Checking *DL-Lite* modularity with QBF solvers

R. Kontchakov[1], V. Ryzhikov[2], F. Wolter[3], and M. Zakharyaschev[1]

[1] School of Computer Science and Information Systems, Birkbeck College, London
{roman,michael}@dcs.bbk.ac.uk
[2] Faculty of Computer Science, Free University of Bozen-Bolzano, Italy
ryzhikov@inf.unibz.it
[3] Department of Computer Science, University of Liverpool, U.K.,
frank@csc.liv.ac.uk

Abstract. We show how the reasoning tasks of checking various versions of conservativity for the description logic $\mathit{DL\text{-}Lite}_{bool}$ can be reduced to satisfiability of quantified Boolean formulas and how *off-the-shelf* QBF solvers perform on a number of typical $\mathit{DL\text{-}Lite}_{bool}$ ontologies.

1 Introduction

Recently, the notion of *conservative extension* and variants thereof have been identified as fundamental for developing, re-using and maintaining ontologies [1–4]. Intuitively, an ontology $\mathcal{T}_{12}$ is a conservative extension of an ontology $\mathcal{T}_1$ w.r.t. a signature Σ if $\mathcal{T}_{12} \supseteq \mathcal{T}_1$ and both $\mathcal{T}_{12}$ and $\mathcal{T}_1$ provide precisely the same information about Σ in the sense that every 'Σ-formula' derivable from $\mathcal{T}_{12}$ is derivable from $\mathcal{T}_1$ as well. If this happens to be the case, then

- an ontology engineer interested only in Σ can use $\mathcal{T}_1$ instead of the possibly much larger $\mathcal{T}_{12}$, and
- an ontology engineer who has added $\mathcal{T}_{21} \setminus \mathcal{T}_1$ to $\mathcal{T}_1$ can be certain that this addition does not change (or 'damage') the meaning assigned to Σ by $\mathcal{T}_1$.

For further discussion and applications the reader is referred to [3, 5].

In the 'definition' of conservativity above, we did not specify the language of Σ-formulas. In fact, different applications may require different languages: one might be only interested in implications between concept names (concept classification), implications between complex concepts, answers to queries when the ontology is used to query databases, etc., which leads to different notions of conservativity. In [6], several such notions were introduced and investigated for the *DL-Lite* family of description logics. It was shown, in particular, that the complexity of checking conservativity sits between coNP and Π_2^p, depending on the member of the *DL-Lite* family and the type of Σ-formulas. Note that for propositional logic deciding conservativity corresponds to deciding validity of quantified Boolean formulas (QBFs) of the form $\forall \boldsymbol{p} \exists \boldsymbol{q} \varphi$, and the lower bounds established in [6] follow from the corresponding lower bounds for QBFs.

The purpose of this paper is to report on (i) how the semantic conservativity criteria found in [6] can be refined in such a way that one can use *off-the-shelf* QBF solvers for deciding conservativity, and (ii) how different QBF solvers perform on a number of 'typical' *DL-Lite* ontologies.

The paper is organised in the following way. In the next section we remind the reader of the logic *DL-Lite*$_{bool}$, briefly discuss the conservativity notions from [6] and illustrate them with an illuminative example. In Section 3 we provide a semantic criterion of deductive conservativity, show how it can be encoded by means of QBFs, and report on the results of our experiments with three standard QBF solvers: sKizzo [7], 2clsQ [8] and Quaffle [9, 10]. Section 4 deals with query conservativity, and in Section 5 we discuss the obtained results and future work.

2 Conservativity in *DL-Lite*$_{bool}$

Our main concern in this paper is the logic *DL-Lite*$_{bool}$ [11] which covers most of the members of the *DL-Lite* family [12, 13, 11]. The language of *DL-Lite*$_{bool}$ has *object names* $a_1, a_2, \ldots$, *concept names* $A_1, A_2, \ldots$, and *role names* $P_1, P_2, \ldots$. Complex *roles* R and *concepts* C are defined as follows:

$$R \quad ::= \quad P_i \quad | \quad P_i^-, \qquad B \quad ::= \quad \bot \quad | \quad A_i \quad | \quad \geq q\, R,$$
$$C \quad ::= \quad B \quad | \quad \neg C \quad | \quad C_1 \sqcap C_2,$$

where $q \geq 1$. (Other usual concept constructs like $\top$, $\exists R$, $\leq q\, R$ and $C_1 \sqcup C_2$ can be used as standard abbreviations.) A *concept inclusion* is of the form $C_1 \sqsubseteq C_2$, where C_1 and C_2 are concepts; a *DL-Lite*$_{bool}$ *TBox* is a finite set of concept inclusions, and an *ABox* is a set of assertions of the form $C(a_i)$, $R(a_i, a_j)$, where C is a concept, R a role, and a_i, a_j are object names. A *knowledge base* (*KB*) is a pair $(\mathcal{T}, \mathcal{A})$ consisting of a TBox $\mathcal{T}$ and an ABox $\mathcal{A}$.

The semantics of *DL-Lite*$_{bool}$ is defined in the usual way using interpretations $\mathcal{I} = (\Delta^{\mathcal{I}}, A_1^{\mathcal{I}}, \ldots, P_1^{\mathcal{I}}, \ldots a_1^{\mathcal{I}}, \ldots)$. An (*essentially positive*) *existential query* $q(x_1, \ldots, x_n)$ is a first-order formula of the form

$$\exists y_1 \ldots \exists y_m \varphi(x_1, \ldots, x_n, y_1, \ldots, y_m),$$

where φ is constructed, using only $\wedge$ and $\vee$, from atoms of the form $C(s)$ and $P(s_1, s_2)$, with C being a concept, P a role, and s_i either a variable from the list $x_1, \ldots, x_n, y_1, \ldots, y_m$ or an object name. Given a KB $(\mathcal{T}, \mathcal{A})$ and a query $q(\boldsymbol{x})$, with $\boldsymbol{x} = x_1, \ldots, x_n$, we say that an n-tuple $\boldsymbol{a}$ of object names is an *answer* to $q(\boldsymbol{x})$ w.r.t. $(\mathcal{T}, \mathcal{A})$ and write $(\mathcal{T}, \mathcal{A}) \models q(\boldsymbol{a})$ if, for every model $\mathcal{I}$ for $(\mathcal{T}, \mathcal{A})$, we have $\mathcal{I} \models q(\boldsymbol{a})$. The data complexity of the query answering problem for *DL-Lite*$_{bool}$ KBs is coNP-complete [11].

A *signature* Σ is a finite set of concept and role names. Given a concept (role, TBox, ABox, query) E, we denote by *sig*(E) the *signature* of E, i.e., the set of concept and role names that occur in E. Note that $\bot$ and $\top$ are regarded as *logical symbols*. A concept (role, TBox, ABox, query) E is a Σ-*concept* (*role*, *TBox*, *ABox*, *query*, respectively) if *sig*$(E) \subseteq \Sigma$. Thus, P^- is a Σ-role iff $P \in \Sigma$.

We now introduce four types of conservativity we deal with in this paper; for further details and discussion we refer the reader to [6].

Definition 1. Let $\mathcal{T}_1 \subseteq \mathcal{T}_{12}$ be *DL-Lite*$_{bool}$ TBoxes and Σ a signature.

- $\mathcal{T}_{12}$ is called a *deductive conservative extension of* $\mathcal{T}_1$ *w.r.t.* Σ if, for every concept inclusion $C_1 \sqsubseteq C_2$ with $sig(C_1 \sqsubseteq C_2) \subseteq \Sigma$, we have $\mathcal{T}_1 \models C_1 \sqsubseteq C_2$ whenever $\mathcal{T}_{12} \models C_1 \sqsubseteq C_2$.
- $\mathcal{T}_{12}$ is a *query conservative extension of* $\mathcal{T}_1$ *w.r.t.* Σ if, for every ABox $\mathcal{A}$ with $sig(\mathcal{A}) \subseteq \Sigma$, every query q with $sig(q) \subseteq \Sigma$, and every tuple $\boldsymbol{a}$ of object names from $\mathcal{A}$, we have $(\mathcal{T}_1, \mathcal{A}) \models q(\boldsymbol{a})$ whenever $(\mathcal{T}_{12}, \mathcal{A}) \models q(\boldsymbol{a})$.
- $\mathcal{T}_{12}$ is a *strong deductive* (*query*) *conservative extension* of $\mathcal{T}_1$ in w.r.t. Σ if $\mathcal{T}_{12} \cup \mathcal{T}$ is a deductive (respectively, query) conservative extension of $\mathcal{T}_1 \cup \mathcal{T}$ w.r.t. Σ, for every TBox $\mathcal{T}$ with $sig(\mathcal{T}) \cap sig(\mathcal{T}_{12}) \subseteq \Sigma$.

