Assertion Absorption in Object Queries over Knowledge Bases*

Jiewen Wu and **Alexander Hudek** and **David Toman** and **Grant Weddell**

D. R. Cheriton School of Computer Science
University of Waterloo, Waterloo, Canada
{*j55wu,akhudek,david,gweddell*}*@cs.uwaterloo.ca*

Abstract

We develop a novel absorption technique for large collections of factual assertions about individual objects. These assertions are commonly accompanied by implicit background knowledge and form a knowledge base. Both the assertions and the background knowledge are expressed in a suitable language of Description Logic and queries over such knowledge bases can be expressed as assertion retrieval queries. The proposed absorption technique significantly improves the performance of such queries, in particular in cases where a large number of object features are known for the objects represented in such a knowledge base. In addition to the absorption technique we present the results of a preliminary experimental evaluation that validates the efficacy of the proposed optimization.

1 Introduction

Computing certain answers to queries over RDF data sets in the context of an ontology that captures background knowledge of application domains has become an important service in information systems. In this paper, we consider a *description logic* (DL) based representation of such data sets and ontologies. In particular, we assume a data set and an ontology are given in the form of a *knowledge base* $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ over some choice of DL dialect $\mathcal{L}$, in which $\mathcal{T}$ is a terminology (or TBox) that captures general ontological knowledge, and in which $\mathcal{A}$ is a set of assertions (or ABox) that identifies objects of interest to some agent and asserts facts about those objects. In this setting, an object corresponds to an individual name occurring in an ABox.

Perhaps the most basic reasoning tasks that underlie the computation of certain answers to queries over a DL knowledge base concern *knowledge base consistency*, the problem of determining if a given knowledge base $\mathcal{K}$ is consistent, and *assertion membership*, the problem of determining if a given assertion $a : C$, stating that object a occurring in $\mathcal{A}$ belongs to concept C, is a logical consequence of a consistent $\mathcal{K}$, written $\mathcal{K} \models a : C$. Note that these tasks are often combined in the literature in the sense that the latter is assume to include the former. However, in a more practical setting, we believe that a typical workload for a general DL reasoner will include far more instances of assertion membership tasks than of knowledge base consistency tasks. Thus, this separation of concerns can enable technology that is far more efficient for such workloads, particularly so in the case of "non-Horn" DLs that preclude the possibility of computing so-called canonical ABoxes (such as DLs that include disjunction). In this paper, we contribute to this development by introducing a novel absorption technique for knowledge bases and demonstrate that the technique is efficacious for workloads that contain many thousands of assertion membership tasks.

To date, work on absorption has focused on the *concept satisfaction* problem, a simple case of the assertion membership problem for knowledge bases with an ABox consisting of a single assertion $a : \top$. Indeed, it has been known for some time in this case that *lazy unfolding* is an important optimization technique in model building algorithms for satisfiability (Baader et al. 1994). It is also imperative for a large TBox to be manipulated by an *absorption generation* process to maximize the benefits of lazy unfolding in such algorithms, thereby reducing the combinatorial effects of disjunction in underlying chase procedures (Horrocks 1998).

We build on earlier work reported at the description logics workshop (Hudek and Weddell 2006) that proposed a generalization of the absorption theory and algorithms developed in (Horrocks and Tobies 2000a; 2000b) for the problem of concept satisfaction. The generalization makes it possible for lazy unfolding to be used for parts of terminologies not handled by earlier absorption algorithms and theory.

Binary absorption combines two key ideas. The first is the possibility of avoiding the need to *internalize* (at least some of the) terminological axioms of the form $(A_1 \sqcap A_2) \sqsubseteq C$, where the A_i denote *primitive concepts* and C a general concept. The second is an idea relating to *role absorptions* developed by Tsarkov and Horrocks (Tsarkov and Horrocks 2004). These ideas, in combination and when coupled with standard equivalences, make it possible for an algorithm to completely absorb, e.g., the TBox definition

$$\text{GOODCLIENT} \doteq \text{CLIENT} \sqcap (\exists Recommend^{-}.\ \text{BANK}) \sqcap (\exists Buy.(\text{COSTLY} \sqcup \text{PROFITABLE}))$$

as the set of TBox inclusion dependencies

*An extended version of the paper is available as a technical report (Wu et al. 2012).

$$
\begin{aligned}
\text{GOODCLIENT} &\sqsubseteq \text{CLIENT}\\
\text{GOODCLIENT} &\sqsubseteq \exists Buy.\ (\text{COSTLY} \sqcup \text{PROFITABLE})\\
\text{GOODCLIENT} &\sqsubseteq \exists Recommend^{-}.\ \text{BANK}\\
\text{COSTLY} &\sqsubseteq \text{A1}\\
\text{PROFITABLE} &\sqsubseteq \text{A1}\\
\text{A1} &\sqsubseteq \forall Buy^{-}.\text{A2}\\
\text{CLIENT} \sqcap \text{A2} &\sqsubseteq \text{A3}\\
\text{BANK} &\sqsubseteq \forall Recommend.\text{A4}\\
\text{A3} \sqcap \text{A4} &\sqsubseteq \text{CLIENT}
\end{aligned}
$$

in which concepts A1, A2, A3 and A4 are fresh atomic concepts introduced by the absorption algorithm.

