Preferred Arguments are Harder to Compute than Stable Extensions

Yannis Dimopoulos
University of Cyprus
Dept. of Computer Science
75 Kallipoleos Street
Nicosia P.O. Box 537
Cyprus
yannis@cs.ucy.ac.cy

Bernhard Nebel
Universität Freiburg
Institut für Informatik
Am Flughafen 17
D-79110 Freiburg
Germany
nebel@informatik.uni-freiburg.de

Francesca Toni
Imperial College
Dept. of Computing
180 Queen's Gate
London SW7 2BZ
United Kingdom
ft@doc.ic.ac.uk

Abstract

Based on an abstract framework for nonmonotonic reasoning, Bondarenko *et al.* have extended the logic programming semantics of *admissible* and *preferred arguments* to other nonmonotonic formalisms such as circumscription, auto-epistemic logic and default logic. Although the new semantics have been tacitly assumed to mitigate the computational problems of nonmonotonic reasoning under the standard semantics of *stable extensions*, it seems questionable whether they improve the worst-case behaviour. As a matter of fact, we show that credulous reasoning under the new semantics in propositional logic programming and propositional default logic has the same computational complexity as under the standard semantics. Furthermore, sceptical reasoning under the admissibility semantics is easier – since it is trivialised to monotonic reasoning. Finally, sceptical reasoning under the preferability semantics is harder than under the standard semantics.

1 Introduction

Bondarenko *et al.* [1997] show that many logics for nonmonotonic reasoning, in particular default logic (DL) [Reiter, 1980] and logic programming (LP), can be understood as special cases of a single abstract framework. The standard semantics of these logics can be understood in terms of *stable extensions* of a given theory, where a stable extension is a set of assumptions that does not *attack* itself and it attacks every assumption not in the set. In abstract terms, an assumption is attacked if its *contrary* can be proven, in some appropriate underlying monotonic logic, possibly with the aid of other conflicting assumptions.

Bondarenko *et al.* [1997] also propose two new semantics generalising, respectively, the *admissibility semantics* [Dung, 1991] and the *semantics of preferred extensions* [Dung, 1991] and *partial stable models* [Saccà and Zaniolo, 1990] for LP. In abstract terms, a set of assumptions is an *admissible argument* of a given theory, iff it does not attack itself and it attacks all sets of assumptions which attack it. A set of assumptions is a *preferred argument* iff it is a maximal (wrt. set inclusion) admissible argument.

The new semantics are more general than the stability semantics since every stable extension is a preferred (and admissible) argument, but not every preferred argument is a stable extension. Moreover, the new semantics are more liberal because for most concrete logics for nonmonotonic reasoning, admissible and preferred arguments are always guaranteed to exist, whereas stable extensions are not. Finally, reasoning under the new semantics appears to be computationally easier than reasoning under the stability semantics. Intuitively, to show that a given sentence is justified by a stable extension, it is necessary to perform a global search amongst all the assumptions, to determine for each such assumption whether it or its contrary can be derived, independently of the sentence to be justified. For the semantics of admissible and preferred arguments, however, a "local" search suffices. First, one has to construct a set of assumptions which, together with the given theory, (monotonically) derives the sentence to be justified, and then one has to augment the constructed set with further assumptions to defend it against all attacks.

However, from a complexity-theoretic point of view, it seems unlikely that the new semantics lead to better lower bounds than the standard semantics since all the "sources of complexity" one has in nonmonotonic reasoning are present. There are potentially exponentially many assumption sets sanctioned by the semantics. Further, in order to test whether a sentence is entailed by a particular argument one has to reason in the underlying monotonic logic. For this reason, one would expect that reasoning under the new semantics has the same complexity as under the stability semantics, i.e., it is on the first level of the polynomial hierarchy for LP and on the second level for logics with full propositional logic as the underlying logic [Cadoli and Schaerf, 1993]. However, previous results on the expressive power of DATALOG$^{\neg}$ queries by Saccà [1997] suggest that this is not the case for LP. Indeed, these results imply that reasoning under the preferability semantics for LP is at the second level of the polynomial hierarchy.

