From query answering to satisfiability in nonmonotonic logics

Riccardo Rosati
Dipartimento di Informatica e Sistemistica
Università di Roma "La Sapienza"
Via Salaria 113, 00198 Roma, Italy
rosati@dis.uniroma1.it

Abstract

We propose a unifying view of negation as failure, integrity constraints, and epistemic queries in nonmonotonic reasoning. Specifically, we study the relationship between satisfiability and logical implication in nonmonotonic logics, showing that, in many nonmonotonic formalisms, it is possible and easy to reduce logical implication to satisfiability. This result not only allows for establishing new complexity results for the satisfiability problem in nonmonotonic logics, but also establishes a clear relationship between the studies on epistemic queries and integrity constraints in monotonic knowledge bases with the work on negation by default in nonmonotonic reasoning and logic programming. From the perspective of the design of knowledge representation systems, such a reduction allows for defining both a simple method for answering epistemic queries in knowledge bases with nonmonotonic abilities, and a procedure for identifying integrity constraints in the knowledge base, which can be employed for optimizing reasoning in such systems.

1 Introduction

Research in logical formalization of nonmonotonic reasoning has come up with a variety of proposals, which try to overcome the limitations of classical logic with respect to the representation of forms of commonsense reasoning by identifying suitable ways for completing the knowledge of a rational agent. This corresponds to making assumptions and drawing conclusions on the basis of information not specified to the agent, which is not possible in a monotonic, first-order logic setting. On the one hand, nonmonotonicity allows for representing forms of commonsense reasoning in terms of a logical theory; on the other hand, some interesting properties of classical logic are generally missing in a nonmonotonic framework. In particular, the deduction theorem, namely the property

$$\Sigma \models \varphi \text{ iff } \models \Sigma \supset \varphi$$

does not hold in nonmonotonic logics, with the major exception of Levesque's logic of only-knowing[1] [7]. This property is important not only from a theoretical viewpoint, but also for the design of deductive methods for such logics. In fact, the deduction theorem implies that it is possible to turn logical implication (and hence all the basic reasoning tasks) into satisfiability.

In this paper we analyze in detail the relationship between epistemic queries and knowledge bases in nonmonotonic logics. The results of this study provide a unifying view of the notions of negation as failure, integrity constraints, and epistemic queries in nonmonotonic reasoning.

Specifically, we first study the relationship between logical implication and satisfiability in several nonmonotonic formalisms: Lifschitz's logic of minimal knowledge and negation as failure MKNF [8], McDermott and Doyle's modal logics [12, 13], Moore's autoepistemic logic [14], Halpern and Moses' modal logic of minimal knowledge [3], and Reiter's default logic [16]. We show the existence of easy reductions in all these formalisms: in particular, we provide a reduction of query answering to unsatisfiability in MKNF, which highlights the relationship between the notion of epistemic query [6, 7, 17] and the notion of negation as failure [2]. Such a reduction also allows for establishing the computational complexity of the satisfiability

[1] Actually, such a logic is monotonic, although it allows for representing knowledge closure.

problem in MKNF.

Notably, the existence of such reductions allows for defining simple methods for answering epistemic queries in knowledge representation systems with nonmonotonic abilities. Such methods rely on usual non-epistemic query answering abilities of the system, reducing the problem of computing answers to epistemic queries to the problem of checking satisfiability of a knowledge base. In particular, we show that it is possible to reduce epistemic query answering to satisfiability in any knowledge representation system which allows for expressing default rules.

Based on the above result, we then provide a generalization of the notion of epistemic integrity constraint introduced by Reiter [17, 1], which in turn highlights the deep relationship between the notions of integrity constraint and negation as failure. A clear connection is thus established between the studies on epistemic queries and integrity constraints in first-order, monotonic knowledge bases with the work on negation by default in nonmonotonic reasoning and logic programming. In particular, we prove that this generalized, epistemic view of integrity constraints corresponds to a form of nonmonotonic reasoning which is nicely captured by the notion of negation as failure.

Finally, the above characterization of integrity constraints can be exploited in order to speed up the deduction process in nonmonotonic knowledge bases. Specifically, we define an algorithm for establishing whether a formula is an integrity constraint in the setting of MKNF.

The paper is organized as follows. First, we recall nonmonotonic modal logics, in particular Lifschitz's MKNF and McDermott and Doyle's logics. Then, we deal with the problem of reducing query answering to unsatisfiability in nonmonotonic logics. Finally, we study the relationship between negation as failure and Reiter's epistemic interpretation of integrity constraints.

2 Preliminaries

In this section we briefly recall Lifschitz's logic MKNF [8] and McDermott and Doyle's (MDD) logics [13, 11].

