# Classifying Description Logics

**Natasha Kurtonina**
Center for Logic, Ghent University
Rozier 44, 9000 Ghent, Belgium
Natasha.Kurtonina@rug.ac.be

**Maarten de Rijke**
Dept. of Computer Science, University of Warwick
Coventry CV4 7AL, England
mdr@dcs.warwick.ac.uk

## Abstract

We describe a method for characterizing the expressive power of description logics. The method is essentially model-theoretic in nature, and it is applied to obtain expressiveness results for a wide range of logics in the well-known $\mathcal{FL}^-$ and $\mathcal{AL}$ hierarchies. As a corollary we obtain a complete classification of the relative expressive power of these logics.

## 1 Introduction

In the field of description logics, and indeed, in Knowledge Representation in general, one of the guiding themes in the design of representation formalisms has been the slogan 'feasibility vs. expressiveness.' Or: the more expressive a formalism is, the higher the computational costs of the reasoning tasks that can be performed in it. The complexity of satisfiability and subsumption problems for description logics has been studied extensively [Donini *et al.*, 1997], but the issue of expressiveness has hardly been addressed in a formal and rigorous manner; we are aware of only two publications on this topic [Baader, 1996; Borgida, 1996].

While we don't claim to have the final analysis of the interplay between feasibility and expressiveness, we do hope to contribute to a better understanding of the issue by offering a formal yet intuitive explanation in model-theoretic terms of the expressive power of a wide range of description logics. Briefly, we identify description logics as fragments of a common background logic, characterize these fragments semantically in terms of preservation results, and then compare description logics by comparing the corresponding fragments. The main result is a complete classification of a wide range of description logics.

We proceed as follows. Section 2 recalls the logics we will consider, as well as their semantics. Section 3 explains our methods, and Section 4 summarizes our main results. In Section 5 we briefly discuss related work, and in Section 6 we mention our ongoing work.

## 2 Preliminaries

The description logics we consider are built up using the constructors in Table 1. Recall that $\mathcal{FL}^-$ has universal quantification, conjunction and unqualified existential quantification $\exists R.\top$; see [Brachman and Levesque, 1984]. To simplify the formulation of our results we will assume that $\mathcal{FL}^-$ contains top and bottom. $\mathcal{AL}$ extends $\mathcal{FL}^-$ by negation of (atomic) concept names. Extensions of $\mathcal{FL}^-$ and $\mathcal{AL}$ are denoted by postfixing the name of the constructors being added.

| Constructor | Syntax | Semantics |
|---|---|---|
| concept name | $A$ | $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ |
| top | $\top$ | $\Delta^{\mathcal{I}}$ |
| bottom | $\bot$ | $\emptyset$ |
| conjunction | $C \sqcap D$ | $C^{\mathcal{I}} \cap D^{\mathcal{I}}$ |
| disjunction ($\mathcal{U}$) | $C \sqcup D$ | $C^{\mathcal{I}} \cup D^{\mathcal{I}}$ |
| negation ($\mathcal{C}$) | $\neg C$ | $\Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$ |
| univ. quant. | $\forall R.C$ | $\{d_1 \mid \forall d_2\, (d_1, d_2) \in R^{\mathcal{I}} \to d_2 \in C^{\mathcal{I}}\}$ |
| exist. quant. ($\mathcal{E}$) | $\exists R.C$ | $\{d_1 \mid \exists d_2\, (d_1, d_2) \in R^{\mathcal{I}} \wedge d_2 \in C^{\mathcal{I}}\}$ |
| number | $(\geq nR)$ | $\{d_1 \mid \lvert\{(d_1, d_2) \in R^{\mathcal{I}}\}\rvert \geq n\}$ |
| restriction ($\mathcal{N}$) | $(\leq nR)$ | $\{d_1 \mid \lvert\{(d_1, d_2) \in R^{\mathcal{I}}\}\rvert \leq n\}$ |
| role name | $R$ | $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ |
| role conj. ($\mathcal{R}$) | $Q \sqcap R$ | $Q^{\mathcal{I}} \cap R^{\mathcal{I}}$ |

Table 1: Constructors in First-Order Description Logics

We refer the reader to [Donini *et al.*, 1996] for a recent overview of the area.

