Data & Knowledge Engineering 96–97 (2015) 57–77

Contents lists available at ScienceDirect

Data & Knowledge Engineering

journal homepage: www.elsevier.com/locate/datak

Editorial

# Towards richer rule languages with polynomial data complexity for the Semantic Web

Linh Anh Nguyen [a,b,*], Thi-Bich-Loc Nguyen [c], Andrzej Szałas [b,d]

[a] Division of Knowledge and System Engineering for ICT, Faculty of Information Technology, Ton Duc Thang University, Ho Chi Minh City, Viet Nam
[b] Institute of Informatics, University of Warsaw, Banacha 2, 02-097 Warsaw, Poland
[c] Department of Information Technology, Hue University of Sciences, 77 Nguyen Hue, Hue city, Viet Nam
[d] Department of Computer and Information Science, Linköping University, SE-581 83 Linköping, Sweden

ARTICLE INFO

Available online 11 April 2015

*Keywords:*
Description logics
Semantic Web
Rule-based languages
Horn fragments
Horn-DL

ABSTRACT

We introduce a Horn description logic called Horn-DL, which is strictly and essentially richer than Horn-$\mathcal{R}eg^{I}$, Horn-$\mathcal{SHIQ}$ and Horn-$\mathcal{SROIQ}$, while still has PTime data complexity. In comparison with Horn-$\mathcal{SROIQ}$, Horn-DL additionally allows the universal role and assertions of the form irreflexive$(s)$, $\neg s(a, b)$, $a \dot{\neq} b$. More importantly, in contrast to all the well-known Horn fragments $\mathcal{EL}$, DL-Lite, DLP, Horn-$\mathcal{SHIQ}$, and Horn-$\mathcal{SROIQ}$ of description logics, Horn-DL allows a form of the concept constructor "universal restriction" to appear at the left hand side of terminological inclusion axioms. Namely, a universal restriction can be used in such places in conjunction with the corresponding existential restriction. We develop the first algorithm with PTime data complexity for checking satisfiability of Horn-DL knowledge bases.


## 1. Introduction

The Semantic Web is a rapidly growing research area that has received lots of attention in the last decade (e.g., [39,13,14,36,1,31]). One of the layers of the Semantic Web is OWL (Web Ontology Language). The first version of OWL is based on the description logic $\mathcal{SHIQ}$, the second version OWL 2, recommended by W3C in 2009, is based on the description logic $\mathcal{SROIQ}$ [17]. These logics are highly expressive but have intractable combined complexity (ExpTime-complete and N2ExpTime-complete, respectively) and data complexity (NP-hard) for basic reasoning problems. Thus, W3C also recommended profiles OWL 2 EL, OWL 2 QL and OWL 2 RL, which are restricted sublanguages of OWL 2 Full with PTime data complexity.[1]

Description logics (DLs) are formal languages suitable for representing terminological knowledge. They are of particular importance in providing a logical formalism for ontologies and the Semantic Web. DLs represent the domain of interest in terms of concepts, individuals, and roles. A concept is interpreted as a set of individuals, while a role is interpreted as a binary relation among individuals. A knowledge base in a DL consists of axioms about roles (grouped into an RBox), terminology axioms (grouped into a TBox), and assertions about individuals (grouped into an ABox). A DL is usually specified by: i) a set of constructors that allow building complex concepts and complex roles from concept names, role names and individual names and ii) allowed forms of axioms and assertions. The basic DL $\mathcal{ALC}$ allows basic concept constructors listed in Table 1, but does not allow role constructors nor role axioms. The

* Corresponding author at: Division of Knowledge and System Engineering for ICT, Faculty of Information Technology, Ton Duc Thang University, No. 19, Nguyen Huu Tho Street, Tan Phong Ward, District 7, Ho Chi Minh City, Vietnam.
*E-mail addresses:* nguyenanhlinh@tdt.edu.vn, nguyen@mimuw.edu.pl (L.A. Nguyen), ntbichloc@hueuni.edu.vn (T.-B.-L. Nguyen), andsz@mimuw.edu.pl (A. Szałas).

[1] The full names of the complexity classes mentioned in this paper are as follows: PTime means *deterministic polynomial time*, NP means *nondeterministic polynomial time*, ExpTime means *deterministic exponential time*, N2ExpTime means *nondeterministic double exponential time*. The data complexity is measured in the size of the ABox, which is in the "reduced" form, while assuming that the RBox, the TBox and the query are fixed.