Theorem 1 ([6]). *For any two DL-Lite$_{bool}$ TBoxes $\mathcal{T}_1 \subseteq \mathcal{T}_{12}$ and signature Σ, $\mathcal{T}_{12}$ is a query conservative extension of $\mathcal{T}_1$ w.r.t. Σ iff $\mathcal{T}_{12}$ is a strong deductive conservative extension of $\mathcal{T}_1$ w.r.t. Σ iff $\mathcal{T}_{12}$ is a strong query conservative extension of $\mathcal{T}_1$ w.r.t. Σ. Every query conservative extension is a deductive conservative extension, but not the other way round. The problems of deciding deductive and query conservativity are both Π_2^p-complete.*

We illustrate this theorem by an example which shows that checking conservativity is a non-trivial task, even for a transparent ontology with 10 axioms.

Example 1. Let $\Sigma = \{\mathsf{teaches}\}$, $\mathcal{T}_1$ contain the axioms

$$\begin{array}{llll}
\mathsf{ResearchStaff} \sqcap \mathsf{Visiting} \sqsubseteq \bot, & & \mathsf{Academic} \sqsubseteq \exists \mathsf{teaches} \sqcap \neg \geq 2\,\mathsf{teaches}, \\
\exists \mathsf{teaches} \sqsubseteq \mathsf{Academic} \sqcup \mathsf{ResearchStaff}, & & \exists \mathsf{writes} \sqsubseteq \mathsf{Academic} \sqcup \mathsf{ResearchStaff}, \\
\mathsf{ResearchStaff} \sqsubseteq \exists \mathsf{worksIn}, & & \exists \mathsf{worksIn}^- \sqsubseteq \mathsf{Project}, \\
\mathsf{Project} \sqsubseteq \exists \mathsf{manages}^-, & & \exists \mathsf{manages} \sqsubseteq \mathsf{Academic} \sqcup \mathsf{Visiting},
\end{array}$$

and let $\mathcal{T}_{12} = \mathcal{T}_1 \cup \{\mathsf{Visiting} \sqsubseteq\, \geq 2\,\mathsf{writes}\}$. It is not hard to see that in $\mathcal{T}_{12}$ we can derive new inclusions like $\mathsf{Visiting} \sqsubseteq \exists \mathsf{teaches} \sqcap \neg \geq 2\,\mathsf{teaches}$, but nothing new in the signature Σ. It follows that $\mathcal{T}_{12}$ is a deductive conservative extension of $\mathcal{T}_1$ w.r.t. Σ. Consider now the ABox $\mathcal{A} = \{\mathsf{teaches}(a, b), \mathsf{teaches}(a, c)\}$ and the query $q = \exists x \,\big((\exists \mathsf{teaches})(x) \land (\neg \geq 2\,\mathsf{teaches})(x)\big)$, that is, 'is there anybody who teaches precisely one module?' Clearly, $(\mathcal{T}_1, \mathcal{A}) \not\models q$ because $\mathcal{I} \models (\mathcal{T}_1, \mathcal{A})$ and $\mathcal{I} \not\models q$, for $\mathcal{I}$ with domain $\{a, b, c, u, v\}$ and $\mathsf{Academic}^{\mathcal{I}} = \emptyset$, $\mathsf{ResearchStaff}^{\mathcal{I}} = \{a\}$, $\mathsf{Visiting}^{\mathcal{I}} = \{v\}$, $\mathsf{Project}^{\mathcal{I}} = \{u\}$, $\mathsf{teaches}^{\mathcal{I}} = \{(a, b), (a, c)\}$, $\mathsf{worksIn}^{\mathcal{I}} = \{(a, u)\}$, $\mathsf{manages}^{\mathcal{I}} = \{(v, u)\}$. On the other hand, $(\mathcal{T}_{12}, \mathcal{A}) \models q$. Indeed, let $\mathcal{I} \models (\mathcal{T}_{12}, \mathcal{A})$. Then $a \in \mathsf{ResearchStaff}^{\mathcal{I}}$, and so there is u such that $(a, u) \in \mathsf{worksIn}^{\mathcal{I}}$ and $u \in \mathsf{Project}^{\mathcal{I}}$. Then we have some v with $(v, u) \in \mathsf{manages}^{\mathcal{I}}$ and $v \in (\mathsf{Academic} \sqcup \mathsf{Visiting})^{\mathcal{I}}$. Clearly, $\mathcal{T}_{12} \models \mathsf{Visiting} \sqsubseteq \mathsf{Academic}$, from which $v \in \mathsf{Academic}^{\mathcal{I}}$. It follows that there is w such that $(v, w) \in \mathsf{teaches}^{\mathcal{I}}$ and that such a point is unique. Therefore, $\mathcal{T}_{12}$ is not a query conservative extension of $\mathcal{T}_1$ w.r.t. Σ.

Note that the query q above contains the *negated* concept $\neg \geq 2\,\mathsf{teaches}$. This is permitted by our definition of query conservativity, although one might argue that most query languages used in DL only allow *positive existential queries*. The reason we adopt a more 'liberal' definition is that it gives us a more robust notion of conservativity, which is *stable* under the addition of 'abbreviations' to an ontology (e.g., the addition of $A \equiv \neg \geq 2\,\mathsf{teaches}$ to $\mathcal{T}_1$ and A to Σ should not affect conservativity). For our definition of queries, this is trivially the case, but it not so for any smaller class of queries.

By Theorem 1, $\mathcal{T}_{12}$ should not be a strong deductive conservative extension of $\mathcal{T}_1$ w.r.t. Σ either. Indeed, let $\mathcal{T} = \{\exists\mathsf{teaches} \sqsubseteq \geq 2\,\mathsf{teaches}\}$. Then $\mathcal{T}_1 \cup \mathcal{T}_2 \cup \mathcal{T}$ is not a deductive conservative extension of $\mathcal{T}_1 \cup \mathcal{T}$ w.r.t. Σ because $\exists\mathsf{teaches}$ is not satisfiable w.r.t. $\mathcal{T}_{12} \cup \mathcal{T}$, but is satisfiable w.r.t. $\mathcal{T}_1 \cup \mathcal{T}$.

3 Deductive conservativity

Let us fix TBoxes $\mathcal{T}_1 \subseteq \mathcal{T}_{12}$ and a signature $\Sigma \subseteq \Sigma_1 \subseteq \Sigma_{12}$, where $\Sigma_1 = sig(\mathcal{T}_1)$ and $\Sigma_{12} = sig(\mathcal{T}_{12})$.[4] Let m_0 and $m_{\mathcal{T}}$ be, respectively, the number of role names in Σ and $\mathcal{T}$, for $\mathcal{T} \in \{\mathcal{T}_1, \mathcal{T}_{12}\}$. Denote by Q the set of all numerical parameters (together with 1) that occur in $\mathcal{T}_{12}$.[5]

Without loss of generality we will assume that both $\mathcal{T}_1$ and $\mathcal{T}_{12}$ contain all the axioms of the form $\geq q'\,R \sqsubseteq\ \geq q\,R$, for all roles R in $\mathcal{T}$ and $q, q' \in Q$ such that q is the immediate predecessor of q'. We will also assume that our TBoxes contain only axioms of the form $D_1 \sqsubseteq D_2$, where the D_i are conjunctions of concepts of the form B or $\neg B$ from the definition of *DL-Lite*$_{bool}$ concepts.