There are other reasons that binary absorption proves useful, beyond the well-documented advantages of reducing the need for internalization of general terminological axioms. For one, it works very well for the parts of a terminology that are Horn-like. A second reason is a key contribution of this paper. Our contributions are as follows:

1. We propose a generalization of the absorption theory and algorithms pioneered by Horrocks and Tobies (Horrocks and Tobies 2000a; 2000b). The generalization applies for any DL dialect that includes $\mathcal{ALCI}$ as a fragment for concept satisfaction tasks.
2. We introduce the notion of role and concrete feature guards in the context of a knowledge base for the DL dialect $\mathcal{ALCI}(\mathbb{D})$. In particular, we show how assertion membership tasks in this dialect can map to concept satisfaction problems in the dialect $\mathcal{ALCIO}(\mathbb{D})$, but where binary absorption in combinations with guards can usefully avoid reasoning about, e.g., irrelevant ABox individuals.
3. We report on the results of a preliminary experimental evaluation that validates the efficacy of the proposed optimization.

2 Preliminaries

We study the proposed technique for the logic $\mathcal{ALCI}(\mathbb{D})$; for technical reasons, however, we need to use a (controlled subset of) $\mathcal{ALCIO}(\mathbb{D})$ defined as follows:

Definition 1 (Description Logic $\mathcal{ALCIO}(\mathbb{D})$) *Let $\mathcal{ALCIO}(\mathbb{D})$ be a DL based on disjoint infinite sets of atomic concepts $\mathrm{N_c}$, atomic roles $\mathrm{N_r}$, concrete features $\mathrm{N_f}$ and nominals $\mathrm{N_o}$. We require that if $A \in \mathrm{N_c}$, $R \in \mathrm{N_r}$, $o \in \mathrm{N_o}$, $\mathrm{N_f}$ contains f, g and C_1 and C_2 are concept descriptions, then a, A, $\neg C_1$, $C_1 \sqcap C_2$, $C_1 \sqcup C_2$, $\top$, $\bot$, $\exists R.C_1$, $\exists R^{-}.C_1$, $\forall R.C_1$, $\forall R^{-}.C_1$, $f = k$ and $f < g$, where k is a finite string are also concept descriptions.*

An interpretation $\mathcal{I}$ is a pair $\mathcal{I} = (\Delta^{\mathcal{I}} \uplus \mathbb{D}^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where $\Delta^{\mathcal{I}}$ is a non-empty set, $\mathbb{D}^{\mathcal{I}}$ a disjoint concrete domain of finite strings, and $\cdot^{\mathcal{I}}$ is a function mapping each feature f to a total function $f^{\mathcal{I}} : \Delta \to \mathbb{D}$, the "=" symbol to the equality relation over $\mathbb{D}$, the "<" symbol to the binary relation for an alphabetic ordering of $\mathbb{D}$, a finite string k to itself, $\mathrm{N_c}$ to subsets of $\Delta^{\mathcal{I}}$, $\mathrm{N_r}$ to subsets of $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$, and $\mathrm{N_o}$ to singleton sets of $\Delta^{\mathcal{I}}$. The interpretation is extended to compound descriptions in the standard way.

We also define the fragment $\mathcal{ALCI}(\mathbb{D})$ that disallows the use of the nominal concept constructor.

Definition 2 (TBox, ABox, and KB Satisfiability) *A* TBox $\mathcal{T}$ *is a finite set of axioms of the form $C_1 \sqsubseteq C_2$ or $C_1 \doteq C_2$. A TBox $\mathcal{T}$ is called* primitive *iff it consists entirely of axioms of the form $A \doteq C$ with $A \in \mathrm{N_c}$, each $A \in \mathrm{N_c}$ appears in at most one left hand side of an axiom, and $\mathcal{T}$ is* acyclic. *$A \in \mathrm{N_c}$ is defined in $\mathcal{T}$ if $\mathcal{T}$ contains $A \sqsubseteq C$ or $A \doteq C$. An* ABox $\mathcal{A}$ *is a finite set of assertions of the form $a : A$, $a : (f\ \mathit{op}\ k)$ and $R(a, b)$.*

Let $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ be an $\mathcal{ALCIO}(\mathbb{D})$ knowledge base *(KB). An interpretation $\mathcal{I}$ is a* model *of $\mathcal{K}$, written $\mathcal{I} \models \mathcal{T}$, iff $C_1^{\mathcal{I}} \subseteq C_2^{\mathcal{I}}$ holds for each $C_1 \sqsubseteq C_2 \in \mathcal{T}$, $C_1^{\mathcal{I}} = C_2^{\mathcal{I}}$ holds for each $C_1 \doteq C_2 \in \mathcal{T}$, $a^{\mathcal{I}} \in A^{\mathcal{I}}$ for $a : A \in \mathcal{A}$, $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}}$, and $f^{\mathcal{I}}(a^{\mathcal{I}})\ \mathit{op}\ k$ for $a : (f\ \mathit{op}\ k) \in \mathcal{A}$. A concept C is* satisfiable *with respect to a knowledge base $\mathcal{K}$ iff there is an $\mathcal{I}$ such that $\mathcal{I} \models \mathcal{K}$ and such that $C^{\mathcal{I}} \neq \emptyset$.*