In this paper, we extend these results and show that for LP and DL

- *credulous* reasoning under the *admissibility* and *preferability semantics* has the same complexity as under the stability semantics,
- *sceptical* reasoning under the *admissibility semantics* is easier than under the stability semantics – since it reduces to monotonic reasoning with the given theory, and, finally,
- *sceptical* reasoning under the *preferability semantics* is harder than under the stability semantics. In other words, here intuition seems to clash severely with the complexity-theoretic results.

The paper is organised as follows. Section 2 summarises the abstract framework introduced by Bondarenko *et al.* [1997], its semantics and concrete instances capturing LP and DL. Section 3 gives complexity theory background and introduces the reasoning problems. Section 4 gives abstract upper bounds for credulous and sceptical reasoning, parametric wrt. the complexity of the underlying monotonic logic. Section 5 gives the completeness results. Section 6 discusses the results and concludes.

2 Default Reasoning via Argumentation

Assume a **deductive system** $(\mathcal{L}, \mathcal{R})$ where $\mathcal{L}$ is some formal language with countably many sentences and $\mathcal{R}$ is a set of inference rules inducing a monotonic derivability notion $\vdash$. Given a theory $T \subseteq \mathcal{L}$ and a formula $\alpha \in \mathcal{L}$, $Th(T) = \{\alpha \in \mathcal{L} \mid T \vdash \alpha\}$ is the deductive closure of T. Then, an **abstract (assumption-based) framework** is a triple $\langle T, A, \overline{} \rangle$, where $T, A \subseteq \mathcal{L}$ and $\overline{}$ is a mapping from A into $\mathcal{L}$. T, the **theory**, is a (possibly incomplete) set of beliefs, formulated in the underlying language, and can be extended by subsets of A, the set of **assumptions**. An **extension** of an abstract framework $\langle T, A, \overline{} \rangle$ is a theory $Th(T \cup \Delta)$, with $\Delta \subseteq A$ (sometimes an extension is referred to simply as $T \cup \Delta$ or Δ). Finally, given an assumption $\alpha \in A$, $\overline{\alpha}$ denotes its **contrary**.

LP is the instance of the abstract framework $\langle T, A, \overline{} \rangle$ where T is a logic program, the assumptions in A are all negations $not\, p$ of atomic sentences p, and the contrary $\overline{not\, p}$ of an assumption $not\, p$ is p. $\vdash$ is Horn logic provability, with assumptions, $not\, p$, understood as new atoms, p^*.

DL is the instance of the abstract framework $\langle T, A, \overline{} \rangle$ where the monotonic logic is classical logic augmented with domain-specific inference rules of the form

$$\frac{\alpha_1, \ldots, \alpha_m, M\beta_1, \ldots, M\beta_n}{\gamma}$$

where α_i, β_j, γ are sentences in classical logic. T is a classical theory and A consists of all expressions of the form $M\beta$ where β is a sentence of classical logic. The contrary $\overline{M\beta}$ of an assumption $M\beta$ is $\neg\beta$.

In the remainder of the paper, without loss of generality, we will assume that the set of assumptions A in the abstract framework for DL consists of all assumptions $M\beta$ occurring in the domain-specific inference rules.

Given an abstract framework $\langle T, A, \overline{} \rangle$ and an assumption set $\Delta \subseteq A$, Δ **attacks an assumption** $\alpha \in A$ iff $\overline{\alpha} \in Th(T \cup \Delta)$ and Δ **attacks an assumption set** $\Delta' \subseteq A$ iff Δ attacks some assumption $\alpha \in \Delta'$.

The standard semantics of extensions of DL [Reiter, 1980] and stable models of LP [Gelfond and Lifschitz, 1988] correspond to the "stability" semantics of abstract frameworks, where an assumption set $\Delta \subseteq A$ is **stable** iff

1. Δ does not attack itself, and
2. Δ attacks each assumption $\alpha \notin \Delta$.

A **stable extension** is an extension $Th(T \cup \Delta)$ for some stable assumption set Δ.

Bondarenko *et al.* define new semantics for the abstract framework, e.g., by generalising the admissibility semantics originally proposed for LP by Dung [1991]. An assumption set $\Delta \subseteq A$ is **admissible** iff

1. Δ does not attack itself, and
2. for all $\Delta' \subseteq A$, if Δ' attacks Δ then Δ attacks Δ'.