Let $\Sigma_0 \in \{\Sigma, \Sigma_1, \Sigma_{12}\}$. A *$\Sigma_0 Q$-concept* is any concept of the form $\perp$, A_i, $\geq q\,R$, or its negation, for some $A_i \in \Sigma_0$, Σ_0-role R and $q \in Q$. A *$\Sigma_0 Q$-type* is a set $\mathfrak{t}$ of $\Sigma_0 Q$-concepts containing $\top$ and such that the following holds:

- for every $\Sigma_0 Q$-concept C, either $C \in \mathfrak{t}$ or $\neg C \in \mathfrak{t}$ (but not both),
- if $q < q'$ are both in Q and $\geq q'\,R \in \mathfrak{t}$ then $\geq q\,R \in \mathfrak{t}$,
- if $q < q'$ are both in Q and $\neg(\geq q\,R) \in \mathfrak{t}$ then $\neg(\geq q'\,R) \in \mathfrak{t}$.

For a type $\mathfrak{t}$, there is always an interpretation $\mathcal{I}$ and a point x in it such that $x \in C^{\mathcal{I}}$, for all $C \in \mathfrak{t}$. In this case we say that $\mathfrak{t}$ is *realised at x in $\mathcal{I}$*.

Definition 2. A set Ξ of $\Sigma_0 Q$-types is *$\mathcal{T}$-realisable* if there is a model for $\mathcal{T}$ realising *all* types in Ξ. Ξ is *precisely $\mathcal{T}$-realisable* if there is a model $\mathcal{I}$ for $\mathcal{T}$ such that $\mathcal{I}$ realises all types in Ξ and every $\Sigma_0 Q$-type realised in $\mathcal{I}$ is in Ξ.

The following semantic conservativity criterion was proved in [6]:

Theorem 2. *$\mathcal{T}_{12}$ is a deductive conservative extension of $\mathcal{T}_1$ w.r.t. Σ iff every $\mathcal{T}_1$-realisable ΣQ-type is $\mathcal{T}_{12}$-realisable.*

We now refine the criterion of Theorem 2 with the aim of encoding it by means of QBFs. $\mathcal{T}$-realisability of a type $\mathfrak{t}$ means that there is a precisely $\mathcal{T}$-realisable set Ξ of types at least one of which expands $\mathfrak{t}$. And it turns out that one can

---

[4] As shown in [6], *DL-Lite*$_{bool}$ has the interpolation property and, therefore, we can always assume that $\Sigma \subseteq sig(\mathcal{T}_1)$.

[5] In the QBF translations below, instead of Q we use the sets Q_R of numerical parameters, for each individual role R.

always find such a Ξ of size $\leq m_{\mathcal{T}}+1$. Moreover, we can order the types in Ξ in such a way that its i's type $\mathfrak{t}_i$ 'takes care of the role P_i.' To make this claim more precise we need a definition. For a ΣQ-type $\mathfrak{t}$, a sequence $\Theta_{\mathfrak{t}}^{\mathcal{T}} = \mathfrak{t}_0, \mathfrak{t}_1, \dots, \mathfrak{t}_{m_{\mathcal{T}}}$ of (not necessarily distinct) $sig(\mathcal{T})Q$-types is called a *$\mathcal{T}$-witness set for* $\mathfrak{t}$ if

($\mathbf{a_1}$) $\mathfrak{t} \subseteq \mathfrak{t}_0$;
($\mathbf{b_1}$) each type in $\mathfrak{t}_0, \mathfrak{t}_1, \dots, \mathfrak{t}_{m_{\mathcal{T}}}$ is $\mathcal{T}$-realisable;
($\mathbf{c_1}$) $\exists P_i \in \mathfrak{t}_j$, for some j, iff $\exists P_i \in \mathfrak{t}_i$ or $\exists P_i^- \in \mathfrak{t}_i$, for each of the role names P_i in $\mathcal{T}$, $1 \leq i \leq m_{\mathcal{T}}$.

Theorem 3. *A ΣQ-type $\mathfrak{t}$ is $\mathcal{T}$-realisable iff there is a $\mathcal{T}$-witness set for $\mathfrak{t}$. So $\mathcal{T}_{12}$ is a deductive conservative extension of $\mathcal{T}_1$ w.r.t. Σ iff, for every ΣQ-type $\mathfrak{t}$, whenever there is a $\mathcal{T}_1$-witness set for $\mathfrak{t}$ then there is also a $\mathcal{T}_{12}$-witness set for $\mathfrak{t}$.*

To translate the criterion of Theorem 3 into QBF, with each basic $\Sigma_0 Q$-concept (different from $\bot$) we associate a propositional variable. Fix some linear order on the set of all *DL-Lite$_{bool}$* basic concepts, and let $B_1, \dots, B_n$ be the induced list of $\Sigma_0 Q$-concepts. Then any vector $\boldsymbol{t} = (b_1, \dots, b_n)$ of distinct propositional variables b_i can be used to encode $\Sigma_0 Q$-types: every classical assignment $\mathfrak{a}$ (of the truth values F and T to propositional variables) gives rise to the $\Sigma_0 Q$-type $\mathfrak{t}^{\mathfrak{a}}(\boldsymbol{t})$ such that $B_i \in \mathfrak{t}^{\mathfrak{a}}(\boldsymbol{t})$ iff $\mathfrak{a}(b_i) = \mathsf{T}$ (and so if $\mathfrak{a}(b_i) = \mathsf{F}$ then $\neg B \in \mathfrak{t}^{\mathfrak{a}}(\boldsymbol{t})$). We will call $\boldsymbol{t}$ a *$\Sigma_0 Q$-vector* and $\mathfrak{t}^{\mathfrak{a}}(\boldsymbol{t})$ the *$\Sigma_0 Q$-type of $\boldsymbol{t}$ under* $\mathfrak{a}$. We also set $\boldsymbol{t}(B_i) = b_i$ and extend this map inductively to complex $\Sigma_0 Q$-concepts:

$$\boldsymbol{t}(\bot) = \bot, \qquad \boldsymbol{t}(\neg C) = \neg \boldsymbol{t}(C), \qquad \boldsymbol{t}(C_1 \sqcap C_2) = \boldsymbol{t}(C_1) \wedge \boldsymbol{t}(C_2).$$

We use concatenation $\boldsymbol{t}_0 \cdot \boldsymbol{t}_1$ of types $\boldsymbol{t}_0$, $\boldsymbol{t}_1$ (when extending $\Sigma_0 Q$-types to $\Sigma_0' Q$-types, $\Sigma_0 \subset \Sigma_0'$) and projection $\boldsymbol{t}{\restriction}_{\{B_1,\dots,B_k\}} = (\boldsymbol{t}(B_1), \dots, \boldsymbol{t}(B_k))$ (not a $\Sigma_0 Q$-vector, in general). A sequence $\boldsymbol{t}^n, \dots, \boldsymbol{t}^m$ of ΣQ_0-vectors is denoted by $\boldsymbol{t}^{n..m}$.

Let $\boldsymbol{t}_0^0$ be a ΣQ-vector, $\hat{\boldsymbol{t}}_1^0$ a $(\Sigma_1 \setminus \Sigma)Q$-vector, $\boldsymbol{t}_1^{1..m_{\mathcal{T}_1}}$ a sequence of $\Sigma_1 Q$-vectors, $\hat{\boldsymbol{t}}_{12}^0$ a $(\Sigma_{12} \setminus \Sigma)Q$-vector, and $\boldsymbol{t}_{12}^{1..m_{\mathcal{T}_{12}}}$ a sequence of $\Sigma_{12} Q$-vectors. By Theorem 3, the condition '$\mathcal{T}_{12}$ *is a deductive conservative extension of* $\mathcal{T}_1$' can be represented by means of the following closed quantified Boolean formula

$$\forall \boldsymbol{t}_0^0 \Big[\exists \hat{\boldsymbol{t}}_1^0 \boldsymbol{t}_1^{1..m_{\mathcal{T}_1}} \phi_{\mathcal{T}_1}(\boldsymbol{t}_0^0 \cdot \hat{\boldsymbol{t}}_1^0, \boldsymbol{t}_1^{1..m_{\mathcal{T}_1}}) \ \rightarrow\ \exists \hat{\boldsymbol{t}}_{12}^0 \boldsymbol{t}_{12}^{1..m_{\mathcal{T}_{12}}} \phi_{\mathcal{T}_{12}}(\boldsymbol{t}_0^0 \cdot \hat{\boldsymbol{t}}_{12}^0, \boldsymbol{t}_{12}^{1..m_{\mathcal{T}_{12}}}) \Big], \quad (1)$$

