The student – with no preferences – wants to know whether he/she must attend a specific course. The weakness of this service is in that a non-mandatory course could be true only in *curricula* that the student would never accept.

Problem 4. Decide whether a course is preferred (preferences decided in advance). Same as above, but the student has preferences.

This sophisticated form of reasoning is mostly relevant is this situation: the lazy student might take course c, but he/she does not want to commit until he/she is completely convinced that c is preferred. In fact not all courses suggested by an answer to Problem 2 are preferred.

The complexity of the above problems has been already studied in the literature (when T is in 2-CNF):

Problem 1. is polynomial (Even, Itai, & Shamir 1976);

Problem 2. is polynomial (Cadoli 1992);

Problem 3. is polynomial (Even, Itai, & Shamir 1976);

Problem 4. is co-NP-complete (Cadoli & Lenzerini 1990).

In the above scenario NMR seems to do exactly the form of reasoning the student needs. In fact a solution to Problem 4 gives some extra information that the three other problems miss, therefore we may be willing to accept its extra complexity. Furthermore this extra complexity does not seem to be really dangerous: since the set of courses the student could ever take is limited – let's say they are $c_1, \ldots, c_n$ – he/she might compute which of them is preferred, i.e. decide whether $CIRC(T; P; Z) \models c_i$ holds for each i ($1 \leq i \leq n$). Even if this amounts to solve n co-NP-complete problems – one for each query – it is not necessary to find the answer when the student asks a query. More precisely the queries can be posed off-line and their answers cached; then on-line query-answering just amounts to table look-up, which is clearly polynomial.

We call *compilation* any off-line process that makes on-line reasoning polynomial.

The next step is now to consider the scenario where *several* students are interested in preparing their *curricula*. In this case, students may have the same preferences, or may not.

Problem 4.1. Decide (repeatedly) whether a course is preferred, when all students have the same preferences;

Problem 4.2. Decide (repeatedly) whether a course is preferred, when each student has his/her own preferences.

The above argument proves the following property of Problem 4.1.

Proposition 1 *It is possible to compile Problem 4.1.*

Compilation for Problem 4.1 can be done simply by caching. Even if such caching cannot be done for Problem 4.2 (there are exponentially many different preferences) one may wonder if the problem is compilable in some smarter way.

However, we are able to show that it is very unlikely that such a compilation may exist. To do this we resort on the notion of *non-uniform* computation. A problem Π is in the class non-uniform P if there exists a function $f()$ that for each instance π of Π maps the *size* of π into a polynomial-time algorithm that solves π (Johnson 1990, p. 116). We remark that there is no restriction on $f()$, which may be even non-recursive. Intuitively, $f()$ represents the off-line computation. The relations between non-uniform P and uniform complexity classes, such as NP, have been studied in the literature. In particular it has been shown (Karp & Lipton 1980) that NP $\subseteq$ non-uniform P would imply some unlikely consequences on complexity classes.

Theorem 2 *Unless NP $\subseteq$ non-uniform P, there is no data structure representing a 2-CNF formula T such that, given a set P of atoms to minimize and a set Z of varying atoms, deciding whether $CIRC(T; P; Z) \models z$ (with $z \in Z$) can be answered in polynomial time.*

Proof. We first prove a key lemma. Its proof is based on a reduction given in (Cadoli & Lenzerini 1990, Theorem 5), which showed that inference in 2-positive-CNF under CIRC is coNP-hard. The major difference is that we now need to code every possible 3-CNF over n atoms in one theory.

Lemma 3 *For any integer n, there exists a 2-positive-CNF formula T_n of polynomial size wrt n, such that given any 3-CNF formula π using n atoms, there are particular P_π and Z_π such that π is unsatisfiable iff $CIRC(T_n; P_\pi; Z_\pi) \models z$.*

Proof (sketch). We use the following conventions. We denote with a, b, c, d propositional atoms; if a is an atom, $\overline{a}$ is its negation. We denote with w, x, y literals (i.e. either atoms or negated atoms). If x is the literal $\overline{a}$, then $\overline{x}$ is a.