MKNF is a modal logic with two epistemic operators: a "minimal knowledge" modality K and a "negation as failure" (also called "negation by default") modality *not*; however, for ease of notation, in the following we will use the symbol A (introduced in [10]), which stands for $\neg not$. We use $\mathcal{L}$ to denote a fixed propositional language built in the usual way from an alphabet $\mathcal{A}$ of propositional symbols (atoms), the symbols true, false, and the propositional connectives $\wedge, \neg$ ($\vee, \supset$ are defined as usual in terms of $\wedge, \neg$). Formulas from $\mathcal{L}$ are called *objective*. We denote with $\mathcal{L}_M$ the modal extension of $\mathcal{L}$ with the modalities K and A (and call *MKNF formula* a formula from $\mathcal{L}_M$), with $\mathcal{L}_K$ the set of formulas from $\mathcal{L}_M$ in which the modality A does not occur (called *K-formulas*), and with $\mathcal{L}_A$ the set of formulas from $\mathcal{L}_M$ in which K does not occur (called *A-formulas*). We call φ a *subjective* formula if each propositional symbol in φ appears in the scope of at least one modality. Moreover, we call *modal atom* any formula of the form $K\psi$ or $A\psi$; the set $MA(\varphi)$ of modal atoms of $\varphi \in \mathcal{L}_M$ is the set of modal atoms which are subformulas of φ, and $MA(\Sigma)$, when $\Sigma \subseteq \mathcal{L}_M$, is the union of the modal atoms of all formulas from Σ.

We now recall the notion of MKNF model. An *interpretation* I is a subset of propositional symbols from $\mathcal{A}$ ($p \in \mathcal{A}$ is true in I iff $p \in I$). Satisfiability of a formula $\varphi \in \mathcal{L}$ in I (which we denote as $I \models \varphi$) is defined in the usual way. A set of interpretations M is called *cluster*, since modal formulas are evaluated in M as in the Kripke model in which each world corresponds to an interpretation in M and the accessibility relation between worlds is universal. We denote with $Th(M)$ the set of modal formulas from $\mathcal{L}_K$ which are satisfied in M. An MKNF structure is a triple (I, M_k, M_a), where I is an interpretation and M_k, M_a are clusters, which are denoted respectively as the K-cluster and the A-cluster of (I, M_k, M_a). Satisfiability of a formula in an MKNF structure is inductively defined as follows:

1. if φ is an atom, φ is true in (I, M_k, M_a) iff $\varphi \in I$;
2. $\neg\varphi$ is true in (I, M_k, M_a) iff φ is not true in (I, M_k, M_a);
3. $\varphi_1 \wedge \varphi_2$ is true in (I, M_k, M_a) iff φ_1 is true in (I, M_k, M_a) and φ_2 is true in (I, M_k, M_a);
4. $K\varphi$ is true in (I, M_k, M_a) iff, for every $J \in M_k$, φ is true in (J, M_k, M_a);
5. $A\varphi$ is true in (I, M_k, M_a) iff, for every $J \in M_a$, φ is true in (J, M_k, M_a).

Given a pair (M_k, M_a) of clusters, and a formula $\varphi \in \mathcal{L}_M$, we write $(M_k, M_a) \models \varphi$ (and say that φ is satisfied in (M_k, M_a)) iff for each $I \in M_k$, φ is true in (I, M_k, M_a). When $\varphi \in \mathcal{L}_K$ sometimes we abbreviate $(M_k, M_a) \models \varphi$ with $M_k \models \varphi$, since the A-cluster M_a is not considered in the evaluation of K-formulas. If $\Sigma \subseteq \mathcal{L}_M$, we write $(I, M_k, M_a) \models \Sigma$ iff $(I, M_k, M_a) \models \varphi$ for each $\varphi \in \Sigma$.

Definition 1 *A cluster M is a* MKNF model *(or simply* model*) for $\Sigma \subseteq \mathcal{L}_M$ iff $(M, M) \models \Sigma$ and, for each* *cluster M', if $(M', M) \models \Sigma$ then $M' \not\supset M$.*

We say that a formula $\varphi \in \mathcal{L}_K$ is logically implied by $\Sigma \in \mathcal{L}_M$ in MKNF (and write $\Sigma \models \varphi$) iff φ is true in every MKNF model for Σ. We say that $\Sigma \subseteq \mathcal{L}_M$ is MKNF-satisfiable if Σ has an MKNF model (MKNF-unsatisfiable otherwise). When dealing with the entailment problem $\Sigma \models \varphi$, we call *knowledge base* the formula Σ, and *query* the formula φ. Notice that only K-formulas are allowed as queries, since each MKNF model corresponds to a structure where the A-cluster coincides with the K-cluster, therefore A would be interpreted in the same way as K in the query.