where, for a TBox $\mathcal{T}$ and $N \geq m_{\mathcal{T}}$,

$$\begin{aligned}
\phi_{\mathcal{T}}(\boldsymbol{t}^{0..N}) \ &= \ \bigwedge_{j=0}^{N} \theta_{\mathcal{T}}(\boldsymbol{t}^j) \quad \wedge \quad \bigwedge_{i=1}^{m_{\mathcal{T}}} \varrho_{P_i,i}(\boldsymbol{t}^{0..N}{\restriction}_{\{\exists P_i, \exists P_i^-\}}),\\
\theta_{\mathcal{T}}(\boldsymbol{t}) \ &= \ \bigwedge_{D_1 \sqsubseteq D_2 \in \mathcal{T}} \big(\boldsymbol{t}(D_1) \rightarrow \boldsymbol{t}(D_2)\big),\\
\varrho_{P,i}(\boldsymbol{p}^{0..N}) \ &= \ \big(\boldsymbol{p}^i(\exists P) \rightarrow \bigvee_{j=0}^{N} \boldsymbol{p}^j(\exists P^-)\big) \quad \wedge \quad \big(\boldsymbol{p}^i(\exists P^-) \rightarrow \bigvee_{j=0}^{N} \boldsymbol{p}^j(\exists P)\big)\\
&\qquad \wedge \big(\neg \boldsymbol{p}^i(\exists P) \wedge \neg \boldsymbol{p}^i(\exists P^-) \rightarrow \bigwedge_{\substack{j=0\\ j\neq i}}^{N} \neg \boldsymbol{p}^j(\exists P) \wedge \bigwedge_{\substack{j=0\\ j\neq i}}^{N} \neg \boldsymbol{p}^j(\exists P^-)\big).
\end{aligned}$$

Theorem 4. *For each assignment* $\mathfrak{a}$*, we have* $\mathfrak{a}(\phi_{\mathcal{T}}(\boldsymbol{t}^{0..N})) = \mathsf{T}$ *iff the set* $\{\mathfrak{t}^{\mathfrak{a}}(\boldsymbol{t}^0), \dots, \mathfrak{t}^{\mathfrak{a}}(\boldsymbol{t}^N)\}$ *of sig*$(\mathcal{T})$*Q-types is precisely* $\mathcal{T}$*-realisable in a model* $\mathcal{I}$ *where* $P_i^{\mathcal{I}} \neq \emptyset$ *iff* $\mathfrak{a}(\boldsymbol{t}^i(\exists P_i)) = \mathsf{T}$ *or* $\mathfrak{a}(\boldsymbol{t}^i(\exists P_i^-)) = \mathsf{T}$*, for* $1 \leq i \leq m_{\mathcal{T}}$*. In particular,* $\mathcal{T}_{12}$ *is a deductive conservative extension of* $\mathcal{T}_1$ *w.r.t.* Σ *iff QBF* (1) *is satisfiable.*

There are different ways of transforming (1) into a prenex CNF, which is a standard input to QBF solvers (see `http://dcs.bbk.ac.uk/~roman/qbf` for some options). One of the versions we used in our experiments is of the form

$$\forall \boldsymbol{t}_0^0 \forall \hat{\boldsymbol{t}}_1^0 \boldsymbol{t}_1^{1..m_{\mathcal{T}_1}} \quad \exists \hat{\boldsymbol{t}}_{12}^0 \boldsymbol{t}_{12}^{1..m_{\mathcal{T}_{12}}} \quad \exists \boldsymbol{u}_1 \dots \boldsymbol{u}_{m_1} \exists \boldsymbol{w}^{0..m_{\mathcal{T}_1}} \exists p$$
$$\Big[\ \phi'_{\mathcal{T}_1}(\boldsymbol{t}_0^0 \cdot \hat{\boldsymbol{t}}_1^0, \boldsymbol{t}_1^{1..m_{\mathcal{T}_1}}, \boldsymbol{u}_1 \dots \boldsymbol{u}_{m_{\mathcal{T}_1}}, \boldsymbol{w}^{0..m_{\mathcal{T}_1}}, p) \ \wedge \ \ \phi''_{\mathcal{T}_{12}}(\boldsymbol{t}_0^0 \cdot \hat{\boldsymbol{t}}_{12}^0, \boldsymbol{t}_{12}^{1..m_{\mathcal{T}_{12}}}, p)\Big],$$

where $\boldsymbol{u}_1, \dots, \boldsymbol{u}_{m_1}$, $\boldsymbol{w}^{0..m_{\mathcal{T}_1}}$ and p are K auxiliary variables, $K = (m_{\mathcal{T}_1}+1)C_{\mathcal{T}_1} + 3m_{\mathcal{T}_1} + 1$ and $C_{\mathcal{T}}$ is the number of axioms in $\mathcal{T}$. In total the prenex QBF has $(m_{\mathcal{T}_1}+1)W_{\mathcal{T}_1}$ universal and $(m_{\mathcal{T}_{12}}+1)W_{\mathcal{T}_{12}} - W_0 + K$ existential variables, where $W_{\mathcal{T}}$ and W_0 are the numbers of basic concepts in $\mathcal{T}$ and Σ, respectively. CNFs $\phi'_{\mathcal{T}}(\boldsymbol{t}^{0..N}, \boldsymbol{u}_1 \dots \boldsymbol{u}_{m_{\mathcal{T}}}, \boldsymbol{w}^{0..N}, p)$ and $\phi''_{\mathcal{T}}(\boldsymbol{t}^{0..N}, p)$ contain $(N+1)B_{\mathcal{T}} + 1 + (2N+7)m_{\mathcal{T}}$ and $(N+1)(C_{\mathcal{T}} + B'_{\mathcal{T}}) + 2(N+1)m_{\mathcal{T}}$ clauses, where $B_{\mathcal{T}}$ and $B'_{\mathcal{T}}$ are the numbers of basic concepts in the left- and right-hand sides in $\mathcal{T}$, respectively.

The order of the variables in the prefix has a strong impact on the solvers' performance (as is well-known in the QBF community), and usually one can fine-tune it depending on the solver. Another important parameter, which has not been studied comprehensively yet by the QBF community, is the structure of the prefix. For example, some of the existential quantifiers can be moved right after the universal ones they depend on, which gives a prefix of the form $\forall\exists \dots \forall\exists$. The impact of this transformation is not completely clear. However, our experiments show—especially for the more complex query conservativity—that the structure of the prefix may become crucial for a solver to succeed.

3.1 Experimentation

We experimented with several variants of the above translation. As our benchmarks, we considered three series of '3D' instances of the form $(\Sigma, \mathcal{T}_1, \mathcal{T}_{12})$, with Σ containing 1–10 roles and 5–52 basic concepts, $\mathcal{T}_1$ containing 8–25 roles, 47–122 basic concepts, 59–154 axioms, and $\mathcal{T}_{12}$ 9–30 roles, 49–147 basic concepts and 74–198 axioms. In all instances of the first series, the *NN-series*, $\mathcal{T}_{12}$ is not a deductive conservative extension of $\mathcal{T}_1$ w.r.t. Σ; in the *YN-series*, it is a deductive but not a query conservative extension; and in the *YY-series*, $\mathcal{T}_{12}$ is a query conservative extension of $\mathcal{T}_1$ w.r.t. Σ. The reader can find the benchmarks (as both LaTeX and .qdimacs files, as well as a .qdimacs translator for LaTeX files) at `http://dcs.bbk.ac.uk/~roman/qbf`. It is to be noted that our ontologies are *not* randomly generated. On the contrary, we use 'typical' *DL-Lite* ontologies available on the Web: extensions of *DL-Lite*$_{bool}$ fragments of the standard 'department ontology' (as in Example 1) as well as *DL-Lite*$_{bool}$ representations of the ER diagrams used in the QuOnto system (`http://www.dis.uniroma1.it/~quonto/`). The number of clauses in the prenex QBFs ranges over the interval 2000–18300;