Let L be the alphabet of n atoms used in the 3-CNF formulas. Let L' be the alphabet $L \cup \overline{L} \cup C \cup \{z\}$, where $\overline{L} = \{\overline{a} \mid a \in L\}$, and C is a set of atoms one-to-one with possible three-literals clauses of L, i.e., $C = \{c_i \mid \gamma_i$ is a three-literals clause of $L\}$. Observe that now $\overline{a}$ denotes also a sintactic symbol of L': we retain this ambiguity to simplify the notation in the reduction.

We define T on the alphabet L' according to the following rules:

1. for each letter a of L, there is a clause $a \vee \overline{a}$ in T;
2. for each clause $\gamma_i = w \vee x \vee y$, where w, x, y are literals, there are four clauses in T. The first three are $c_i \vee w, c_i \vee x, c_i \vee y$; observe that if x is a negated literal in π, then x is a (syntactically equal) atom in T. The fourth clause is $c_i \vee z$.

Notice that the size of T is $O(n^3)$, and T is a 2-positive-CNF formula.

Let π be a 3-CNF formula over L, and let $c_1, \ldots, c_h$ be all the atoms of L' corresponding to the clauses in π. Define $P_\pi = \{c_1, \ldots, c_h\} \cup L \cup \overline{L}$, and $Z_\pi = L' - P_\pi$. The theorem can be proven by showing that:

1. given a model of π, we can build a $(P_\pi; Z_\pi)$-minimal model MM of T such that z is false in MM: Let $M \subseteq L$ be a model for π, and let $\overline{M}$ be a subset of $\overline{L}$ such that if $a \in M$ then $\overline{a} \in \overline{M}$. Define MM as $(L - M) \cup \overline{M} \cup C$. It can be verified that MM is a model of T, because it includes C, and for every atom $a \in L$ it contains either a or $\overline{a}$, but not both. Moreover, z is false in MM. One can prove that MM is indeed $(P_\pi; Z_\pi)$-minimal.
2. given a $(P_\pi; Z_\pi)$-minimal model MM of T such that z is false in MM, we can build a model M of π: Let MM be a $(P_\pi; Z_\pi)$-minimal model of T, such that z is false in MM. Since for each $c \in C$, the clause $c \vee z$ is in T, $C \subseteq MM$. Consider an atom $c_i \in P$. There are in T the three clauses $c_i \vee w, c_i \vee x, c_i \vee y$. One can prove that at least one between w, x, y is not in MM, and that $a \in MM$ iff $\overline{a} \notin MM$. Define $M = \{a \mid \overline{a} \in MM\}$. □

Let T_n be as in the above lemma. If for each n there is a data structure representing T_n, such that given P and Z, $CIRC(T, P; Z) \models z$ can be answered in polynomial time, one could give an infinite class of algorithms $A_1, \ldots, A_i, \ldots$, each one embedding the data structure for T_i, and each one working in polynomial time. Now given a 3-CNF formula π of size n, π is satisfiable iff A_n answers "No" on inputs P_π, Z_π, where P_π, Z_π are chosen as in the above lemma. Then 3-CNF-SAT $\in$ non-uniform P, and since 3-CNF-SAT is complete for NP, NP $\subseteq$ non-uniform P. A similar technique has been used in (Kautz & Selman 1992). □

The above theorem shows that Problem 4.2 is not compilable.

So far we considered atomic queries. Non-atomic clauses are nevertheless necessary for posing more complex queries such as disjunction (e.g. is it necessary to take at least one course in a given set?) or implication (e.g. is it necessary to take course a provided course b is taken?). Theorem 2 implies that Problem 4.2 is not compilable also for non-atomic queries. Instead, since in its proof it is crucial that P, Z are given as part of the input, one could think that when P, Z are fixed – like in Problem 4.1 – compilation of all non-atomic queries is still possible. Next theorem shows that this conjecture is very unlikely to hold.