Maximal (wrt. set inclusion) admissible assumption sets are called **preferred**. In this paper we use the following terminology: an **admissible (preferred) argument** is an extension $Th(T \cup \Delta)$ for some admissible (preferred) assumption set Δ. Bondarenko *et al.* show that preferred arguments correspond to preferred extensions [Dung, 1991] and partial stable models [Saccà and Zaniolo, 1990] for LP.

In order to illustrate the effects of the different semantics, let us consider the following logic program:

$$p \leftarrow not\, p; \qquad r \leftarrow not\, q;$$
$$s \leftarrow not\, r; \qquad q \leftarrow not\, r.$$

This logic program has no stable extension, two preferred arguments ($\{not\, r\}$ and $\{not\, q, not\, s\}$) and four admissible arguments (additionally $\emptyset$ and $\{not\, q\}$). If we drop the clause "$p \leftarrow not\, p$," we get the same admissible and preferred arguments. In addition, the preferred arguments are also stable.

In [Bondarenko *et al.*, 1997], the definition of stable and admissible sets includes a third condition, namely, that the set Δ must be **closed**, i.e., $\Delta = A \cap Th(T \cup \Delta)$, and in part 2 of the definition of admissible sets all Δ' are required to be closed. Here we omit these conditions because in the LP and DL instances of the framework every set is guaranteed to be closed. Frameworks with this property are called **flat**.

In the sequel we will use the following properties:

Prop$_1$: Every preferred assumption set is (trivially) admissible and every admissible assumption set is a subset of some preferred assumption set [Bondarenko *et al.*, 1997, Theorem 4.8];

Prop$_2$: The empty assumption set is always admissible, trivially, for all concrete LP and DL frameworks.

3 Reasoning Problems and Complexity

We will analyse the *computational complexity* of the following reasoning problems for the propositional variants of the frameworks for LP and DL under admissibility and preferability semantics:

- the **credulous reasoning problem**, i.e., the problem of deciding for any given sentence φ in the set of possible queries whether $\varphi \in Th(T \cup \Delta)$ for *some* assumption set Δ sanctioned by the semantics;
- the **sceptical reasoning problem**, i.e., the problem of deciding for any given sentence φ in the set of possible queries whether $\varphi \in Th(T \cup \Delta)$ for *all* assumption sets Δ sanctioned by the semantics.

The set of possible queries consists of (variable-free conjunctions of) literals in the LP case and formulas in propositional logic in the DL case.

Instead of the sceptical reasoning problem, we will often consider its complementary problem, i.e.

- the **co-sceptical reasoning problem**, i.e, the problem of deciding for any given sentence φ (in a set of possible queries) whether $\varphi \notin Th(T \cup \Delta)$ for *some* assumption set Δ sanctioned by the semantics.

The computational complexity[1] of the above problems for all frameworks and semantics we consider is located at the lower end of the *polynomial hierarchy*. This is an infinite hierarchy of complexity classes above NP defined by using *oracle machines*, i.e. Turing machines that are allowed to call a subroutine—the *oracle*—deciding some fixed problem in constant time. Let $\mathcal{C}$ be a class of decision problems. Then, for any class $\mathcal{X}$ defined by resource bounds, $\mathcal{X}^{\mathcal{C}}$ denotes the class of problems decidable on a Turing machine with an oracle for a problem in $\mathcal{C}$ and a resource bound given by $\mathcal{X}$. Based on these notions, the sets Δ_k^p, Σ_k^p, and Π_k^p are defined as follows:

$\Delta_0^p = \Sigma_0^p = \Pi_0^p = \mathsf{P}$

$\Delta_{k+1}^p = \mathsf{P}^{\Sigma_k^p}, \quad \Sigma_{k+1}^p = \mathsf{NP}^{\Sigma_k^p}, \quad \Pi_{k+1}^p = \mathsf{co\text{-}NP}^{\Sigma_k^p}.$

The "canonical" complete problems are SAT for Σ_1^p=NP and k-QBF for Σ_k^p $(k > 1)$, where k-QBF is the problem of deciding whether the quantified boolean formula

$$\underbrace{\exists \vec{p}\, \forall \vec{q} \ldots}_{k \text{ alternating quantifiers starting with } \exists} \Phi(\vec{p}, \vec{q}, \ldots).$$

is true. The complementary problem, denoted by co-k-QBF, is complete for Π_k^p.