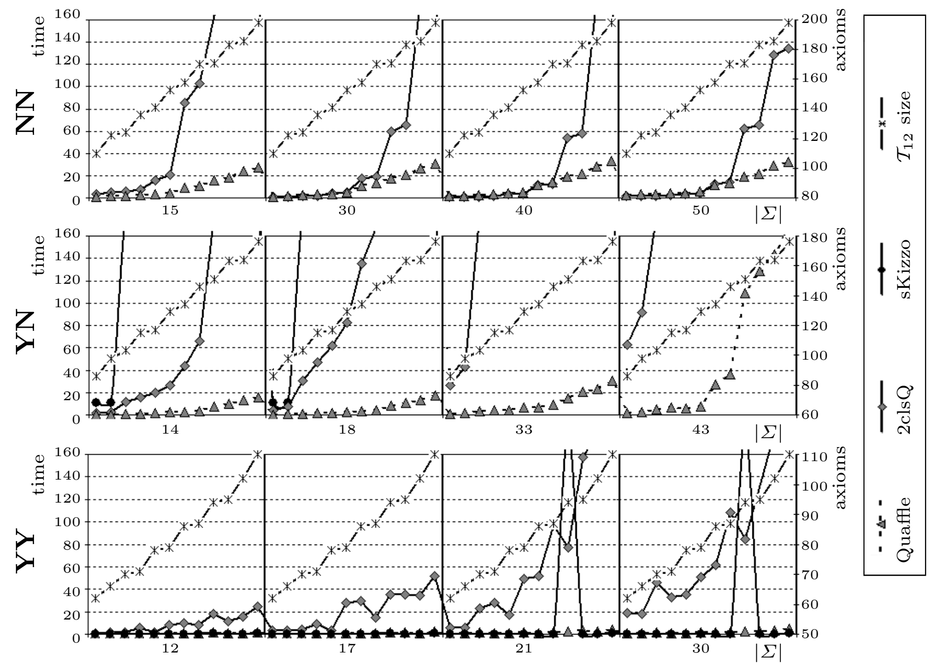

Fig. 1. Run-times for deductive conservativity.

the number of universal variables ranges over 340–3200, and the number of existential variables over 900–8600.

We checked conservativity for our benchmarks with the help of three state-of-the-art QBF solvers: sKizzo [7], 2clsQ [8] and Quaffle [9, 10]. The tests were conducted on a P4 3GHz machine with 2GB memory (in fact, $\leq$ 300MB was required). The detailed results of the experiments are available at `http://dcs.bbk.ac.uk/~roman/qbf`. Here we only give a very brief summary; see Fig. 1.

The only solver to cope with all 828 instances was Quaffle. In the most complex cases, Quaffle needed 32.8 seconds to solve an NN instance with $|\Sigma| = 52$, $|\mathcal{T}_{12}| = 198$ and 18277 clauses, 11809 variables in the translation (3172 universal and 8585 existential); and 224.4 seconds to solve an YN instance with $|\Sigma| = 45$, $|\mathcal{T}_{12}| = 191$ and 17424 clauses, 11374 variables in the translation ($\forall^{3000}\exists^{8329}$). The big difference in the run-time for these two instances may be explained by the fact that the former only required a counterexample, while the latter needed an analysis of the whole search space. The overall performance of Quaffle was by far the best in the case of deductive conservativity. On the YY series, sKizzo performed much better than both Quaffle and 2clsQ (except two instances), while on the NN and YN series sKizzo was rather poor.

4 Query conservativity

The worst-case complexity of query conservativity is the same as the complexity of deductive conservativity: both are Π_2^p-complete. However, the query conservativity criterion from [6] looks much more involved: unlike Theorem 2 dealing with realisability of individual *types*, now we have to deal with *sets* of types.

Theorem 5 ([6]). *$\mathcal{T}_{12}$ is a query conservative extension of $\mathcal{T}_1$ w.r.t. Σ iff every precisely $\mathcal{T}_1$-realisable set of ΣQ-types is precisely $\mathcal{T}_{12}$-realisable.*

To make this criterion more efficient, we observe first that a set Ξ of $sig(\mathcal{T})Q$-types is precisely $\mathcal{T}$-realisable iff every type in Ξ has a $\mathcal{T}$-witness set within Ξ. So the following conditions are equivalent:

- $\mathcal{T}_{12}$ is a query conservative extension of $\mathcal{T}_1$ w.r.t. Σ;
- for every $\mathcal{T}_1$-witness set $\Theta_{\mathfrak{t}}^{\mathcal{T}_1}$ for a ΣQ-type $\mathfrak{t}$, the set $\Theta_{\mathfrak{t}}^{\mathcal{T}_1} \restriction \Sigma$ is precisely $\mathcal{T}_{12}$-realisable, where $\Theta_{\mathfrak{t}}^{\mathcal{T}_1} \restriction \Sigma$ is the set of restrictions of types in $\Theta_{\mathfrak{t}}^{\mathcal{T}_1}$ to Σ.

Intuitively, this result means that we do not have to consider arbitrary sets of $\Sigma_1 Q$-types, but only those of size $\leq m_{\mathcal{T}_1} + 1$ that are 'generated' by a ΣQ-type $\mathfrak{t}$ and ordered in such a way that a certain type $\mathfrak{t}_i$ in the ordering 'takes care of P_i.' Now we extend the notion of a $\mathcal{T}$-witness set as follows. For a $\mathcal{T}_1$-witness set $\Theta_{\mathfrak{t}}^{\mathcal{T}_1} = \mathfrak{t}_0, \mathfrak{t}_1, \dots, \mathfrak{t}_{m_{\mathcal{T}_1}}$ and $M = m_{\mathcal{T}_{12}} - m_0$, call a sequence $\Theta_{\mathfrak{t}}^{\mathcal{T}_1 \mathcal{T}_{12}} = \hat{\mathfrak{t}}_0, \hat{\mathfrak{t}}_1, \dots, \hat{\mathfrak{t}}_{m_{\mathcal{T}_1}}, \mathfrak{s}_1, \dots, \mathfrak{s}_M$ of $\Sigma_{12} Q$-types a $\mathcal{T}_{12}$-*witness set for* $\Theta_{\mathfrak{t}}^{\mathcal{T}_1}$ if

($\mathbf{a_2}$) for each $1 \leq i \leq m_{\mathcal{T}_1}$, $\mathfrak{t}_i \restriction \Sigma \subseteq \hat{\mathfrak{t}}_i$,
($\mathbf{a_2'}$) for each $1 \leq j \leq M$, there is $1 \leq k \leq m_{\mathcal{T}_1}$ with $\mathfrak{t}_k \restriction \Sigma \subseteq \mathfrak{s}_j$,
($\mathbf{b_2}$) each type in $\hat{\mathfrak{t}}_0, \hat{\mathfrak{t}}_1, \dots, \hat{\mathfrak{t}}_{m_{\mathcal{T}_1}}, \mathfrak{s}_1, \dots, \mathfrak{s}_M$ is $\mathcal{T}_{12}$-realisable,
($\mathbf{c_2}$) $\exists P_i \in \hat{\mathfrak{t}}_j$, for some $1 \leq j \leq m_{\mathcal{T}_1}$, or $\exists P_i \in \mathfrak{s}_k$, for some $1 \leq k \leq M$, iff $\exists P_i \in \mathfrak{s}_i$ or $\exists P_i^- \in \mathfrak{s}_i$, for each role name P_i in $\Sigma_{12} \setminus \Sigma$, $1 \leq i \leq M$.

Theorem 6. *$\mathcal{T}_{12}$ is a query conservative extension of $\mathcal{T}_1$ w.r.t. Σ iff, for every $\mathcal{T}_1$-witness set $\Theta_{\mathfrak{t}}^{\mathcal{T}_1}$ for some ΣQ-type $\mathfrak{t}$, there is a $\mathcal{T}_{12}$-witness set for $\Theta_{\mathfrak{t}}^{\mathcal{T}_1}$.*

In the criterion of Theorem 3, we had to take a ΣQ-type $\mathfrak{t}$, (i) extend $\mathfrak{t}$ to a $\Sigma_1 Q$-type, (ii) check whether there are 'witnesses' for all the roles in that type and the types providing those witnesses, and if this is the case, we finally had to repeat steps (i) and (ii) again for Σ_{12} in place of Σ_1. The criterion of Theorem 6 is much more complex not only because now we have to start with a set of $(m_{\mathcal{T}_1} + 1)$ $\Sigma_1 Q$-types rather than a single type. More importantly, now the $\mathcal{T}_{12}$ witnesses we choose for these types are not *arbitrary* but must have the same Σ-restrictions as the original $\Sigma_1 Q$-types. This last condition makes the QBF translation much more complex (see below) and, consequently, computationally more costly.

