

# On The Complexities Of Horn Description Logics <sup>★</sup>

Markus Krötzsch, Sebastian Rudolph, Pascal Hitzler

*AIFB, Universität Karlsruhe (TH)*

---

## Abstract

Description Logics are currently advancing to become a very prominent paradigm for representing knowledge bases in a variety of application areas. Central to leveraging them for corresponding systems is the provision of a favourable balance between expressivity of the knowledge representation formalism on the one hand, and runtime performance of reasoning algorithms on the other. Due to this, Horn description logics (Horn-DLs) have recently started to attract attention since their (worst-case) data complexities are in general lower than their overall (i.e. combined) complexities, which makes them attractive for reasoning with large sets of instance data (ABoxes). However, the natural question whether Horn-DLs also provide advantages for schema (TBox) reasoning has hardly been addressed so far. In this paper, we therefore provide a thorough and comprehensive analysis of the combined complexities of Horn-DLs. While the combined complexity for many Horn-DLs turns out to be the same as for their non-Horn counterparts, we identify subboolean DLs where Hornness simplifies reasoning. We also provide convenient normal forms for Horn-DLs.

*Key words:* description logics, Horn logic, computational complexity

---

## 1 Introduction

One of the driving motivations behind description logic (DL) research is to design languages which maximise the availability of expressive language features for the knowledge modelling process, while at the same time striving for the most inexpensive languages in terms of computational complexity. A particularly prominent

---

<sup>★</sup> This work was supported by the Deutsche Forschungsgemeinschaft (DFG) under the ReaSem project, and by the EU in the IST project NeOn (IST-2006-027595, <http://www.neon-project.org/>)

*Email addresses:* [mak@aifb.uni-karlsruhe.de](mailto:mak@aifb.uni-karlsruhe.de) (Markus Krötzsch),  
[rudolph@aifb.uni-karlsruhe.de](mailto:rudolph@aifb.uni-karlsruhe.de) (Sebastian Rudolph),  
[hitzler@aifb.uni-karlsruhe.de](mailto:hitzler@aifb.uni-karlsruhe.de) (Pascal Hitzler).

case in point is the DL-based Web Ontology Language OWL,<sup>1</sup> which is a W3C recommended standard since 2004. OWL (more precisely, OWL DL) is indeed among the most expressive known knowledge representation languages which are also decidable.

Of particular interest for practical investigations are obviously tractable DLs [3,6,10]. While not being boolean closed, and thus relatively inexpressive, they recently receive increasing attention as they promise to provide a good trade-off between expressivity and scalability. This is also reflected by the fact that the currently undergoing revision of the OWL standard through the OWL Working Group<sup>2</sup> of the World Wide Web consortium (W3C) strongly considers to adopt one or more tractable DLs as designated important fragments of OWL.

At the same time, Horn-DLs have been introduced [10,15], as their generally lower data complexities make them a natural and efficient choice for reasoning with large numbers of individuals, i.e. for ABox reasoning. As such, they are also under investigation by the OWL Working Group. However, the natural question whether Horn-DLs also provide advantages for TBox reasoning – in terms of combined complexity – has hardly been addressed so far.

In this paper, we therefore provide a thorough and comprehensive analysis of the combined complexities of Horn-DLs. While the combined complexity for many Horn-DLs turns out to be the same as for their non-Horn counterparts – which is no surprise –, we are also able to identify subboolean DLs where the Hornness restriction improves reasoning complexity.

As a secondary contribution of our work, we also present alternative characterisations for many Horn-DLs, including one for Horn-*SHIQ*, which is much simpler than the original definition.

To the best of our knowledge, this paper contains the first systematic investigation of Horn-DLs.

The paper is structured as follows. After recalling some preliminaries on DLs and some preparations concerning normal forms for Horn-DLs (Section 2), we deal in turn with the Horn versions of  $\mathcal{FL}_0$ ,  $\mathcal{FL}^-$  and  $\mathcal{FLE}$  and some of their variants (Sections 3-5). We will see that these provide us with a fairly complete picture of the complexities of Horn-DLs. We close with some related work (Section 6) and a summary with open questions (Section 7).

A brief summary of the results in this paper has been published in [17].

---

<sup>1</sup> <http://www.w3.org/2004/OWL/>

<sup>2</sup> [http://www.w3.org/2007/OWL/wiki/OWL\\_Working\\_Group](http://www.w3.org/2007/OWL/wiki/OWL_Working_Group)

Table 1

Concept constructors in  $\mathcal{SHOIQ}_\circ$ . Semantics refers to an interpretation  $\mathcal{I}$  with domain  $\mathcal{D}$ .

Name	Syntax	Semantics
inverse role	$R^-$	$\{(x, y) \mid (y, x) \in R^{\mathcal{I}}\}$
top	$\top$	$\mathcal{D}$
bottom	$\perp$	$\emptyset$
nominal	$\{i\}$	$\{i^{\mathcal{I}}\}$
negation	$\neg C$	$\mathcal{D} \setminus C^{\mathcal{I}}$
conjunction	$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
disjunction	$C \sqcup D$	$C^{\mathcal{I}} \cup D^{\mathcal{I}}$
univ. restriction	$\forall R.C$	$\{x \in \mathcal{D} \mid (x, y) \in R^{\mathcal{I}} \text{ implies } y \in C^{\mathcal{I}}\}$
exist. restriction	$\exists R.C$	$\{x \in \mathcal{D} \mid \text{for some } y \in \mathcal{D}, (x, y) \in R^{\mathcal{I}} \text{ and } y \in C^{\mathcal{I}}\}$
qualified number	$\leq n R.C$	$\{x \in \mathcal{D} \mid \#\{y \in \mathcal{D} \mid (x, y) \in R^{\mathcal{I}} \text{ and } y \in C^{\mathcal{I}}\} \leq n\}$
restriction	$\geq n R.C$	$\{x \in \mathcal{D} \mid \#\{y \in \mathcal{D} \mid (x, y) \in R^{\mathcal{I}} \text{ and } y \in C^{\mathcal{I}}\} \geq n\}$

## 2 Preparations

We briefly recall some basic definitions of DLs and introduce our notation. We start with the rather expressive description logic  $\mathcal{SHOIQ}_\circ$  and define other DLs as restrictions thereof. We then define Horn-DLs in general, and relate our definition to the known definition of Horn- $\mathcal{SHIQ}$ .

**Definition 2.1** *A knowledge base of the description logic  $\mathcal{SHOIQ}_\circ$  is based on a set  $\mathbf{N}_R$  of role names, a set  $\mathbf{N}_C$  of concept names, and a set  $\mathbf{I}$  of individual names. The set of  $\mathcal{SHOIQ}_\circ$  atomic concepts  $\mathbf{C}$  consists of all concept names and all expressions of the form  $\{i\}$  with  $i \in \mathbf{N}_I$ . The set of  $\mathcal{SHOIQ}_\circ$  (abstract) roles is  $\mathbf{R} = \mathbf{N}_R \cup \{R^- \mid R \in \mathbf{N}_R\}$ , and we set  $\text{Inv}(R) = R^-$  and  $\text{Inv}(R^-) = R$ . In the following, we leave this vocabulary implicit and assume that  $A, B$  are atomic concepts,  $a, b$  are individual names, and  $R, S$  are abstract roles.*

*A  $\mathcal{SHOIQ}_\circ$  knowledge base consists of three finite sets of axioms that are referred to as  $\text{RBox}$ ,  $\text{TBox}$ , and  $\text{ABox}$ . A  $\mathcal{SHOIQ}_\circ$   $\text{RBox}$  may contain axioms of the form  $S \sqsubseteq R$  iff it also contains  $\text{Inv}(S) \sqsubseteq \text{Inv}(R)$ , and axioms of the form  $\text{Trans}(R)$  iff it also contains  $\text{Trans}(\text{Inv}(R))$ . By  $\sqsubseteq^*$  we denote the reflexive-transitive closure of  $\sqsubseteq$ . A role  $R$  is transitive whenever there is a role  $S$  such that  $\text{Trans}(S)$ ,  $R \sqsubseteq^* S$  and  $S \sqsubseteq^* R$ .  $R$  is simple if it has no transitive subroles, i.e., if  $S \sqsubseteq^* R$  implies that  $S$  is not transitive. Roles that are not simple are also called complex. Moreover, an  $\text{RBox}$  can contain axioms of the form  $S_1 \circ \dots \circ S_n \sqsubseteq R$ .*

A  $\mathcal{SHOIQ}_\circ$  TBox consists of axioms of the form  $C \sqsubseteq D$ , where  $C$  and  $D$  are concept expressions constructed from concept names, role names, and individual names by the operators shown in Table 1. A  $\mathcal{SHOIQ}_\circ$  ABox consists of axioms of the form  $A(a)$ ,  $R(a, b)$ , and  $a \approx b$ .

The above definition is fairly standard, except that we restrict ABox concept statements to atomic concepts. Our ABoxes thus are *extensionally reduced*, but it is known that this does not restrict the expressivity of the logic since complex ABox statements can easily be moved into the TBox by introducing auxiliary concept names. Moreover, we do not explicitly consider concept/role equivalence  $\equiv$ , since it can be modelled via mutual concept/role inclusions.  $\mathcal{SHOIQ}_\circ$  is very closely related to the emerging revision of the OWL DL standard known as OWL 1.1,<sup>3</sup> which is based on  $\mathcal{SROIQ}$  [14].

We adhere to the common model-theoretic semantics for  $\mathcal{SHOIQ}_\circ$  with general concept inclusion axioms: an interpretation  $\mathcal{I}$  consists of a set  $\mathcal{D}$  called *domain* together with a function  $\cdot^{\mathcal{I}}$  mapping

- individual names to elements of  $\mathcal{D}$ ,
- class names to subsets of  $\mathcal{D}$ , and
- role names to subsets of  $\mathcal{D} \times \mathcal{D}$ .

This function is inductively extended to roles and concept descriptions as shown in Table 1. An interpretation  $\mathcal{I}$  *satisfies* an axiom  $F$ , written  $\mathcal{I} \models F$ , if one of the following conditions hold:

- $\mathcal{I} \models S \sqsubseteq R$  if  $S^{\mathcal{I}} \subseteq R^{\mathcal{I}}$
- $\mathcal{I} \models S_1 \circ \dots \circ S_n \sqsubseteq R$  if  $S_1^{\mathcal{I}} \circ \dots \circ S_n^{\mathcal{I}} \subseteq R^{\mathcal{I}}$  (where  $\circ$  is the relational product)
- $\mathcal{I} \models \text{Trans}(S)$  if  $S^{\mathcal{I}}$  is a transitive relation
- $\mathcal{I} \models C \sqsubseteq D$  if  $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$
- $\mathcal{I} \models A(a)$  if  $a^{\mathcal{I}} \in A^{\mathcal{I}}$
- $\mathcal{I} \models R(a, b)$  if  $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}}$
- $\mathcal{I} \models a \approx b$  if  $a^{\mathcal{I}} = b^{\mathcal{I}}$

We will be specifically interested in (variants of) the following subboolean fragments of  $\mathcal{SHOIQ}_\circ$ . Those definitions and naming conventions can also be found in [2].

**Definition 2.2** *Restricting the syntax of  $\mathcal{SHOIQ}_\circ$ , we define the following description logics:*

- $\mathcal{FL}\mathcal{E}$  is the fragment of  $\mathcal{SHOIQ}_\circ$  using only the constructors  $\top$ ,  $\perp$ ,  $\sqcap$ ,  $\exists$ , and  $\forall$ .
- $\mathcal{FL}^-$  is the fragment of  $\mathcal{FL}\mathcal{E}$  for which all existential role restrictions have the form  $\exists R.\top$ .

<sup>3</sup> <http://www.w3.org/TR/owl11-syntax>

- $\mathcal{FL}_0$  is the fragment of  $\mathcal{FL}^-$  that does not contain existential role restrictions.

In the presence of GCIs, all of those logics are known to have a combined complexity that is  $\text{ExpTime}$ -complete. To prevent this effect in our below investigation of their Horn-fragments, we impose suitable restrictions that ensure that the syntactically forbidden constructors do not sneak in through the back door.

## 2.1 Horn-DLs

Now we define the class of Horn-DLs as we consider it in this paper. Before we do this, however, let us briefly discuss some of the conceptual difficulties underlying such a definition.

Originally, the property of being Horn is defined by a syntactic restriction on first-order predicate logic. Although DL knowledge bases can be translated into semantically equivalent first-order theories (with some extensions such as equality and concrete datatypes), they are not *per se* such theories. Starting from the original definition of Hornness, the definition of Horn-DLs thus rests on the question how the translation into first-order predicate logic is performed.

Even when allowing only *naive* transformations, which have been used for defining Description Logic Programs (DLPs) [10,26], it remains unclear which transformations are *naive*. To illustrate this, consider the DL axiom  $A \sqcup B \sqsubseteq C$ . Translations to first-order predicate logic could result e.g. in the single statement  $(\forall x)(A(x) \vee B(x) \rightarrow C(x))$ , which is not Horn, or to the theory consisting of the two statements  $(\forall x)(A(x) \rightarrow C(x))$  and  $(\forall x)(B(x) \rightarrow C(x))$ , which is Horn. The original papers describing DLP indeed remain somewhat ambiguous as to the exact definition, but this does not concern us here.<sup>4</sup>

One could also attempt to define Horn-DLs by means of semantic properties, e.g. by taking the existence of a least model as characterising property. Such an approach, though, remains to be worked out, and it is not clear at this stage whether it would result in a reasonable definition of Horn-DLs.

In this paper, we follow an approach laid out in [15,20] for  $\mathcal{SHIQ}$  – resulting in the DL known as Horn- $\mathcal{SHIQ}$  –, which is more sophisticated than the DLP approach mentioned above. Indeed the transformation to first-order predicate logic in this case is of  $\text{ExpTime}$  worst-case complexity. However, since it suffices to perform this transformation on the TBox, Data complexity remains that for Horn clauses, which is polynomial in time.

We do this by first defining Horn- $\mathcal{SHOIQ}_o$ , and then identifying suitable (syn-

<sup>4</sup> A proposal how to rectify this is contained in [11,12].

Table 2

A grammar for defining Horn- $\mathit{SHOIQ}_\circ$ .  $\mathbf{A}$ ,  $\mathbf{R}$ , and  $\mathbf{S}$  denote the sets of all atomic concepts, abstract roles, and simple role names, respectively. The presentation is slightly simplified by exploiting associativity and commutativity of  $\sqcap$  and  $\sqcup$ , and by omitting  $\geq 1 R.C$  if  $\exists R.C$  is present. The grammar for Horn- $\mathit{SHIQ}$  is indeed identical: the difference lies in allowing nominal expressions of the form  $\{i\}$ ,  $i \in \mathbf{N}_I$  as atomic concepts, and in allowing RBox expressions of the form  $S_1 \circ \dots \circ S_n \sqsubseteq R$ .

$$\mathbf{C}_1^+ ::= \top \mid \perp \mid \neg \mathbf{C}_1^- \mid \mathbf{C}_1^+ \sqcap \mathbf{C}_1^+ \mid \mathbf{C}_0^+ \sqcup \mathbf{C}_1^+ \mid \exists \mathbf{R}.\mathbf{C}_1^+ \mid \forall \mathbf{S}.\mathbf{C}_1^+ \mid \forall \mathbf{R}.\mathbf{C}_0^+ \mid \geq n \mathbf{R}.\mathbf{C}_1^+ \mid \leq 1 \mathbf{R}.\mathbf{C}_0^+ \mid \mathbf{A}$$

$$\mathbf{C}_1^- ::= \top \mid \perp \mid \neg \mathbf{C}_1^+ \mid \mathbf{C}_0^- \sqcap \mathbf{C}_1^- \mid \mathbf{C}_1^- \sqcup \mathbf{C}_1^- \mid \exists \mathbf{S}.\mathbf{C}_1^- \mid \exists \mathbf{R}.\mathbf{C}_0^- \mid \forall \mathbf{R}.\mathbf{C}_1^- \mid \geq 2 \mathbf{R}.\mathbf{C}_0^- \mid \leq n \mathbf{R}.\mathbf{C}_1^+ \mid \mathbf{A}$$

$$\mathbf{C}_0^+ ::= \top \mid \perp \mid \neg \mathbf{C}_0^- \mid \mathbf{C}_0^+ \sqcap \mathbf{C}_0^+ \mid \mathbf{C}_0^+ \sqcup \mathbf{C}_0^+ \mid \forall \mathbf{R}.\mathbf{C}_0^+$$

$$\mathbf{C}_0^- ::= \top \mid \perp \mid \neg \mathbf{C}_0^+ \mid \mathbf{C}_0^- \sqcap \mathbf{C}_0^- \mid \mathbf{C}_0^- \sqcup \mathbf{C}_0^- \mid \exists \mathbf{R}.\mathbf{C}_0^- \mid \mathbf{A}$$

tactic) fragments of it. We will show formally that the fragment corresponding to Horn- $\mathit{SHIQ}$  corresponds to the original definition from [15,20].