http://dx.doi.org/10.1016/j.datak.2015.04.005
0169-023X/

**Table 1**
Concept constructors for $\mathcal{ALC}$ and some additional constructors/features of other DLs.

| Concept constructors of $\mathcal{ALC}$ | | |
|---|---|---|
| **Constructor** | **Syntax** | **Example** |
| Complement | $\neg C$ | $\neg$ *Male* |
| Intersection | $C \sqcap D$ | *Human* $\sqcap$ *Male* |
| Union | $C \sqcap D$ | *Doctor* $\sqcup$ *Lawyer* |
| Existential restriction | $\exists r.C$ | $\exists$*hasChild.Male* |
| Universal restriction | $\forall r.C$ | $\forall$*hasChild.Female* |
| Additional constructors/features of other DLs | | |
| **Constructor/feature** | **Syntax** | **Example** |
| Inverse roles ($\mathcal{I}$) | $r^-$ | *hasChild*$^-$ (i.e., *hasParent*) |
| Qualified number | $\geq n\,R.C$ | $\geq$3*hasChild.Male* |
| Restrictions ($\mathcal{Q}$) | $\leq n\,R.C$ | $\leq$*hasParent*.$\top$ |
| Nominals ($\mathcal{O}$) | $\{a\}$ | {*John*} |
| Hierarchies of roles ($\mathcal{H}$) | $R \sqsubseteq S$ | *hasChild* $\sqsubseteq$ *hasDescendant* |
| Transitive roles ($\mathcal{S}$) | $R \circ R \sqsubseteq R$ | *hasDescendant* $\circ$ *hasDescendant* $\sqsubseteq$ *hasDescendant* |

most common additional features for extending $\mathcal{ALC}$ are also listed in Table 1: $\mathcal{I}$ is a role constructor, $\mathcal{Q}$ and $\mathcal{O}$ are concept constructors, while $\mathcal{H}$ and $\mathcal{S}$ are allowed forms of role axioms.

### 1.1. Rule languages in description logics

Rule languages in DLs have attracted lots of attention due to their applications to the Semantic Web. The profiles OWL 2 EL, OWL 2 QL and OWL 2 RL of OWL 2 are based on the families of DLs $\mathcal{EL}$ [2,3], DL-Lite [5] and DLP [16], which are monotonic rule languages with PTime data complexity. Such monotonic rule languages are Horn fragments of the corresponding full languages with appropriate restrictions adopted to eliminate nondeterminism.

A number of Horn fragments of DLs with PTime data complexity have also been investigated in [18,21,23,37,30,35]. The fragments Horn-$\mathcal{SHIQ}$ [18] and Horn-$\mathcal{SROIQ}$ [35] are notable, with considerable rich sets of allowed constructors and features, where Horn-$\mathcal{SROIQ}$ is richer than Horn-$\mathcal{SHIQ}$. The combined complexities of Horn fragments of DLs were considered, among others, in [22]. Some tractable Horn fragments of DLs without ABoxes have also been isolated in [2,4]. Combinations of rule languages like Datalog or its extensions with DLs have also been studied in a considerable number of works [8,24,26,38,20,25,9,12,6].

To eliminate nondeterminism, all $\mathcal{EL}$ [2,3], DL-Lite [5], DLP [16], Horn-$\mathcal{SHIQ}$ [18] and Horn-$\mathcal{SROIQ}$ [35] disallow (any form of) the universal restriction at the left hand side of $\sqsubseteq$ in terminological axioms. The problem is that roles are not required to be serial (i.e., satisfying the condition $\forall x \exists y\, R(x, y)$) and the general Horn fragment of the basic DL $\mathcal{ALC}$ that allows $\forall R.C$ at the left hand side of $\sqsubseteq$ has NP-complete data complexity [30].

In [28] Nguyen introduced the deterministic Horn fragment of $\mathcal{ALC}$, in which the constructor $\forall R.C$ is allowed at the left hand side of $\sqsubseteq$ in the combination with $\exists R.C$ (in the form $\forall R.C \sqcap \exists R.C$, denoted by $\forall\exists R.C$ [4]). He proved that that fragment has PTime data complexity by providing a bottom-up method for constructing a (logically) least model for a given deterministic positive knowledge base in the restricted language. In [30] Nguyen applied the method of [28] to regular DL $\mathcal{R}$eg, which extends $\mathcal{ALC}$ with regular role inclusion axioms characterized by finite automata. Let us denote the Horn fragment of $\mathcal{R}$eg that allows the constructor $\forall\exists R.C$ at the left hand side of $\sqsubseteq$ by Horn-$\mathcal{R}$eg. As not every positive Horn-$\mathcal{R}$eg knowledge base has a (logically) least model, Nguyen [30] proposed to approximate the instance checking problem in Horn-$\mathcal{R}$eg by using weakenings with PTime data complexity.

In [33] we studied a Horn fragment called Horn-$\mathcal{R}eg^I$ of the regular DL with inverse $\mathcal{R}eg^I$ and provided an algorithm with PTime data complexity for checking satisfiability of Horn-$\mathcal{R}eg^I$ knowledge bases. This fragment extends Horn-$\mathcal{R}$eg with inverse roles. Our work [33] overcomes the difficulties encountered in [28,30] by using the top-down rather than bottom-up approach, and thus enables to show that both Horn-$\mathcal{R}$eg and Horn-$\mathcal{R}eg^I$ have PTime data complexity, solving an open problem of [30].

### 1.2. Constructivism in defining rules

We now justify the usefulness of the concept constructor $\forall\exists R.C$. It is related to constructivism in defining rules [30]. Consider the TBox $\mathcal{T}$ consisting of the following rules:

$$\exists hasChild.\top \sqsubseteq Parent \tag{1}$$

$$\forall hasChild.Happy \sqsubseteq HappyParent \tag{2}$$

$$HappyParent \sqsubseteq Happy. \tag{3}$$