It has been proved [8] that MKNF is able to embed many of the best known formalisms for nonmonotonic reasoning, e.g. default logic, autoepistemic logic, circumscription, and extended disjunctive logic programs. Such a logic has therefore been considered as a unifying framework for nonmonotonic reasoning.

Example 2 Let $\Sigma = \{Kp\}$. The only MKNF model for Σ is $M = \{I : I \models p\}$. Hence, $\Sigma \models Kp$, and $\Sigma \models \neg K\psi$ for each $\psi \in \mathcal{L}$ such that $p \supset \psi$ is not a propositional tautology. Therefore, the agent modeled by Σ has minimal knowledge, in the sense that she only knows p and the objective facts logically implied by p. Also, let $\Sigma = \{\neg Ap \supset Kq\}$. It is easy to see that the only MKNF model for Σ is $M = \{I : I \models q\}$, since p can be assumed not to hold by the agent modeled by Σ, which is then able to conclude q.

It is easy to show that, when restricting to K-formulas, MKNF corresponds to the modal logic of minimal knowledge due to Halpern and Moses [3], also known as ground nonmonotonic modal logic $\mathsf{S5}_G$. The semantics of K-formulas can be given by simplifying Definition 1 as follows [22]: M is a model for $\Sigma \in \mathcal{L}_K$ iff $M \models \Sigma$ and, for each M', if $M' \models \Sigma$ then $M' \not\supset M$. In other words, when $\Sigma \in \mathcal{L}_K$, a cluster M satisfying Σ is compared with all other clusters satisfying Σ, while in the case $\Sigma \in \mathcal{L}_M$ the cluster M is only compared with those clusters M' such that (M', M) satisfies Σ. Hence, the main difference between MKNF and $\mathsf{S5}_G$ lies in the fact that in $\mathsf{S5}_G$ all models are maximal wrt set containment (or minimal wrt the objective knowledge which holds in the model), while in MKNF this property is not generally true. E.g., the theory $\Sigma = \{Aq \supset Kq\}$ has two models: M_1, corresponding to the set of all interpretations (which represents the case in which Aq is not assumed to hold); and $M_2 = \{I : I \models q\}$. Namely, if Aq is assumed to hold, then Kq must be assumed to hold in order to satisfy Σ: that is, the initial assumption is *justified* by the knowledge derived on the basis of such assumption [10].

Finally, we briefly recall MDD logics, a family of unimodal nonmonotonic modal formalisms which are among the most studied logics for nonmonotonic reasoning [12, 13, 11]. Here we report a characterization of MDD logics in terms of a preference semantics on Kripke models [20], which is better suited to our purposes than the original, equivalent characterization in terms of expansions (i.e. sets of modal formulas).

Definition 3 *Given a modal logic $\mathcal{S}$, a cluster M is an $\mathcal{S}_{MDD}$ model for $\Sigma \subseteq \mathcal{L}_K$ if: (i) $M \models \Sigma$; (ii) for each Kripke model $\mathcal{N}$ for the logic $\mathcal{S}$, if $\mathcal{N} \models \Sigma \cup \{\neg K\psi : \psi \in \mathcal{L}_K - Th(M)\}$, then $Th(\mathcal{N}) = Th(M)$.*

We write $\Sigma \models_{\mathcal{S}_{MDD}} \varphi$ iff φ is satisfied in all $\mathcal{S}_{MDD}$ models for Σ. We say that Σ is $\mathcal{S}_{MDD}$-satisfiable if it has an $\mathcal{S}_{MDD}$ model. We remark that the logic $\mathsf{KD45}_{MDD}$ corresponds to well-known Moore's autoepistemic logic (AEL) [14]. It can also be shown that AEL in turn corresponds to the K-free fragment of MKNF [10, 19]. We denote with Σ^K the theory obtained by replacing in Σ all occurrences of the modality A with K in all formulas from Σ.

Proposition 4 *Let $\Sigma \subseteq \mathcal{L}_A$. Then, M is an MKNF model for Σ iff M is a $\mathsf{KD45}_{MDD}$ model for Σ^K.*

The previous property implies that the modality A has in MKNF exactly the *same* interpretation of the modal operator of autoepistemic logic.

3 From query answering to satisfiability

In this section we study the problem of reducing logical implication (or query answering) to unsatisfiability in nonmonotonic logics. In particular, we analyze this problem in the following settings: MKNF, MDD logics, AEL, $\mathsf{S5}_G$, and default logic.