**Definition 2.3** *The description logic Horn- $\mathit{SHOIQ}_\circ$  is defined as  $\mathit{SHOIQ}_\circ$  except that the only allowed concept inclusions are of the form  $\mathbf{C}_0^- \sqsubseteq \mathbf{C}_1^+$  or  $\mathbf{C}_1^- \sqsubseteq \mathbf{C}_0^+$  according to the grammar in Table 2.*

## 2.2 Normal Form Transformation

To facilitate further considerations and proofs, we now show that any Horn- $\mathit{SHOIQ}_\circ$  knowledge base can be transformed into an equisatisfiable Horn- $\mathit{SHOIQ}_\circ$  knowledge base without negations and disjunctions.

As a first facilitation, note that any GCI  $C \sqsubseteq D$  with  $C \in \mathbf{C}_1^-$  and  $D \in \mathbf{C}_0^+$  is equivalent to the GCI  $\neg D \sqsubseteq \neg C$ . Since  $\neg D \in \mathbf{C}_1^+$  and  $\neg C \in \mathbf{C}_0^-$  we will in the following assume any GCI to be of the form  $\mathbf{C}_0^- \sqsubseteq \mathbf{C}_1^+$ . For a given concept description, we recursively define the negation normal form (NNF) as usual by:

$\text{NNF}(C)$	$:= C$ for all $C \in \{\top, \perp, A, \neg A\}$
$\text{NNF}(\neg\top)$	$:= \perp$
$\text{NNF}(\neg\perp)$	$:= \top$
$\text{NNF}(\neg\neg C)$	$:= \text{NNF}(C)$
$\text{NNF}(C \sqcap D)$	$:= \text{NNF}(C) \sqcap \text{NNF}(D)$
$\text{NNF}(\neg(C \sqcap D))$	$:= \text{NNF}(\neg C) \sqcup \text{NNF}(\neg D)$
$\text{NNF}(C \sqcup D)$	$:= \text{NNF}(C) \sqcup \text{NNF}(D)$
$\text{NNF}(\neg(C \sqcup D))$	$:= \text{NNF}(\neg C) \sqcap \text{NNF}(\neg D)$
$\text{NNF}(\forall R.C)$	$:= \forall R.\text{NNF}(C)$
$\text{NNF}(\neg\forall R.C)$	$:= \exists R.\text{NNF}(\neg C)$
$\text{NNF}(\exists R.C)$	$:= \exists R.\text{NNF}(C)$
$\text{NNF}(\neg\exists R.C)$	$:= \forall R.\text{NNF}(\neg C)$
$\text{NNF}(\leq nR.C)$	$:= \leq nR.\text{NNF}(C)$
$\text{NNF}(\neg \leq nR.C)$	$:= \geq (n + 1)R.\text{NNF}(C)$
$\text{NNF}(\geq nR.C)$	$:= \geq nR.\text{NNF}(C)$
$\text{NNF}(\neg \geq nR.C)$	$:= \leq (n - 1)R.\text{NNF}(C)$

Obviously, calculating the negation normal form of a concept description does not change its semantics (see, e.g., [20]). As an auxiliary lemma, we will show that converting a concept expression to its NNF does not change its grammar type due to Table 2.

**Lemma 2.4** *Let  $C \in \mathbf{D}$  be a concept description with  $\mathbf{D} \in \{\mathbf{C}_1^+, \mathbf{C}_1^-, \mathbf{C}_0^+, \mathbf{C}_0^-\}$ . Then  $\text{NNF}(C) \in \mathbf{D}$  as well.*

**Proof:** The proof can be done by induction over the formula depth. Note that for every  $\mathbf{D}$ , we just have to check the case  $\neg\mathbf{D}'$  since in the other cases the proposition follows directly from the induction hypothesis. Moreover one can skip the cases where double negation occurs, since it can be just eliminated directly (and in the presented grammar, for any  $\mathbf{D}$ ,  $\neg\neg C \in \mathbf{D}$  implies  $C \in \mathbf{D}$ ).

The base cases (i.e.,  $C \in \top, \perp, A, \neg A$ ) are clear since NNF does not change them at all. From the remaining cases, we will just exemplarily give one, the others can be done in an analogue way.

Thus consider  $\mathbf{D} = \mathbf{C}_1^+$  and  $C = \neg(D \sqcap E)$  with  $D \in \mathbf{C}_0^-$  and  $E \in \mathbf{C}_1^-$ . This directly implies  $\neg D \in \mathbf{C}_0^+$  and  $\neg E \in \mathbf{C}_1^+$ . Due to the induction hypothesis, we then also

Table 3

Reduced grammar for defining Horn- $\mathcal{SHOIQ}_o$  via the NNF.

$$\mathbf{C}_1^+ ::= \top \mid \perp \mid \mathbf{C}_1^+ \sqcap \mathbf{C}_1^+ \mid \mathbf{C}_0^+ \sqcup \mathbf{C}_1^+ \mid \exists \mathbf{R}.\mathbf{C}_1^+ \mid \forall \mathbf{S}.\mathbf{C}_1^+ \mid \forall \mathbf{R}.\mathbf{C}_0^+ \mid \geq n \mathbf{R}.\mathbf{C}_1^+ \mid \leq 1 \mathbf{R}.\mathbf{C}_0^- \mid \mathbf{A} \mid \neg \mathbf{A}$$

$$\mathbf{C}_0^+ ::= \top \mid \perp \mid \mathbf{C}_0^+ \sqcap \mathbf{C}_0^+ \mid \mathbf{C}_0^+ \sqcup \mathbf{C}_0^+ \mid \forall \mathbf{R}.\mathbf{C}_0^+ \mid \neg \mathbf{A}$$

$$\mathbf{C}_0^- ::= \top \mid \perp \mid \mathbf{C}_0^- \sqcap \mathbf{C}_0^- \mid \mathbf{C}_0^- \sqcup \mathbf{C}_0^- \mid \exists \mathbf{R}.\mathbf{C}_0^- \mid \mathbf{A}$$

have  $\text{NNF}(\neg D) \in \mathbf{C}_0^+$  and  $\text{NNF}(\neg E) \in \mathbf{C}_1^+$ . Hence,  $\text{NNF}(\neg(D \sqcap E)) = \text{NNF}(\neg D) \sqcup \text{NNF}(\neg E) \in \mathbf{C}_1^+$  as a look to the grammar immediately shows.  $\square$

Note that any concept expression of any DL allowing arbitrary negation can be transformed into NNF, while for DLs not allowing negation (or only on the atomic concept level) any concept expression trivially is already in negation normal form. Hence we will without loss of generality assume that all concept expressions we deal with are in NNF.

This directly reduces the grammar from Table 2 to the one presented in Table 3. The assumptions underlying this reduction have an important effect on subsequent syntactic restrictions. On the one hand, the transformation to negation normal form may introduce different logical operators. On the other hand, the aforementioned transformation from  $\mathbf{C}_1^- \sqsubseteq \mathbf{C}_0^+$  to  $\mathbf{C}_0^- \sqsubseteq \mathbf{C}_1^+$  may also have this effect. For example, the  $\mathcal{FL}_0$  axiom  $\forall \mathbf{R}.C \sqsubseteq \forall \mathbf{S}.\perp$  is of the form  $\mathbf{C}_1^- \sqsubseteq \mathbf{C}_0^+$ . Its equivalent form  $\exists \mathbf{S}.\top \sqsubseteq \exists \mathbf{R}.C$  in turn can be stated only in (Horn-) $\mathcal{FL}\mathcal{E}$ . This effect is due to the presence of GCIs, and is also the reason why the distinction of  $\mathcal{FL}_0$ ,  $\mathcal{FL}^-$ , and  $\mathcal{FL}\mathcal{E}$  is not of interest in this general case [2]. Since it is our goal to identify description logic fragments that are sufficiently restricted to have smaller worst-case complexities, we prevent the above effect by restricting to  $\mathcal{FL}_0$  and  $\mathcal{FL}^-$  axioms to the normal form of Table 3.

**Definition 2.5** *A knowledge base is in Horn- $\mathcal{FL}_0$  (Horn- $\mathcal{FL}^-$ , Horn- $\mathcal{FL}\mathcal{E}$ ) whenever all its TBox axioms  $F$  satisfy the following requirements:*

- $F$  is of the form  $\mathbf{C}_0^- \sqsubseteq \mathbf{C}_1^+$  of Table 3, and
- $F$  is in  $\mathcal{FL}_0$  ( $\mathcal{FL}^-$ ,  $\mathcal{FL}\mathcal{E}$ ).

For defining the Horn fragments of all DLs that are Boolean closed, one can as well consider axioms of all forms given in Table 2 – as for the case of Horn- $\mathcal{SHIQ}$ . While the above negation normal form thus is a true restriction for some description logics, we will now show that one can safely extend all of the Horn fragments we consider with negations and (some of the) disjunctions allowed in Table 3.

**Definition 2.6** *A Horn- $\mathcal{SHOIQ}_o$  knowledge base is in normal form if it contains only axioms of the forms shown in Table 4.*

Table 4

Normal form for Horn- $\mathcal{SHOIQ}_\circ$ .  $A$ ,  $B$ , and  $C$  are names of atomic concepts,  $R$ ,  $S$ , and  $T$  (possibly inverse) role names, and  $c$  and  $d$  individual names.

$\top \sqsubseteq A$	$A \sqcap B \sqsubseteq C$	$\exists R.A \sqsubseteq B$	$A(c)$	$R \sqsubseteq T$
$A \sqsubseteq \perp$	$A \sqsubseteq \exists R.B$	$A \sqsubseteq \forall S.B$	$R(c, d)$	$R \circ S \sqsubseteq T$
$A \sqsubseteq \geq n R.A$	$A \sqsubseteq \leq 1 R.A$	$A \sqsubseteq C$	$c \approx d$	

Table 5

Normal form transformation for Horn- $\mathcal{SHOIQ}_\circ$ .  $A$ ,  $B$ ,  $C$ , possibly endowed with  $\dot{D}$  or  $\hat{D}$ , are concept expressions, where symbols with  $\cdot$  must be concept names or nominals while symbols with  $\hat{\cdot}$  are neither concept names nor nominals.  $\dot{D}$  is a fresh concept name.  $R_i$ ,  $S$ , and  $U$  are (possibly inverse) role names, where  $U$  is fresh.

P1:	$R_1 \circ \dots \circ R_{n-1} \circ R_n \sqsubseteq S$	$\mapsto \{R_1 \circ \dots \circ R_{n-1} \sqsubseteq U, U \circ R_n \sqsubseteq S\}$
	$\hat{A} \sqsubseteq \hat{C}$	$\mapsto \{\hat{A} \sqsubseteq \dot{D}, \dot{D} \sqsubseteq \hat{C}\}$
	$\hat{A} \sqcap B \sqsubseteq C$	$\mapsto \{\hat{A} \sqsubseteq \dot{D}, \dot{D} \sqcap B \sqsubseteq C\}$
	$B \sqcap \hat{A} \sqsubseteq C$	$\mapsto \{\hat{A} \sqsubseteq \dot{D}, \dot{D} \sqcap B \sqsubseteq C\}$
	$\dot{A} \sqsubseteq B \sqcap C$	$\mapsto \{\dot{A} \sqsubseteq B, \dot{A} \sqsubseteq C\}$
	$\hat{A} \sqcup B \sqsubseteq \dot{C}$	$\mapsto \{\hat{A} \sqsubseteq \dot{C}, B \sqsubseteq \dot{C}\}$
	$B \sqcup \hat{A} \sqsubseteq \dot{C}$	$\mapsto \{\hat{A} \sqsubseteq \dot{C}, B \sqsubseteq \dot{C}\}$
	$A \sqsubseteq \hat{B} \sqcup \hat{C}$	$\mapsto \begin{cases} \{\dot{D} \sqsubseteq \hat{B}, \dot{A} \sqsubseteq \dot{D} \sqcup \hat{C}\} & \text{if } B \in \mathbf{C}_1^+ \\ \{\dot{D} \sqsubseteq \hat{C}, \dot{A} \sqsubseteq \dot{D} \sqcup \hat{B}\} & \text{otherwise} \end{cases}$
	$\dot{A} \sqsubseteq B \sqcup \dot{C}$	$\mapsto \{\text{NNF}(\neg B) \sqsubseteq \dot{D}, \dot{A} \sqcap \dot{D} \sqsubseteq \dot{C}\}$
	$\dot{A} \sqsubseteq \dot{C} \sqcup B$	$\mapsto \{\text{NNF}(\neg B) \sqsubseteq \dot{D}, \dot{A} \sqcap \dot{D} \sqsubseteq \dot{C}\}$
	$\exists R.\hat{A} \sqsubseteq \hat{B}$	$\mapsto \{\hat{A} \sqsubseteq \dot{D}, \exists R.\dot{D} \sqsubseteq \hat{B}\}$
	$\dot{A} \sqsubseteq \exists R.\hat{C}$	$\mapsto \{\dot{A} \sqsubseteq \exists R.\dot{D}, \dot{D} \sqsubseteq \hat{C}\}$
	$\dot{A} \sqsubseteq \forall R.\hat{C}$	$\mapsto \{\dot{A} \sqsubseteq \forall R.\dot{D}, \dot{D} \sqsubseteq \hat{C}\}$
	$\dot{A} \sqsubseteq \geq n R.\hat{B}$	$\mapsto \{\dot{A} \sqsubseteq \geq n R.\dot{D}, \dot{D} \sqsubseteq \hat{B}\}$
	$\dot{A} \sqsubseteq \leq 1 R.\hat{B}$	$\mapsto \{\dot{A} \sqsubseteq \leq 1 R.\dot{D}, \hat{B} \sqsubseteq \dot{D}\}$
	$\perp \sqsubseteq C$	$\mapsto \emptyset$
	$A \sqsubseteq \top$	$\mapsto \emptyset$
P2:	$\dot{A} \sqsubseteq \neg \hat{B}$	$\mapsto \{\dot{A} \sqcap \hat{B} \sqsubseteq \dot{D}, \dot{D} \sqsubseteq \perp\}$

The following shows that we can restrict to knowledge bases in normal form for checking satisfiability.

**Theorem 2.7** *Checking satisfiability of a Horn- $\mathcal{SHOIQ}_\circ$  knowledge base can be reduced in linear time to checking satisfiability of a Horn- $\mathcal{SHOIQ}_\circ$  knowledge*

base that is in normal form.

**Proof:** Consider the transformation rules in Table 5, where each rule replaces one axiom by a set of derived axioms. A transformation algorithm is given by always exhaustively applying all P1 rules and finally exhaustively applying the P2 rule. We have to show the following propositions:

- The algorithm terminates after at most a linear number of steps.
- The result of this transformation is a knowledge base in normal form.
- The algorithm preserves satisfiability.

First note that every transformation rule – if applied to Horn- $\mathcal{SHOIQ}_\circ$  axioms – yields only axioms which are again in Horn- $\mathcal{SHOIQ}_\circ$ .

We now show that the algorithm terminates after finally many steps. Regarding the P1 rules, we assign to a knowledge base  $KB$  a natural number

$$\sigma(KB) = \sum_{G \sqsubseteq H \in KB} \sigma(G \sqsubseteq H)$$

where

$$\sigma(G \sqsubseteq H) := \begin{cases} 0 & \text{if } G \sqsubseteq H \text{ matches one of the axiom forms from Table 4} \\ \max(0, 2 \cdot (\text{junctors}(G) + \text{junctors}(H)) - 1) & \text{otherwise,} \end{cases}$$

where  $\text{junctors}$  sums up all occurrences of the symbols  $\circ, \exists, \forall, \geq n, \leq n, \sqcap, \sqcup, \top, \perp$  in a concept or role expression (note that  $\neg$  is excluded here).

Obviously,  $\sigma(KB)$  is linearly bounded by the size of  $KB$ . Moreover, note that every single application of a P1 rule to a knowledge base decreases  $\sigma(KB)$  by at least one. Hence the exhaustive application of P1 rules also terminates after linearly many steps.

Finally, the P2 rule can be applied at most once to every axiom, producing only axioms in normal form, which is also clearly linear.

That the resulting knowledge base is in normal form can be easily seen: for any axiom being not in normal form, one of the transformation rules applies (note that we can presume that everything is negation normalized and in Horn- $\mathcal{SHOIQ}_\circ$ ). Termination of the process has been shown above, so the only possible axioms left must be in normal form.