All problems in the polynomial hierarchy can be solved in polynomial time iff $\mathsf{P} = \mathsf{NP}$. Further, all these problems can be solved by worst-case exponential time algorithms. Thus, the polynomial hierarchy might not seem too meaningful. However, different levels of the hierarchy differ considerably in practice, e.g. methods working for moderately sized instances of NP-complete problems do not work for Σ_2^p-complete problems.

The complexity of the problems we are interested in has been extensively studied for existing logics for nonmonotonic reasoning under the standard, stability semantics [Cadoli and Schaerf, 1993; Gottlob, 1992; Niemelä, 1990; Marek and Truszczynski, 1993; Stillman, 1992]. In particular, the credulous reasoning problem is NP-complete for LP and Σ_2^p-complete for DL, and the sceptical reasoning problem is $\mathsf{co\text{-}NP}$-complete for LP and Π_2^p-complete for DL.

4 Generic Upper Bounds

We identify upper bounds for the credulous and sceptical reasoning problems by exploiting the following *guess-and-verify algorithm* that, in order to decide these problems, guesses an assumption set, verifies that it is sanctioned by the semantics, and verifies that the formula under consideration is derivable or not derivable, respectively, from the set of assumptions and the given theory in the underlying monotonic logic. The upper bounds are parametric on the complexity of the **derivability problem** in the underlying monotonic logic. Moreover, the upper bounds are determined by exploiting upper bounds for their sub-problem that an assumption set is sanctioned by the semantics, called the **assumption set verification problem**.

In LP, the underlying logic is (propositional) Horn logic, hence the derivability problem is P-complete (under log-space reductions) [Papadimitriou, 1994, p.176]. In DL, the underlying logic is classical (propositional) logic extended with domain-specific inference rules. However, these extra inference rules do not increase the complexity of reasoning. It is known (e.g. see [Gottlob, 1995, p.90]) that for any DL like propositional monotonic rule system S, checking whether $S \not\models \varphi$ is NP-complete. Therefore, the following proposition follows immediately.

Proposition 1 *Given a DL framework $\langle T, A, \overline{} \rangle$, deciding for a sentence $\varphi \in \mathcal{L}$ and an assumption set $\Delta \subseteq A$ whether $\varphi \in Th(T \cup \Delta)$ is* $\mathsf{co\text{-}NP}$*-complete.*

We now prove the basic membership result for flat frameworks in general and LP and DL in particular. In fact, flatness seems to be a computationally important property. For non-flat frameworks, the assumption set verification problem under the admissibility and preferability semantics seems to become harder in general.

Theorem 2 *Given a flat framework with derivability problem in $\mathcal{C}$, the* assumption set verification problem *is*

1. *in $\mathsf{P}^{\mathcal{C}}$ under the stability semantics,*
2. *in $\mathsf{P}^{\mathcal{C}}$ under the admissibility semantics, and*
3. *in $\mathsf{co\text{-}NP}^{\mathcal{C}}$ under the preferability semantics.*

Proof: 1. Only polynomially many $\mathcal{C}$-oracle calls are needed to verify that a given assumption set Δ does not attack itself and it attacks all assumptions $\alpha \notin \Delta$.

2. The following deterministic, polynomial-time algorithm using a $\mathcal{C}$-oracle decides whether a given assumption set Δ is admissible:

(a) Verify that Δ does not attack itself
($|\Delta|$ calls to a $\mathcal{C}$-oracle).

[1] For the following, we assume that the reader is familiar with the basic concepts of complexity theory [Papadimitriou, 1994], i.e., the complexity classes P, NP, and $\mathsf{co\text{-}NP}$ and the notion of completeness with respect to log-space reductions.

(b) Compute $A^* = \{\alpha \in A - \Delta | \Delta \text{ does not attack } \alpha\}$ ($|A - \Delta|$ calls to a $\mathcal{C}$-oracle).

(c) Verify that $A^* \cup \Delta$ does not attack Δ. (Polynomially many $\mathcal{C}$-oracle calls).

If test (c) fails, then Δ is not admissible, since $A^* \cup \Delta$ attacks Δ but, by (b), Δ does not attack A^* and, by (a), Δ does not attack itself.[2] Otherwise, let Δ' be any attack against Δ. If $\Delta' \subseteq A^* \cup \Delta$, then, by monotonicity of the underlying derivability, $A^* \cup \Delta$ attacks Δ, thus contradicting that test (c) succeeds. Therefore, $\Delta' \not\subseteq A^* \cup \Delta$. Let $\alpha \in \Delta' - A^* - \Delta$. By (b), Δ attacks α. Thus, Δ attacks Δ' and, by (a), Δ is admissible.