Theorem 4 *Unless NP $\subseteq$ non-uniform P, there is no data structure representing a 2-CNF T and two sets P, Z of atoms, such that given a clause F, deciding whether $CIRC(T, P; Z) \models F$ can be answered in polynomial time.*

The proof can be obtained with techniques similar to those of the previous theorem.

At a first sight, the above two theorems seem to give just a negative result: NMR is not compilable. But the result is in fact twofold: the theorems show that if one wants to represent the circumscription of a 2-CNF KB K with a new KB K', either inference in K' is intractable, or (if inference has to be kept tractable) K' has exponential size w.r.t. K. In other words, circumscription allows one to derive an exponential number of new consequences not derivable from K with classical inference: If the number of new consequences were polynomial, they could be simply cached. It might be possible that such consequences could be compacted in one formula of polynomial size; but in this case, Theorems 2 and 4 show that extracting consequences from such a formula would very probably be an intractable task. Hence the positive aspect of Theorems 2 and 4 is that circumscription is an extremely powerful tool for representing problems in a compact way.

We summarize the results in Table 1, where we divide cases between the two services, and between "short" and "long" queries. By "short" we mean clauses of fixed length (e.g. length 1 to ask for preferred exams in the lazy student example), while "long" means clauses of arbitrary length.

	all queries with same preferences	different preferences
"short" queries	compilable Prop. 1	non-compilable Theo. 2
"long" queries	non-compilable Theo. 4	non-compilable Theo. 2, 4

Table 1: Are 2-CNF knowledge bases compilable under CIRC?

We remark that "non-compilable" also means that CIRC allows one to represent in a compact way an exponential number of new consequences.

Unsound and fast inference with NMR

As mentioned in the Introduction, it has been frequently argued that one of the expected features of NMR was that it could account for a form of unsound, but *fast*, inference. The results on the computational complexity of NMR seem to contradict the possibility of NMR of being faster than classical reasoning. In this section we show that NMR is more efficient in

some situations than classical reasoning even according to worst-case analysis. We introduce this aspect by means of an example.

Example 2: (The cautious student) The faculty members decide the requirements needed to attend a course, which may be represented with a set of dependencies R:

$$\begin{aligned}
\texttt{NonMonotonicReasoning} &\longrightarrow \texttt{Algebra}\\
\texttt{NonMonotonicReasoning} &\longrightarrow \texttt{Logic}\\
\texttt{DataBases} &\longrightarrow \texttt{Algebra}\\
\texttt{ComputerArchitectures} &\longrightarrow \texttt{Algebra}\\
\texttt{ReasoningAboutKnowledge} &\longrightarrow \texttt{Logic}\\
\texttt{ComputationalComplexity} &\longrightarrow \texttt{Logic}
\end{aligned}$$

Throughout this section we assume that such formulae are always Definite Horn. A set such as $C = \{$**NonMonotonicReasoning**, **ComputerArchitectures**$\}$ can represent courses the student has attended, or has committed to attend. The models of $C \cup R$ represent all admissible completions of the *curriculum*.

A conjunct like $g_1 =$ **Algebra**$\wedge$ **ReasoningAboutKnowledge**$\wedge \neg$ **DataBases**$\wedge \neg$ **ComputationalComplexity** represents courses the student may be interested in taking (positive literals) or avoiding (negative literals), but he/she has not committed yet. The student may also have alternative plans, such as $g_2=$ **Logic**$\wedge$ **DataBases**$\wedge$ **ComputationalComplexity**$\wedge \neg$ **ReasoningAboutKnowledge**, or $g_3=$ **Algebra**$\wedge$ **ComputerArchitectures**$\wedge \neg$ **ReasoningAboutKnowledge**.

A plan g may be satisfied by a model of $C \cup R$ or not. The student wants to know if in all models at least one of his/her plans will be satisfied. This could be represented as a goal $G = g_1 \vee g_2 \vee g_3$.