Description logics are interpreted on *interpretations* $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where $\Delta^{\mathcal{I}}$ is a non-empty domain, and $\cdot^{\mathcal{I}}$

is a function assigning subsets of $\Delta^{\mathcal{I}}$ to concept names and binary relations over $\Delta^{\mathcal{I}}$ to role names; complex concepts and roles are interpreted using the recipes in Table 1. The *semantic value* of an expression $E$ in an interpretation $\mathcal{I}$ is the set $E^{\mathcal{I}}$; two expressions are *equivalent* if they have equal semantic values in every interpretation.

## 3 The Method

Let $\mathcal{L}_1$ and $\mathcal{L}_2$ be description logics. We say that $\mathcal{L}_1$ is *at least as expressive as* $\mathcal{L}_2$ if for every expression in $\mathcal{L}_2$ there is an equivalent expression in $\mathcal{L}_1$ that uses (at most) the same concept and role names.

Our method for characterizing and comparing the expressive power of description logics consists of the following:

I a mapping taking description logics to fragments of a 'background logic';

II model-theoretic characterizations of such fragments; and

III comparisons of description logics via comparisons of the corresponding fragments, using the semantic characterizations to separate logics.

Let us consider each of these in turn, starting with item I. As is well-known, each extension of $\mathcal{FL}^-$ that is defined using the constructors from Table 1 can be viewed as a fragment of first-order logic over a suitable vocabulary; thus, in this paper we use first-order logic as our background logic.

Item II is the heart of the method. With each fragment corresponding to a description logic $\mathcal{L}$ that is defined using the constructors in Table 1 we will associate an $\mathcal{L}$-relation between interpretations that characterizes $\mathcal{L}$ in the following way. Roughly speaking, a first-order formula is (equivalent to) an $\mathcal{L}$-concept if, and only if, it remains true under passing from an interpretation to an $\mathcal{L}$-related interpretation. These characteristic $\mathcal{L}$-relations are all derived from the notion of bisimulation known from concurrency theory and modal logic.

As to item III, the notion of preservation gives us the tools we need to distinguish between description logics. To show that a description logic $\mathcal{L}_1$ is *more* expressive than $\mathcal{L}_2$, we first show that for every $\mathcal{L}_2$-concept there is an equivalent $\mathcal{L}_1$-concept, and, next, that there exists an $\mathcal{L}_1$-concept that is not equivalent to an $\mathcal{L}_2$-concept as it is not preserved by the characteristic $\mathcal{L}_2$-relations between interpretations. In all cases considered here the latter can be established using small interpretations and simple concepts.

## 4 Main Results

Our results come in two kinds. First, we establish the model-theoretic characterizations announced above for every extension of $\mathcal{FL}^-$ and $\mathcal{AL}$ that can be defined using the constructors from Table 1. Second, we use these characterizations to separate description logics. By way of example we consider $\mathcal{FL}^-$ and define the notion of an $\mathcal{FL}^-$-simulation; after that we indicate how characteristic relations for other logics may be built on top of the one for $\mathcal{FL}^-$ by adding further clauses or conditions.

Given a binary relation $S$ on objects, and sets of objects $X$ and $Y$, we write $XS^{\uparrow}Y$ if $\forall x \in X \exists y \in Y\, Sxy$, and $XS_{\downarrow}Y$ if $\forall y \in Y \exists x \in X\, Sxy$.

**Definition 1** Let $\mathcal{I}$ and $\mathcal{J}$ be two interpretations. An $\mathcal{FL}^-$*-simulation* is a non-empty relation $Z \subseteq \mathcal{P}(\Delta^{\mathcal{I}}) \times \Delta^{\mathcal{J}}$ such that the following hold.