3. The following nondeterministic, polynomial-time algorithm using a $\mathsf{P}^{\mathcal{C}}$-oracle decides whether a given assumption set Δ is not preferred:

- Verify that Δ is admissible (one call to a $\mathsf{P}^{\mathcal{C}}$-oracle, by part (2)). If not, succeed, otherwise
- Guess a set $\Delta' \supset \Delta$.
- Verify that Δ' is admissible (one call to a $\mathsf{P}^{\mathcal{C}}$-oracle, by part (2)). If it is, succeed, else fail. ∎

The guess-and-verify algorithm and Theorem 2 directly give upper bounds for the credulous and (co-)sceptical reasoning problems. However, properties Prop_1 and Prop_2 in section 2 allow to reduce these upper bounds. Indeed, by Prop_1, credulous reasoning under the preferability semantics is equivalent to credulous reasoning under the admissibility semantics, and the two problems have the same upper bounds. Moreover, by Prop_2, the sceptical reasoning problem under the admissibility semantics reduces to the underlying derivability problem. As a consequence, the following upper bounds hold, for flat frameworks with a derivability problem in $\mathcal{C}$:

	Stable	Admissibility	Preferability
Credulous	$\mathsf{NP}^{\mathcal{C}}$	$\mathsf{NP}^{\mathcal{C}}$	$\mathsf{NP}^{\mathcal{C}}$
Sceptical	$\mathsf{co\text{-}NP}^{\mathcal{C}}$	$\mathcal{C}$	$\mathsf{co\text{-}NP}^{\mathsf{NP}^{\mathcal{C}}}$

In particular, the credulous reasoning problem for stability, admissibility and preferability semantics is in NP for LP and in Σ_2^p for DL, the sceptical reasoning problem for the stability semantics is in $\mathsf{co\text{-}NP}$ for LP and Π_2^p for DL, the sceptical reasoning problem for the admissibility semantics is in P for LP and $\mathsf{co\text{-}NP}$ for DL, and the sceptical reasoning problem for the preferability semantics is in Π_2^p for LP and Π_3^p for DL. The results summarised in section 3 imply that these upper bounds are tight for the stability semantics. Since the sceptical reasoning problem for the admissibility semantics reduces to the derivability problems in the underlying monotonic logic for LP and DL, and these are P-complete and $\mathsf{co\text{-}NP}$-complete, respectively, the corresponding upper bounds are also tight. In the next section we will prove that the remaining upper bounds are tight as well.

[2]See that if the framework is not flat, the set of assumptions $A^* \cup \Delta$ need not to be closed. Therefore, even if (c) fails, Δ can be still admissible, since it can be the case that Δ attacks an assumption that is *derivable* from $A^* \cup \Delta$.

5 Completeness Results

By instantiating the generic upper bounds of the previous section to the concrete reasoning problems we consider in the following, we obtain the necessary membership results. Therefore, in order to show completeness, it is sufficient to prove hardness.

The next two results show that credulous reasoning under the admissibility and preferability semantics is as hard as under the stability semantics. Intuitively, this is the case because the number of assumption sets that need to be considered under the admissibility semantics is not smaller than under the stability semantics, and it can be, as in the case of stability, exponentially large.

The following theorem is a direct corollary of the result by Saccà [1997] that the expressive power of DATALOG$^\neg$ queries under the "possible M-stable semantics" (corresponding to credulous reasoning under the preferability semantics) is DB-NP, i.e., such queries characterise all collections of databases that are recognisable in NP. From that it is immediate that credulous reasoning in propositional logic programs is NP-complete.

Theorem 3 *Credulous reasoning in LP under the admissibility and preferability semantics is* NP*-complete.*

As one would expect, reasoning in DL increases the computational complexity to the second level of the polynomial hierarchy.