That the algorithm preserves satisfiability follows from the fact, that any of the transformation steps does so. Hence one has to show for every transformation rule that applying it to an according axiom in a Horn knowledge base  $KB$  one obtains an equisatisfiable knowledge base  $KB'$ . We will show a stronger proposition, namely

that for any model  $\mathcal{I}$  of KB we find a model  $\mathcal{I}'$  of KB' where  $\cdot^{\mathcal{I}'}$  coincides with  $\cdot^{\mathcal{I}}$  on the original sets of concept and role names – and vice versa: any model  $\mathcal{I}'$  of KB' gives rise to a model  $\mathcal{I}$  of KB with this property. The line of argumentation herein is quite straightforward: on one hand one provides a canonical extension from  $\mathcal{I}$  to  $\mathcal{I}'$  by letting the newly introduced concept have the same extension as the complex concept it substitutes. On the other hand one shows that in any model  $\mathcal{I}'$  the axiom removed by the transformation rule is satisfied and hence  $\mathcal{I}'$  can also serve as a model of KB.  $\square$

Clearly, the above transformation algorithm does not affect the containment of a set of axioms in a syntactic DL fragment, as long as the negation normal form transformations  $\text{NNF}(\neg B)$  and  $\text{NNF}(\neg C)$  do not introduce axioms that are outside the given fragment. The structure of the concepts  $B$  and  $C$  above is in turn only depending on  $\mathbf{C}_0^+$ , and we can easily identify the following admissible extensions of subboolean Horn logics.

**Corollary 2.8** *Consider the following alternative definitions of  $\mathbf{C}_0^+$  in Table 3:*

- (a)  $\mathbf{C}_0^{+'} := \top \mid \perp \mid \mathbf{C}_0^{+'} \sqcap \mathbf{C}_0^{+'} \mid \mathbf{C}_0^{+'} \sqcup \mathbf{C}_0^{+'} \mid \neg \mathbf{A}$
- (b)  $\mathbf{C}_0^{+''} := \top \mid \perp \mid \mathbf{C}_0^{+''} \sqcap \mathbf{C}_0^{+''} \mid \mathbf{C}_0^{+''} \sqcup \mathbf{C}_0^{+''} \mid \forall \mathbf{R}.\perp \mid \neg \mathbf{A}$

Moreover, let  $\mathbf{C}_1^{+'}$  and  $\mathbf{C}_1^{+''}$  denote the rules obtained by replacing  $\mathbf{C}_0^+ \sqcup \mathbf{C}_1^+$  in the definition of  $\mathbf{C}_1^+$  by  $\mathbf{C}_0^{+'} \sqcup \mathbf{C}_1^+$  and  $\mathbf{C}_0^{+''} \sqcup \mathbf{C}_1^+$ , respectively.

*Checking satisfiability of a knowledge base that consists of  $\mathcal{FL}_0$  ( $\mathcal{FL}^-$ ) axioms of the form  $\mathbf{C}_0^- \sqsubseteq \mathbf{C}_1^{+'}$  ( $\mathbf{C}_0^- \sqsubseteq \mathbf{C}_1^{+''}$ ) can be reduced in linear time to checking satisfiability of a Horn- $\mathcal{FL}_0$  (Horn- $\mathcal{FL}^-$ ) knowledge base that is in the reduced normal form of Table 4.*

Knowing that they can be reduced to the standard notions, we will not consider extensions of the above form in the rest of this paper. Similar restricted forms of disjunction and atomic negation are admissible in many Horn-fragments. For example, note that also the description logic  $\mathcal{EL}^{++}$  [1] can be extended with Horn atomic negations and some forms of Horn disjunctions (arbitrary disjunction in  $\mathbf{C}_0^-$  and disjunction with quantifier-free  $\mathbf{C}_0^+$  as part of  $\mathbf{C}_1^+$ ), thus obtaining an even more expressive tractable description logic.

The principles underlying the above reduction of  $\sqcup$  are easily seen to be closely related to *Lloyd-Topor transformations* that are well-known in (Horn) logic programming. Reductions of atomic negations are less common, since many logic programming paradigms do not support  $\perp$  and classical negations.

Table 6

Positions in a concept (left) and their polarity (right).

$C _{\epsilon}$	$= C$	$\text{pol}(C, \epsilon)$	$= 1$
$(\neg C) _{1p}$	$= C _p$	$\text{pol}(\neg C, 1p)$	$= -\text{pol}(C, p)$
$(C_1 \circ C_2) _{ip}$	$= C_i _p$	$\text{pol}(C_1 \circ C_2, ip) = \text{pol}(C_i, p)$ for $\circ \in \{\sqcap, \sqcup\}$ and $i \in \{1, 2\}$	
$\diamond R.C _{2p}$	$= C _p$	$\text{pol}(\diamond R.C, 2p) = \text{pol}(C, p)$ for $\diamond \in \{\forall, \exists\}$	
$\leq n R.C _{3p}$	$= C _p$	$\text{pol}(\leq n R.C, 3p) = -\text{pol}(C, p)$	
$\geq n R.C _{3p}$	$= C _p$	$\text{pol}(\geq n R.C, 3p) = \text{pol}(C, p)$	

### 2.3 Horn-SHIQ

We next show that Horn-SHIQ as defined in [15,20] is indeed a fragment of Horn-SHOIQ<sub>o</sub>. The proof is somewhat involved since the original definition of Horn-SHIQ is in a very different and less accessible format.

We recall the definition of Horn-SHIQ as given in [15], which requires us to introduce a number of auxiliary concepts first. Subconcepts of some description logic concept are denoted by specifying their *position*. Formally, a position  $p$  is a finite sequence of integers, where  $\epsilon$  denotes the empty position. Given a concept  $D$ ,  $D|_p$  denotes the *subconcept* of  $D$  at position  $p$ , defined recursively as in Table 6 (left). In this paper, we consider only positions that are defined according to this table, and the set of *all positions in a concept*  $D$  is understood accordingly. Given a concept  $D$  and a position  $p$  in  $D$ , the *polarity*  $\text{pol}(D, p)$  of  $D$  at position  $p$  is defined as in Table 6 (right). Using this notation, we can state the following definition of Horn knowledge bases.

**Definition 2.9 ([15, Definition 1])** Let  $\text{pl}^+$  and  $\text{pl}^-$  denote mutually recursive functions that map a SHIQ concept  $D$  to a non-negative integer as specified in Table 7, where  $\text{sgn}(0) = 0$  and  $\text{sgn}(n) = 1$  for  $n > 0$ . We define a function  $\text{pl}$  that assigns to each SHIQ-concept  $D$  and position  $p$  in  $D$  a non-negative integer by setting:

$$\text{pl}(D, p) = \begin{cases} \text{pl}^+(D|_p) & \text{if } \text{pol}(D, p) = 1, \\ \text{pl}^-(D|_p) & \text{if } \text{pol}(D, p) = -1, \end{cases}$$

A concept  $C$  is Horn if  $\text{pl}(C, p) \leq 1$  for every position  $p$  in  $C$  (including the empty position  $\epsilon$ ). An extensionally reduced<sup>5</sup>  $\mathcal{ALCHIQ}$  knowledge base  $KB$  is Horn if  $\neg C \sqcup D$  is Horn for each axiom  $C \sqsubseteq D$  of  $KB$ .

While suitable as a criterium for *checking* the Hornness of single axioms or knowl-

<sup>5</sup> A knowledge base is *extensionally reduced* if its ABox contains only literal concepts. This can always be achieved by introducing new names for complex concept terms.

Table 7  
Definition of  $\text{pl}^+(D)$  and  $\text{pl}^-(D)$ .

$D$	$\text{pl}^+(D)$	$\text{pl}^-(D)$
$\perp$	0	0
$\top$	0	0
$A$	1	0
$\neg C$	$\text{pl}^-(C)$	$\text{pl}^+(C)$
$\sqcap C_i$	$\max_i \text{sgn}(\text{pl}^+(C_i))$	$\sum_i \text{sgn}(\text{pl}^-(C_i))$
$\sqcup C_i$	$\sum_i \text{sgn}(\text{pl}^+(C_i))$	$\max_i \text{sgn}(\text{pl}^-(C_i))$
$\exists R.C$	1	$\text{sgn}(\text{pl}^-(C))$
$\forall R.C$	$\text{sgn}(\text{pl}^+(C))$	1
$\geq n R.C$	1	$\frac{n(n-1)}{2} + n \text{sgn}(\text{pl}^-(C))$
$\leq n R.C$	$\frac{n(n+1)}{2} + (n+1) \text{sgn}(\text{pl}^-(C))$	1

edge bases, this definition is not particularly suggestive as a description of the class of Horn knowledge bases as a whole. Indeed, it is not readily seen for which formulae  $\text{pl}$  yields values smaller or equal to 1 for all possible positions in the formula. On the other hand, Definition 2.9 is still overly detailed as  $\text{pl}$  calculates the *exact* number of positive literals being introduced when transforming some (sub)formula.

In order to derive a more convenient characterization, observe that, since we require the value of  $\text{pl}$  to be smaller or equal to 1 at *all* positions of the concept, there cannot be any sub-concepts of higher values, even though the value of the subconcept is not decisive for some cases of the calculation of  $\text{pl}$  (e.g. for  $\text{pl}^+(\exists R.C)$ ). Thus we can generally restrict to concepts with a  $\text{pl}$ -value  $\leq 1$ . To do so, one has to consider four different classes of concepts, having a  $\text{pl}$ -value either =0 or  $\leq 1$  when occurring either at a positive or negative position in a formula. Appropriate classes of  $\mathcal{ALCHI}Q$  concepts are then defined in Table 2.

Intuitively, the classes  $\mathbf{C}_0^+$  and  $\mathbf{C}_1^+$  define exactly those concepts for which the value of  $\text{pl}$  is smaller or equal to 0 and 1, respectively. In particular,  $\mathbf{C}_1^+$  denotes the class of all Horn concepts. Let us now show this formally.

**Lemma 2.10** *A SHIQ concept  $D$  is in  $\mathbf{C}_0^+$  ( $\mathbf{C}_0^-$ ) iff we find, for every position  $p$  in  $D$  (in  $\neg D$ ), that  $\text{pl}(D, p) = 0$  ( $\text{pl}(\neg D, p) = 0$ ).*

**Proof:** Observe that using  $\neg D$  in the condition for  $\mathbf{C}_0^-$  reflects the fact that those concepts occur only at negative subpositions in concepts of type  $\mathbf{C}_0^+$ . The proof proceeds by induction over the structure of concepts. For the base cases  $\perp$ ,  $\top$ , and  $A$ , the claim is obvious. Now let  $D = \neg C$ .

It is easy to see that  $D \in \mathbf{C}_0^+$  iff  $C \in \mathbf{C}_0^-$ . By the induction hypothesis, this is

equivalent to  $\text{pl}(D, p) = \text{pl}(\neg C, p) = 0$  for any  $p$ . Conversely,  $D \in \mathbf{C}_0^-$  iff  $C \in \mathbf{C}_0^+$ , which is equivalent to  $\text{pl}(C, p) = 0$  for every  $p$  in  $C$ . By the definition of  $\text{pl}^-$ , it is easy to see that this is equivalent to  $\text{pl}(\neg D, p) = 0$  for every  $p$  in  $\neg D$ .

The remaining cases are very similar, and the arguments for  $\mathbf{C}_0^+$  and  $\mathbf{C}_0^-$  are mostly symmetric. We exemplify the required reasoning by considering the case  $D = \exists R.C$ . Clearly,  $\text{pl}(D, \epsilon) \neq 0$ , so  $D$  is not in  $\mathbf{C}_0^+$ . On the other hand, we find that  $D \in \mathbf{C}_0^-$  iff  $C \in \mathbf{C}_0^-$  iff  $\text{pl}(\neg C, p) = 0$  for every  $p$ . By the definition of  $\text{pl}^-$ , this clearly is equivalent to  $\text{pl}(\neg D, p) = 0$  for every  $p$ . All other cases are shown analogously.  $\square$

The following is the crucial result for our characterization.

**Proposition 2.11** *A SHIQ concept  $D$  is in  $\mathbf{C}_1^+$  ( $\mathbf{C}_1^-$ ) iff we find, for every position  $p$  in  $D$  (in  $\neg D$ ), that  $\text{pl}(D, p) \leq 1$  ( $\text{pl}(\neg D, p) \leq 1$ ).*

**Proof:** The proof proceeds as in Lemma 2.10, so we only consider some specific cases of the induction. So assume that  $D = C_1 \sqcup C_2$ . Then  $D \in \mathbf{C}_1^+$  iff  $C_1 \in \mathbf{C}_0^+$  and  $C_2 \in \mathbf{C}_1^+$ . By the induction hypothesis and the definition of  $\text{pl}^+$ , we obtain  $\text{pl}(D, p) \leq 1$  for all  $p$  in  $D$ .

Conversely, assume that  $\text{pl}(D, p) \leq 1$  for all  $p$  in  $D$ . For this to hold at all positions other than  $\epsilon$ ,  $C_1$  and  $C_2$  must be in  $\mathbf{C}_1^+$ . In addition,  $\text{pl}(D, \epsilon) = \text{pl}^+(C_1) + \text{pl}^+(C_2) \leq 1$  implies that  $\text{pl}^+(C_1) = 0$  or  $\text{pl}^+(C_2) = 0$ . Without loss of generality, we assume that  $\text{pl}^+(C_1) = 0$ . By the definition of  $\text{pl}^+$  and  $\text{pl}^-$ , it is easy to see that this implies  $\text{pl}(C_1, p) = 0$  for all  $p$  in  $C_1$ . Thus, by Lemma 2.10,  $C_1 \in \mathbf{C}_0^+$ . The case for  $D = C_1 \sqcup C_2$  and  $\mathbf{C}_1^-$  is simpler, since values of  $C_1$  and  $C_2$  are combined with the max operation here. All other cases are shown analogously.  $\square$

Now we can sum up our results in the following corollary.

**Corollary 2.12** *An extensionally reduced  $\mathcal{ALCHIQ}$  knowledge base  $KB$  is Horn iff, for all axioms  $C \sqsubseteq D$  of  $KB$ , one finds that  $(\neg C \sqcup D) \in \mathbf{C}_1^+$ .*

We note that for  $E$  of the form  $\neg C \sqcup D$ , it is equivalent to say that  $E \in \mathbf{C}_1^+$  (as in Corollary 2.12) and to say that  $E$  is of the form of the form  $\mathbf{C}_0^- \sqsubseteq \mathbf{C}_1^+$  or  $\mathbf{C}_1^- \sqsubseteq \mathbf{C}_0^+$  (as in Definition 2.3).

We argue that this definition is more easy to comprehend than the original characterization. For example, it is now readily seen that the axiom  $\geq 2 R.(C \sqcap \exists R.D) \sqsubseteq \forall R.\neg E$  is of the form  $\mathbf{C}_1^- \sqsubseteq \mathbf{C}_0^+$  and is thus Horn, whereas  $\forall R.\neg E \sqsubseteq \geq 2 R.(C \sqcap \exists R.D)$  is not. This is less obvious when considering the original definition. Also note that Corollary 2.12 only depends on Table 2, but does not require the definition of *position*, *polarity*, or any of the auxiliary functions  $\text{pl}^{(+/-)}$ .

So far, we only considered  $\mathcal{ALCHIQ}$  knowledge bases, i.e. we excluded tran-

Table 8

Definition of  $\text{clos}(\mathcal{KB})$ .  $\text{NNF}(C)$  denotes the negation normal form of some concept  $C$ . For details see [20].

- If  $C \sqsubseteq D \in \mathcal{KB}$ , then  $\text{NNF}(\neg C \sqcup D) \in \text{clos}(\mathcal{KB})$ ,
- If  $C(a) \in \mathcal{KB}$ , then  $\text{NNF}(C) \in \text{clos}(\mathcal{KB})$ ,
- If  $C \in \text{clos}(\mathcal{KB})$  and  $D$  is a subconcept of  $C$ , then  $D \in \text{clos}(\mathcal{KB})$ ,
- If  $\leq n R.C \in \text{clos}(\mathcal{KB})$ , then  $\text{NNF}(\neg C) \in \text{clos}(\mathcal{KB})$ ,
- If  $\forall R.C \in \text{clos}(\mathcal{KB})$ ,  $S \sqsubseteq^* R$ , and  $\text{Trans}(S) \in \mathcal{KB}$ , then  $\forall S.C \in \text{clos}(\mathcal{KB})$ .

sitivity from our treatment. The reason is that transitivity axioms of  $\mathit{SHIQ}$  are replaced by  $\mathit{ALCHIQ}$  axioms before axioms are transformed into clausal normal form. These additional axioms can actually lead to non-Hornness as well. For showing that our following definition of Horn- $\mathit{SHIQ}$  is correct, we first briefly repeat this transformation procedure.

For a  $\mathit{SHIQ}$  knowledge base  $\mathcal{KB}$ , a set of concept terms  $\text{clos}(\mathcal{KB})$  is defined recursively as shown in Table 8. Now  $\mathcal{KB}$  is transformed into an  $\mathit{ALCHIQ}$  knowledge base  $\Omega(\mathcal{KB})$  by

- eliminating all transitivity axioms  $\text{Trans}(S)$ , and by
- adding the axiom  $\forall R.C \sqsubseteq \forall S.(\forall S.C)$ , for every concept  $\forall R.C \in \text{clos}(\mathcal{KB})$  and role  $S$ , such that  $S \sqsubseteq^* R$  and  $\text{Trans}(S) \in \mathcal{KB}$ .