Let $M = m_{\mathcal{T}_{12}} - m_0$, $\boldsymbol{t}_0^{0..m_{\mathcal{T}_1}}$ be ΣQ-vectors, $\hat{\boldsymbol{t}}_1^{0..m_{\mathcal{T}_1}}$ $(\Sigma_1 \setminus \Sigma) Q$-vectors, $\hat{\boldsymbol{t}}_{12}^{0..m_{\mathcal{T}_1}}$, $\boldsymbol{s}_{12}^{1..M}$ be $(\Sigma_{12} \setminus \Sigma) Q$-vectors. By Theorem 6, the condition '$\mathcal{T}_{12}$ *is a query conservative extension of* $\mathcal{T}_1$' can be expressed by the following closed QBF

$$\forall \boldsymbol{t}_0^{0..m_{\mathcal{T}_1}} \Big[\exists \hat{\boldsymbol{t}}_1^{0..m_{\mathcal{T}_1}} \phi_{\mathcal{T}_1}((\boldsymbol{t}_0 \cdot \hat{\boldsymbol{t}}_1)^{0..m_{\mathcal{T}_1}}) \quad \rightarrow \quad \exists \hat{\boldsymbol{t}}_{12}^{0..m_{\mathcal{T}_1}} \exists \boldsymbol{s}_{12}^{1..M} \quad \beta_{\mathcal{T}_{12}}(\boldsymbol{t}_0^{0..m_{\mathcal{T}_1}}, \hat{\boldsymbol{t}}_{12}^{0..m_{\mathcal{T}_1}}, \boldsymbol{s}_{12}^{1..M}) \Big], \quad (2)$$

where $\beta_{\mathcal{T}_{12}}(\boldsymbol{t}_0^{0..m_{\mathcal{T}_1}}, \hat{\boldsymbol{t}}_{12}^{0..m_{\mathcal{T}_1}}, \boldsymbol{s}_{12}^{1..M})$ is the formula

$$\bigwedge_{j=0}^{m_{\mathcal{T}_1}} \theta_{\mathcal{T}_{12}}(\boldsymbol{t}_0^j \cdot \hat{\boldsymbol{t}}_{12}^j) \quad\wedge\quad \bigwedge_{j=1}^{M} \bigvee_{k=0}^{m_{\mathcal{T}_1}} \theta_{\mathcal{T}_{12}}(\boldsymbol{t}_0^k \cdot \boldsymbol{s}_{12}^j)$$
$$\wedge \quad \bigwedge_{i=1}^{M} \varrho_{P_{m_0+i},i}((\boldsymbol{s}_{12}^{1..M}){\restriction_{\{\exists P_i, \exists P_i^-\}}}, (\hat{\boldsymbol{t}}_{12}^{0..m_{\mathcal{T}_1}}){\restriction_{\{\exists P_i, \exists P_i^-\}}}),$$

and $\phi_{\mathcal{T}}$, $\theta_{\mathcal{T}}$ and $\varrho_{P,i}$ are defined as before (here we assume that all concepts $\exists R$ for Σ-roles precede those for $\Sigma_{12} \setminus \Sigma$-roles).

Theorem 7. *$\mathcal{T}_{12}$ is a query conservative extension of $\mathcal{T}_1$ w.r.t. Σ iff* (2) *is satisfiable.*

It can be checked that (2) is equivalent to the prenex QBF

$$\forall \boldsymbol{t}_0^{0..m_{\mathcal{T}_1}} \exists \hat{\boldsymbol{t}}_{12}^{0..m_{\mathcal{T}_1}} \exists \boldsymbol{s}_{12}^{1..M} \exists \boldsymbol{q}^{0..m_{\mathcal{T}_1}} \exists p$$
$$\forall \hat{\boldsymbol{t}}_1^0 \exists \boldsymbol{w}^0 \cdots \forall \hat{\boldsymbol{t}}_1^{m_{\mathcal{T}_1}} \exists \boldsymbol{w}^{m_{\mathcal{T}_1}} \exists \boldsymbol{u}_1 \dots \boldsymbol{u}_{m_{\mathcal{T}_1}}$$
$$\Big[\phi'_{\mathcal{T}_1}((\boldsymbol{t}_0^0 \cdot \hat{\boldsymbol{t}}_1)^{0..m_{\mathcal{T}_1}}, \boldsymbol{u}_1 \dots \boldsymbol{u}_{m_{\mathcal{T}_1}}, \boldsymbol{w}^{0..m_{\mathcal{T}_1}}, p) \wedge$$
$$\beta''_{\mathcal{T}_{12}}(\boldsymbol{t}_0^{0..m_{\mathcal{T}_1}}, \hat{\boldsymbol{t}}_{12}^{0..m_{\mathcal{T}_1}}, \boldsymbol{s}_{12}^{1..M}, \boldsymbol{q}^{0..m_{\mathcal{T}_1}}, p)\Big],$$

where $\boldsymbol{q}^j = (q_1^j, \dots, q_M^j)$, for $0 \leq j \leq m_{\mathcal{T}_1}$, $\phi_{\mathcal{T}}$ is as before and $\beta''_{\mathcal{T}}$ is a CNF equivalent to $(p \to \beta_{\mathcal{T}})$. The latter CNF contains $(M+1)(N+1)(C_{\mathcal{T}} + B'_{\mathcal{T}}) + 2M(M+N+1) + M$ clauses, where $N = m_{\mathcal{T}_1}$, which is quadratic in $m_{\mathcal{T}_{12}}$, the number of roles in Σ_{12} (unlike $\phi''_{\mathcal{T}_{12}}$, which is only linear in $m_{\mathcal{T}_{12}}$).

4.1 Experimentation

We checked query conservativity of the same three series (NN, YN, YY) of ontologies as in Section 3.1, where only in the YY series $\mathcal{T}_{12}$ was a query conservative extension of $\mathcal{T}_1$ w.r.t. Σ. Thus, the 828 instances $(\Sigma, \mathcal{T}_1, \mathcal{T}_{12})$ are precisely the same as before. However, their QBF translations are quite different in the query conservativity case: now they have 9239–153497 clauses with 352–3172 universal and 1409–11098 existential variables. Unfortunately, not all of the instances have been solved by the three solvers. For the detailed results of the tests the reader is again referred to `http://dcs.bbk.ac.uk/~roman/qbf`, while here we confine ourselves to a brief summary; see Fig. 2.

Quaffle, the deductive conservativity 'champion,' could not solve a single query conservativity instance in 300 sec. 2clsQ showed a reasonable and, more importantly, robust performance in the NN and YN cases. The most complex YN instance it solved in 1172 sec. had the following parameters: $|\Sigma| = 45$, $|\mathcal{T}_{12}| = 176$, 100360 clauses in the QBF translation with 2576 universal and 7654 existential variables. On the other hand, 2clsQ could not solve any YY instances with timeout 300 sec. sKizzo's performance was poor on both NN and YN instances, where only very few instances were solved. However, quite unexpectedly, for some variant of the translation, sKizzo managed to solve about 40% of YY instances, where the two other solvers completely failed.