Generally speaking, it is not immediate to perform such a reduction in nonmonotonic logics, since the deduction theorem does not hold in these formalisms. In order to reduce query answering to unsatisfiability, we look for a translation $\mathcal{E}(\varphi)$ of the query φ such that, for each knowledge base Σ, $\Sigma \cup \{\mathcal{E}(\varphi)\}$ is unsatisfiable iff φ is satisfied in all models for Σ. The basic idea is to obtain a formula $\mathcal{E}(\varphi)$ from φ such that the effect of adding $\mathcal{E}(\varphi)$ to Σ is to rule out all those models for Σ satisfying φ, while preserving all other models for Σ. It turns out that such a translation is extremely easy to define for many nonmonotonic logics.

We start by analyzing the case of MDD logics.

Theorem 5 *Let* $\mathsf{K} \subseteq \mathcal{S} \subseteq \mathsf{S5}, \Sigma \subseteq \mathcal{L}_K, \varphi \in \mathcal{L}_K$. *Then,* $\Sigma \models_{\mathcal{S}_{\mathrm{MDD}}} \varphi$ *iff the theory* $\Sigma \cup \{\neg K\varphi\}$ *is* $\mathcal{S}_{MDD}$*-unsatisfiable.*

The proof of this theorem can be obtained by showing that adding a formula of the form $\neg K\varphi$ to Σ not only rules out all $\mathcal{S}_{MDD}$ models for Σ satisfying φ, but also cannot give rise to any model for $\Sigma \cup \{\varphi\}$ different from a model for Σ. In fact, suppose M is a cluster satisfying both Σ and $\neg K\varphi$. It is easy to see that if M is an $\mathcal{S}_{MDD}$ model for Σ, then M is an $\mathcal{S}_{MDD}$ model for $\Sigma \cup \{\neg K\varphi\}$. Now suppose M is not an $\mathcal{S}_{MDD}$ model for Σ. Then, by Definition 3 there exists a Kripke model $\mathcal{N}$ such that $\mathcal{N} \models \Sigma \cup \{\neg K\psi : \psi \in \mathcal{L}_K - Th(M)\}$, and since $M \not\models \varphi$, it follows that $\neg K\varphi \in \{\neg K\psi : \psi \in \mathcal{L}_K - Th(M)\}$. Consequently, the properties of model $\mathcal{N}$ imply that Condition 2. in Definition 3 is false, hence M is not an $\mathcal{S}_{MDD}$ model for $\Sigma \cup \{\neg K\varphi\}$.

An analogous result can be obtained in the case of MKNF, through the translation function $\mathcal{E}_{MKNF}(\cdot)$.

Definition 6 *Let* $\varphi \in \mathcal{L}_K$. *Then,* $\mathcal{E}_{MKNF}(\varphi) \in \mathcal{L}_M$ *is the formula obtained from* φ *by substituting each occurrence of the modality* K *in* φ *with* A.

The proof of the following property follows from the semantic definition of MKNF, in a way similar to the proof of Theorem 5.

Theorem 7 *Let* $\Sigma \subseteq \mathcal{L}_M$, $\varphi \in \mathcal{L}_K$. *Then,* $\Sigma \models_{\mathrm{MKNF}} \varphi$ *iff the theory* $\Sigma \cup \{\mathcal{E}_{MKNF}(\neg K\varphi)\}$ *is MKNF-unsatisfiable.*

The above theorem establishes a precise relationship between the notions of epistemic query and negation as failure: more precisely, it implies a duality between the epistemic modality used in query expressions and the negation as failure (or autoepistemic) operator of MKNF. Informally, a formula can pass from the right-hand side to the left-hand side of the logical implication symbol in MKNF by negating it and replacing all its modalities with the A operator.

Notice that both Theorem 5 and Theorem 7 imply the same translation for autoepistemic logic (corresponding to the case $\mathcal{S} = \mathsf{KD45}$ in Theorem 5 and to the use of the only modality A in MKNF).

On the other hand, it is easy to show that it is impossible to easily reduce query answering to unsatisfiability in $\mathsf{S5}_G$, the A-free fragment of MKNF. In fact, the following computational result holds: the problem $\Sigma \models_{\mathsf{S5}_G} \varphi$ is Π^p_3-complete,[2] while $\mathsf{S5}_G$-unsatisfiability is coNP-complete. Hence, unless the polynomial hierarchy collapses, it is not possible to find a polynomial-time translation of φ for turning query answering into unsatisfiability in $\mathsf{S5}_G$.