It was shown in [20] that  $\mathcal{KB}$  is satisfiable iff  $\Omega(\mathcal{KB})$  is satisfiable. A similar reduction was already introduced in [24, Chapter 6], but we focus on the transformation used for defining Horn- $\mathit{SHIQ}$ . Based on the prior definition of Horn- $\mathit{ALCHIQ}$ , a Horn- $\mathit{SHIQ}$  knowledge base in [15] was defined as a  $\mathit{SHIQ}$  knowledge base  $\mathcal{KB}$  for which  $\Omega(\mathcal{KB})$  is in Horn- $\mathit{ALCHIQ}$ . We are now ready to provide a simpler formulation.

**Proposition 2.13** *We say that a  $\mathit{SHIQ}$  axiom  $C \sqsubseteq D$  is Horn if the concept expression  $\neg C \sqcup D$  has the form  $\mathbf{C}_1^+$  as defined by the context-free grammar in Table 2.*

*A  $\mathit{SHIQ}$  knowledge base with an extensionally reduced ABox is in Horn- $\mathit{SHIQ}$  iff all of its TBox axioms are Horn.*

**Proof:** In Corollary 2.12 it was shown that a knowledge base is in Horn- $\mathit{ALCHIQ}$  iff its TBox consists of  $\mathit{ALCHIQ}$ -axioms that are Horn in the above sense. Here we only show that the components with complex roles account for the additional axioms that can be constructed in Horn- $\mathit{SHIQ}$ . This is achieved by analysing the axioms that are introduced by the above transformation. Indeed, axioms of the form  $\forall R.C \sqsubseteq \forall S.(\forall S.C)$  might fail to be Horn since they correspond to expressions  $\exists R. \neg C \sqcup \forall S.(\forall S.C)$ . The latter are generally not Horn, since disjunctions in  $\mathbf{C}_1^+$

must have the form  $C_0^+ \sqcup C_1^+$ . Since  $\exists R.\neg C$  cannot be of the form  $C_0^+$ , this requires that  $\forall S.(\forall S.C)$  is in  $C_0^+$ . But this can only be the case if  $C$  is in  $C_0^+$  as well.  $\exists R.\neg C$  in this case also is in  $C_1^+$ , since  $C_0^- \subseteq C_1^+$  and  $C_0^+ \subseteq C_1^-$ . This can be shown by an easy induction over the structure of  $C_0^-$  which we omit here (the base case is  $\mathbf{A}$ ; the mutual dependency between  $C_0^+$  and  $C_0^-$  is not problematic during the induction steps).

We thus have described the axioms that can be introduced without problems during transitivity elimination. A closer look at the elimination procedure reveals that the introduction of axioms depends on the existence of formulae of the form  $\forall R.C \in \text{clos}(\mathcal{KB})$ , where  $R$  has a transitive subrole, i.e.  $R$  is not simple. We must ensure that  $C$  is in  $C_0^+$  in this case. The last two lines of Table 8 obviously cannot directly contribute to the inclusion of formulae  $\forall R.C$  in  $\text{clos}(\mathcal{KB})$  (unless another problematic axiom is already present). Moreover, since we restrict to extensionally reduced ABoxes, the second line is not relevant either. Consequently, a formula  $\forall R.C$  is in  $\text{clos}(\mathcal{KB})$  iff it is a subconcept of the negation normal form of some concept  $\neg D \sqcup E$  with  $D \sqsubseteq E \in \mathcal{KB}$ .

Now consider a *SHIQ* knowledge base  $\mathcal{KB}$  which has a TBox in Horn-*ALCHIQ* when ignoring any transitivity axioms. From the above considerations we conclude:  $\mathcal{KB}$  is in Horn-*SHIQ* iff, for every TBox axiom  $D \sqsubseteq E$ , every non-simple role  $R$ , and every subconcept  $\forall R.C$  of  $\text{NNF}(\neg D \sqcup E)$ , we find that  $C$  is in  $C_0^+$ . For subconcepts of positive polarity, this is exactly captured by the distinction between  $\forall S.C_1^+$  and  $\forall R.C_0^+$  in the definition of  $C_1^+$ . Subconcepts of the form  $C_1^-$  have negative polarity in the constructed axiom, so the dual descriptions  $\exists S.C_1^-$  and  $\exists R.C_0^-$  characterise the required restrictions. Clearly, no further restrictions are required, and the given restrictions cannot be relaxed without introducing non-Horn axioms during the elimination procedure.  $\square$

The advantage of the above definition, besides its simplicity and brevity, is that it provides a local criterion for checking Hornness by investigating the structure of single axioms. The original definition hides this locality by relying on a transitivity elimination procedure that operates on the whole knowledge base. We adopt the definition of Proposition 2.13 to characterise the Horn-version of fragments of *SHIQ*, such as Horn-*FLE*, as well. Note that Horn-*SHIQ* includes all of  $\mathcal{EL}$ , i.e. Horn- $\mathcal{EL}$  is just  $\mathcal{EL}$ .

As an example of a Horn-*SHIQ* knowledge base, consider the ontology in Table 9. It shows some of the expressivity possible in this fragment.

#### 2.4 Reducibility of reasoning problems in the Horn case

Finally, we observe that the following standard reasoning tasks are mutually reducible even when restricting to Horn knowledge bases:

Table 9  
An example ontology in Horn-*SHIQ*.

**TBox/RBox**

Parent  $\equiv \exists \text{ hasChild}.\top$   
 Person  $\sqsubseteq \exists \text{ childOf}.\text{Person}$   
 ManyChildren  $\sqsubseteq \geq 2 \text{ hasChild}.\top$   
 NoSiblings  $\sqsubseteq \text{Person} \sqcap \forall \text{ childOf} . (\leq 1 \text{ hasChild}.\top)$   
 childOf  $\equiv \text{hasChild}^{-1}$

**ABox**

hasChild(Elaine, Sir Lancelot)  
 noSiblings(Lancelot du Lac)  
 childOf(Lancelot du Lac, Elaine)

**Knowledge base satisfiability.** We call a knowledge base *satisfiable*, if it has a model, i.e., if there exists an interpretation  $\mathcal{I}$  satisfying all axioms of the knowledge base.

**Instance checking.** For a given individual  $a$  and a given concept description  $C$  of form  $C_0^-$ , we ask whether  $C(a)$  is satisfied in all models of the knowledge base KB. This task can be reduced to the knowledge base satisfiability problem in the following way: Letting  $A$  be a new, unused concept name, check whether the knowledge base  $\text{KB} \cup \{A(a), A \sqcap C \sqsubseteq \perp\}$  is unsatisfiable.

**Entailment of TBox axioms.** A TBox axiom (GCI)  $C \sqsubseteq D$  is *entailed* by a knowledge base KB if it is satisfied by all interpretations that satisfy the knowledge base. If  $C$  is of the form  $C_1^+$  and  $D$  is of the form  $C_0^-$ , this problem can be reduced to the instance checking problem: let  $A, B$  be concept names not already present in the knowledge base KB and  $a$  be a new individual name. Then instance check for  $B(a)$  in  $\text{KB} \cup \{A \sqsubseteq C, D \sqsubseteq B, A(a)\}$ .

**Concept satisfiability.** A concept description  $C$  is *satisfiable* (with respect to a given knowledge base) if the knowledge base has a model  $\mathcal{I}$  with  $C^{\mathcal{I}} \neq \emptyset$ . If  $C$  has the form  $C_1^+$ , this can be reduced to the preceding problem by checking whether  $C \sqsubseteq \perp$  is entailed by the considered knowledge base.

Hence, we have shown that all reasoning problems can be reduced to knowledge base satisfiability. Querying a knowledge base for some statement is equivalent to checking whether the negation of this statement entails unsatisfiability, which explains why the above (Horn) restrictions on queries are in a sense dual to the restrictions on Horn axioms.

Finally mark that a knowledge base is satisfiable if and only if the concept  $\top$  is satisfiable. This closes the circle and shows that also in the Horn case all mentioned reasoning tasks are reducible to each other.

### 3 Horn- $\mathcal{FL}_0$

The description logic  $\mathcal{FL}_0$  is indeed very simple:  $\top$ ,  $\perp$ ,  $\sqcap$ , and  $\forall$  are the only operators allowed. Yet, checking the satisfiability of  $\mathcal{FL}_0$  knowledge bases is already EXPTIME-complete [1]. In this section, we show that Horn- $\mathcal{FL}_0$  is in P, and thus is much simpler than its non-Horn counterpart. In fact, we can even extend the logic with various operations without sacrificing tractability.

**Definition 3.1** *The description logic  $\mathcal{FL}_0^+$  is the extension of  $\mathcal{FL}_0$  with*

- *nominals,*
- *role hierarchies,*
- *role composition, and*
- *inverse roles.*

*The logic Horn- $\mathcal{FL}_0^+$  is the restriction of  $\mathcal{FL}_0^+$  to TBox axioms of the form  $C_0 \sqsubseteq C_1^+$  as defined in Table 4.*

To show that Horn- $\mathcal{FL}_0^+$  is in P, we will reduce satisfiability checking for Horn- $\mathcal{FL}_0^+$  to satisfiability checking in the 3-variable fragment of function-free Horn logic. A Horn-clause is a disjunction of atomic formulae or negations thereof, which contains at most one non-negated atom, and with all variables quantified universally. Horn-clauses are commonly written as implications (with possibly empty head or body), and without explicitly specifying the quantifiers. The following is straightforward.

**Proposition 3.2** *Satisfiability of a logical theory which consists of function-free Horn-clauses with a bounded number of variables can be checked in time polynomial w.r.t. the size of the theory.*

**Proof:** Due to the absence of function symbols, the theory is equivalent to its grounding (assuming, w.l.o.g., that the language has at least one constant symbol). The latter is a theory of propositional Horn-logic that is polynomially bounded in the size of the input theory. Satisfiability of propositional Horn-logic theories can easily be checked in polynomial time.  $\square$

The following is an easy restriction of Theorem 2.7 to Horn- $\mathcal{FL}_0^+$ .

**Lemma 3.3** *Checking satisfiability of a Horn- $\mathcal{FL}_0^+$  knowledge base can be reduced in linear time to checking satisfiability of a Horn- $\mathcal{FL}_0^+$  knowledge base that*

Table 10

Normal form for Horn- $\mathcal{FL}_0^+$ .  $A$ ,  $B$ , and  $C$  are names of atomic concepts or nominal classes,  $R$ ,  $S$ , and  $T$  (possibly inverse) role names, and  $c$  and  $d$  individual names.

$A \sqsubseteq C$	$\top \sqsubseteq C$	$A(c)$	$R \sqsubseteq T$
$A \sqcap B \sqsubseteq C$	$A \sqsubseteq \perp$	$R(c, d)$	$R \circ S \sqsubseteq T$
$A \sqsubseteq \forall R.C$		$c \approx d$	

contains only axioms in the normal form given in Table 10.

The normal form transformation is necessary to ensure that at most three distinct variables are needed within the first-order version of every axiom.

**Lemma 3.4** *Every Horn- $\mathcal{FL}_0^+$  knowledge base in normal form is semantically equivalent to a logical theory in the 3-variable fragment of function-free Horn-logic.*

**Proof:** The translation is straightforward for most cases. Axioms of the form  $A \sqsubseteq \forall R.C$  are translated into Horn-clauses  $\forall x. \forall y. (\neg A(x) \vee \neg R(x, y) \vee C(y))$ . For nominal classes  $\{c\}$ , we write  $x \approx c$  instead of  $\{c\}(x)$ . Equality statements from this transformation and from ABox statements are taken into account by explicitly axiomatising equality in Horn-logic. The following axioms are added

$$\begin{array}{l}
 \rightarrow x \approx x \quad C(x) \vee x \approx y \rightarrow C(y) \\
 x \approx y \quad \rightarrow y \approx x \quad R(x, z) \vee x \approx y \rightarrow R(y, z) \\
 x \approx y \vee y \approx z \rightarrow x \approx z \quad R(z, x) \vee x \approx y \rightarrow R(z, y)
 \end{array}$$

instantiated for every concept and role name in place of  $C$  and  $R$ , respectively. Furthermore, axioms of the form

$$R(x, y) \rightarrow R^{-1}(y, x) \quad R^{-1}(x, y) \rightarrow R(y, x)$$

are added for every role name  $R$ . The additional axioms obviously increase the size of the knowledge base only linearly. It is easy to see that the resulting Horn-theory is semantically equivalent to the original knowledge base.  $\square$

Summing up, we obtain the following.

**Theorem 3.5** *Deciding satisfiability for the description logic Horn- $\mathcal{FL}_0^+$  is in P.*

**Proof:** Combine Lemmas 3.3 and 3.4 with Proposition 3.2.  $\square$

This also shows that decidability in Horn- $\mathcal{FL}_0$  can be checked in polynomial time, which is an interesting contrast to the EXPTIME-completeness of  $\mathcal{FL}_0$ .

The well-known DLP-fragment of *SHIQ* mentioned before [10,26] does indeed allow for a similar reduction to 3-variable Horn logic, and thus has an at most polynomial time complexity. To the best of our knowledge, this result has not been spelled out before. To see it, we need to look at a syntactic definition that allows for a suitable normal form transformation. The following is taken from [11,12].

**Proposition 3.6** *All description logic programs (DLP) can be transformed into a semantically equivalent set of function-free Horn rules with at most three-variables.*

**Proof:** The claim follows immediately from Theorem 2 of [11], together with the normal forms given in Table 1 loc. cit.  $\square$

**Corollary 3.7** *Deciding satisfiability for description logic programs (DLP) is in P.*

As discussed in [26,11], extensions of DLP with nominals are also admissible. In fact, their use of enumerated concepts of the form  $\{o_1, o_2, \dots, o_n\}$  is a special case of the reduction of disjunctions in  $\mathbf{C}_0^-$  established by Theorem 2.7.

## 4 Horn- $\mathcal{FL}^-$

Horn- $\mathcal{FL}^-$  is the Horn fragment of  $\mathcal{ALC}$  that allows  $\top$ ,  $\perp$ ,  $\sqcap$ ,  $\sqcup$ , and unqualified  $\exists$  (i.e. concept expressions of the form  $\exists R.\top$ ). Although Horn- $\mathcal{FL}^-$  is only a very small extension of Horn- $\mathcal{FL}_0$ , we will see that it is PSPACE-complete. Moreover, not all of the extensions that could be added to Horn- $\mathcal{FL}_0^+$  can also be added to Horn- $\mathcal{FL}^-$  without further increasing the complexity. The extension of  $\mathcal{FL}^-$  that we will consider below is defined as follows.

**Definition 4.1** *The description logic  $\mathcal{FLOH}^-$  is the extension of  $\mathcal{FL}^-$  with*

- *nominals, and*
- *role hierarchies.*

*The logic Horn- $\mathcal{FLOH}^-$  is the restriction of  $\mathcal{FLOH}^-$  to TBox axioms of the form  $\mathbf{C}_0 \sqsubseteq \mathbf{C}_1^+$  as defined in Table 4.*

We will show that all logics between Horn- $\mathcal{FL}^-$  and Horn- $\mathcal{FLOH}^-$  are PSPACE-complete.

### 4.1 Hardness

We directly show that Horn- $\mathcal{FL}^-$  is PSPACE by reducing the halting problem for polynomially space-bounded Turing machines to checking unsatisfiability in Horn-

$\mathcal{FL}^-$ .

**Definition 4.2** A deterministic Turing machine (TM)  $\mathcal{M}$  is a tuple  $(Q, \Sigma, \Delta, q_0)$  where

- $Q$  is a finite set of states,
- $\Sigma$  is a finite alphabet that includes a blank symbol  $\square$ ,
- $\Delta \subseteq (Q \times \Sigma) \times (Q \times \Sigma \times \{l, r\})$  is a transition relation that is deterministic, i.e.  $(q, \sigma, q_1, \sigma_1, d_1), (q, \sigma, q_2, \sigma_2, d_2) \in \Delta$  implies  $q_1 = q_2, \sigma_1 = \sigma_2$ , and  $d_1 = d_2$ .
- $q_0 \in Q$  is the initial state, and
- $Q_A \subseteq Q$  is a set of accepting states.

A configuration of  $\mathcal{M}$  is a word  $\alpha \in \Sigma^* Q \Sigma^*$ . A configuration  $\alpha'$  is a successor of a configuration  $\alpha$  if one of the following holds:

- (1)  $\alpha = w_l q \sigma \sigma_r w_r, \alpha' = w_l \sigma' q' \sigma_r w_r$ , and  $(q, \sigma, q', \sigma', r) \in \Delta$ ,
- (2)  $\alpha = w_l q \sigma, \alpha' = w_l \sigma' q' \square$ , and  $(q, \sigma, q', \sigma', r) \in \Delta$ ,
- (3)  $\alpha = w_l \sigma_l q \sigma w_r, \alpha' = w_l q' \sigma_l \sigma' w_r$ , and  $(q, \sigma, q', \sigma', l) \in \Delta$ ,

where  $q \in Q$  and  $\sigma, \sigma', \sigma_l, \sigma_r \in \Sigma$  as well as  $w_l, w_r \in \Sigma^*$ . Given some natural number  $s$ , the possible transitions in space  $s$  are defined by additionally requiring that  $|\alpha'| \leq s + 1$ .