3 ABox Transformation

In this section we convert an $\mathcal{ALCI}(\mathbb{D})$ knowledge base $\mathcal{K}$ to a TBox by representing individuals in $\mathcal{K}$'s ABox by nominals (a *controlled* fragment of $\mathcal{ALCIO}(\mathbb{D})$) as follows:

Definition 3 (ABox Conversion) *Let $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ be a knowledge base. We define a TBox $\mathcal{T}_{\mathcal{A}}$ for the ABox of $\mathcal{K}$ as follows:*

$$
\begin{aligned}
\mathcal{T}_{\mathcal{A}} = &\{a \sqcap \mathrm{Def}_a \sqsubseteq \mathrm{A} \mid a : \mathrm{A} \in \mathcal{A}\}\\
&\cup \{a \sqcap \mathrm{Def}_f \sqsubseteq (f\ \mathit{op}\ k) \mid a : (f\ \mathit{op}\ k) \in \mathcal{A}\}\\
&\cup \{a \sqcap \mathrm{Def}_R \sqsubseteq \exists R.(b \sqcap \mathrm{Def}_b),\\
&\quad \{b \sqcap \mathrm{Def}_{R^-} \sqsubseteq \exists R^{-}.(a \sqcap \mathrm{Def}_a) \mid R(a, b) \in \mathcal{A}\}
\end{aligned}
$$

All axioms resulting from ABox assertions are *guarded* by auxiliary primitive concepts of the form Def_a, Def_R, and Def_f. These concepts, when coupled with an appropriate absorption, allow a reasoner to *ignore* parts of the original ABox: all the constants for which Def_a is not *set*, yielding considerable performance gains. For this idea to work we require w.l.o.g. that the TBox only uses universal restrictions of the form $\mathrm{A} \sqsubseteq \forall R.\mathrm{B}$ where A and B are (negated) primitive concepts. Subsumptions of the form $\exists R.\mathrm{A} \sqsubseteq \mathrm{B}$ (considered to be *universal restrictions*) and nested universal restrictions must be rewritten as well. Thus every $\mathcal{ALCI}(\mathbb{D})$ TBox can be transformed to an equivalent TBox that satisfies this restriction by introducing new concept names. Then the following assertions can manipulate the guards:

Definition 4 (TBox Conversion) *Let $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ be a knowledge base. We define a TBox $\mathcal{T}_{\mathcal{T}}$ for the ABox of $\mathcal{K}$ as follows:*

$$
\begin{aligned}
\mathcal{T}_{\mathcal{T}} = &\{\mathrm{A} \sqsubseteq \mathrm{Def}_R, \neg\mathrm{B} \sqsubseteq \mathrm{Def}_{R^-} \mid \mathrm{A} \sqsubseteq \forall R.\mathrm{B} \in \mathcal{T}\}\\
&\cup \{(t_1\ \mathit{op}\ t_2) \sqsubseteq \mathrm{Def}_f \mid f \text{ appears in } t_1 \text{ or in } t_2,\\
&\qquad (t_1\ \mathit{op}\ t_2) \text{ appears in } \mathcal{T}\}.
\end{aligned}
$$

In the following we use $\mathcal{T}_{\mathcal{K}}$ for $\mathcal{T} \cup \mathcal{T}_{\mathcal{T}} \cup \mathcal{T}_{\mathcal{A}}$.

Theorem 5 *Let $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ be a consistent knowledge base. Then*

$$\mathcal{K} \models a : \mathrm{C} \text{ if and only if } \mathcal{T}_{\mathcal{K}} \models a \sqcap \mathrm{D} \sqsubseteq \mathrm{C},$$

where $\mathrm{D} = \mathrm{Def}_a \sqcap (\bigsqcap_{f \text{ appears in } \mathrm{C}} \mathrm{Def}_f)$.

Proof. A sketch. Assume that there is an interpretation I_0 that satisfies $\mathcal{T}_{\mathcal{K}}$ such that $(a)^{I_0} \subseteq (\mathrm{D})^{I_0}$ but $(a)^{I_0} \cap (\mathrm{C})^{I_0} = \emptyset$ and an interpretation I_1 that satisfies $\mathcal{K}$ in which *all existential restrictions are fulfilled by anonymous objects*. We

assume w.l.o.g. both I_0 and I_1 are tree-shaped outside of the converted ABox. We construct an interpretation J for $\mathcal{K} \cup \{a : \neg \mathrm{C}\}$ as follows: Let Γ^{I_0} be the set of objects $o \in \Delta^{I_0}$ such that either $o \in (a)^{I_0}$ and $(a)^{I_0} \subseteq (\mathrm{Def}_a)^{I_0}$ or o is an anonymous object in Δ^{I_0} rooted by such an object. Similarly let Γ^{I_1} be the set of objects $o \in \Delta^{I_1}$ such that either $o = (a)^{I_1}$ and $(a)^{I_0} \cap (\mathrm{Def}_a)^{I_0} = \emptyset$ or o is an anonymous object in Δ^{I_1} rooted by such an object. We set