Finally, we deal with the problem of reducing query answering to unsatisfiability in Reiter's default logic. First, we briefly recall Reiter's default logic (DL in the following). A default theory is a pair (D, W) such that $W \in \mathcal{L}$ and D is a set of default rules, each one of the form

$$d = \frac{\alpha : \beta_1, \ldots, \beta_n}{\gamma} \tag{1}$$

where $n \geq 0$ and $\alpha, \beta_1, \ldots, \beta_n, \gamma \in \mathcal{L}$. We recall that a default extension is a deductively closed set of propositional formulas, which can be constructed starting from $\{W\}$ and increasing such a set by applying the special inference rules in D. In order to apply a default rule of the form (1) (i.e. adding the conclusion γ to the extension), the prerequisite α must be implied by the set of formulas representing the part of the extension constructed so far, while each negated justification $\neg\beta_i$ must not be implied set of formulas obtained after applying all defaults. We refer to [16] for a detailed definition of the semantics of DL.

In order to define epistemic queries over default theories, we establish a natural correspondence between extensions and clusters: we say that $\varphi \in \mathcal{L}_K$ is logically implied by a default theory (D, W) iff, for each extension E of (D, W), $M(E) \models \varphi$, where $M(E)$ is the cluster $\{I : I \models E\}$.

We now define a translation $\mathcal{E}_{DL}(\cdot)$ of epistemic queries into default rules. Let $\varphi \in \mathcal{L}_K$. We denote with $DNF(\varphi)$ the formula corresponding to a modal disjunctive normal form of φ, precisely the following formula (which is equivalent to $K\varphi$ in modal logic $\mathsf{S5}$):

$$DNF(\varphi) = \bigwedge_i \neg K\alpha^i \vee K\beta^i_1 \vee \ldots \vee K\beta^i_{n_i} \tag{2}$$

where all $\alpha^i, \beta^i_j \in \mathcal{L}$. Any formula $\varphi \in \mathcal{L}_K$ can be put in such a normal form (e.g. through the transformation reported in [4]).

Given a formula φ such that $DNF(\varphi)$ has the form (2), $\mathcal{E}_{DL}(\varphi)$ is the set of default rules

$$\bigcup_i \left\{ \frac{\alpha^i : \neg\beta^i_1, \ldots, \neg\beta^i_{n_i}}{\mathsf{false}} \right\}$$

In the following, we call *conclusion-free* default a default rule whose conclusion is the formula false. Observe that the translation $\mathcal{E}_{DL}(\cdot)$ is polynomial if φ is

[2] We refer to [5] for the definition of the complexity classes mentioned in the paper.

in modal conjunctive normal form, i.e.:

$$\varphi = \bigvee_i K\alpha^i \wedge \neg K\beta_1^i \wedge \ldots \wedge \neg K\beta_{n_i}^i$$

Theorem 8 *Let (D, W) be a default theory, and let $\varphi \in \mathcal{L}_K$. Then, φ is logically implied by (D, W) iff the default theory $(D \cup \mathcal{E}_{DL}(\neg K\varphi), W)$ has no extensions.*

The above results show that in general it is quite easy to reduce query answering to unsatisfiability in nonmonotonic logics. Moreover, the reductions presented can be employed in the realization of epistemic query answering in nonmonotonic knowledge representation systems: in fact, a simple method for adding epistemic queries to a system with the ability of expressing negation as failure consists of first adding the translation of the epistemic query to the knowledge base, and then checking unsatisfiability of the knowledge base thus obtained through the non-epistemic query false. E.g., such a method can be employed in those knowledge representation systems which provide the ability of expressing defaults for answering epistemic queries (in modal conjunctive normal form) in such systems, by first translating each epistemic query in terms of a set of conclusion-free defaults, then adding such defaults to the knowledge base, and finally checking whether the new knowledge base implies the query false.

Finally, the above results can be used, together with known computational analyses of query answering in nonmonotonic logics, for establishing the computational characterizations of the satisfiability problem in such formalisms. The following theorem characterizes the computational complexity of satisfiablity in MKNF.

Theorem 9 *Let $\Sigma \subseteq \mathcal{L}_M$. Then, the problem of establishing whether Σ is MKNF-unsatisfiable is Π_3^p-complete.*

The proof of the above property directly follows from the following facts: (i) query answering in MKNF is Π_3^p-complete [18]; (ii) $\mathcal{E}_{MKNF}(\varphi)$ can be computed in time linear wrt the size of φ; (iii) validity of Theorem 7.

4 Integrity constraints in nonmonotonic logics

In this section we deal with the representation of integrity constraints (ICs) in nonmonotonic logics. We start by recalling Reiter's epistemic interpretation of ICs in first-order knowledge bases [17]: an IC is not a statement about the world, it is a statement about what the knowledge base is said to know. Under this interpretation, an IC φ is not analogous to a usual piece of information in the knowledge base Σ, which causes inconsistency of $\Sigma \cup \{\varphi\}$ iff φ contradicts other information in Σ: it is a property which *must hold* in Σ in order to preserve consistency of Σ.