1. If $X_1 Z d_2$ then, for every (atomic) concept name $A$, if $X_1 \subseteq A^{\mathcal{I}}$, then $d_2 \in A^{\mathcal{J}}$.
2. For every role name $R$, if $X_1 (R^{\mathcal{I}})^{\uparrow} Y_1$ and $X_1 Z d_2$, then there exists $e_2 \in \Delta^{\mathcal{J}}$ with $R^{\mathcal{J}} d_2 e_2$.
3. For every role name $R$, if $R^{\mathcal{J}} d_2 e_2$ and $X_1 Z d_2$, then there exists $Y_1 \subseteq \Delta^{\mathcal{I}}$ with $X_1 (R^{\mathcal{I}})_{\downarrow} Y_1$ and $Y_1 Z e_2$.

A first-order formula $\alpha(x)$ is *preserved under* $\mathcal{FL}^-$*-simulations* if for all interpretations $\mathcal{I}$ and $\mathcal{J}$, all sets $X \subseteq \Delta^{\mathcal{I}}$ and objects $d_2 \in \Delta^{\mathcal{J}}$, and all $\mathcal{FL}^-$-simulations $Z$ between $\mathcal{I}$ and $\mathcal{J}$, we have that if $XZd_2$ and for all $d_1 \in X$, $\mathcal{I} \models \alpha[d_1]$, then $\mathcal{J} \models \alpha[d_2]$.

It is easily verified by induction on concepts that all (first-order formulas that are equivalent to) $\mathcal{FL}^-$-concepts are preserved under $\mathcal{FL}^-$-simulations. But the converse of this statement is also true: if a first-order formula is preserved under $\mathcal{FL}^-$-simulations, then it must be (equivalent to) an $\mathcal{FL}^-$-concept. It is in this sense that $\mathcal{FL}^-$-simulations characterize $\mathcal{FL}^-$.

To obtain analogous characterizations for richer logics in the $\mathcal{FL}^-$-hierarchy, the definition of an $\mathcal{FL}^-$-simulation has to be strengthened so as to identify fewer interpretations and thereby ensure preservation of the additional constructors. For each constructor ($\mathcal{C}$, $\mathcal{E}$, $\mathcal{U}$, $\mathcal{N}$, $\mathcal{R}$) one has to amend the existing clauses or add one or more new clauses. Without going into details, the main ideas are the following:

($\mathcal{E}$) To ensure preservation of qualified existential quantifications $\exists R.C$ we only need to add a conjunct 'and $Y_1 Z e_2$' to clause 2 in Definition 1, thus establishing a symmetry between clauses 2 and 3.

($\mathcal{U}$) To characterize logics with disjunction of concepts we have to change the format of our simulation relation by linking objects to objects rather than sets of objects to objects. Put another way, one can arrive at simulations for $\mathcal{FLU}^-$ by requiring, essentially, that the sets $X_1$, $Y_1$ in Definition 1 are singletons.

($\mathcal{C}$) To ensure preservation of negated (atomic) concept names, we change clause 1 in Definition 1 by demanding that both atomic and negated atomic concepts are preserved. For the more general case where we add negations of arbitrary concepts to $\mathcal{FL}^-$, disjunctions become definable, of course, and by the previous item, the format should be changed to a relation linking objects to objects. Moreover, the first clause in Definition 1 should become an equivalence: if $d_1 Z d_2$, then, for every (atomic) concept name $A$, $d_1 \in A^{\mathcal{I}}$ iff $d_2 \in A^{\mathcal{J}}$.

($\mathcal{N}$) Guaranteeing that number restrictions are preserved requires more substantial changes. In the presence of $\mathcal{N}$ our simulation relation becomes a tuple $(Z_0, Z_1, Z_2, \ldots)$, where $Z_0$ is just like the $\mathcal{FL}^-$-simulations defined in Definition 1, and the relations $Z_i$ $(i > 0)$ relate sets of size $i$ to each other. Suitable back-and-forth conditions for the relations $Z_i$ $(i > 0)$ ensure that number restrictions $(\leq i\ R)$ and $(\geq i\ R)$ are preserved.