1. $\Delta^J = \Gamma^{I_0} \cup \Gamma^{I_1}$;
2. $(a)^J \in (a)^{I_0}$ for $(a)^J \in \Gamma^{I_0}$ and $(a)^J = (a)^{I_1}$ for $(a)^J \in \Gamma^{I_1}$;
3. $o \in A^J$ if $o \in A^{I_0}$ and $o \in \Gamma^{I_0}$ or if $o \in A^{I_1}$ and $o \in \Gamma^{I_1}$ for $\mathrm{A} \in \mathrm{N_c}$ (similarly for concrete domain concepts);
4. $(o_1, o_2) \in (R)^J$ if
   (a) $(o_1, o_2) \in R^{I_0}$ (resp. R^{I_1}) and $o_1, o_2 \in \Gamma^{I_0}$ (resp. $o_1, o_2 \in \Gamma^{I_1}$), or
   (b) $o_1 \in (a)^{I_0} \cap (\mathrm{Def}_a)^{I_0}$, $o_2 \in (b)^{I_1}$ and $R(a, b) \in \mathcal{A}$ (or vice versa).

Then $(a)^J \cap (\mathrm{C})^J = \emptyset$ holds trivially. To show $J \models \mathcal{K}$ we only consider those R edges in (4b) that *cross* between the two interpretations, i.e., when $o_1 \in (\{a\})^{I_0}$, $o_2 = (b)^{I_1}$ and $R(a, b) \in \mathcal{A}$. These edges are not needed to fulfill existential restrictions, which are fulfilled by anonymous objects. For universal restrictions expressed as $\mathrm{A} \sqsubseteq \forall R.\mathrm{B} \in \mathcal{T}$, we can conclude that $o_1 \notin (\mathrm{A})^{I_0}$ as otherwise $o_1 \in (\mathrm{Def}_R)^{I_0}$ by Definition 4 and thus $o_2 \in (\mathrm{Def}_b)^{I_0}$ by Definition 3 which contradicts the assumption $(b)^{I_0} \cap (\mathrm{Def}_b)^{I_0} = \emptyset$. Hence the inclusion dependency is satisfied vacuously. The edges in (4a) satisfy all dependencies in $\mathcal{K}$ as the remainder of the interpretation J is copied from one of the two interpretations; hence $J \models \mathcal{K}$. The proof of the other direction is trivial. □

The proof idea can be extended to handle number restrictions by treating *at most* restrictions in a similar way as universal restrictions.

4 Binary Absorptions

Model building algorithms for checking the satisfaction of a concept C operate by manipulating an internal data structure (e.g., in the form of a node and edge labelled rooted tree with "back edges"). The data structure "encodes" a *partial description* of (eventual) interpretations $\mathcal{I}$ for which $C^{\mathcal{I}}$ will be non-empty. Such a partial description will almost always abstract details on class membership for hypothetical elements of $\Delta^{\mathcal{I}}$ and on details relating to the interpretation of roles. To talk formally about absorption and lazy evaluation, it is necessary to codify the idea of a partial description. Horrocks and Tobies have done this by introducing the following notion of a *witness*, of an interpretation that *stems* from a witness, and of what it means for a witness to be *admissible* with respect to a given terminology.

Definition 6 (Witness) *Let* L *be a DL and* $C \in \mathrm{L}$ *a concept. A* witness $\mathcal{W} = (\Delta^{\mathcal{W}}, \cdot^{\mathcal{W}}, \mathcal{L}^{\mathcal{W}})$ *for* C *consists of a non-empty set* $\Delta^{\mathcal{W}}$*, a function* $\cdot^{\mathcal{W}}$ *that maps* $\mathrm{N_r}$ *to subsets of* $\Delta^{\mathcal{W}} \times \Delta^{\mathcal{W}}$*, and a function* $\mathcal{L}^{\mathcal{W}}$ *that maps* $\Delta^{\mathcal{W}}$ *to subsets of* L *such that:*

(W1) there is some $x \in \Delta^{\mathcal{W}}$ *with* $C \in \mathcal{L}^{\mathcal{W}}(x)$*,*

(W2) there is an interpretation $\mathcal{I} \in \mathrm{Int}(\mathrm{L})$ *that* stems *from* $\mathcal{W}$*, and*

(W3) for each interpretation $\mathcal{I} \in \mathrm{Int}(\mathrm{L})$ *that* stems *from* $\mathcal{W}$*,* $x \in C^{\mathcal{I}}$ *if* $C \in \mathcal{L}^{\mathcal{W}}(x)$*.*

An interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ *is said to* stem *from* $\mathcal{W}$ *if* $\Delta^{\mathcal{I}} = \Delta^{\mathcal{W}}$*,* $\cdot^{\mathcal{I}}|_{\mathrm{N_r}} = \cdot^{\mathcal{W}}$*, for each* $A \in \mathrm{N_c}$*,* $A \in \mathcal{L}^{\mathcal{W}}(x)$ *implies* $x \in A^{\mathcal{I}}$ *and* $\neg A \in \mathcal{L}^{\mathcal{W}}(x)$ *implies* $x \notin A^{\mathcal{I}}$*, for each* $a \in \mathrm{N_o}$*,* $a \in \mathcal{L}^{\mathcal{W}}(x)$ *implies* $x \in a^{\mathcal{I}}$ *and* $\neg a \in \mathcal{L}^{\mathcal{W}}(x)$ *implies* $x \notin a^{\mathcal{I}}$*, for each* $(f \ \mathsf{op}\ k)$*,* $(f \ \mathsf{op}\ k) \in \mathcal{L}^{\mathcal{W}}(x)$ *implies* $x \in (f \ \mathsf{op}\ k)^{\mathcal{I}}$ *and* $\neg(f \ \mathsf{op}\ k) \in \mathcal{L}^{\mathcal{W}}(x)$ *implies* $x \notin (f \ \mathsf{op}\ k)^{\mathcal{I}}$*.*

A witness $\mathcal{W}$ *is called* admissible *with respect to a TBox* $\mathcal{T}$ *if there is an interpretation* $\mathcal{I} \in \mathrm{Int}(\mathrm{L})$ *that stems from* $\mathcal{W}$ *with* $\mathcal{I} \models \mathcal{T}$*.*

The properties satisfied by a witness have been captured by the original lemmas 2.6 and 2.7 in (Horrocks and Tobies 2000b). We further extend binary absorption (Hudek and Weddell 2006) to allow for assertions absorbed in Section 3.

Definition 7 (Binary Absorption) *Let* L *and* $\mathcal{K}$=$\{\mathcal{T}, \mathcal{A}\}$ *be a DL and a KB, respectively. A* binary absorption *of* $\mathcal{T}$ *is a pair of TBoxes* $(\mathcal{T}_u, \mathcal{T}_g)$ *such that* $\mathcal{T} \equiv \mathcal{T}_u \cup \mathcal{T}_g$ *and* $\mathcal{T}_u$ *contains axioms of the form* $A_1 \sqsubseteq C$*,* $\neg A_1 \sqsubseteq C$*, and* $a \sqsubseteq C$ *or the form* $(A_1 \sqcap A_2) \sqsubseteq C$*, and* $a \sqcap A \sqsubseteq C$ *where* $\{A, A_1, A_2\} \subseteq \mathrm{N_c}$*, and* $a \in \mathrm{N_o}$*.*

A binary absorption $(\mathcal{T}_u, \mathcal{T}_g)$ *of* $\mathcal{T}$ *is called* correct *if it satisfies the following condition: For each witness* $\mathcal{W}$ *and* $x \in \Delta^{\mathcal{W}}$*, if all conditions in Figure 1 are satisfied, then* $\mathcal{W}$ *is admissible w.r.t.* $\mathcal{T}$*. A witness that satisfies the above property will be called* unfolded*.*

The distinguishing feature of our extension is the addition of the first three implications in Figure 1. Binary absorption allows additional axioms in $\mathcal{T}_u$ to be dealt with in a deterministic manner, as we illustrate in our introductory example. Assertion absorption, treating ABox assertions as axioms, also contributes to binary absorption via nominals.

5 A Binary Absorption Algorithm

In this section, we extend binary absorptions (Hudek and Weddell 2006) that derive from the absorption algorithm for the FaCT system outlined in earlier work (Baader et al. 2003; Horrocks and Tobies 2000a; 2000b). In particular, when coupled with assertion absorption the extended algorithm makes it possible to retain guarding constraints as much as possible by prioritizing binary absorptions.

The algorithm is given a $\mathcal{T}_{\mathcal{K}}$ that consists of arbitrary axioms. It proceeds by constructing four TBoxes such that: $\mathcal{T} \equiv \mathcal{T}_g \cup \mathcal{T}_{prim} \cup \mathcal{T}_{uinc} \cup \mathcal{T}_{binc}$, $\mathcal{T}_{prim}$ is primitive, $\mathcal{T}_{uinc}$ consists of axioms of the form $A_1 \sqsubseteq C$, and $a \sqsubseteq C$, and $\mathcal{T}_{binc}$ consists of axioms of the form $(A_1 \sqcap A_2) \sqsubseteq C$ and $(a \sqcap A) \sqsubseteq C$ and none of the above primitive concept are defined in $\mathcal{T}_{prim}$. Here, $\mathcal{T}_{uinc}$ contains unary absorptions and $\mathcal{T}_{binc}$ contains binary absorptions.

In the first phase, we move as many axioms as possible from $\mathcal{T}$ into $\mathcal{T}_{prim}$. We initialize $\mathcal{T}_{prim} = \emptyset$ and process each axiom $X \in \mathcal{T}$ as follows.