Reiter pointed out that this kind of ICs could be formalized in terms of epistemic queries to first-order knowledge bases, if the knowledge base is interpreted according to a minimal knowledge semantics (what is not logically implied by Σ is assumed as *not known* by Σ). Under this semantics, any consistent first-order knowledge base Σ has one (epistemic) model, which consists of the set of all first-order interpretations satisfying Σ. An integrity constraint ψ can be checked by verifying whether the property ψ (which is in general an epistemic formula) is satisfied in this model: if it is not the case, then Σ is inconsistent wrt ψ.

Example 10 Let φ, ψ be propositional formulas, and let $\mathcal{I}$ be the IC: "If φ is known to hold, then ψ must be known to hold", which expresses the fact that those models for Σ in which φ holds and ψ does not hold should be ruled out. $\mathcal{I}$ can be expressed as an epistemic query by the formula $\mathcal{I}' = K\varphi \supset K\psi$, which can be used for discarding such models during the evaluation of another query ξ against Σ, since it can be shown that $\Sigma \models_{MKNF} \xi$ iff $\Sigma \models_{MKNF} (\mathcal{I}' \supset \xi)$.

In a nonmonotonic logic setting, a knowledge base has in general many epistemic models. The natural generalization of Reiter's interpretation of integrity constraints to this case is to consider a formula as an IC iff the effect of adding it to a knowledge base reduces the set of models of the knowledge base, without producing any new model. That is, φ is an IC iff for each knowledge base Σ, $MOD(\Sigma) \supseteq MOD(\Sigma \cup \{\varphi\})$, where $MOD(\Sigma)$ represents the set of models of Σ. In other words, the effect of φ is just to rule out those models for Σ which do not satisfy φ.

We formally generalize the above notion of integrity constraint to the case of MKNF through the following definition (in which $MOD(\Sigma)$ denotes the set of MKNF models for Σ), which can be easily instantiated to the case of all other nonmonotonic logics dealt with by the paper.

Definition 11 *A formula $\varphi \in \mathcal{L}_M$ is an integrity constraint iff for each $\Sigma \subseteq \mathcal{L}_M$, $MOD(\Sigma) \supseteq MOD(\Sigma \cup \{\varphi\})$.*

Let us now go back to the reduction of query answering to satisfiability in MKNF. It is not hard to see that, given any query $\psi \in \mathcal{L}_K$, the formula $\mathcal{E}_{MKNF}(\psi)$ is an IC. This shows the perfect duality between the above notion of integrity constraint and the notion of epistemic query: not only any IC can be checked through an epistemic query (as shown by Reiter), but also any epistemic query has a corresponding IC. This property, together with the relationship between negation as failure and epistemic queries previously illustrated, establishes a tight connection between integrity constraints and negation as failure: any IC in MKNF is equivalent to a subjective A-formula. That is, the negation as failure (or autoepistemic) modality is *the* epistemic operator for expressing ICs.

The above results show that epistemic queries and integrity constraints are dual notions, and both can be naturally expressed by the negation as failure modality.

Example 12 (10 continued) We now show that $\mathcal{I}$ can directly be expressed as a formula inside the MKNF knowledge base, by simply substituting in $\mathcal{I}'$ all occurrences of K with A: $\mathcal{I}'' = A\varphi \supset A\psi$. Indeed, the models for $\Sigma \cup \{\mathcal{I}''\}$ are exactly those models for Σ verifying $\mathcal{I}$ (and $\mathcal{I}'$). Adding $\mathcal{I}'$ to Σ does *not* correctly formalize $\mathcal{I}$: informally, $\mathcal{I}'$ rules out only those models for Σ which satisfy φ and are inconsistent with ψ (e.g. which satisfy $\neg\psi$), while it *forces* knowledge of ψ in those models for Σ satisfying φ and consistent with ψ, which is not the intended meaning of the IC.

As for reasoning in nonmonotonic knowledge bases, recognizing ICs may help to speed up the deduction process in such systems. For example, it would be very useful to establish that a piece of information that has to be added to a knowledge base is an IC, since in this case the knowledge base shows a "monotonic" behaviour. Suppose e.g. that Σ has to be updated with a formula φ. Now, if φ is an IC, then (i) it could be processed by an ad-hoc procedure for ICs: if e.g. the models of Σ have been stored, then it is sufficient to evaluate φ against such models in order to find the set of models of $\Sigma \cup \{\varphi\}$ (those models of Σ satisfying φ); (ii) all queries true in Σ are also true in $\Sigma \cup \{\varphi\}$, which can be helpful for reusing previously computed answers to queries.