The scenario could be modified if the faculty add new requirements to R or if the student makes further commitments, thus adding atoms to C. $\diamond$

Let us formalize the computational services the student may be interested in:

Problem 5. Decide whether the set of courses in C plus the courses required by R satisfy the goal G. This service provides information on the current situation but it does not give any guarantee on the future. It amounts to decide whether $CWA(C \cup R) \models G$.

Problem 6. Decide whether the goal G will be satisfied no matter which new requirements are imposed (in addition to R) by the faculty and which new courses (in addition to C) the student decides to attend. This service provides information on the current and the future situation. It amounts to decide whether $C \cup R \models G$.

Coming back to the example, we have that $CWA(C \cup R) \models G$ while $C \cup R \not\models G$. Therefore, if the faculty do not change the set of requirements and the student does not decide to take additional courses, the goal will be satisfied.

Let's consider an alternative goal of the student: $G' = g'_1 \vee g'_2 \vee g'_3$, where $g'_1 =$ **Algebra**$\wedge$ **Logic**$\wedge$ **ReasoningAboutKnowledge**$\wedge \neg$ **ComputationalComplexity**, $g'_2 =$ **Logic**$\wedge$ **ComputationalComplexity**$\wedge \neg$ **ReasoningAboutKnowledge**, $g'_3 =$ **Algebra**$\wedge \neg$ **ReasoningAboutKnowledge**. Both $CWA(C \cup R) \models G'$ and $C \cup R \models G'$ hold. As a consequence, he/she is sure that, whatever new requirements and courses are added, the goal G' will always be satisfied.

We consider now the complexities of the above problems.

Problem 5. is polynomial: First compute the minimal model M of the Horn formula $R \wedge C$, then check whether $M \models G$. Both steps can be accomplished in polynomial time.

Problem 6. is co-NP-complete: hardness follows from the co-NP-completeness of tautology checking of a DNF formula.

NMR is faster than classical reasoning in this specific case. Therefore NMR can be seen as a fast, complete but unsound approximation of classical reasoning, as $\Sigma \models \gamma$ implies $CWA(\Sigma) \models \gamma$.

This behavior of NMR is not restricted to this particular situation. Let's take a further example from the logic programming field. Let P be a propositional general logic program – where negation is allowed in the body of the rules – and γ be a clause. Deciding whether γ is true in all the (classical) models of P, i.e. $P \models \gamma$ interpreting **not** as classical negation, is a co-NP-complete problem. On the other hand, deciding whether γ is a consequence of P under the well-founded semantics (van Gelder, Ross, & Schlipf 1991), i.e. γ is satisfied by the well-founded model of P ($WF(P) \models \gamma$), is a polynomial time problem.

Even in this case NMR accounts for a fast and complete, although unsound, approximation of classical reasoning, as $P \models \gamma$ implies $WF(P) \models \gamma$.

Conclusions

Recent theoretical results on the computational complexity of NMR seem to contradict the main reasons for the development of non-monotonic formalisms, namely that defeasible assumptions should allow for: 1) faster, although unsound, inference and 2) more compact representation of knowledge. We have shown in this paper that, to some extent, NMR has fulfilled its goal.

Regarding the second goal, we have proven that circumscription does indeed allow for more compact representation of knowledge. The results can also be extended to circumscription of general propositional formulae (Cadoli, Donini, & Schaerf 1994). Results about intractability of NMR can therefore be interpreted as the price one has to pay for having extra-compact representation of knowledge. It is therefore unfair to say that NMR is harder than classical reasoning. In fact

the input of a NM inference problem could be exponentially smaller than the input of a classical inference problem. On the other hand the implicit assumption in saying that NMR is harder is that the sizes of the inputs are the same.

Regarding the first goal, we have given prototypical scenarios where closed-world reasoning accounts for a faster and unsound approximation of classical reasoning. Therefore NMR is not always computationally harder, and may even be simpler than classical reasoning.