The set of accepting configurations is the least set which satisfies the following conditions. A configuration  $\alpha$  is accepting iff

- $\alpha = w_l q w_r$  and  $q \in Q_A$ , or
- at least one the successor configurations of  $\alpha$  are accepting.

$\mathcal{M}$  accepts a given word  $w \in \Sigma^*$  (in space  $s$ ) iff the configuration  $q_0 w$  is accepting (when restricting to transitions in space  $s$ ).

The complexity class PSPACE is defined as follows.

**Definition 4.3** A language  $L$  is accepted by a polynomially space-bounded TM iff there is a polynomial  $p$  such that, for every word  $w \in \Sigma^*$ ,  $w \in L$  iff  $w$  is accepted in space  $p(|w|)$ .

In this section, we exclusively deal with polynomially space-bounded TMs, and so we omit additions such as “in space  $s$ ” when clear from the context.

In the following, we consider a fixed TM  $\mathcal{M}$  denoted as in Definition 4.2, and a polynomial  $p$  that defines a bound for the required space. For any word  $w \in \Sigma^*$ , we construct a Horn- $\mathcal{FLOH}^-$  knowledge base  $K_{\mathcal{M}, w}$  and show that  $w$  is accepted by  $\mathcal{M}$  iff  $K_{\mathcal{M}, w}$  is unsatisfiable. Intuitively, the elements of an interpretation domain

Table 11

Knowledge base  $K_{\mathcal{M},w}$  simulating a polynomially space-bounded TM. The axioms are instantiated for all  $q, q' \in Q$ ,  $\sigma, \sigma' \in \Sigma$ ,  $i, j \in \{0, \dots, p(|w|) - 1\}$ , and  $\delta \in \Delta$ .

**(1) Left and right transition rules:**

$$A_q \sqcap H_i \sqcap C_{\sigma,i} \sqsubseteq \exists S. \top \sqcap \forall S. (A_{q'} \sqcap H_{i+1} \sqcap C_{\sigma',i})$$

$$\text{with } \delta = (q, \sigma, q', \sigma', r), i < p(|w|) - 1$$

$$A_q \sqcap H_i \sqcap C_{\sigma,i} \sqsubseteq \exists S. \top \sqcap \forall S. (A_{q'} \sqcap H_{i-1} \sqcap C_{\sigma',i}) \quad \text{with } \delta = (q, \sigma, q', \sigma', l), i > 0$$

**(2) Memory:**

$$H_j \sqcap C_{\sigma,i} \sqsubseteq \forall S. C_{\sigma,i} \quad i \neq j$$

**(3) Failure:**

$$F \sqcap A_q \sqsubseteq \perp \quad q \in Q_A$$

**(4) Propagation of failure:**

$$F \sqsubseteq \forall S. F$$

of  $K_{\mathcal{M},w}$  represent possible configurations of  $\mathcal{M}$ , encoded by the following concept names

- $A_q$  for  $q \in Q$ : the TM is in state  $q$ ,
- $H_i$  for  $i = 0, \dots, p(|w|) - 1$ : the TM is at position  $i$  on the storage tape,
- $C_{\sigma,i}$  with  $\sigma \in \Sigma$  and  $i = 0, \dots, p(|w|) - 1$ : position  $i$  on the storage tape contains symbol  $\sigma$ .

Based on those concepts, elements in each interpretation of a knowledge base encode certain states of the Turing machine. A role  $S$  will be used to encode the successor relationship between states. The initial configuration for word  $w$  is described by the concept expression  $I_w$ :

$$I_w := A_{q_0} \sqcap H_0 \sqcap C_{\sigma_0,0} \sqcap \dots \sqcap C_{\sigma_{|w|-1},|w|-1} \sqcap C_{\square,|w|} \sqcap \dots \sqcap C_{\square,p(|w|)-1},$$

where  $\sigma_i$  denotes the symbol at the  $i$ th position of  $w$ .

It is not hard to describe runs of the TM with Horn- $\mathcal{FL}^-$  axioms, but formulating the acceptance condition is slightly more difficult. The reason is that in absence of statements like  $\exists S.A$  and  $\forall S.A$  in the condition part of Horn-axioms, one cannot propagate acceptance from the final accepting configuration back to initial configuration. The solution is to introduce a new concept  $F$  that states that a state is *not* accepting, and to propagate this assumption forwards along the runs to provoke an inconsistency as soon as an accepting configuration is reached. Thus we arrive at the axioms given in Table 11.

Next we need to investigate the relationship between elements of an interpretation that satisfies  $K_{\mathcal{M},w}$  and configurations of  $\mathcal{M}$ . Given an interpretation  $\mathcal{I}$  of  $K_{\mathcal{M},w}$ , we say that an element  $e$  of the domain of  $\mathcal{I}$  *represents* a configuration  $\sigma_1 \dots \sigma_{i-1} q \sigma_i \dots \sigma_m$  if  $e \in A_q^{\mathcal{I}}$ ,  $e \in H_i^{\mathcal{I}}$ , and, for every  $j \in \{0, \dots, p(|w|) - 1\}$ ,

$e \in C_{\sigma,j}^I$  whenever

$$j \leq m \text{ and } \sigma = \sigma_m \quad \text{or} \quad j > m \text{ and } \sigma = \square.$$

Note that we do not require uniqueness of the above, so that a single element might in fact represent more than one configuration. As we will see below, this does not affect our results. If  $e$  represents a configuration as above, we will also say that  $e$  has state  $q$ , position  $i$ , symbol  $\sigma_j$  at position  $j$  etc.

**Lemma 4.4** *Consider some interpretation  $\mathcal{I}$  that satisfies  $K_{\mathcal{M},w}$ . If some element  $e$  of  $\mathcal{I}$  represents a configuration  $\alpha$  and some transition  $\delta$  is applicable to  $\alpha$ , then  $e$  has an  $S^{\mathcal{I}}$ -successor that represents the (unique) result of applying  $\delta$  to  $\alpha$ .*

**Proof:** Consider an element  $e$ , state  $\alpha$ , and transition  $\delta$  as in the claim. Then one of the axioms (1) applies, and  $e$  must also have an  $S^{\mathcal{I}}$ -successor. This successor represents the correct state, position, and symbol at position  $i$  of  $e$ , again by the axioms (1). By axiom (2), symbols at all other positions are also represented by all  $S^{\mathcal{I}}$ -successors of  $e$ .  $\square$

**Lemma 4.5** *A word  $w$  is accepted by  $\mathcal{M}$  iff  $\{I_w(i), F(i)\} \cup K_{\mathcal{M},w}$  is unsatisfiable, where  $i$  is a new constant symbol.*

**Proof:** Let  $\mathcal{I}$  be a model of  $\{I_w(i), F(i)\} \cup K_{\mathcal{M},w}$ .  $\mathcal{I}$  being a model for  $I_w(i)$ ,  $i^{\mathcal{I}}$  clearly represents the initial configuration of  $\mathcal{M}$  with input  $w$ . By Lemma 4.4, for any further configuration reached by  $\mathcal{M}$  during computation,  $i^{\mathcal{I}}$  has a (not necessarily direct)  $S^{\mathcal{I}}$  successor representing that configuration.

Since  $\mathcal{I}$  satisfies  $F(i)$  and axiom (4) of Table 11, a simple induction argument shows that  $F^{\mathcal{I}}$  contains all  $S^{\mathcal{I}}$  successors of  $i^{\mathcal{I}}$ . But then  $\mathcal{I}$  satisfies axiom (3) only if none of the configurations that are reached have an accepting state. Since  $\mathcal{I}$  was arbitrary,  $\{I_w(i), F(i)\} \cup K_{\mathcal{M},w}$  can only have a satisfying interpretation if  $\mathcal{M}$  does not reach an accepting state.

It remains to show the converse: if  $\mathcal{M}$  does not accept  $w$ , there is some interpretation  $\mathcal{I}$  satisfying  $\{I_w(i), F(i)\} \cup K_{\mathcal{M},w}$ . To this end, we define a canonical interpretation  $\mathcal{M}$  as follows. The domain of  $\mathcal{M}$  is the set of all configurations of  $\mathcal{M}$  that have size  $p(|w|) + 1$  (i.e. that encode a tape of length  $p(|w|)$ , possibly with trailing blanks). The interpretations for the concepts  $A_q$ ,  $H_i$ , and  $C_{\sigma,i}$  are defined as expected so that every configuration represents itself but no other configuration. Especially,  $I_w^{\mathcal{M}}$  is the singleton set containing the initial configuration. Given two configurations  $\alpha$  and  $\alpha'$ , and a transition  $\delta$ , we define  $(\alpha, \alpha') \in S^{\mathcal{M}}$  iff there is a transition  $\delta$  from  $\alpha$  to  $\alpha'$ .  $F^{\mathcal{M}}$  is defined to be the set of all configurations that are reached during the run of  $\mathcal{M}$  on  $w$ .

Table 12

Normal form for Horn- $\mathcal{FLOH}^-$ .  $A, B$ , and  $C$  are names of atomic concepts or nominal classes,  $R, S$ , and  $T$  role names, and  $c$  and  $d$  individual names.

$A \sqsubseteq C$	$\exists R.T \sqsubseteq C$	$A \sqcap B \sqsubseteq C$	$A(c)$
$\top \sqsubseteq C$	$A \sqsubseteq \exists R.T$	$R \sqsubseteq S$	$R(c, d)$
$A \sqsubseteq \perp$	$A \sqsubseteq \forall R.C$		$c \approx d$

It is easy to see that  $M$  satisfies the axioms (1), (2), and (3) of Table 11. Axiom (4) is satisfied since, by our initial assumption, none of the configurations reached by  $M$  is in an accepting state.  $\square$

Thus checking satisfiability of Horn- $\mathcal{FL}^-$  knowledge bases is PSPACE-hard.

## 4.2 Containment

To show that inferencing for Horn- $\mathcal{FLOH}^-$  is in PSPACE, we develop a tableau algorithm for deciding the satisfiability of a Horn- $\mathcal{FLOH}^-$  knowledge base. To this end, we first present a normal form transformation similar to the one in Section 3. Afterwards, we present the tableau construction and show its correctness, and demonstrate that it can be executed in polynomial space.

The reduction is established by first transforming each Horn- $\mathcal{FLOH}^-$  knowledge base into a normal form, again by restricting Theorem 2.7 accordingly.

**Lemma 4.6** *Checking satisfiability of a Horn- $\mathcal{FLOH}^-$  knowledge base can be reduced in linear time to checking satisfiability of a Horn- $\mathcal{FLOH}^-$  knowledge base that contains only axioms in the normal form given in Table 12.*

Next, we are going to present a procedure for checking satisfiability of Horn- $\mathcal{FL}^-$  knowledge bases. In the following we assume without loss of generality that the  $\mathcal{FL}^-$  language in consideration has at least one individual symbol.

**Definition 4.7** *Consider a Horn- $\mathcal{FLOH}^-$  knowledge base  $KB$  in normal form, with  $\mathbf{C}$  the set of atomic concepts and nominal names,  $\mathbf{R}$  the set of role names, and  $\mathbf{I}$  the set of individual names. A set of relevant concept expressions is defined by setting*

$$\text{cl}(KB) = \mathbf{C} \cup \{QR.C \mid R \in \mathbf{R}, C \in \mathbf{C}, Q \in \{\exists, \forall\}\} \cup \{\top, \perp\}.$$

Given a set  $I$  of individual names, a set  $\mathcal{T}_I$  of ABox expressions is defined as follows:

$$\mathcal{T}_I := \{C(e) \mid C \in \text{cl}(KB), e \in I\} \cup \{R(e, f) \mid R \in \mathbf{R}, e, f \in I\}$$

For a set  $T \subseteq \mathcal{T}_I$  and individuals  $e, f \in I$ , we use  $T_{e \mapsto f}$  to denote the set

$$\{C(f) \mid C(e) \in T\} \cup \{R(f, g) \mid R(e, g) \in T, g \in I\} \cup \{R(g, f) \mid R(g, e) \in T, g \in I\}.$$

For the special case that  $e = f$ , we use the abbreviation  $T_e := T_{e \mapsto e}$ . A tableau for  $KB$  is given by a (possibly infinite) set  $I$  of individual names, and a set  $T \subseteq \mathcal{T}_I$  such that  $\mathbf{I} \subseteq I$  and the following conditions hold:

- (1) if  $e \in I$ , then  $\top(e) \in T$  and, if  $e \in \mathbf{I}$ ,  $\{e\} \in T$ ,
- (2) if  $A(e) \in KB$  ( $R(e, f) \in KB$ ), then  $A(e) \in T$  ( $R(e, f) \in T$ ),
- (3) if  $e \approx f \in KB$ , then  $\{f\}(e) \in T$  and  $\{e\}(f) \in T$ ,
- (4) if  $\{f\}(e) \in T$ , then  $C(e) \in T$  iff  $C(f) \in T$ ,  $R(e, g) \in T$  iff  $R(f, g) \in T$ , and  $R(g, e) \in T$  iff  $R(g, f) \in T$ , for all  $C \in \mathbf{C}$ ,  $R \in \mathbf{R}$ , and  $g \in I$ ,
- (5) if  $A \sqsubseteq C \in KB$  and  $A(e) \in T$ , then  $C(e) \in T$ ,
- (6) if  $A \sqcap B \sqsubseteq C \in KB$ ,  $A(e) \in T$ , and  $B(e) \in T$ , then  $C(e) \in T$ ,
- (7) if  $R \sqsubseteq S \in KB$  and  $R(e, f) \in T$ , then  $S(e, f) \in T$ ,
- (8)  $\exists R. \top(e) \in T$  iff  $R(e, f) \in T$  for some  $f \in I$ ,
- (9) if  $\forall R. C(e) \in T$ , then  $C(f) \in T$  for all  $f \in I$  with  $R(e, f) \in T$ ,

A tableau is said contain a clash if it contains a statement of the form  $\perp(e)$ .

**Proposition 4.8** A Horn- $\mathcal{FLOH}^-$  knowledge base  $KB$  is satisfiable iff it has a clash-free tableau.

**Proof:** Assume that  $KB$  has a clash-free tableau  $(I, T)$ . An interpretation  $\mathcal{I}$  is defined as follows. Due to condition 4 in Definition 4.7, we can define an equivalence relation  $\sim$  on  $I$  by setting  $e \sim f$  iff there is some  $g \in \mathbf{I}$  with  $\{\{g\}(e), \{g\}(f)\} \subseteq T$ . The domain  $I_\sim$  of  $\mathcal{I}$  is the set of equivalence classes of  $\sim$ . The interpretation function is defined by setting  $e^{\mathcal{I}} = [e]_\sim$ ,  $e^{\mathcal{I}} \in C^{\mathcal{I}}$  iff  $C(e) \in T$ , and  $(e^{\mathcal{I}}, f^{\mathcal{I}}) \in R^{\mathcal{I}}$  iff  $R(e, f) \in T$ , for all elements  $e, f \in I$ , concept names  $C$ , and role names  $R$ . It is easy to see that  $\mathcal{I}$  satisfies  $KB$ .

For the converse, assume that  $KB$  is satisfiable, and that it thus has some model  $\mathcal{I}$ . We define a tableau  $(I, T)$  where  $I$  is the domain of  $\mathcal{I}$ . Further, we set  $C(e) \in T$  iff  $e \in C^{\mathcal{I}}$ , and  $R(e, f) \in T$  iff  $(e, f) \in R^{\mathcal{I}}$ , where  $C \in \text{cl}(KB)$ , and  $R$  some role name. Again, it is easy to see that this meets the conditions of Definition 4.7.  $\square$

As is evident by the Turing machine construction in the previous section, some Horn- $\mathcal{FLOH}^-$  knowledge bases may require a model to contain an exponential number of individuals, even within single paths of the computation. Detecting clashes in polynomial space thus requires special care. In particular, a standard tableau procedure with blocking does not execute in polynomial space. Therefore, we first provide a procedural description of a *canonical tableau* which will form the basis for our below decision algorithm.

**Definition 4.9** Consider an algorithm that computes a tableau-like structure  $(I, T)$ .

Table 13

An algorithm for constructing tableaux for Horn- $\mathcal{FLOH}^-$  knowledge bases. Role statements computed by the algorithm might be marked *inactive* to better control the deduction. All other role statements are *active*.