A first and straightforward way to identify ICs is by looking at their syntactic form. E.g. in MKNF, the fact that φ is a subjective A-formula implies that φ is an IC; in DL, any conclusion-free default is an IC. However, these are just sufficient conditions for establishing whether a piece of information is an IC: e.g., it can be shown that the formula $K\varphi \supset A\psi$ with $\varphi, \psi \in \mathcal{L}$ is an IC in MKNF, while in DL any default rule whose conclusion is propositionally equivalent to false is an IC. Such examples show that, in general, it is not immediate to check a formula against the necessary and sufficient condition reported in Definition 11.

In the following we study the problem of establishing whether a formula is an IC in the setting of MKNF. We remark that in principle the techniques employed in this case can also be used for other nonmonotonic modal logics. First, we provide an alternative characterization of *non*-ICs in MKNF, which is easily derived from the definition of MKNF model.

Theorem 13 *A formula $\varphi \in \mathcal{L}_M$ is not an integrity constraint iff there exist two clusters M, M' such that the following conditions hold: (i) $(M, M) \models \varphi$; (ii) $M' \supset M$; (iii) $(M', M) \not\models \varphi$.*

In fact, it can be shown that the existence of two clusters M, M' verifying the conditions in the above theorem imply the existence of a $\Sigma \subseteq \mathcal{L}_M$, which is both satisfied in (M, M) and in (M', M) (and hence M is not an MKNF model for Σ), while M is an MKNF model for $\Sigma \cup \{\varphi\}$.

Based on the above theorem, it is possible to define an algorithm for checking whether a formula $\varphi \in \mathcal{L}_M$ is an IC. The technique used in the algorithm for verifying the conditions reported in Theorem 13 is based on a finite characterization of clusters which is widely used in deduction methods for nonmonotonic modal logics (see e.g. [15, 11]). Specifically, it is possible to reason about the infinite set of all possible clusters by analyzing the finite set of all possible partitions of modal atoms of φ, and to represent a single cluster through a propositional formula, called *objective knowledge* of the cluster, such that all interpretations in the cluster satisfy such a formula.

In order to define the algorithm, we need some auxiliary definitions. Let P, N be disjoint subsets of modal atoms from $\mathcal{L}_M$. Then, $\varphi(P, N)$ is the formula obtained by replacing in φ each occurrence of a modal atom from P with true and each occurrence of a modal atom from N with false. We also give the following definitions:

$$
\begin{aligned}
Obj^-(P,N) &= \bigwedge_{K\psi \in P} \psi(P,N) \\
Obj_A(P,N) &= \bigwedge_{A\psi \in P} \psi(P,N) \\
Obj(\varphi,P,N) &= Obj^-(P,N) \wedge \varphi(P,N)
\end{aligned}
$$

Let $\varphi \in \mathcal{L}_M$, and let (P, N) be a partition of $MA(\varphi)$. Then, (P, N) is *consistent* with $\varphi \in \mathcal{L}_M$ iff:

(i) the propositional formula $Obj(\varphi, P, N)$ is consistent;
(ii) the propositional formula $Obj_A(P, N)$ is consistent;
(iii) for each $K\psi \in N$, $Obj(\varphi, P, N) \not\vdash \psi(P, N)$;
(iv) for each $A\psi \in N$, $Obj_A(P, N) \not\vdash \psi(P, N)$.

In the following, φ^K denotes the formula obtained from φ by replacing all modalities A with K, and $\vdash$ denotes logical implication in propositional logic. The algorithm is reported below.

Algorithm Not-IC(φ)
Input: formula $\varphi \in \mathcal{L}_M$;
Output: true if φ is not an IC, false otherwise.
if there exist partition (P, N) of $MA(\varphi^K)$
and partition (P', N') of $MA(\varphi)$
such that
(a) (P, N) is consistent with φ^K **and**
(b) (P', N') is consistent with true **and**
(c) $Obj(\varphi, P, N) \vdash Obj^-(P', N')$ **and**
(d) $Obj^-(P', N') \not\vdash Obj(\varphi, P, N)$ **and**
(e) $Obj^-(P', N') \not\vdash \varphi(P', N')$ **and**
(f) **for each** $A\psi \in MA(\varphi)$,
$A\psi \in P'$ **iff** $Obj(\varphi, P, N) \vdash \psi(P', N')$
then return true
else return false

Correctness of the above algorithm can be proved by exploiting previous results on finite characterizations of clusters [11, 18]. An intuitive explanation of the algorithm can be given as follows. Roughly speaking, partition (P, N) represents the MKNF structure (M, M), while (P', N') represents (M', M). $Obj(\varphi, P, N)$ represents the objective knowledge of M, while $Obj^-(P', N')$ represents the objective knowledge of M'. The three conditions reported in Theorem 13 are checked by the algorithm as follows. (i) $(M, M) \models \varphi$ is verified through condition (a); (ii) $M' \supset M$ is verified through conditions (c) and (d); (iii) $(M', M) \not\models \varphi$ is verified through condition (e). Finally, condition (b) is necessary in order to detect when (P', N') is a self-contradictory guess on modal formulas, while condition (f) guarantees that the two structures identified by (P, N) and (P', N') have the same A-cluster.