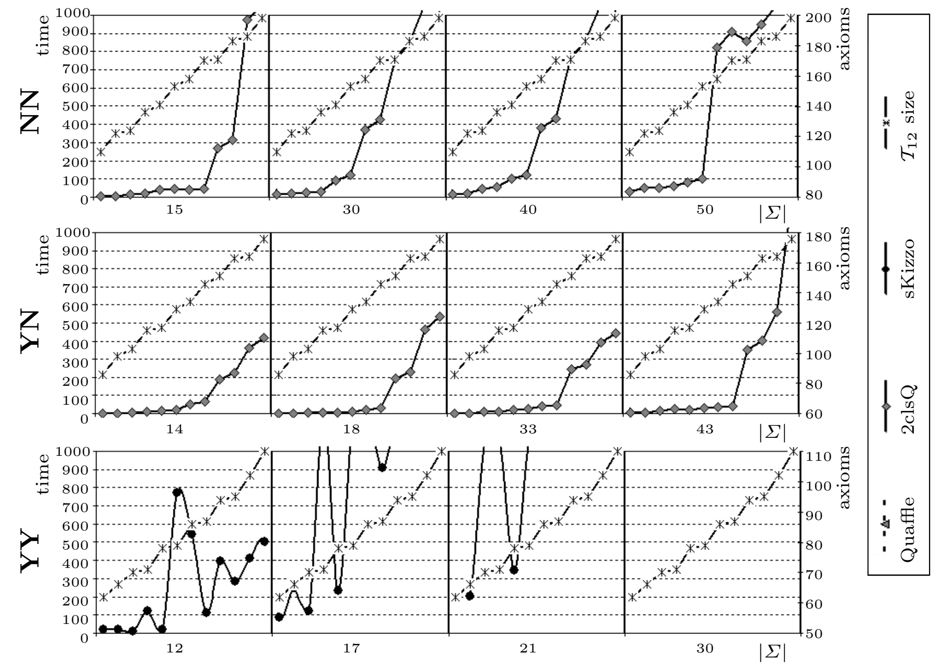

Fig. 2. Run-times for query conservativity.

5 Conclusion

The empirical results presented above indicate that it is indeed possible to automatise conservativity checking for *DL-Lite* ontologies using off-the-shelf QBF solvers. The main application area of the *DL-Lite* family of logics is conceptual data modelling and data integration, where typical *DL-Lite* ontologies do not contain more than a few hundred axioms. We have seen that modern QBF solvers can easily check deductive conservativity for ontologies of this size, even without any strategies, variable orderings, or other techniques developed specially for this particular case. It turned out, however, that query conservativity is usually more time demanding and would clearly benefit from some special QBF techniques for quantifier ordering, variable elimination or BDD ordering. (In fact, the QBF instances generated from our ontologies can form new and unusual benchmarks for QBF solvers.) For example, we plan to experiment with the AQME solver, which can inductively learn its solver selection strategy [14]. Other interesting directions for future research include the following:

- It was shown in [6] that for sub-Boolean variants of *DL-Lite* deciding conservativity becomes 'only' coNP-complete. It would be of interest to consider algorithms for this simpler case as well as other notions of conservativity.
- One can also consider sound but incomplete algorithms such as the locality-based approach of [3]. Unfortunately, very few of the test cases considered

in this paper are in the scope of the locality-based approach, yet we believe that new approximations can be developed.
– We plan to extend our results to the slightly more general case where $\mathcal{T}_1$ and $\mathcal{T}_{12}$ are incomparable. Checking whether they have the same consequences over a given signature can then be regarded as a logic-based generalisation of the standard `diff` operator for different versions of texts.
– In applications, it is important not only to know that $\mathcal{T}_{12}$ is not a conservative extension of $\mathcal{T}_1$, but also to have a corresponding counterexample. It remains to be investigated how such counterexamples can be generated using the algorithms presented in this paper.

Acknowledgements. We are grateful to Marco Benedetti for his help.

References

1. Grau, B.C., Parsia, B., Sirin, E., Kalyanpur, A.: Modularity and Web ontologies. In: Proc. of KR. (2006) 198–209
2. Seidenberg, J., Rector, A.L.: Web ontology segmentation: analysis, classification and use. In: WWW. (2006) 13–22
3. Grau, B.C., Horrocks, I., Kazakov, Y., Sattler, U.: A logical framework for modularity of ontologies. In: Proc. of IJCAI. (2007) 298–303
4. Lutz, C., Walther, D., Wolter, F.: Conservative extensions in expressive description logics. In: Proc. of IJCAI. (2007) 453–458
5. Ghilardi, S., Lutz, C., Wolter, F.: Did I damage my ontology? A case for conservative extensions in description logic. In: Proc. of KR. (2006) 187–197
6. Kontchakov, R., Wolter, F., Zakharyaschev, M.: Modularity in DL-Lite. In: Proc. of DL. (2007) 76–87
7. Benedetti, M.: sKizzo: A suite to evaluate and certify QBFs. In Nieuwenhuis, R., ed.: Proc. of CADE–20. Vol. 3632 of LNCS. (2005) 369–376
8. Samulowitz, H., Bacchus, F.: Binary clause reasoning in QBF. In Biere, A., Gomes, C.P., eds.: Proc. of SAT. Vol. 4121 of LNCS. (2006) 353–367
9. Zhang, L., Malik, S.: Towards a symmetric treatment of satisfaction and conflicts in quantified Boolean formula evaluation. In Hentenryck, P.V., ed.: Proc. of CP. Vol. 2470 of LNCS. (2002) 200–215
10. Zhang, L., Malik, S.: Conflict driven learning in a quantified Boolean satisfiability solver. In Pileggi, L.T., Kuehlmann, A., eds.: Proc. of IEEE/ACM Conf. on Computer-aided Design. (2002) 442–449
11. Artale, A., Calvanese, D., Kontchakov, R., Zakharyaschev, M.: DL-Lite in the light of first-order logic. In: Proc. of AAAI. (2007) 361–366
12. Calvanese, D., De Giacomo, G., Lembo, D., Lenzerini, M., Rosati, R.: DL-Lite: Tractable description logics for ontologies. In: Proc. of AAAI. (2005) 602–607
13. Calvanese, D., De Giacomo, G., Lembo, D., Lenzerini, M., Rosati, R.: Data complexity of query answering in description logics. In: Proc. of KR. (2006) 260–270
14. Pulina, L., Tacchella, A.: A multi-engine solver for quantified Boolean formulas. In Bessiere, C., ed.: Proc. of CP. Vol. 4741 of LNCS. (2007) 574–589

A Transformation into prenex CNF

A.1 Deductive conservativity

We transform (1) into a prenex CNF (which is the standard input of QBF solvers) by using the following:

Proposition 1. *For every pair of formulas ψ_1 and ψ_2,*

$$\psi_1 \to \psi_2 \quad \equiv \quad \exists p \ ((\psi_1 \to p) \land (p \to \psi_2)).$$

Then, for a TBox $\mathcal{T}$ and $N \geq m_{\mathcal{T}}$, we take a fresh variable for each conjunct of $\phi_{\mathcal{T}}(\boldsymbol{t}^{0..N})$ and thus we have

$$\big(\phi_{\mathcal{T}}(\boldsymbol{t}^{0..N}) \to p\big) \ \equiv \ \exists \boldsymbol{u}_1 \dots \boldsymbol{u}_{m_{\mathcal{T}}} \exists \boldsymbol{w}^{0..N} \ \ \phi'_{\mathcal{T}}(\boldsymbol{t}^{0..N}, \boldsymbol{u}_1 \dots \boldsymbol{u}_{m_{\mathcal{T}}}, \boldsymbol{w}^{0..N}, p),$$

where

$$\begin{aligned}
\phi'_{\mathcal{T}}(\boldsymbol{t}^{0..N}, \boldsymbol{u}_1 \dots \boldsymbol{u}_{m_{\mathcal{T}}}, \boldsymbol{w}^{0..N}, p) \quad = \quad & \\
\Big(\bigwedge_{j=0}^{N} \bigwedge_{D_1 \sqsubseteq D_2 \,\in\, \mathcal{T}} w^j_{D_1,D_2} \ \land \ \bigwedge_{i=1}^{m_{\mathcal{T}}} (u_i^0 \land u_i^1 \land u_i^2) \quad \to \quad p\Big) & \\
\land \ \bigwedge_{j=0}^{N} \theta'_{\mathcal{T}}(\boldsymbol{t}^j, \boldsymbol{w}^j) \ \land \ \bigwedge_{i=1}^{m_{\mathcal{T}}} \varrho'_{P_i,i}(\boldsymbol{t}^{0..N}{\restriction}_{\{\exists P_i, \exists P_i^-\}}, \boldsymbol{u}_i), &
\end{aligned}$$

where, for each $1 \leq i \leq m_{\mathcal{T}}$, $\boldsymbol{u}_i = (u_i^1, u_i^2, u_i^3)$ and, for each $1 \leq i \leq N$, $\boldsymbol{w}^j$ contains a variable $w^j_{D_1,D_2}$, for each $D_1 \sqsubseteq D_2$ in $\mathcal{T}$, and