- (T1)  $T := T \cup \{\top(e)\}$
- (T2) if  $e \in \mathbf{I}$  is a named individual,  $T := T \cup \{\{e\}(e)\}$
- (T3) for each  $A(e) \in KB$ ,  $T := T \cup \{A(e)\}$
- (T4) for each  $R(e, f) \in KB$ ,  $T := T \cup \{R(e, f)\}$
- (T5) for each  $e \approx f \in KB$ ,  $T := T \cup \{\{f\}(e)\}$  and  $T := T \cup \{\{e\}(f)\}$
- (T6) for each  $\{f\}(e) \in T$ 
  - (T6a) for each  $C(f) \in T$ ,  $T := T \cup \{C(e)\}$ ,
  - (T6b) for each  $g \in I$  and each  $R(f, g) \in T$ ,  $T := T \cup \{R(e, g)\}$ ;  $R(e, g)$  is marked inactive,
  - (T6c) for each  $g \in I$  and each  $R(g, f) \in T$ ,  $T := T \cup \{R(g, e)\}$ ;  $R(g, e)$  is marked inactive,
  - (T6d) for each  $C(e) \in T$ ,  $T := T \cup \{C(f)\}$ ,
  - (T6e) for each  $g \in I$  and each  $R(e, g) \in T$ ,  $T := T \cup \{R(f, g)\}$ ;  $R(f, g)$  is marked inactive,
  - (T6f) for each  $g \in I$  and each  $R(g, e) \in T$ ,  $T := T \cup \{R(g, f)\}$ ;  $R(g, f)$  is marked inactive
- (T7) for each  $A \sqsubseteq C \in KB$ , if  $A(e) \in T$  then  $T := T \cup \{C(e)\}$
- (T8) for each  $A \sqcap B \sqsubseteq C \in KB$ , if  $A(e) \in T$  and  $B(e) \in T$  then  $T := T \cup \{C(e)\}$
- (T9) for each  $R \sqsubseteq S \in KB$ , do the following:
  - (T9a) for each  $f \in I$ , if  $R(e, f) \in T$  and  $R(e, f)$  is not inactive, then  $T := T \cup \{S(e, f)\}$ ,
  - (T9b) if  $\exists R.\top(e) \in T$  then  $T := T \cup \{\exists S.\top(e)\}$
- (T10) for each  $f \in I$  and  $R(e, f) \in T$  with  $R(e, f)$  not inactive,  $T := T \cup \{\exists R.\top(e)\}$
- (T11) for each  $\forall R.C(e) \in T$  and each  $f \in I$  with  $R(e, f) \in T$ , if  $R(e, f)$  is not inactive, then  $T := T \cup \{C(f)\}$
- ( $\exists$ ) for each  $\exists R.\top(e) \in T$ , if  $R(e, f) \notin T$  for all  $f \in I$  then  $I := I \cup \{g\}$  and  $T := T \cup \{R(e, g)\}$ , where  $g$  is a fresh individual

Initially, we set  $I := \mathbf{I}$ , and  $T := \emptyset$ . The algorithm execute the following steps:

- (1) Iterate over all individuals  $e \in I$ . To each such  $e$ , apply rules (T1) to (T11) of Table 13.
- (2) If  $T$  was changed in the previous step, goto (1).
- (3) Apply rule ( $\exists$ ) of Table 13 to all existing elements  $e \in I$ .
- (4) If  $T$  was changed by the previous step, goto (1).
- (5) Halt.

While most rules should be obvious, some require explanations. The rules (T6) are used to ensure that individuals  $e$  satisfying a nominal class are synchronised with the respective named individual  $f \in \mathbf{I}$ . The six sub-rules are needed since one generally cannot add  $\{e\}(f)$  to  $T$  as  $e$  might not be an element of  $\mathbf{I}$ . On the other hand, role statements that are inferred in this way need not be taken into account as premises in other deduction rules, since they are guaranteed to have an active original. Whatever could be inferred using copied role statements and rules (T9a), (T10), or (T11), can as well be inferred via the active original from which the inactive role was initially created (note that this argument involves an induction over the number of applications of rule (T6)).

Rule (T9) is also special. In principle, one could omit (T9b), and use rules (T9a) and (T10) instead. This inference, however, is the only case where a role-successor of some individual  $e$  might contribute to the classes inferred for  $e$ . By providing rule (T9b), the class expressions containing  $e$  can be computed without considering any role successor, and rule (T10) is essential only when role expressions have been inferred from ABox statements. In combination with the delayed application of rule ( $\exists$ ), this ensures that concepts are indeed inferred by (T9b) rather than by (T9a)+(T10), which will be exploited in the proof of Lemma 4.13 below.

Also note that the algorithm of Definition 4.9 is not a decision procedure, since we do not require the algorithm to halt. What we are interested in, however, is the (possibly infinite) tableau that the algorithm constructs in the limit. The existence of this limit is evident from the fact that all completion rules are finitary, and that each rule monotonically increases the size of the computed structure. It is easy to see that there is a correspondence between the rules of Table 13 and the conditions of Definition 4.7, so that the limit structure will indeed meet all the requirements imposed on a tableau. For a given knowledge base  $KB$ , we write  $(\bar{I}_{KB}, \bar{T}_{KB})$  to denote the *canonical tableau* constructed by the above algorithm from  $KB$ , where the subscripts are omitted when understood. It is easy to see that, whenever the canonical tableau contains a clash, this must be the case for all possible tableaux.

The algorithm of Definition 4.9 can be viewed as a “breadth-first” construction of a canonical tableau. Due to the explicit procedural description of tableau rules, any role and class expression of the canonical tableau is first computed after a well-defined number of computation steps.<sup>6</sup> Accordingly, we define a total order  $<$  on  $\bar{T}$  by setting  $F < G$  iff  $F$  is computed before  $G$ .

The canonical tableau and the order  $<$  are the main ingredients for showing the correctness of following nondeterministic decision algorithm.

---

<sup>6</sup> For this to be true, one must also specify the order for the involved iterations, e.g. by ordering elements lexicographically and adopting a naming scheme for newly introduced elements. We assume that such an order was chosen.

**Definition 4.10** Consider a Horn- $\mathcal{FL}\mathcal{OH}^-$  knowledge base  $KB$  with canonical tableau  $(\bar{I}, \bar{T})$ . A set of individuals  $I$  is defined as  $I := \mathbf{I} \cup \{a, b\}$ , where  $a, b \notin \bar{I}$ . Nondeterministically select one element  $g \in I$ , and initialise  $T \subseteq \mathcal{T}_1$  by setting  $T := \{\perp(g)\}$ .

The algorithm repeatedly modifies  $T$  by nondeterministically applying one of the following rules:

- (N1) Given any  $X \in \mathcal{T}_1$ , set  $T := T \cup \{X\}$ . If  $X$  is a role statement, decide nondeterministically whether  $X$  is marked inactive.
- (N2) If there is some individual  $e \in I$  and  $X \in T$  such that  $X$  can be derived from  $T \setminus \{X\}$  using one of the rules (T1) to (T11) in Table 13, set  $T := T \setminus \{X\}$ . Rules (T6b), (T6c), (T6e), and (T6f) can only be used if  $X$  is marked inactive.
- (N3) If  $T_a = \{R(e, a)\}$  for some  $e \in I \setminus \{a\}$  such that  $\exists R.\top(e) \in T$ , set  $T := T \setminus T_a$ .
- (N4) If  $T_a = \emptyset$ , set  $T := (T \cup T_{b \rightarrow a}) \setminus T_b$ .
- (N5) If  $T = \emptyset$ , return “unsatisfiable.”

**Lemma 4.11** The algorithm of Definition 4.10 can be executed in polynomially bounded space.

**Proof:** Since  $|I|$ ,  $|C|$ , and  $|R|$  are polynomially bounded by the size of the knowledge base, so is  $\text{cl}(KB)$  and thus  $T$ .  $\square$

**Lemma 4.12** If there is a sequence of choices such that the algorithm of Definition 4.10 returns “unsatisfiable” after some finite time,  $KB$  is indeed unsatisfiable.

**Proof:** Intuitively, the nondeterministic algorithm applies rules of the algorithm in Definition 4.9 in reverse order, deleting a conclusion whenever it can be derived from the remaining statements. The anonymous individuals  $a$  and  $b$  are used to dynamically represent (various) elements from the canonical tableau. For a formal proof, assume that the algorithm terminates within finitely many steps, and, without loss of generality, that each step involves a successful application of one of the rules (N1) to (N5). We use  $T^n$  to denote the state of the algorithm  $n$  steps before termination. In particular,  $T^0 = \emptyset$ .

We claim that for each  $T^n$  there are individuals  $e, f \in \bar{I}$ , such that  $T^n_{a \rightarrow e, b \rightarrow f} \subseteq \bar{T}$ . This is verified by induction over the number of steps executed by the algorithm. Since  $T^0 = \emptyset$ , the claim for  $T^0$  holds for any  $e, f \in \bar{I}$ .

For the induction step, assume that  $T^n_{a \rightarrow e, b \rightarrow f} \subseteq \bar{T}$ . To show the claim for  $T^{n+1}$ , we distinguish by the transformation rule that was applied to obtain  $T^n$  from  $T^{n+1}$ :

- (N1) Since  $T^{n+1} \subset T^n$ , we conclude  $T^{n+1}_{a \rightarrow e, b \rightarrow f} \subseteq \bar{T}$ .
- (N2)  $T^{n+1} = T^n \cup \{X\}$ , where  $X$  can be derived from  $T^n$  by one of the rules (T1) to (T11). Since those rules have been applied exhaustively in  $\bar{T}$ , we find  $T^{n+1}_{a \rightarrow e, b \rightarrow f} \subseteq \bar{T}$ .

- (N3) We find  $T_a^n = \emptyset$  and, for some  $g \in I \setminus \{a\}$  and  $R \in \mathbf{R}$ ,  $T^{n+1} = T^n \cup \{R(g, a)\}$  and  $\exists R. \top(g) \in T^n$ . Define  $g' := f$  if  $g = b$ , and  $g' = g$  otherwise. We conclude that  $\exists R. \top(g') \in \bar{T}$  and thus there is some individual  $e' \in \bar{I}$  with  $R(g', e')$ . We conclude that  $T_{a \rightarrow e', b \rightarrow f}^{n+1} \subseteq \bar{T}$ .
- (N4) This rule merely exchanges  $b$  with (the unused)  $a$ . Thus we have  $T_{a \rightarrow f, b \rightarrow e}^{n+1} \subseteq \bar{T}$ .

Applying the above induction to the initial state  $\{\perp(g)\}$ , we find that  $\{\perp(g)\}_{a \rightarrow e, b \rightarrow f} \in \bar{T}$ . Hence  $\bar{T}$  indeed contains a clash and  $KB$  is unsatisfiable.  $\square$

**Lemma 4.13** *Whenever  $KB$  is unsatisfiable, there is a sequence of choices such that the algorithm of Definition 4.10 returns “unsatisfiable” after some finite time.*

**Proof:** We first specify a possible sequence of choices, and then show its correctness. If  $KB$  is unsatisfiable, there is some element  $e \in \bar{I}$  in the canonical tableau such that  $\perp(e) \in \bar{T}$ . Pick one such  $e$ . We use  $a'$  and  $b'$  to denote the elements of  $\bar{I}$  that are currently simulated by  $a$  and  $b$ . Initially, we set  $a' = b' = \square$  for some element  $\square \notin \bar{I}$ . Rule (N1) of the algorithm will repeatedly be used to close  $T$  under relevant inferences that are  $<$ -smaller than some statement  $X$ . Given  $X \in \bar{T}$ , we define:

$$\downarrow X = \left\{ C(f) \in \bar{T} \mid C(f) \leq X, f \in \mathbf{I} \cup \{a', b'\} \right\}_{a' \mapsto a, b' \mapsto b} \cup \left\{ R(f, g) \in \bar{T} \mid R(f, g) \text{ not inactive}, R(f, g) \leq X, f, g \in \mathbf{I} \cup \{a', b'\} \right\}_{a' \mapsto a, b' \mapsto b}.$$

This selects all elements in  $\bar{T}$  that can be represented using the elements from  $I$  with the current representation of  $a'$  as  $a$ , and  $b'$  as  $b$ . Throughout the below computation, the following property will be preserved:

$$T_{a \rightarrow a', b \rightarrow b'} \subseteq \bar{T} \quad (\dagger)$$

Now if  $e \in \mathbf{I}$ , set  $a' := e$ . Using the nondeterministic initialisation and rule (N1), the algorithm of Definition 4.10 can now compute  $T = \downarrow \{\perp(e)\}$ . The algorithm now repeatedly executes steps according to the following choice strategy.

**Single step choice strategy.** If  $T_a$  is non-empty, let  $X$  be the  $<$ -largest element of  $T_a$ . Else, let  $X$  be the  $<$ -largest element of  $T$ . By property  $(\dagger)$ , there is some  $X' \in \bar{T}$  with  $\{X\}_{a \rightarrow a', b \rightarrow b'} = \{X'\}$ . Applying rule (N1), the algorithm first computes  $T := T \cup \downarrow X$  (\*). The algorithm nondeterministically guesses the rule of Table 13 that was used to infer  $X'$ , and proceeds accordingly:

- If  $X'$  was inferred by one of the rules (T1), (T2), (T3), (T4), (T5), (T7), (T8),

(T9a), (T9b), and (T10), the premises of a respective rule application in  $T$  have been computed in (\*). This is so since the required premises are  $\prec$ -smaller and not inactive, and since they only involve individuals that are also found in  $X$ , i.e. which are represented by  $I$  with the current choice of  $a'$  and  $b'$ . Hence the algorithm can apply rule (N2) to reduce  $X$ .

- If  $X'$  was inferred by one of the rules of (T6), then one of the premises used was of the form  $\{f\}(e)$ , and thus  $f \in \mathbf{I}$ . Since inactive roles are not generated by any of the given choices, rules (T6b), (T6c), (T6e), and (T6f) are not relevant. If  $X'$  was inferred by rule (T6a) then  $X$  can directly be reduced by applying rule (N2). The existence of the premises in  $T$  follows again from (\*).

If  $X'$  was inferred by rules (T6d), then  $X'$  is of the form  $C(f)$  and thus  $T_a = \emptyset$ . If the individual  $e$  in the premise is in  $\mathbf{I}$ , then  $X$  again can be reduced by rule (N2). If  $e \notin \mathbf{I}$ , set  $a' = e$  and use rule (N1) to compute  $T_a = \{\{f\}(e), C(e)\}$ . Apply (N2) to reduce  $X$ .

- If  $X'$  was inferred by rule (T11), then  $X' = C(g)$  for some element  $g$ , and there is some element  $e$  such that  $\{\forall R.C(e), R(e, g)\} \subseteq \bar{T}$ . We distinguish two cases:
  - If  $g \in \mathbf{I}$ , then  $X = C(g)$  and  $T_a = \emptyset$ . Set  $a' = e$  and use rule (N1) to compute  $T_a = \{\forall R.C(a), R(a, g)\}$ . Use rule (N2) to reduce  $X$ .
  - If  $g \notin \mathbf{I}$ , then  $X = C(a)$  and  $e \neq a'$ . If  $e \in \mathbf{I} \cup \{b'\}$ , then  $\{\forall R.C(e), R(e, a)\} \subseteq T$  by (\*). Use rule (N2) to reduce  $X$ . If  $e \notin \mathbf{I} \cup \{b'\}$ , then  $b' = \square$  and  $T_b = \emptyset$ , as we will show below. Set  $b' = e$  and use rule (N1) to compute  $T_b = \{\forall R.C(b), R(b, g)\}$ . Use rule (N2) to reduce  $X$ .

We claimed that  $b' = \square$  whenever it is not equal to the predecessor  $e$ . This is so, since  $a' \notin \mathbf{I}$  is ensured by each step of the algorithm, and since elements that are not in  $\mathbf{I}$  are involved in active role statements of exactly one predecessor (the one which generated  $a'$ ). This is easily verified by inspecting the rules that can create role statements.

- If  $X'$  was inferred by rule ( $\exists$ ), we have  $X' = R(e, g)$  for some newly introduced element  $g \notin \mathbf{I}$ . Thus  $X$  is of the form  $R(e', a)$ , and, since  $X$  was selected to be  $\prec$ -maximal,  $T_a = \{X\}$ . Thus we can apply rule (N3) to reduce  $X$ . In addition, the algorithm applies rule (4) to copy  $b$  to the (now empty)  $a$ , and we set  $a' := b'$  and  $b' := \square$ .

With the above choices, the algorithm instantiates elements  $a$  on demand, and repeatedly reduces the statements of those elements. The individual rules show that this reduction might require another (predecessor) individual  $b$  to be considered, but that no further element is needed. Also note that rule (T9b) is required to ensure that all concept expressions in  $T_a$  can be reduced without generating any role successors for  $a$ . Hence, it is evident that the above choice strategy ensures that exactly one of the above reductions is applicable in each step.

Finally, we need to show that the algorithm terminates. This claim is established by defining a well-founded termination order. For details on such approaches and the related terminology, see [4]. Now considering  $T$  as a multiset, the multiset-extension of the well-founded order  $\prec$  is a suitable termination order, which is easy

to see since in every reduction step, the element  $X$  is deleted, and possibly replaced by one or more elements that are strictly smaller than  $X$ .  $\square$

The above lemmata establish an  $\text{NPSpace}$  decision procedure for detecting unsatisfiability of Horn- $\mathcal{FLOH}^-$  knowledge bases. But  $\text{NPSpace}$  is known to coincide with  $\text{PSPACE}$ , and we can conclude the main theorem of this section.