Theorem 14 *Let $\varphi \in \mathcal{L}_M$. Then, φ is an IC iff Not-IC(φ) returns* false*.*

It can be shown that the above algorithm, when considered as a non-deterministic procedure, has a cost of Σ_2^p. Hence, the problem of establishing whether an MKNF formula is an IC is in Π_2^p.

5 Conclusions

The main contributions of this paper can be summarized as follows. First, logical implication can be easily reduced to satisfiability in many nonmonotonic logics. Second, there exists a tight relationship between the notions of epistemic query, integrity constraints, and negation as failure. Third, it is possible to exploit the above results for improving automated nonmonotonic reasoning methods. In particular, the relationship between negation as failure and epistemic queries suggests a new reading of the computational and epistemological properties of logic programs with negation as failure. We are currently addressing this research topic.

Acknowledgments

The author is thankful to Joeri Engelfriet, Daniele Nardi, and Francesco M. Donini for stimulating discussions on the subject of the paper.

References

[1] R. Demolombe and A. Jones. Integrity constraints revisited. *Journal of the IGPL*, 4(3):369–383, 1996.

[2] M. Gelfond and V. Lifschitz. Classical negation in logic programs and deductive databases. *New generation computing*, 9:365–385, 1991.

[3] J. Y. Halpern and Y. Moses. Towards a theory of knowledge and ignorance: Preliminary report. Technical Report CD-TR 92/34, IBM, 1985.

[4] G. E. Hughes and M. J. Cresswell. An introduction to modal logic. Methuen, London, 1968.

[5] D. S. Johnson. A catalog of complexity classes. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume A, chapter 2. Elsevier, 1990.

[6] H.J. Levesque. Foundations of a functional approach to knowledge representation. *AI Journal*, 23:155–212, 1984.

[7] H.J. Levesque. All I know: a study in autoepistemic logic. *AI Journal*, 42:263–310, 1990.

[8] V. Lifschitz. Nonmonotonic databases and epistemic queries. In *Proc. of IJCAI-91*, 381–386, 1991.

[9] V. Lifschitz. Minimal belief and negation as failure. *AI Journal*, 70:53–72, 1994.

[10] F. Lin and Y. Shoham. A logic of knowledge and justified assumptions. *AI Journal*, 57:271–289, 1992

[11] W. Marek and M. Truszczyński. *Nonmonotonic Logics – Context-Dependent Reasoning*. Springer-Verlag, 1993.

[12] D. McDermott and J. Doyle. Non-monotonic logic I. *AI Journal*, 13:41–72, 1980.

[13] D. McDermott. Non-monotonic logic II: Non-monotonic modal theories. *Journal of the ACM*, 29:33–57, 1982.

[14] R. C. Moore. Semantical considerations on non-monotonic logic. *AI Journal*, 25:75–94, 1985.

[15] I. Niemelä. Decision procedure for autoepistemic logic. In *Proc. of CADE-88*, 675–684, 1988.

[16] R. Reiter. A logic for default reasoning. *AI Journal*, 13:81–132, 1980.

[17] R. Reiter. What should a database know? *Journal of Logic Programming*, 14:127–153, 1990.

[18] R. Rosati. Reasoning with minimal belief and negation as failure: algorithms and complexity. In *Proc. of AAAI-97*, 430–435, 1997.

[19] R. Rosati. Embedding minimal knowledge into autoepistemic logic. *Proc. of the 6th Conf. of the Italian Association for AI (AI*IA-97)*, 231–241, LNAI 1321. Springer-Verlag, 1997.

[20] G. Schwarz. Minimal model semantics for non-monotonic modal logics. In *Proc. of LICS-92*, 34–43. IEEE Press, 1992.

[21] G. Schwarz and M. Truszczyński. Minimal knowledge problem: a new approach. *AI Journal*, 67:113–141, 1994.

[22] Y. Shoham. Nonmonotonic logics: Meaning and utility. In *Proc. of IJCAI-87*, pp. 388–392, 1987.