1. If X is of the form $A \doteq C$, A is not defined in $\mathcal{T}_{prim}$, and $\mathcal{T}_{prim} \cup \{X\}$ is primitive, then move X to $\mathcal{T}_{prim}$.

$$
\begin{array}{rll}
a \in \mathcal{L}^{\mathcal{W}}(x) \text{ and } a \in \mathcal{L}^{\mathcal{W}}(y) & \text{implies} & x = y\,, \\
\{a, A\} \subseteq \mathcal{L}^{\mathcal{W}}(x)\,, \text{and } (a \sqcap A) \sqsubseteq C \in \mathcal{T}_u & \text{implies} & C \in \mathcal{L}^{\mathcal{W}}(x)\,, \\
\{a\} \in \mathcal{L}^{\mathcal{W}}(x) \text{ and } \{a\} \sqsubseteq C \in \mathcal{T}_u & \text{implies} & C \in \mathcal{L}^{\mathcal{W}}(x)\,, \\
\{A_1, A_2\} \subseteq \mathcal{L}^{\mathcal{W}}(x) \text{ and } (A_1 \sqcap A_2) \sqsubseteq C \in \mathcal{T}_u & \text{implies} & C \in \mathcal{L}^{\mathcal{W}}(x)\,, \\
A \in \mathcal{L}^{\mathcal{W}}(x) \text{ and } A \sqsubseteq C \in \mathcal{T}_u & \text{implies} & C \in \mathcal{L}^{\mathcal{W}}(x)\,, \\
\neg A \in \mathcal{L}^{\mathcal{W}}(x) \text{ and } \neg A \sqsubseteq C \in \mathcal{T}_u & \text{implies} & C \in \mathcal{L}^{\mathcal{W}}(x)\,, \\
C_1 \sqsubseteq C_2 \in \mathcal{T}_g & \text{implies} & \neg C_1 \sqcup C_2 \in \mathcal{L}^{\mathcal{W}}(x)\,, \\
C_1 \doteq C_2 \in \mathcal{T}_g & \text{implies} & \neg C_1 \sqcup C_2 \in \mathcal{L}^{\mathcal{W}}(x) \text{ and} \\
C_1 \doteq C_2 \in \mathcal{T}_g & \text{implies} & C_1 \sqcup \neg C_2 \in \mathcal{L}^{\mathcal{W}}(x),
\end{array}
$$

Figure 1: Absorption Witness Conditions

2. If X is of the form $A \doteq C$, then remove X from $\mathcal{T}$ and replace it with axioms $A \sqsubseteq C$ and $\neg A \sqsubseteq \neg C$.
3. Otherwise, leave X in $\mathcal{T}$.

In the second phase, we process axioms in $\mathcal{T}$, either by simplifying them or by placing absorbed components in either $\mathcal{T}_{uinc}$ or $\mathcal{T}_{binc}$. We place components that cannot be absorbed in $\mathcal{T}_g$. We let $\mathbf{G} = \{C_1, \ldots, C_n\}$ represent the axiom $\top \sqsubseteq (C_1 \sqcup \ldots \sqcup C_n)$. Axioms are automatically converted to (out of) set notation.