($\mathcal{R}$) Similarly, characterizing logics with role conjunctions also requires a substantial change to Definition 1. Simulations now become triples $(Z_0, Z_1, Z_2)$ where $Z_0$ is just like an $\mathcal{FL}^-$-simulation, and $Z_1$, $Z_2$ link pairs of (sets of) objects to pairs of objects. Basically, they record the roles satisfied by pairs on either side of the relation.

The above remarks sketch what changes have to be made to Definition 1 if the constructors mentioned are added to $\mathcal{FL}^-$. By combining the various changes, appropriate notions of simulation can be found for richer logics in a modular way.

To summarize, then, using simulation relations based on the above ideas, we can obtain the following collection of characterization results.

**Theorem 2 (Characterization)** *Let $\mathcal{L}$ be a description logic that can be obtained by adding zero or more of the constructors $\mathcal{U}$, $\mathcal{C}$, $\mathcal{E}$, $\mathcal{N}$, or $\mathcal{R}$ to $\mathcal{FL}^-$ or $\mathcal{AL}$. Then there exists a characteristic $\mathcal{L}$-relation on interpretations (similar to the above $\mathcal{FL}^-$-simulations) such that a first-order formula $\alpha(x)$ is preserved under $\mathcal{L}$-relations iff it is (equivalent to) an $\mathcal{L}$-concept.*

Some comments about the proofs of the characterization results summarized in Theorem 2 are in order. The proofs consist of two parts: an easy induction to show that $\mathcal{L}$-concepts are indeed preserved under $\mathcal{L}$-relations, and a non-trivial proof of the converse. The latter follows a strategy that is familiar from a wide range of preservation results in first-order logic. Let $\alpha(x)$ be a first-order formula that is preserved under $\mathcal{L}$-relations; to show that $\alpha(x)$ is (equivalent to) an $\mathcal{L}$-concept, we reason as follows. Let $\mathcal{L}$-$CONS(\alpha)$ denote the collection of $\mathcal{L}$-consequences of $\alpha$: $\{\beta \mid \alpha \models \beta$ and $\beta$ is an $\mathcal{L}$-concept$\}$. Then, by the compactness theorem it suffices to show that $\mathcal{L}$-$CONS(\alpha) \models \alpha(x)$. This is where we use the assumption that $\alpha$ is preserved under $\mathcal{L}$-relations. First, take an interpretation $\mathcal{I}$ with an object $d$ such that $d \in (\mathcal{L}\text{-}CONS(\alpha))^{\mathcal{I}}$; we need to show $d \in (\alpha(x))^{\mathcal{I}}$. Second, we construct an interpretation $\mathcal{J}$ with an object $e \in \Delta^{\mathcal{J}}$ such that $e \in (\alpha(x))^{\mathcal{J}}$ and $e \in (\mathcal{L}\text{-}CONS(\alpha))^{\mathcal{J}}$. The main step in the proof is where we use the latter property to construct elementary extensions $\mathcal{I}^*$ and $\mathcal{J}^*$ of $\mathcal{I}$ and $\mathcal{J}$, respectively, in such a way that there exists an $\mathcal{L}$-relation between $e$, or a set containing $e$ (in $\mathcal{J}^*$), and $d$ (in $\mathcal{I}^*$). This step uses tools from first-order model-theory in an essential way, and it allows us to conclude that $d \in (\alpha(x))^{\mathcal{I}}$, as required.

How can Theorem 2 be used to compare the expressive power of description logics? As announced in Section 3, to separate two description logics $\mathcal{L}_1$ and $\mathcal{L}_2$, we show that there exists an $\mathcal{L}_1$-concept $C$ that is not preserved under $\mathcal{L}_2$-simulations. This is done by presenting two interpretations $\mathcal{I}$ and $\mathcal{J}$ such that there exists an $\mathcal{L}_2$-simulation linking (a set of objects in) $\mathcal{I}$ to (an object in) $\mathcal{J}$, while all objects in the set in $\mathcal{I}$ satisfy $C$, whereas the object in $\mathcal{J}$ does not. Although the task of finding two such interpretations is, in general, an undecidable problem, in practice one can separate description logics using very small interpretations.