**Theorem 4.14** *Unsatisfiability of a Horn- $\mathcal{FLOH}^-$  knowledge base  $KB$  can be decided in space that is polynomially bounded by the size of  $KB$ .*

**Proof:** Combine the lemmata 4.11, 4.12, and 4.13 to obtain a nondeterministic time-polynomial decision procedure for detecting unsatisfiability. Apply *Savitch's Theorem to do: reference* to show the existence of an according  $\text{PSPACE}$  algorithm.  $\square$

!!!!

Summing up the result from the previous two sections, we obtain the following.

**Theorem 4.15** *Deciding knowledge base satisfiability in any description logic between Horn- $\mathcal{FL}^-$  and Horn- $\mathcal{FLOH}^-$  is  $\text{PSPACE}$ -complete.*

**Proof:** Combine Lemma 4.5 and Theorem 4.14.  $\square$

## 5 Horn- $\mathcal{FL}\mathcal{E}$

$\mathcal{FL}\mathcal{E}$  further extends  $\mathcal{FL}^-$  by allowing arbitrary existential role quantifications, which turns out to raise the complexity of Horn- $\mathcal{FL}\mathcal{E}$  to  $\text{ExpTime}$ . Note that inclusion in  $\text{ExpTime}$  is obvious since  $\mathcal{FL}\mathcal{E}$  is a fragment of  $\mathcal{SHIQ}$  which is also in  $\text{ExpTime}$  [24]. To show that Horn- $\mathcal{FL}\mathcal{E}$  is  $\text{ExpTime}$ -hard, we reduce the halting problem of polynomially space-bounded alternating Turing machines, defined next, to the concept subsumption problem.

### 5.1 Alternating Turing machines

**Definition 5.1** *An alternating Turing machine (ATM)  $\mathcal{M}$  is a tuple  $(Q, \Sigma, \Delta, q_0)$  where*

- $Q = U \dot{\cup} E$  is the disjoint union of a finite set of universal states  $U$  and a finite set of existential states  $E$ ,
- $\Sigma$  is a finite alphabet that includes a blank symbol  $\square$ ,
- $\Delta \subseteq (Q \times \Sigma) \times (Q \times \Sigma \times \{l, r\})$  is a transition relation, and
- $q_0 \in Q$  is the initial state.

A (universal/existential) configuration of  $\mathcal{M}$  is a word  $\alpha \in \Sigma^* Q \Sigma^* (\Sigma^* U \Sigma^* / \Sigma^* E \Sigma^*)$ . A configuration  $\alpha'$  is a successor of a configuration  $\alpha$  if one of the following holds:

- (1)  $\alpha = w_l q \sigma \sigma_r w_r$ ,  $\alpha' = w_l \sigma' q' \sigma_r w_r$ , and  $(q, \sigma, q', \sigma', r) \in \Delta$ ,
- (2)  $\alpha = w_l q \sigma$ ,  $\alpha' = w_l \sigma' q' \square$ , and  $(q, \sigma, q', \sigma', r) \in \Delta$ ,
- (3)  $\alpha = w_l \sigma_l q \sigma w_r$ ,  $\alpha' = w_l q' \sigma_l \sigma' w_r$ , and  $(q, \sigma, q', \sigma', l) \in \Delta$ ,

where  $q \in Q$  and  $\sigma, \sigma', \sigma_l, \sigma_r \in \Sigma$  as well as  $w_l, w_r \in \Sigma^*$ . Given some natural number  $s$ , the possible transitions in space  $s$  are defined by additionally requiring that  $|\alpha'| \leq s + 1$ .

The set of accepting configurations is the least set which satisfies the following conditions. A configuration  $\alpha$  is accepting iff

- $\alpha$  is a universal configuration and all its successor configurations are accepting, or
- $\alpha$  is an existential configuration and at least one of its successor configurations is accepting.

Note that universal configurations without any successors here play the rôle of accepting final configurations, and thus form the basis for the recursive definition above.

$\mathcal{M}$  accepts a given word  $w \in \Sigma^*$  (in space  $s$ ) iff the configuration  $q_0 w$  is accepting (when restricting to transitions in space  $s$ ).

This definition is inspired by the complexity classes NP and co-NP, which are characterised by non-deterministic Turing machines that accept an input if either at least one or all possible runs lead to an accepting state. An ATM can switch between these two modes and indeed turns out to be more powerful than classical Turing machines of either kind. In particular, ATMs can solve EXP TIME problems in polynomial space [7].

**Definition 5.2** A language  $L$  is accepted by a polynomially space-bounded ATM iff there is a polynomial  $p$  such that, for every word  $w \in \Sigma^*$ ,  $w \in L$  iff  $w$  is accepted in space  $p(|w|)$ .

**Fact 1** The complexity class APSPACE of languages accepted by polynomially space-bounded ATMs coincides with the complexity class EXP TIME.

We thus can show EXP TIME-hardness of Horn-SHIQ by polynomially reducing the halting problem of ATMs with a polynomially bounded storage space to inferencing in Horn-SHIQ. In the following, we exclusively deal with polynomially space-bounded ATMs, and so we omit additions such as “in space  $s$ ” when clear from the context.

## 5.2 Simulating ATMs in Horn- $\mathcal{FL}\mathcal{E}$

In the following, we consider a fixed ATM  $\mathcal{M}$  denoted as in Definition 5.1, and a polynomial  $p$  that defines a bound for the required space. For any word  $w \in \Sigma^*$ , we construct a Horn- $\mathcal{FL}\mathcal{E}$  knowledge base  $K_{\mathcal{M},w}$  and show that acceptance of  $w$  by the ATM  $\mathcal{M}$  can be decided by inferencing over this knowledge base.

In detail,  $K_{\mathcal{M},w}$  depends on  $\mathcal{M}$  and  $p(|w|)$ , and has an empty ABox.<sup>7</sup> Acceptance of  $w$  by the ATM is reduced to checking concept subsumption, where one of the involved concepts directly depends on  $w$ . Intuitively, the elements of an interpretation domain of  $K_{\mathcal{M},w}$  represent possible configurations of  $\mathcal{M}$ , encoded by the following concept names:

- $A_q$  for  $q \in Q$ : the ATM is in state  $q$ ,
- $H_i$  for  $i = 0, \dots, p(|w|) - 1$ : the ATM is at position  $i$  on the storage tape,
- $C_{\sigma,i}$  with  $\sigma \in \Sigma$  and  $i = 0, \dots, p(|w|) - 1$ : position  $i$  on the storage tape contains symbol  $\sigma$ ,
- $A$ : the ATM accepts this configuration.

This approach is pretty standard, and it is not too hard to axiomatise a successor relation  $S$  and appropriate acceptance conditions in  $\mathcal{ALC}$  (see, e.g., [19]). But this reduction is not applicable in Horn- $\mathcal{SHIQ}$ , and it is not trivial to modify it accordingly.

One problem that we encounter is that the acceptance condition of existential states is a (non-Horn) disjunction over possible successor configurations. To overcome this, we encode individual transitions by using a distinguished successor relation for each translation in  $\Delta$ . This allows us to explicitly state which conditions must hold for a particular successor without requiring disjunction. For the acceptance condition, we use a recursive formulation as employed in Definition 5.1. In this way, acceptance is propagated backwards from the final accepting configurations.

In the case of  $\mathcal{ALC}$ , acceptance of the ATM is reduced to concept satisfiability, i.e. one checks whether an accepting initial configuration can exist. This requires that acceptance is faithfully propagated to successor states, so that any model of the initial concept encodes a valid trace of the ATM. Axiomatising this requires many exclusive disjunctions, such as “The ATM always is in *exactly* one of its states  $H_i$ .” Since it is not clear how to model this in a Horn-DL, we take a dual approach: reducing acceptance to concept subsumption, we require the initial state to be accepting in *all* possible models. We therefore may focus on the task of propagating properties to successor configurations, while not taking care of disallowing additional statements to hold. Our encoding ensures that, whenever the initial configuration is not accepting, there is at least one “minimal” model that reflects this.

<sup>7</sup> The RBox is empty for  $\mathcal{FL}\mathcal{E}$  anyway.

Table 14

Knowledge base  $K_{\mathcal{M},w}$  simulating a polynomially space-bounded ATM. The rules are instantiated for all  $q, q' \in Q$ ,  $\sigma, \sigma' \in \Sigma$ ,  $i, j \in \{0, \dots, p(|w|) - 1\}$ , and  $\delta \in \Delta$ .

**(1) Left and right transition rules:**

$$A_q \sqcap H_i \sqcap C_{\sigma,i} \sqsubseteq \exists S_{\delta}.(A_{q'} \sqcap H_{i+1} \sqcap C_{\sigma',i}) \quad \text{with } \delta = (q, \sigma, q', \sigma', r), i < p(|w|) - 1$$

$$A_q \sqcap H_i \sqcap C_{\sigma,i} \sqsubseteq \exists S_{\delta}.(A_{q'} \sqcap H_{i-1} \sqcap C_{\sigma',i}) \quad \text{with } \delta = (q, \sigma, q', \sigma', l), i > 0$$

**(2) Memory:**

$$H_j \sqcap C_{\sigma,i} \sqsubseteq \forall S_{\delta}.C_{\sigma,i} \quad i \neq j$$

**(3) Existential acceptance:**

$$A_q \sqcap \exists S_{\delta}.A \sqsubseteq A \quad \text{for all } q \in E$$

**(4) Universal acceptance:**

$$A_q \sqcap H_i \sqcap C_{\sigma,i} \sqcap \prod_{\delta \in \tilde{\Delta}} (\exists S_{\delta}.A) \sqsubseteq A \quad q \in U, x \in \{r \mid i < p(|w|) - 1\} \cup \{l \mid i > 0\}$$

$$\tilde{\Delta} = \{(q, \sigma, q', \sigma', x) \in \Delta\}$$

After this informal introduction, consider the knowledge base  $K_{\mathcal{M},w}$  given in Table 14. The roles  $S_{\delta}$ ,  $\delta \in \Delta$ , describe a configuration's successors using the transition  $\delta$ . The initial configuration for word  $w$  is described by the concept expression  $I_w$ :

$$I_w := A_{q_0} \sqcap H_0 \sqcap C_{\sigma_0,0} \sqcap \dots \sqcap C_{\sigma_{|w|-1},|w|-1} \sqcap C_{\square,|w|} \sqcap \dots \sqcap C_{\square,p(|w|)-1},$$

where  $\sigma_i$  denotes the symbol at the  $i$ th position of  $w$ . We will show that checking whether the initial configuration is accepting is equivalent to checking whether  $I_w \sqsubseteq A$  follows from  $K_{\mathcal{M},w}$ . The following is obvious from the characterisation given in Table 2.

**Lemma 5.3**  $K_{\mathcal{M},w}$  and  $I_w \sqsubseteq A$  are in Horn- $\mathcal{FL}\mathcal{E}$ .

Next we need to investigate the relationship between elements of an interpretation that satisfies  $K_{\mathcal{M},w}$  and configurations of  $\mathcal{M}$ . Given an interpretation  $\mathcal{I}$  of  $K_{\mathcal{M},w}$ , we say that an element  $e$  of the domain of  $\mathcal{I}$  represents a configuration  $\sigma_1 \dots \sigma_{i-1} q \sigma_i \dots \sigma_m$  if  $e \in A_q^{\mathcal{I}}$ ,  $e \in H_i^{\mathcal{I}}$ , and, for every  $j \in \{0, \dots, p(|w|) - 1\}$ ,  $e \in C_{\sigma,j}^{\mathcal{I}}$  whenever

$$j \leq m \text{ and } \sigma = \sigma_m \quad \text{or} \quad j > m \text{ and } \sigma = \square.$$

Note that we do not require uniqueness of the above, so that a single element might in fact represent more than one configuration. As we will see below, this does not affect our results. If  $e$  represents a configuration as above, we will also say that  $e$  has state  $q$ , position  $i$ , symbol  $\sigma_j$  at position  $j$  etc.

**Lemma 5.4** Consider some interpretation  $\mathcal{I}$  that satisfies  $K_{\mathcal{M},w}$ . If some element  $e$  of  $\mathcal{I}$  represents a configuration  $\alpha$  and some transition  $\delta$  is applicable to  $\alpha$ , then  $e$  has an  $S_{\delta}^{\mathcal{I}}$ -successor that represents the (unique) result of applying  $\delta$  to  $\alpha$ .

**Proof:** Consider an element  $e$ , state  $\alpha$ , and transition  $\delta$  as in the claim. Then one of the axioms (1) applies, and  $e$  must also have an  $S_\delta^{\mathcal{I}}$ -successor. This successor represents the correct state, position, and symbol at position  $i$  of  $e$ , again by the axioms (1). By axiom (2), symbols at all other positions are also represented by all  $S_\delta^{\mathcal{I}}$ -successors of  $e$ .  $\square$

**Lemma 5.5** *A word  $w$  is accepted by  $\mathcal{M}$  iff  $I_w \sqsubseteq A$  is a consequence of  $K_{\mathcal{M},w}$ .*

**Proof:** Consider an arbitrary interpretation  $\mathcal{I}$  that satisfies  $K_{\mathcal{M},w}$ . We first show that, if any element  $e$  of  $\mathcal{I}$  represents an accepting configuration  $\alpha$ , then  $e \in A^{\mathcal{I}}$ .

We use an inductive argument along the recursive definition of acceptance. If  $\alpha$  is a universal configuration then all successors of  $\alpha$  are accepting, too. By Lemma 5.4, for any  $\delta$ -successor  $\alpha'$  of  $\alpha$  there is a corresponding  $S_\delta^{\mathcal{I}}$ -successor  $e'$  of  $e$ . By the induction hypothesis for  $\alpha'$ ,  $e'$  is in  $A^{\mathcal{I}}$ . Since this holds for all  $\delta$ -successors of  $\alpha$ , axiom (4) implies  $e \in A^{\mathcal{I}}$ . Especially, this argument covers the base case where  $\alpha$  has no successors.

If  $\alpha$  is an existential configuration, then there is some accepting  $\delta$ -successor  $\alpha'$  of  $\alpha$ . Again by Lemma 5.4, there is an  $S_\delta^{\mathcal{I}}$ -successor  $e'$  of  $e$  that represents  $\alpha'$ , and  $e' \in A^{\mathcal{I}}$  by the induction hypothesis. Hence axiom (3) applies and also conclude  $e \in A^{\mathcal{I}}$ .

Since all elements in  $I_w^{\mathcal{I}}$  represent the initial configuration of the ATM, this shows that  $I_w^{\mathcal{I}} \subseteq A^{\mathcal{I}}$  whenever the initial configuration is accepting.

It remains to show the converse: if the initial configuration is not accepting, there is some interpretation  $\mathcal{I}$  such that  $I_w^{\mathcal{I}} \not\subseteq A^{\mathcal{I}}$ . To this end, we define a canonical interpretation  $M$  of  $K_{\mathcal{M},w}$  as follows. The domain of  $M$  is the set of all configurations of  $\mathcal{M}$  that have size  $p(|w|) + 1$  (i.e. that encode a tape of length  $p(|w|)$ , possibly with trailing blanks). The interpretations for the concepts  $A_q$ ,  $H_i$ , and  $C_{\sigma,i}$  are defined as expected so that every configuration represents itself but no other configuration. Especially,  $I_w^M$  is the singleton set containing the initial configuration. Given two configurations  $\alpha$  and  $\alpha'$ , and a transition  $\delta$ , we define  $(\alpha, \alpha') \in S_\delta^M$  iff there is a transition  $\delta$  from  $\alpha$  to  $\alpha'$ .  $A^M$  is defined to be the set of accepting configurations.

By checking the individual axioms of Table 14, it is easy to see that  $M$  satisfies  $K_{\mathcal{M},w}$ . Now if the initial configuration is not accepting,  $I_w^M \not\subseteq A^M$  by construction. Thus  $M$  is a counterexample for  $I_w \sqsubseteq A$  which thus is not a logical consequence.  $\square$

We can summarise our results as follows.

**Theorem 5.6** *Checking concept subsumption in any of the description logics between Horn- $\mathcal{FL}\mathcal{E}$  and Horn- $\mathcal{SHIQ}$  is EXP-TIME-complete.*

**Proof:** Inclusion is obvious as Horn- $\mathcal{SHIQ}$  is a fragment of  $\mathcal{ALC}$ , which is in

EXPTIME. Regarding hardness, Lemma 5.5 shows that the word problem for polynomially space-bounded ATMs can be reduced to checking concept subsumption in  $K_{\mathcal{M},w}$ . By Lemma 5.3,  $K_{\mathcal{M},w}$  is in Horn- $\mathcal{FL}\mathcal{E}$ . The reduction is polynomially bounded due to the restricted number of axioms: there are at most  $2 \times |Q| \times p(|w|) \times |\Sigma| \times |\Delta|$  axioms of type (1),  $p(|w|)^2 \times |\Sigma| \times |\Delta|$  of type (2),  $|Q| \times |\Sigma|$  of type (3), and  $|Q| \times p(|w|) \times |\Sigma|$  of type (4).  $\square$

Note that, even in Horn logics, it is straightforward to reduce knowledge base satisfiability to the entailment of the concept subsumption  $\top \sqsubseteq \perp$ . The proof that was used to establish the previous result is suitable for obtaining further complexity results for logical fragments that are not above Horn- $\mathcal{FL}\mathcal{E}$ .