1. If $\mathcal{T}$ is empty, then return the binary absorption $(\{A \sqsubseteq C, \neg A \sqsubseteq \neg C \mid A \doteq C \in \mathcal{T}_{prim}\} \cup \mathcal{T}_{uinc} \cup \mathcal{T}_{binc}, \mathcal{T}_g)$. Otherwise, remove an axiom $\mathbf{G}$ from $\mathcal{T}$.
2. Simplify $\mathbf{G}$.
   (a) If there is some $\neg C \in \mathbf{G}$ such that C is not a primitive concept, then add $(\mathbf{G} \cup \mathrm{NNF}(\neg C) \setminus \{\neg C\}$ to $\mathcal{T}$, where the function $\mathrm{NNF}(\cdot)$ converts concepts to negation normal form. Return to Step 1.
   (b) If there is some $C \in \mathbf{G}$ such that C is of the form $(C_1 \sqcap C_2)$, then add both $(\mathbf{G} \cup \{C_1\}) \setminus \{C\}$ and $(\mathbf{G} \cup \{C_2\}) \setminus \{C\}$ to $\mathcal{T}$. Return to Step 1.
   (c) If there is some $C \in \mathbf{G}$ such that C is of the form $C_1 \sqcup C_2$, then apply associativity by adding $(\mathbf{G} \cup \{C_1, C_2\}) \setminus \{C_1 \sqcup C_2\}$ to $\mathcal{T}$. Return to Step 1.
3. Partially absorb $\mathbf{G}$.
   (a) If $\{\neg a, \neg A\} \subset \mathbf{G}$, and A is a guard, then do the following. If an axiom of the form $(a \sqcap A) \sqsubseteq A'$ is in $\mathcal{T}_{binc}$, add $\mathbf{G} \cup \{\neg A'\} \setminus \{\neg a, \neg A\}$ to $\mathcal{T}$. Otherwise, introduce a new concept $A' \in \mathrm{N_c}$, add $(\mathbf{G} \cup \{\neg A'\}) \setminus \{\neg a, \neg A\}$ to $\mathcal{T}$, and $(a \sqcap A) \sqsubseteq A'$ to $\mathcal{T}_{binc}$. Return to Step 1.
   (b) If $\{\neg A_1, \neg A_2\} \subset \mathbf{G}$, and neither A_1 nor A_2 are defined in $\mathcal{T}_{prim}$, then do the following. If an axiom of the form $(A_1 \sqcap A_2) \sqsubseteq A'$ is in $\mathcal{T}_{binc}$, add $\mathbf{G} \cup \{\neg A'\} \setminus \{\neg A_1, \neg A_2\}$ to $\mathcal{T}$. Otherwise, introduce a new concept $A' \in \mathrm{N_c}$, add $(\mathbf{G} \cup \{\neg A'\}) \setminus \{\neg A_1, \neg A_2\}$ to $\mathcal{T}$, and $(A_1 \sqcap A_2) \sqsubseteq A'$ to $\mathcal{T}_{binc}$. Return to Step 1.
   (c) If $\forall R.\neg A$ (resp. $\forall R^-.\neg A) \in \mathbf{G}$, then do the following. Introduce a new internal primitive concept A' and add both $A \sqsubseteq \forall R^-.A'$ (resp. $A \sqsubseteq \forall R.A'$) and $(\mathbf{G} \cup \{\neg A'\}) \setminus \{\forall R.\neg A\}$ (resp. $\setminus \{\forall R^-.\neg A\}$) to $\mathcal{T}$. Return to Step 1.
4. Unfold $\mathbf{G}$. If, for some $A \in \mathbf{G}$ (resp. $\neg A \in \mathbf{G}$), there is an axiom $A \doteq C$ in $\mathcal{T}_{prim}$, then substitute $A \in \mathbf{G}$ (resp. $\neg A \in \mathbf{G}$) with C (resp. $\neg C$), and add $\mathbf{G}$ to $\mathcal{T}$. Return to Step 1.
5. Absorb $\mathbf{G}$. If $\neg a \in \mathbf{G}$, add $a \sqsubseteq C$ to $\mathcal{T}_{uinc}$ where C is the disjunction of $\mathbf{G} \setminus \{\neg a\}$. Return to Step 1.
6. Absorb $\mathbf{G}$. If $\neg A \in \mathbf{G}$ and A is not defined in $\mathcal{T}_{prim}$, add $A \sqsubseteq C$ to $\mathcal{T}_{uinc}$ where C is the disjunction of $\mathbf{G} \setminus \{\neg A\}$. Return to Step 1.
7. If none of the above are possible ($\mathbf{G}$ cannot be absorbed), add $\mathbf{G}$ to $\mathcal{T}_g$. Return to Step 1.

Termination of our procedure can be established by a counting argument involving concept constructors in $\mathcal{T}$.

Theorem 8 *For any TBox $\mathcal{T}$, the binary absorption algorithm computes a correct absorption of $\mathcal{T}$.*

Proof. The proof is by induction on iterations of our algorithm. We abbreviate the pair $(\{\mathcal{T}_{prim} \cup \mathcal{T}_{uinc} \cup \mathcal{T}_{binc}, \mathcal{T}_g \cup \mathcal{T})$ as $\mathcal{T}$ and claim that this pair is always a correct binary absorption. Initially, $\mathcal{T}_{uinc}$, $\mathcal{T}_{binc}$, and $\mathcal{T}_g$ are empty, primitive axioms are in $\mathcal{T}_{prim}$, and the remaining axioms are in $\mathcal{T}$.

- In Step 3(a) or Step 3(b), a newly introduced primitive concept only appears on the left hand side of an axiom once, hence $\mathcal{T}$ is a correct binary absorption.
- In Step 3(c), $\mathcal{T}$ is a correct binary absorption.
- In any of Steps 1, 2, 5-8, $\mathcal{T}$ is a correct binary absorption as they use only equivalence preserving operations.

Thus, $\mathcal{T}$ is a correct binary absorption by induction. □

6 Experiments

Our empirical studies focus on an important class of search queries, *assertion retrieval*, proposed in earlier work (Pound et al. 2011). Assertion retrieval queries are of the form (C, Pd), in which C is a query concept in some DL $\mathcal{L}$ serving the same role as in instance retrieval and in which Pd is a *projection description* that defines a subset of the concepts of $\mathcal{L}$. Assertion retrieval thus reduces to *instance retrieval* if $Pd = \top$. Assertion queries return assertions of the form $a : C_a$ such that $\mathcal{K} \models a : (C \sqcap C_a)$, where C_a provides the*most informative answer* for which this condition holds. A projection description generalizes the effect of the relational *projection operation* by controlling both the format and information content of query results.

Our experiments were carried out on a digital camera KB, where each camera has a list of over 70 feature-value pairs in specification. This $\mathcal{ALC}$ KB has 15 axioms, 1931 assertions, 1029 instances, and 2117 constants. We evaluated five different assertion queries of the form (C, Pd), which have the

same selection concept C and different Pd's (in increasing difficulty) over the qualifying instances. The KB and queries are available at http://db-tom.cs.uwaterloo.ca.