By way of example, we will now use Theorem 2 to separate $\mathcal{FL}^-$ from other description logics. We only show this for one language, namely $\mathcal{FLE}^-$, and we do so by showing that the concept $\exists R.A$ (which lives in $\mathcal{FLE}^-$ and its extensions) can not be equivalent to an $\mathcal{FL}^-$-concept, as it is not preserved under $\mathcal{FL}^-$-simulations. Consider the interpretations $\mathcal{I}$ and $\mathcal{J}$ depicted in Figure 1 (dotted lines indicate an $\mathcal{FL}^-$-simulation). We have $\{d_1\} \subseteq (\exists R.A)^{\mathcal{I}}$, but $d_2 \notin (\exists R.A)^{\mathcal{J}}$ even though there is an $\mathcal{FL}^-$-simulation relating $\{d_1\}$ and $d_2$. Hence, by Theorem 2, $\exists R.A$ can not be equivalent to an $\mathcal{FL}^-$-concept. As $\mathcal{FL}^-$ is obviously contained in $\mathcal{FLE}^-$, the latter must be strictly more expressive than $\mathcal{FL}^-$.

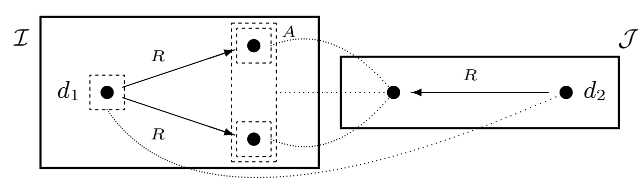


Figure 1: Separating $\mathcal{FL}^-$ and $\mathcal{FLE}^-$

The above separation result is an instance of a more general result which completely classifies the expressive power of all description logics extending $\mathcal{FL}^-$ or $\mathcal{AL}$.

**Theorem 3 (Classification)** *Using the characterizations of Theorem 2 all extensions of $\mathcal{FL}^-$ and $\mathcal{AL}$ obtained by adding the constructors in Table 1 can be completely classified with respect to their expressive power.*

The classification may be presented in a diagram; see Figure 2. Due to space limitations we are only able to depict extensions of $\mathcal{FLE}^-$ instead of the full hierarchy. The diagram should be read as follows. First, the classification is complete in the sense that every extension of $\mathcal{FLE}^-$ coincides with one of the logics shown; if a logic is linked to a logic at a higher level, the former is strictly less expressive than the latter; logics not connected by (sequences of) lines are incomparable with respect to their expressive power.

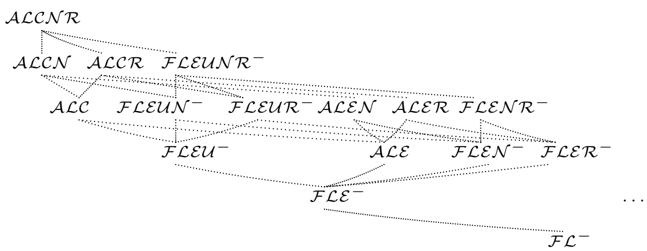


Figure 2: Classifying Description Logics

## 5 Related Work

We see two major lines of work related to this paper, the first one centered around the use of model-theoretic methods similar to the ones we have used, the second one focusing on the expressive power of description logics.

As to the first theme, the technique of Ehrenfeucht-Fraïssé games in first-order logic is closely related to our simulations, and it has been used to obtain numerous separation and preservation results; see [Doets, 1996]. [Immerman and Kozen, 1987] use pebble games to obtain model-theoretic expressivity results about finite variable logics, and related techniques have been used in modal logic as well; for instance, [Kurtonina and de Rijke, 1997] use various kinds of bisimulations to characterize temporal logics with *Since* and *Until*. Also, [Toman and Niwiński, 1997] use similar methods to separate query languages. One of the principle advantages shared by these methods is their explicit and intuitive descriptions of the languages being studied. The results in this paper are different from the above ones, as we are interested in relatively poor languages with limited expressive power and without closure under some of the boolean operators; this special focus necessitates both new notions of simulations and novel techniques for proving the characterization results.