$$\begin{aligned}
\theta'_{\mathcal{T}}(\boldsymbol{t}, \boldsymbol{w}) \quad &= \quad \bigwedge_{D_1 \sqsubseteq D_2 \,\in\, \mathcal{T}} \big(\big(\boldsymbol{t}(D_1) \to \boldsymbol{t}(D_2)\big) \to w_{D_1,D_2}\big),\\
\varrho'_{P,i}(\boldsymbol{p}^{0..N}, \boldsymbol{u}) \quad &= \quad \big(\bigvee_{j=0}^{N} \boldsymbol{p}^j(\exists P^-) \to u^0\big) \quad \land \quad \big(\neg \boldsymbol{p}^i(\exists P) \to u^0\big)\\
&\land \quad \big(\bigvee_{j=0}^{N} \boldsymbol{p}^j(\exists P) \to u^1\big) \quad \land \quad \big(\neg \boldsymbol{p}^i(\exists P^-) \to u^1\big)\\
&\land \quad \big(\boldsymbol{p}^i(\exists P) \to u^2\big) \quad \land \quad \big(\boldsymbol{p}^i(\exists P^-) \to u^2\big)\\
&\land \quad \big(\bigwedge_{\substack{j=0\\ j\neq i}}^{N} \neg \boldsymbol{p}^j(\exists P) \ \land \ \bigwedge_{\substack{j=0\\ j\neq i}}^{N} \neg \boldsymbol{p}^j(\exists P^-) \quad \to \quad u^2\big),
\end{aligned}$$

which gives $(N+1)B_{\mathcal{T}} + 1 + m_{\mathcal{T}}(2(N+1)+5)$ clauses in CNF.

Here $C_{\mathcal{T}}$ is the number of axioms in $\mathcal{T}$,

$$B_{\mathcal{T}} = \sum_{D_1^i \sqsubseteq D_2^i \,\in\, \mathcal{T}} (|D_1^i| + 1) \quad \text{and} \quad B'_{\mathcal{T}} = \sum_{D_1^i \sqsubseteq D_2^i \,\in\, \mathcal{T}} (|D_2^i| - 1),$$

where $|D|$ is defined as follows: $|C_1 \sqcap C_2| = |C_1| + |C_2|$, $|\neg C| = |C|$ and $|A| = 1$, $|\geq q\, R| = 1$ and $|\bot| = 0$.

The formula $\big(p \to \phi_{\mathcal{T}}(\boldsymbol{t}^{0..N})\big)$ can easily be transformed into an equivalent CNF (without introducing any fresh variables) $\phi''_{\mathcal{T}}(\boldsymbol{t}^{0..N})$ containing $(N{+}1)(C_{\mathcal{T}}{+}B'_{\mathcal{T}}) + m_{\mathcal{T}} \cdot 2N$ clauses.

A.2 Query conservativity

We again begin by introducing p. Then the formula $\big(\phi_{\mathcal{T}} \to p\big)$ is transformed into CNF as in the previous section.

In order to transform the formula $\big(p \to \beta_{\mathcal{T}}(\boldsymbol{t}^{0..N}, \hat{\boldsymbol{t}}^{0..N}, \boldsymbol{s}^{1..M})\big)$ into CNF, we introduce fresh variables $\boldsymbol{q}^j = (q^j_1, \dots, q^j_M)$, for $0 \leq j \leq N$, and then

$$(p \to \beta_{\mathcal{T}}(\boldsymbol{t}^{0..N}, \hat{\boldsymbol{t}}^{0..N}, \boldsymbol{s}^{1..M})) \quad = \exists\, \boldsymbol{q}^{0..N} \;\; \beta''_{\mathcal{T}}(\boldsymbol{t}^{0..N}, \hat{\boldsymbol{t}}^{0..N}, \boldsymbol{s}^{1..M}, \boldsymbol{q}^{0..N}, p),$$

where $\beta''_{\mathcal{T}}(\boldsymbol{t}^{0..N}, \hat{\boldsymbol{t}}^{0..N}, \boldsymbol{s}^{1..M}, \boldsymbol{q}^{0..N}, p)$ is the following formula:

$$\begin{aligned}
&\bigwedge_{j=0}^{N} \big(p \to \theta_{\mathcal{T}}(\boldsymbol{t}^j_0 \cdot \hat{\boldsymbol{t}}^j)\big)\\
\wedge\quad &\bigwedge_{j=1}^{M} \big(p \to \bigvee_{k=0}^{N} q^j_k\big) \quad \wedge \quad \bigwedge_{j=1}^{M}\bigwedge_{k=0}^{N} \big(q^j_k \to \theta_{\mathcal{T}}(\boldsymbol{t}^k_0 \cdot \boldsymbol{s}^j)\big)\\
\wedge\quad &\bigwedge_{i=1}^{M} \big(p \to \varrho_{P_{m_0+i},i}((\boldsymbol{s}^{1..M})\!\restriction_{\{\exists P_i, \exists P_i^-\}}, (\hat{\boldsymbol{t}}^{0..N})\!\restriction_{\{\exists P_i, \exists P_i^-\}})\big),
\end{aligned}$$

which gives $(N+1)(C_{\mathcal{T}} + B'_{\mathcal{T}}) + M + M(N+1)(C_{\mathcal{T}} + B'_{\mathcal{T}}) + 2M(M+N+1)$ clauses.

B QBF variants: Unordered roles

We may start with an alternative definition of $\varrho_{P,i}$, which does not impose any order on the roles:

$$\begin{aligned}
\rho_{P,i}(\boldsymbol{p}^{0..N}) \;=\; \Big(\big(\big(\bigwedge_{j=0}^{N} \neg\boldsymbol{p}^j(\exists P)\big) \,\vee\, \big(\bigwedge_{j=0}^{N} \neg\boldsymbol{p}^j(\exists P^-)\big)\big) \;\to\;\\
\big(\bigwedge_{j=0}^{N} \neg\boldsymbol{p}^j(\exists P) \;\wedge\; \bigwedge_{j=0}^{N} \neg\boldsymbol{p}^j(\exists P^-)\big)\Big).
\end{aligned}$$

Then, for $\phi'_{\mathcal{T}}$, we take the following:

$$\begin{aligned}
\rho'_{P,i}(\boldsymbol{p}^{0..N}, \boldsymbol{u}) \;=\; &\bigwedge_{j=0}^{N} \big(\boldsymbol{p}^j(\exists P^-) \to u^0\big) \;\wedge \bigwedge_{j=0}^{N} \big(\boldsymbol{p}^j(\exists P) \to u^1\big) \;\wedge \big(u^0 \wedge u^1 \to u^2\big)\\
&\wedge\;\; \Big(\Big(\bigwedge_{j=0}^{N} \neg\boldsymbol{p}^j(\exists P) \;\wedge\; \bigwedge_{j=0}^{N} \neg\boldsymbol{p}^j(\exists P^-)\Big) \to u^2\Big),
\end{aligned}$$

which gives $2(N+1)+2$ clauses per role (compare with $2(N+1)+5$ for ϱ').

In order to convert $\big(p \to \phi_{\mathcal{T}}(\boldsymbol{t}^{0..N})\big)$ into CNF $\phi''_{\mathcal{T}}(\boldsymbol{t}^{0..N}, p)$, we introduce $m_{\mathcal{T}}$ auxiliary existential variables $e_1, \dots, e_{\mathcal{T}}$, and then we take

$$\begin{aligned}\rho''_{P,i}(\boldsymbol{p}^{0..N}, p, e_i) \ &= \big(p \to \big(\bigwedge_{j=0}^{N} \neg\boldsymbol{p}^j(\exists P) \to e_i\big)\big) \quad \land \quad \big(p \to \big(\bigwedge_{j=0}^{N} \neg\boldsymbol{p}^j(\exists P^-) \to e_i\big)\big)\\ &\land \quad \big(e_i \to \bigwedge_{j=0}^{N} \neg\boldsymbol{p}^j(\exists P) \ \land \ \bigwedge_{j=0}^{N} \neg\boldsymbol{p}^j(\exists P^-)\big),\end{aligned}$$

which gives $2 + 2(N+1)$ clauses per role (compare with $2(N+1)$ for ϱ'').