The times given in Table 1, averaged out over three independent runs on a Macbook with Intel Core 2 Duo processor and 4GB memory, compare five queries under three environments: NG, PG and FG, which refer to *No Guarding* that checks all individuals in $\mathcal{K}$, *Partial Guarding* that checks *relevant* individuals instead, and *Full Guarding* that, in addition to PG, introduces guards for features to reason about only *relevant* domain concepts, respectively. In NG, a Tableaux procedure needs to check all named individuals in $\mathcal{K}$, which makes query evaluation infeasible (timed out after 1000 seconds, as denoted by –). The dramatic differences depicted in Table 1 suggest that, without guarding, evaluating object queries in large datasets often becomes infeasible. The results for these five queries imply that FG reduces the

	Q1	Q2	Q3	Q4	Q5
NG	–	–	–	–	–
PG	29.98	29.80	30.33	32.28	32.34
FG	1.48	1.72	3.64	5.98	7.08

Table 1: Experimental Results (time in seconds)

total running time by at least 75 percent of that employing PG. Particularly, FG effectively trims the graph size by more than an order of magnitude during clash finding in concrete domain, thus leading to a substantial runtime improvement for query answering in KBs that have a considerable number of features. In a nutshell, the proposed optimization in this paper is a promising technique for querying large ABoxes, especially when objects are described by numerous domain concepts.

7 Conclusion

We have developed a technique that allows DL reasoners to avoid exploring a large fraction of individuals in a knowledge base that include a very large ABox in order to perform assertion membership tasks. We show how, with the presumption that the knowledge base is consistent, one can avoid considering irrelevant ABox individuals to the posed question while preserving soundness and completeness of answers. This goal is achieved by *instrumenting* the original ABox with additional *guards* that are represented by auxiliary primitive concepts, and then by developing an extension to absorption theory and algorithms in (Horrocks and Tobies 2000a; 2000b). This extension, called *binary absorption*, originally designed for TBox reasoning alone (Hudek and Weddell 2006), allows terminological axioms of the form $(a \sqcap A) \sqsubseteq C$ to qualify for lazy unfolding in model building satisfaction procedures for description logics, such as those based on tableaux technology. Such lazily unfolded axioms with *binary left-hand sides* are *essential* when (translations of) ABox assertions are to be processed by such algorithms since they prevent exploring concepts and roles associated with irrelevant ABox individuals (indeed, a simple modification to said tableaux algorithms will avoid creating instances of such individuals altogether.) Such an optimization cannot be achieved when only unary absorption is available.

The experiments show that in realistic situations arising, e.g., in implementations of *assertion retrieval* (Pound et al. 2011) in which a number of assertion membership queries are needed to answer a single user query, or in the case of *ontology-based query answering* (Lutz, Toman, and Wolter 2009; Kontchakov et al. 2010; Rosati and Almatelli 2010; Kontchakov et al. 2011), when non-Horn DLs are used (and thus the above techniques cannot be applied), our technique makes querying often feasible. The experiments show, on relatively simple examples, that while using the proposed technique allows answers to be produced in few seconds, attempting the same tasks without the optimization is not feasible.

References

Baader, F.; Franconi, E.; Hollunder, B.; Nebel, B.; and Profitlich, H.-J. 1994. An empirical analysis of optimization techniques for terminological representation systems, or: Making KRIS get a move on. *Applied Artificial Intelligence* 4:109–132.

Baader, F.; Calvanese, D.; McGuinness, D.; Nardi, D.; and Patel-Schneider, P., eds. 2003. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge Univserity Press.

Horrocks, I., and Tobies, S. 2000a. Optimisation of terminological reasoning. In *Description Logics'00*, 183–192.

Horrocks, I., and Tobies, S. 2000b. Reasoning with axioms: Theory and practice. In *KR'00*, 285–296.

Horrocks, I. 1998. Using an Expressive Description Logic: FaCT or Fiction? In *KR'98*, 636–647.

Hudek, A. K., and Weddell, G. E. 2006. Binary absorption in tableaux-based reasoning for description logics. In *Description Logics'06*.

Kontchakov, R.; Lutz, C.; Toman, D.; Wolter, F.; and Zakharyaschev, M. 2010. The combined approach to query answering in DL-Lite. In *KR'10*.

Kontchakov, R.; Lutz, C.; Toman, D.; Wolter, F.; and Zakharyaschev, M. 2011. The combined approach to query answering in DL-Lite. In *IJCAI'11*, 2656–2661.

Lutz, C.; Toman, D.; and Wolter, F. 2009. Conjunctive query answering in the description logic EL using a relational database system. In *IJCAI'09*, 2070–2075.

Pound, J.; Toman, D.; Weddell, G. E.; and Wu, J. 2011. An assertion retrieval algebra for object queries over knowledge bases. In *IJCAI'11*, 1051–1056.

Rosati, R., and Almatelli, A. 2010. Improving query answering over DL-Lite ontologies. In *KR'10*.

Tsarkov, D., and Horrocks, I. 2004. Efficient reasoning with range and domain constraints. In *Description Logics'04*.

Wu, J.; Hudek, A.; Toman, D.; and Weddell, G. 2012. Assertion absorption in object queries over knowledge bases. Technical Report CS-2012-04, University of Waterloo.