As to the second theme — expressiveness of description logics —, we know of only two earlier references: [Baader, 1996] and [Borgida, 1996]. We will briefly discuss each of these. Baader's work is different from ours in two important ways. First, Baader's definition of expressive power differs from ours. Recall that we we define a logic $\mathcal{L}_1$ to be at least as expressive as a logic $\mathcal{L}_2$ if for every $\mathcal{L}_2$-expression there is an equivalent $\mathcal{L}_1$-expression over the same vocabulary. Intuitively, Baader's definition allows $\mathcal{L}_1$ to use additional concepts and roles in finding $\mathcal{L}_1$-equivalents for every $\mathcal{L}_2$-expression. More formally, let $\Gamma$ be a collection of concepts, and let $\mathrm{Voc}(\Gamma)$ denote the collection of all atomic concepts and roles occurring in $\Gamma$. Further, assume that we have a mapping $f : \mathrm{Voc}(\Gamma_1) \to \mathrm{Voc}(\Gamma_2)$, and interpretations $\mathcal{I}_1$ and $\mathcal{I}_2$ that satisfy all of $\mathcal{I}_1$ and $\mathcal{I}_2$, respectively. Then $f$ embeds $\mathcal{I}_1$ in $\mathcal{I}_2$ if for all $S \in \mathrm{Voc}(\Gamma_1)$ we have $S^{\mathcal{I}_1} = f(S)^{\mathcal{I}_2}$. Then, $\Gamma_2$ *can be expressed* by $\Gamma_1$ if there exists $f : \mathrm{Voc}(\Gamma_2) \to \mathrm{Voc}(\Gamma_1)$ such that

1. every interpretation that validates all of $\Gamma_2$ can be embedded by $f$ in some interpretation that validates all of $\Gamma_1$, and
2. for every interpretation $\mathcal{I}_1$ that validates all of $\Gamma_1$ there exists an interpretation $\mathcal{I}_2$ that validates all of $\Gamma_2$ and that can be embedded in $\mathcal{I}_1$ by $f$.

Then, $\mathcal{L}_1$ is at least as expressive as $\mathcal{L}_2$ (according to Baader) if every collection of $\mathcal{L}_2$-concepts can be expressed by some collection of $\mathcal{L}_1$-concepts.

Clearly, this more involved definition allows one to equate more description logics with respect to their expressive power than ours does; for instance, under Baader's definition negation of atomic concepts can be simulated by number restrictions over additional roles, whereas according to our results negations of atomic concepts can't be expressed using number restrictions (over the same vocabulary).[1] While we agree that it may be useful to be able to use additional concepts and roles in finding equivalent expressions, as Baader himself points out, what is lacking from his definition is a measure on how much additional material one may use and on the complexity of the function that maps $\mathcal{L}_2$-expressions to equivalent $\mathcal{L}_1$-expressions over a possibly richer vocabulary.

A second important difference between Baader's work and ours lies in the results that have been obtained. Baader only establishes a small number of separation results, whereas we provide a *complete* classification of all languages definable using the constructors in Table 1. More importantly, our separation results are based on semantic characterizations; this gives a deeper insight into the properties of logics than mere separation results.

In [Borgida, 1996] the author shows that certain description logics have the same expressive power as the two or three variable fragment of first-order logic (over the same vocabulary). Two remarks are in order. First,

[1] As an aside, the difference between our definition and Baader's is analogous to the difference between definability and projective definability in the area of model-theoretic logics; see [Barwise and Feferman, 1985].

it is well-known that there is a correspondence between some description logics and modal logics (see [Schild, 1991]), and modal logicians have considered the links with finite variable fragments for quite some time (see [Gabbay, 1981]). Thus, Borgida's results could also have been obtained this way. Secondly, the description logics considered in this paper are all expressible in the two variable fragment of first-order logic (possibly with counting), however, it may be shown that none coincides with the full two-variable fragment.