**Theorem 5.7** (a) Let  $\mathcal{EL}^{\leq 1}$  denote  $\mathcal{EL}$  extended with number restrictions of the form  $\leq 1 R.\top$ .  
(b) Let  $\mathcal{FL}^{\circ-}$  denote  $\mathcal{FL}^-$  extended with composition of roles.  
(c) Let  $\mathcal{FLI}^-$  denote  $\mathcal{FL}^-$  extended with inverse roles.

Horn- $\mathcal{FL}^{\circ-}$  is EXPTIME-hard, and both Horn- $\mathcal{EL}^{\leq 1}$  and Horn- $\mathcal{FLI}^-$  are EXPTIME-complete.

**Proof:** The results are established by modifying the knowledge base  $K_{\mathcal{M},w}$  to suite the given fragment. We restrict to providing the required modifications; the full proofs are analogous to the proof for Horn- $\mathcal{FL}\mathcal{E}$ .

(a) Replace axioms (2) in Table 14 with the following statements:

$$\top \sqsubseteq \leq 1 S_{\delta}.\top \quad H_j \sqcap C_{\sigma,i} \sqcap \exists S_{\delta}.\top \sqsubseteq \exists S_{\delta}.C_{\sigma,i}, \quad i \neq j$$

(b) Replace axioms (1) with axioms of the form

$$A_q \sqcap H_i \sqcap C_{\sigma,i} \sqsubseteq \exists S_{\delta}.\top \sqcap \forall S_{\delta}.(A_{q'} \sqcap H_{i\pm 1} \sqcap C_{\sigma',i}).$$

Any occurrence of concept  $A$  is replaced by  $\exists R_A.\top$ , with  $R_A$  a new role. Moreover, we introduce roles  $R_{A\delta}$  for each transition  $\delta$ , and replace any occurrence of  $\exists S_{\delta}.A$  with  $\exists R_{A\delta}.\top$ . Finally, the following axioms are added:

$$S_{\delta} \circ R_A \sqsubseteq R_{A\delta} \quad \text{for each } \delta \in \Delta.$$

(c) Axioms (1) are replaced as in (b). Any occurrence of  $\exists S_{\delta}.A$  is now replaced with a new concept name  $A_{S_{\delta}}$ , and the following axioms are added:

$$A \sqsubseteq \forall S_{\delta}^{-1}.A_{S_{\delta}} \quad \text{for each } \delta \in \Delta.$$

It is easy to see that those changes still enable analogous reductions. Inclusion results for Horn- $\mathcal{EL}^{\leq 1}$  and for Horn- $\mathcal{FLI}^-$  are immediate from their inclusion in  $\mathcal{SHIQ}$ .  $\square$

EXPTIME-completeness of  $\mathcal{EL}^{\leq 1}$  was shown in [1], but the above theorem sharpens this result to the Horn case, and provides a more direct proof. Theorems 5.6 and 5.7 thus can be viewed as sharpenings of the hardness results on extensions of  $\mathcal{EL}$ .

## 6 Related Work

To the best of our knowledge, this paper presents the first systematic treatment of Horn-DLs, and thus opens the door for investigating them further in the tradition of DL research. Only some of the Horn-DLs mentioned herein have been identified before in the literature, the most important of which are under investigation for the revision of the OWL standard in the OWL Working Group of the World Wide Web Consortium, as already mentioned.

To this date, there are only very few – and rather recent – publications which take Hornness of these DLs as the decisive feature for further investigations. We are able to identify the tractable DLs  $\mathcal{EL}$  [1] and DL-Lite [6] as Horn-DLs, but they have originally been conceived as tractable description logics, and it remains to be investigated to what extent their recognition as Horn-DLs can be exploited.

For Horn-*SHIQ*, which has already been conceived as a Horn-DL, there is a body of work which draws on its Hornness. The HerMiT<sup>8</sup> reasoner [21], which is based on Hypertableaux, is particularly efficient when dealing with Horn-*SHIQ*. Another line of investigations relates Horn-*SHIQ* to logic programming, showing that theoretically there is a natural way of reasoning with it within Prolog systems [16]. As initial experiments show, however, off-the-shelf Prolog systems are not optimised to handle the kind of knowledge bases which arise from translating Horn-*SHIQ* to Prolog [22]. This point remains to be investigated further, as it appears to be reasonable that algorithms tailored to Hornness, like SLG resolution [8], should be adaptable for efficient handling of such knowledge bases. Another line of research aims at approximate reasoning with OWL through approximations of knowledge bases by Horn-*SHIQ*, and evaluations show that a favorable trade-off between efficiency of reasoning and errors introduced through the approximation can be achieved [13].

Historically, Description Logic Programs (DLPs) [10,26] initiated the investigation of Hornness for DLs. They provide a naive bridge between description logics and rule-based systems and are therefore of importance whenever relations between these two paradigms are investigated. As such, DLPs have also been part of the Web Rule Language WRL [5,9] which has been an initial input for the currently ongoing RIF Working Group<sup>9</sup> of the W3C on standardising a rule interchange format. DLPs have failed in becoming a widely recognised stand-alone ontology language, however, despite some efforts in that direction. Nevertheless, some prominent ontologies, such as the SWRC [23], are DLP-ontologies, and also the usefulness, in principle, of that fragment for relating rules and OWL is obvious.

<sup>8</sup> <http://www.cs.man.ac.uk/~bmotik/HermiT/>

<sup>9</sup> [http://www.w3.org/2005/rules/wiki/RIF\\_Working\\_Group](http://www.w3.org/2005/rules/wiki/RIF_Working_Group)

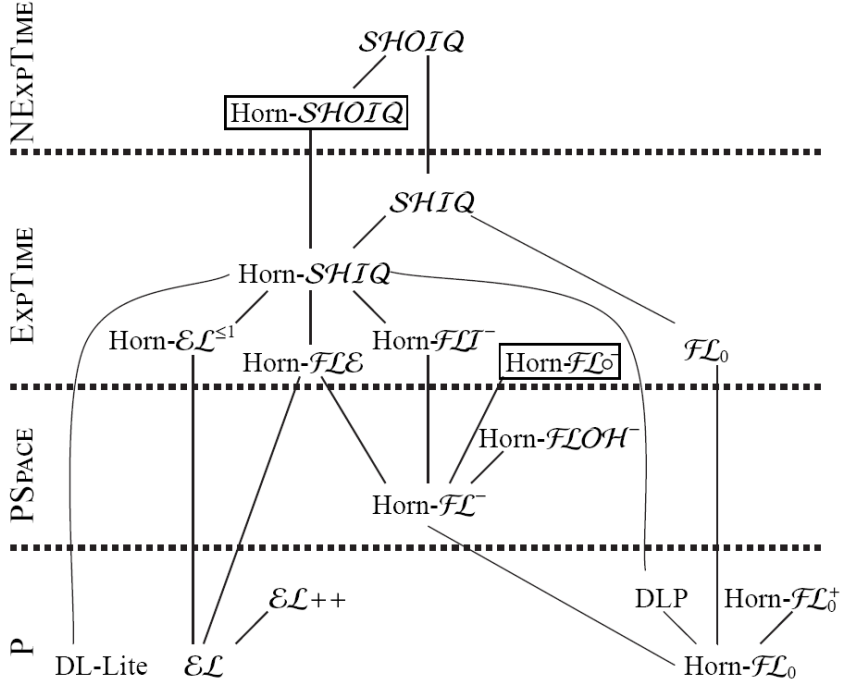


Fig. 1. Overview over complexity boundaries for combined complexity of (Horn) description logics in the presence of GCIs. The exact location of the boxed DLs is conjectured, and has not been shown yet.

## 7 Conclusions and Further Work

Horn logics, while having a long tradition in logic programming, have only recently been studied in the context of description logics, mainly due to their lower data complexities. In this paper, we have investigated the effects of Hornness on the overall complexity of DL reasoning, and we have shown that only the Horn fragments of certain subboolean description logics are actually less complex than their non-Horn versions. On the other hand, the well-known tractable DLs  $\mathcal{EL}^{++}$  [1], DL-Lite [6] and DLP [10] are also recognised as (fragments of) Horn-logics, and we thus obtain a unified picture of combined complexities of some of the most important tractable DLs currently discussed, e.g. in the context of the currently undergoing revision of the OWL standard by the World Wide Web Consortium. The main results of our work are summarised in Figure 1.

While most of the displayed relationships have been verified above, Figure 1 also includes two open conjectures that are left to future research:  $Horn-SHOIQ$  might also turn out to be in EXPTIME, whereas  $Horn-FL_0^-$  could even be NEXPTIME-hard. In addition, some related results have not been included in Figure 1. In particular, we have shown that (Horn) disjunction and atomic negation never increase the complexity of Horn-logics. Accordingly, the extension of  $\mathcal{EL}^{++}$  to  $Horn-\mathcal{EL}^{(n)++}$  is still tractable, while it was shown in [1] that both  $\mathcal{EL}^{(n)}$  and  $\mathcal{EL}^{(n)}$  are EXPTIME-complete. An interesting application of this extension is the use of DLs as query

languages, since the use of disjunctions is not constrained within queries (which are treated as negated axioms) at all. This is exploited, for instance, in the semantic search implementation of Semantic MediaWiki [25,18], which indeed supports a (syntactically adopted) fragment of Horn- $\mathcal{ELU}++$  for querying large scale knowledge bases.

Our results on Horn- $\mathcal{FL}\mathcal{E}$  and Horn- $\mathcal{EL}^{\leq 1}$  sharpen the known results on the non-extensibility of  $\mathcal{EL}$ . On the other hand, various expressive extensions that are known not to increase the complexity of  $\mathcal{EL}$  were also shown to be tolerable in the case of Horn- $\mathcal{FL}_0$  and Horn- $\mathcal{FL}^-$ . In particular, nominals and role hierarchies have had no negative effect on the worst-case complexity of any of the investigated Horn-logics. Yet, it is apparent from the technically more involved proof of Horn- $\mathcal{FLOH}^-$ 's PSPACE-completeness that especially nominals can easily make the reasoning task more complicated. As all other proofs, this proof is established directly, without referring to existing complexity results. While this is often increasing the length of the required argumentation, we believe that direct proofs are often most instructive for analysing the source of increased complexities.

It appears that further systematic studies of Horn-DLs should allow to leverage the considerable body of research undertaken on Horn logic, logic programming, and rule systems for description logics and for the Semantic Web in a fruitful way. They promise to allow a transfer of results, methods, and technologies which should be beneficial in terms of theoretical insights and also in the development of practically useful ontology management systems, in order to bring the Semantic Web vision yet another step closer to its realisation.

## References

- [1] F. Baader, S. Brandt, C. Lutz, Pushing the EL envelope, in: Proc. 19th Int. Joint Conf. on Artificial Intelligence (IJCAI-05), Morgan-Kaufmann Publishers, Edinburgh, UK, 2005.
- [2] F. Baader, D. Calvanese, D. McGuinness, D. Nardi, P. Patel-Schneider (eds.), The Description Logic Handbook: Theory, Implementation and Applications, Cambridge University Press, 2003.
- [3] F. Baader, C. Lutz, B. Suntisrivaraporn, Is tractable reasoning in extensions of the description logic  $\mathcal{EL}$  useful in practice?, in: Journal of Logic, Language and Information, Special Issue on Method for Modality (M4M), 2007, to appear.
- [4] F. Baader, T. Nipkow, Term Rewriting and All That, Cambridge University Press, 1998.
- [5] H. Boley, J. de Bruijn, D. Fensel, P. Hitzler, M. Kifer, R. Krümmenacher, H. Lausen, A. Polleres, R. Studer, Web Rule Language (WRL), W3C Member Submission, 09 September 2005 (2005).

- [6] D. Calvanese, G. D. Giacomo, D. Lembo, M. Lenzerini, R. Rosati, Tractable reasoning and efficient query answering in description logics: The DL-Lite family, *J. Autom. Reasoning* 39 (3) (2007) 385–429.
- [7] A. K. Chandra, D. C. Kozen, L. J. Stockmeyer, Alternation, *J. ACM* 28 (1) (1981) 114–133.
- [8] W. Chen, D. S. Warren, Tabled evaluation with delaying for general logic programs, *J. ACM* 43 (1) (1996) 20–74.
- [9] J. de Bruijn, D. Fensel, P. Hitzler, M. Kifer, A. Polleres, Relationship of WRL to relevant other technologies, W3C Member Submission, 09 September 2005 (2005).
- [10] B. Groszof, I. Horrocks, R. Volz, S. Decker, Description logic programs: Combining logic programs with description logics, in: *Proc. of WWW 2003, Budapest, Hungary, May 2003*, ACM, 2003.
- [11] P. Hitzler, A. Eberhart, Description Logic Programs: Normal forms, in: P. Hitzler, T. Roth-Berghofer, S. Rudolph (eds.), *FAInt-07, Foundations of Artificial Intelligence, Workshop at KI 2007*, vol. 277 of *CEUR Workshop Proceedings*, 2007.
- [12] P. Hitzler, B. Parsia, Ontologies and Rules, in: S. Staab, R. Studer (eds.), *Handbook on Ontologies*, Springer, Berlin, 2008, to appear.
- [13] P. Hitzler, D. Vrandečić, Resolution-based approximate reasoning for OWL DL, in: Y. G. et al. (ed.), *Proceedings of the 4th International Semantic Web Conference, Galway, Ireland, November 2005*, vol. 3729 of *Lecture Notes in Computer Science*, Springer, Berlin, 2005.
- [14] I. Horrocks, O. Kutz, U. Sattler, The even more irresistible SROIQ, in: *Proc. of the 10th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR2006)*, AAAI Press, 2006.
- [15] U. Hustadt, B. Motik, U. Sattler, Data complexity of reasoning in very expressive description logics, in: *Proc. 18th Int. Joint Conf. on Artificial Intelligence (IJCAI-05)*, Morgan-Kaufmann Publishers, Edinburgh, UK, 2005.
- [16] M. Krötzsch, P. Hitzler, D. Vrandečić, M. Sintek, How to reason with OWL in a logic programming system, in: *Proc. 2nd Int. Conf. on Rules and Rule Markup Languages for the Semantic Web (RuleML 2006)*, Springer, Athens, Georgia, USA, 2006.
- [17] M. Krötzsch, S. Rudolph, P. Hitzler, Complexity boundaries for Horn description logics, in: *Proceedings of the 22nd AAAI Conference on Artificial Intelligence*, AAAI Press, Vancouver, British Columbia, Canada, 2007.
- [18] M. Krötzsch, D. Vrandečić, M. Völkel, H. Haller, R. Studer, Semantic Wikipedia, *Journal of Web Semantics* 5 (2007) 251–261.
- [19] C. Lutz, U. Sattler, Description logics, tutorial at the ICCL Summer School 2005, Dresden, Germany. Course material available online. (2005).
- [20] B. Motik, Reasoning in description logics using resolution and deductive databases, Ph.D. thesis, Universität Karlsruhe (TH), Germany (2006).

- [21] B. Motik, R. Shearer, I. Horrocks, Optimized Reasoning in Description Logics using Hypertableaux, in: F. Pfenning (ed.), Proc. of the 21st Conference on Automated Deduction (CADE-21), vol. 4603 of LNAI, Springer, Bremen, Germany, 2007.
- [22] S. Rudolph, M. Krötzsch, P. Hitzler, M. Sintek, D. Vrandečić, Efficient OWL reasoning with logic programs – evaluations, in: M. Marchiori, J. Z. Pan, C. de Sainte Marie (eds.), Proceedings of The First International Conference on Web Reasoning and Rule Systems 2007 (RR2007), vol. 4524 of Springer Lecture Notes in Computer Science, Springer, 2007.
- [23] Y. Sure, S. Bloehdorn, P. Haase, J. Hartmann, D. Oberle, The SWRC ontology - Semantic Web for Research Communities, in: C. Bento, A. Cardoso, G. Dias (eds.), Proceedings of the 12th Portuguese Conference on Artificial Intelligence - Progress in Artificial Intelligence (EPIA 2005), vol. 3803 of LNCS, Springer, Covilha, Portugal, 2005.
- [24] S. Tobies, Complexity results and practical algorithms for logics in knowledge representation, Ph.D. thesis, RWTH Aachen, Germany (2001).
- [25] M. Völkel, M. Krötzsch, D. Vrandečić, H. Haller, R. Studer, Semantic Wikipedia, in: Proc. of the 15th International WWW Conference, Edinburgh, Scotland., 2006.
- [26] R. Volz, Web ontology reasoning with logic databases, Ph.D. thesis, Universität Karlsruhe (TH), Universität Karlsruhe (TH), Institut AIFB, D-76128 Karlsruhe (2004).