Theorem 4 *Credulous reasoning in DL under the admissibility and preferability semantics is* Σ_2^p*-complete.*

Proof: We have seen that credulous reasoning under the preferability semantics coincides with credulous reasoning under the admissibility semantics. We prove the theorem by a straightforward reduction from 2-QBF to the credulous reasoning problem under the admissibility semantics. Assume the quantified boolean formula $\exists p_1, \ldots, p_n \, \forall q_1, \ldots, q_m \Phi$, with Φ a formula in 3CNF over the propositional variables $p_1, \ldots, p_n, q_1, \ldots, q_m$. We construct a default theory $(\emptyset, D)$ such that the given quantified boolean formula is true iff some admissible argument for $(\emptyset, D)$ contains Φ.

The language of $(\emptyset, D)$ consists of atoms $p_1, \ldots, p_n$. D consists of the default rules

$$\frac{Mp_i}{p_i}; \quad \frac{M\neg p_i}{\neg p_i}$$

for each $i = 1, \ldots, n$ (simulating the choice of a truth value for the propositional variable p_i in Φ).

Obviously, this construction of $(\emptyset, D)$ can be done in log-space. Moreover, it is easy to see that the given 2-QBF is true iff there exists an admissible extension of $(\emptyset, D)$ containing Φ. ∎

As noted earlier, sceptical reasoning under the admissibility semantics is "trivial" in the sense that it reduces to the underlying derivability problem. Therefore, the sceptical reasoning problem needs to be considered only for the preferability semantics. Theorem 2 suggests that this problem has higher complexity than the corresponding problem under the stability semantics, since in order to verify that a set is preferred we need to check that none of its supersets is admissible. The following two theorems show that we cannot do better than this.

Again Saccà [1997] has shown that the expressive power of DATALOG$^\neg$ queries under the "definite M-stable semantics" (corresponding to sceptical preferability semantics) coincides with the class DB-Π_2^p. Hence, as a corollary we immediately obtain the following result.

Theorem 5 *Sceptical reasoning in LP under the preferability semantics is Π_2^p-complete.*

We now show that sceptical reasoning has a similar effect on DL.

Theorem 6 *Sceptical reasoning in DL under the preferability semantics is Π_3^p-complete.*

Proof: We show that co-sceptical reasoning is Σ_3^p-hard by a reduction from 3-QBF. Assume the following quantified boolean formula: $\exists p_1,\ldots,p_n \, \forall q_1,\ldots,q_m \, \exists r_1,\ldots,r_k \Phi$, with Φ a formula in 3CNF over the propositional variables $p_1,\ldots,p_n$, $q_1,\ldots,q_m$, $r_1,\ldots,r_k$. We build a default theory $(\emptyset, D)$ such that the given quantified boolean formula is true iff some sentence F is not contained in some preferred argument for $(\emptyset, D)$.

The language of $(\emptyset, D)$ contains atoms $p_1,\ldots,p_n$, $q_1,\ldots,q_m$, and $r_1,\ldots,r_k$ as well as atoms $t_1,\ldots,t_n$, $s_1,\ldots,s_m$, intuitively holding if a truth value for the variables $p_1,\ldots,p_n$, $q_1,\ldots,q_m$ has been chosen. D consists of

$$\frac{M(p_i \wedge t_i)}{p_i \wedge t_i};\ \frac{M(\neg p_i \wedge t_i)}{\neg p_i \wedge t_i};\ \frac{M(q_j \wedge s_j)}{q_j \wedge s_j};\ \frac{M(\neg q_j \wedge s_j)}{\neg q_j \wedge s_j}$$

for each $i = 1,\ldots,n$, $j = 1,\ldots,m$ (to indicate that variables are assigned either true or false, but not both, and that a truth value for p_i and q_j has been chosen),

$$\frac{M\Phi}{\wedge_{j=1,\ldots,m} \neg s_j}$$

(to prohibit truth choices on q_j that render Φ *satisfiable*),

$$\frac{M\neg s_j}{\wedge_{h=1,\ldots,m} \neg s_h};\ \frac{M\neg t_i}{\wedge_{h=1,\ldots,n} \neg t_h \wedge \wedge_{h=1,\ldots,m} \neg s_h}$$

for each $i = 1,\ldots,n$, $j = 1,\ldots,m$ (to enforce that truth value choices are made either for all q_j's or for no q_j and truth value choices are made either for all p_i's or for none of the p_i's and q_j's), and

$$\frac{M\Phi}{\neg\Phi};\ \frac{M\neg t_i}{t_i};\ \frac{M\neg s_j}{s_j}.$$

for each $i = 1,\ldots,n$, $j = 1,\ldots,m$ (to guarantee that no admissible set contains $M\Phi$ or any of $M\neg t_i$ and $M\neg s_j$).