Due to the lack of space we cannot give a detailed comparison of our work with other recently appeared in the literature on off-line reasoning, e.g. Kautz & Selman (1991) and Moses & Tennenholtz (1993). We briefly compare our work with the second one.

Moses & Tennenholtz analyze the possibility of speeding up the complexity of query answering through a previous off-line analysis of the knowledge base. Their goal can be considered as a special case of compilation: They consider a particular subset of all queries, which they call *efficient basis*, whose answers enables to answer all queries in polynomial time. Some query languages may not admit an efficient basis. Our results complement theirs, as we consider any possible preprocessing (even a non-recursive one) with the only restriction that the new representation can answer queries in time polynomial in the size of the original knowledge base. For each entry of Table 1 marked "non-compilable", we proved that not only no efficient basis exists but also that no other compilation is possible.

Acknowledgements

This work has been supported by the ESPRIT Basic Research Action N.6810 (COMPULOG 2) and by the Progetto Finalizzato Sistemi Informatici e Calcolo Parallelo of the CNR (Italian Research Council), LdR "Ibridi".

References

Baader, F., and Hollunder, B. 1992. Embedding defaults into terminological knowledge representation formalisms. In *Proc. of KR-92*, 306–317.

Brachman, R. J. 1990. The future of knowledge representation. In *Proc. of AAAI-90*, 1082–1092.

Cadoli, M., and Lenzerini, M. 1990. The complexity of closed world reasoning and circumscription. In *Proc. of AAAI-90*, 550–555. MIT press. Extended version to appear on *Journal of Computer ans System Sciences*, 1994.

Cadoli, M., and Schaerf, M. 1993. A survey of complexity results for non-monotonic logics. *Journal of Logic Programming* 17:127–160.

Cadoli, M.; Donini, F. M.; and Schaerf, M. 1994. On compact representations of propositional circumscription. Manuscript.

Cadoli, M. 1992. On the complexity of model finding for nonmonotonic propositional logics. In *Proceedings of the Fourth Italian Conference on Theoretical Computer Science*, 125–139. World Scientific Publishing Co.

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

Even, S.; Itai, A.; and Shamir, A. 1976. On the complexity of timetable and multicommodity flow problems. *SIAM Journal of Computing* 5:691–703.

Gottlob, G. 1992. Complexity results for nonmonotonic logics. *Journal of Logic and Computation* 2:397–425.

Johnson, D. S. 1990. A catalog of complexity classes. In van Leeuwen, J., ed., *Handbook of Theoretical Computer Science*, volume A. Elsevier Science Publishers B. V. (North Holland). chapter 2.

Karp, R. M., and Lipton, R. J. 1980. Some connections between non-uniform and uniform complexity classes. In *Proc. of STOC-80*, 302–309.

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

Kautz, H. A., and Selman, B. 1992. Forming concepts for fast inference. In *Proc. of AAAI-92*, 786–793.

Lifschitz, V. 1985. Computing circumscription. In *Proc. of IJCAI-85*, 121–127.

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

McCune, W. 1990. Otter 2.0. In *Proceedings of the Tenth Conference on Automated Deduction (CADE-90)*, 663–664. Springer-Verlag.

Moses, Y., and Tennenholtz, M. 1993. Off-line reasoning for on-line efficiency. In *Proc. of IJCAI-93*, 490–495.

Reiter, R. 1978. On closed world data bases. In Gallaire, H., and Minker, J., eds., *Logic and Data Bases*. Plenum. 119–140.

Schlipf, J. S. 1987. Decidability and definability with circumscription. *Annals of Pure and Applied Logic* 35:173–191.

Selman, B., and Kautz, H. A. 1991. Knowledge compilation using Horn approximations. In *Proc. of AAAI-91*, 904–909.

Sterling, L., and Shapiro, E. 1986. *The Art of Prolog.* The MIT Press.

van Gelder, A.; Ross, K. A.; and Schlipf, J. S. 1991. The well-founded semantics for general logic programs. *Journal of the ACM* 38:620–650.