## 6 Concluding Remarks

In this paper we have introduced a series of model-theoretic tools to capture the expressive power of all description logics in the $\mathcal{FL}^-$-hierarchy. We have used these tools to separate description logics, and our main result is a complete classification of the expressive power of all extensions of $\mathcal{FL}^-$ that are definable using the constructors in Table 1.

Future research in this area will concentrate on the following themes. First, as was pointed out above, the proofs for our characterization results use first-order techniques in an essential way. We aim to avoid these techniques, and thus to extend our methods to description logics with non-first-order features (like transitive closure). Second, we want to gain a better understanding of the difference between our approach and that of [Baader, 1996]. In particular, we want to extend our model-theoretic tools in ways that will characterize the expressive power of description logics in Baader's sense. Third, there is an influential line of work in the database literature that characterizes the expressive power of query languages in terms of the complexity of the recognition problem associated with queries expressible in the language at hand; see, for instance, [Abiteboul *et al.*, 1995]. Can this approach be adapted to description logics? And if it can, would it induce the same classification of description logics as ours? Finally, what is the complexity of separating description logics? It is known from the literature on bisimulation that, in general, even the question whether two given interpretations are bisimilar, is undecidable, but for finite interpretations the question becomes decidable. In our case, the question is not just to check bisimilarity, but to determine whether there exists an $\mathcal{L}_1$-concept that is not preserved under $\mathcal{L}_2$-relations. Are there special cases of this question that become decidable?

**Acknowledgment.** Maarten de Rijke was partially supported by the Research and Teaching Innovation Fund at the University of Warwick.

## References

[Abiteboul *et al.*, 1995] S. Abiteboul, R. Hull, and V. Vianu. *Foundations of Databases*. Addison-Wesley, Reading, MA, 1995.

[Baader, 1996] F. Baader. A formal definition for the expressive power of terminological knowledge representation languages. *J. Logic and Computation*, 6:33–54, 1996.

[Barwise and Feferman, 1985] J. Barwise and S. Feferman, editors. *Model-Theoretic Logics*. Springer-Verlag, 1985.

[Borgida, 1996] A. Borgida. On the relative expressiveness of description logics and predicate logics. *Artificial Intelligence*, 82:353–367, 1996.

[Brachman and Levesque, 1984] R. Brachman and H.J. Levesque. The tractability of subsumption in frame-based description languages. In *AAAI-84*, pages 34–37, 1984.

[Doets, 1996] K. Doets. *Basic Model Theory*. CSLI Publications, 1996.

[Donini *et al.*, 1996] F.M. Donini, M. Lenzerini, D. Nardi, and A. Schaerf. Reasoning in description logics. In G. Brewka, editor, *Principles of Knowledge Representation*, pages 191–236. CSLI Publications, Stanford, 1996.

[Donini *et al.*, 1997] F.M. Donini, M. Lenzerini, D. Nardi, and W. Nutt. The complexity of concept languages. *Information and Computation*, 134:1–58, 1997.

[Gabbay, 1981] D.M. Gabbay. Expressive functional completeness in tense logic. In U. Mönnich, editor, *Aspects of Philosophical Logic*, pages 91–117. Reidel, Dordrecht, 1981.

[Immerman and Kozen, 1987] N. Immerman and D. Kozen. Definability with bounded number of bound variables. In *Proc. LICS 1987*, Washington, 1987. Computer Society Press.

[Kurtonina and de Rijke, 1997] N. Kurtonina and M. de Rijke. Simulating without negation. *J. Logic and Computation*, 7:503–524, 1997.

[Schild, 1991] K. Schild. A correspondence theory for terminological logics. In *Proc. 12th IJCAI*, pages 466–471, 1991.

[Toman and Niwiński, 1997] D. Toman and D. Niwiński. First-order queries over temporal databases inexpressible in temporal logic. Manuscript, 1997.