We will prove that the given quantified boolean formula is true iff there is a preferred argument not containing $F = \wedge_{j=1,\ldots,m} s_j$.

If v is a truth assignment to the p_i's, we denote by Δ_v^p the assumption set $\{M(p_i \wedge t_i) | v(p_i) = true\} \cup \{M(\neg p_i \wedge t_i) | v(p_i) = false\}$. Similarly, we denote by Δ_v^q the assumption set $\{M(q_i \wedge s_i) | v(q_i) = true\} \cup \{M(\neg q_i \wedge s_i) | v(q_i) = false\}$ for some truth assignment v to the q_i's.

First of all, it is obvious that no admissible set can contain any of the assumptions $M\Phi$, $M\neg s_i$, $M\neg t_i$. Furthermore, it is easy to see that for any truth assignment v to the p_i's, the set Δ_v^p is an admissible set. Moreover, every preferred assumption set must contain a set Δ_v^p for some truth assignment v to the p_i's. Finally, if Δ_v^p is not preferred, then there exists a truth assignment u to the q_i's such that $\Delta_v^p \cup \Delta_u^q$ is preferred.

Assume that the quantified boolean formula is true under a particular truth assignment u to the p_i's. We will show that the set Δ_u^p is a preferred assumption set. Suppose that we try to extend Δ_u^p by adding to it the set Δ_k^q for some truth assignment k to the q_i's. If the new set is admissible, it means that it counter attacks the attack $M\Phi$, ie. $\neg\Phi \in Th(D \cup \Delta_u^p \cup \Delta_k^q)$, hence the QBF is not true for the truth assignment u, a contradiction. Hence, $Th(D \cup \Delta_u^p)$ is a preferred argument that does contain $F = \wedge_{j=1,\ldots,m} s_j$.

Conversely, assume that $(\emptyset, D)$ has a preferred argument $P = Th(D, \Delta)$ that does not contain $F = \wedge_{j=1,\ldots,m} s_j$. Clearly $\Delta_v^p \subseteq \Delta$, for some truth assignment v to the p_i's. Since P is preferred and does not contain F, none of the sets $\Delta_v^p \cup \Delta_k^q$, for every possible truth assignment k to the q_i's is admissible. This means that none of these sets of assumptions can counter attack the attack $M\Phi$ and derive $\neg\Phi$, therefore the QBF is true.

Obviously, the above construction can be done in log-space. Thus, the construction of $(\emptyset, D)$ is a log-space reduction from 3-QBF to co-sceptical reasoning in DL under preferred arguments and Σ_3^p-hardness holds. ■

It should be noted that similar results to those for DL have been recently obtained for the case of disjunctive logic programs [Eiter *et al.*, 1998].

6 Discussion

We have shown that credulous reasoning in DL and LP using the admissibility and preferability semantics is as hard as it is under the standard, stability semantics. Moreover, sceptical reasoning under the preferability semantics is harder than under the stability semantics.

There appears to be a clash between these results and the intuition spelled out in the Introduction, namely, that admissibility and preferability arguments are seemingly easier to compute than stable extensions. However, our results are not as surprising as they might appear. Since the admissibility and preferability semantics *do not* restrict the number of extensions, one would expect that nonmonotonic reasoning under these semantics is as hard as under the stability semantics. The higher complexity of the sceptical reasoning problem under the preferability semantics is due to the fact that in order to verify that an assumption set is preferred, one needs to check that *none* of its supersets is admissible.

Of course, our results do not contradict the expectation that in practice constructing admissible arguments is often easier than constructing stable extensions. For example, given the propositional logic program $P \cup \{p\}$, with P any set of clauses not defining the atom p, the empty set is an admissible argument for the query p that can be constructed "locally", without accessing P. Moreover, if $P \cup \{p\}$ is stratified or order-consistent [Bondarenko *et al.*, 1997], p is guaranteed to be a credulous consequence of the program under the stability semantics. Indeed, in all cases where the stability semantics coincides with the preferability semantics (e.g. for stratified and order-consistent abstract frameworks) any sound (and complete) computational mechanism for the admissibility semantics is sound (and complete) for the stability semantics.

The "locality" feature of the admissibility semantics renders it a feasible alternative to the stability semantics in the first-order case, when the propositional version of the given abstract framework is infinite. For example, given the (negation-free) logic program: $\{q(f(X));\ p(0)\}$, the empty set of assumptions is an admissible argument for the query $p(0)$ that can be constructed "locally", even though the propositional version of the corresponding abstract framework is infinite.

The complexity results in this paper discredit sceptical reasoning under admissibility and preferability semantics as trivial and "unnecessarily" complex, respectively. However, this does not seem to matter for the envisioned applications of this semantics, because credulous reasoning only is required for these applications [Kowalski and Toni, 1996]. For example, in argumentation in practical reasoning in general and legal reasoning in particular, unilateral arguments are put forwards and defended against all counterarguments, in a credulous manner. Indeed, these domains appear to be particularly well suited for credulous reasoning under the admissibility semantics.

Acknowledgements

The first author has been partially supported by the DFG as part of the graduate school on *Human and Machine Intelligence* at the University of Freiburg and the third author has been partially supported by the UK EPSRC project "Logic-based multi-agent systems." We would also like to thank the anonymous reviewers for their comments on an earlier version of this paper.

References

[Bondarenko *et al.*, 1997] Andrei Bondarenko, Phan Minh Dung, Robert A. Kowalski, and Francesca Toni. An abstract, argumentation-theoretic framework for default reasoning. *Artificial Intelligence*, 93(1–2):63–101, 1997.

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

[Dung, 1991] Phan Minh Dung. Negation as hypothesis: an abductive foundation for logic programming. In K. Furukawa, editor, *Proceedings of the 8th International Conference on Logic programming*, pages 3–17, Paris, France, 1991. MIT Press.

[Eiter *et al.*, 1998] Thomas Eiter, Nicola Leone, and Domenico Saccà. Expressive power and complexity of partial models for disjunctive deductive databases. *Theoretical Computer Science*, 206:181–218, 1998.

[Gelfond and Lifschitz, 1988] Michael Gelfond and Vladimir Lifschitz. The stable model semantics for logic programming. In K. Bowen and R. A. Kowalski, editors, *Proceedings of the 5th International Conference on Logic programming*, pages 1070–1080, Seattle, WA, 1988. MIT Press.

[Gottlob, 1992] Georg Gottlob. Complexity results for nonmonotonic logics. *Journal for Logic and Computation*, 2(3):397–425, 1992.

[Gottlob, 1995] Georg Gottlob. The complexity of default reasoning under the stationary fixed point semantics. *Information and Computation*, 121:81–92, 1995.

[Kowalski and Toni, 1996] Robert A. Kowalski and Francesca Toni. Abstract argumentation. *Journal of Artificial Intelligence and Law*, 4(3–4):275–296, 1996.

[Marek and Truszczynski, 1993] Victor W. Marek and Miroslaw Truszczynski. *Nonmonotonic logic: context-dependent reasoning.* Springer-Verlag, Berlin, Heidelberg, New York, 1993.

[Niemelä, 1990] Ilkka Niemelä. Towards automatic autoepistemic reasoning. In *Logics in AI*, volume 478 of *Lecture Notes in Artificial Intelligence*. Springer-Verlag, Berlin, Heidelberg, New York, 1990.

[Papadimitriou, 1994] Christos H. Papadimitriou. *Computational Complexity.* Addison-Wesley, Reading, MA, 1994.

[Reiter, 1980] Raymond Reiter. A logic for default reasoning. *Artificial Intelligence*, 13(1):81–132, April 1980.

[Saccà and Zaniolo, 1990] Domenico Saccà and Carlo Zaniolo. Stable models and non-determinism in logic programs with negation. In *Proceedings of the 9th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database-Systems (PODS-90)*, pages 205–217, 1990.

[Saccà, 1997] Domenico Saccà. The expressive power of stable models for bound and unbound DATALOG queries. *Journal of Computer and System Sciences*, 54:441–464, 1997.

[Stillman, 1992] J. Stillman. The complexity of propositional default logics. In *Proceedings of the 10th National Conference of the American Association for Artificial Intelligence (AAAI-92)*, pages 794–799, San Jose, CA, July 1992. MIT Press.