The Bag Semantics of Ontology-Based Data Access*

Charalampos Nikolaou, Egor V. Kostylev, George Konstantinidis,
Mark Kaminski, **Bernardo Cuenca Grau**, and **Ian Horrocks**
Department of Computer Science, University of Oxford, UK

Abstract

Ontology-based data access (OBDA) is a popular approach for integrating and querying multiple data sources by means of a shared ontology. The ontology is linked to the sources using mappings, which assign views over the data to ontology predicates. Motivated by the need for OBDA systems supporting database-style aggregate queries, we propose a bag semantics for OBDA, where duplicate tuples in the views defined by the mappings are retained, as is the case in standard databases. We show that bag semantics makes conjunctive query answering in OBDA CONP-hard in data complexity. To regain tractability, we consider a rather general class of queries and show its rewritability to a generalisation of the relational calculus to bags.

1 Introduction

Ontology-based data access (OBDA) is an increasingly popular approach to enable uniform access to multiple data sources with diverging schemas [Poggi *et al.*, 2008].

In OBDA, an ontology provides a unifying conceptual model for the data sources together with domain knowledge. The ontology is linked to each source by global-as-view (GAV) mappings [Lenzerini, 2002], which assign views over the data to ontology predicates. Users access the data by means of queries formulated using the vocabulary of the ontology; query answering amounts to computing the certain answers to the query over the union of ontology and the materialisation of the views defined by the mappings. The formalism of choice for representing ontologies in OBDA is the description logic $\mathit{DL\text{-}Lite}_{\mathcal{R}}$ [Calvanese *et al.*, 2007], which underpins OWL 2 QL [Motik *et al.*, 2012]. $\mathit{DL\text{-}Lite}_{\mathcal{R}}$ was designed to ensure that queries against the ontology are *first-order rewritable*; that is, they can be reformulated as a set of relational queries over the sources [Calvanese *et al.*, 2007].

Example 1. A company stores data about departments and their employees in several databases. The sales department uses the schema $\mathsf{SalEmployee}(\mathsf{id}, \mathsf{name}, \mathsf{salary}, \mathsf{loc}, \mathsf{mngr})$, where attributes id, name, salary, loc, and mngr stand for employee ID within the department, their name, salary, location, and name of their manager. In turn, the IT department stores data using the schema $\mathsf{ITEmployee}(\mathsf{id}, \mathsf{surname}, \mathsf{salary}, \mathsf{city})$, where managers are not specified. To integrate employee data, the company relies on an ontology with TBox $\mathcal{T}_{ex}$, which defines unary predicates such as SalEmp, ITEmp, and Mngr, and binary predicates such as $\mathsf{hasMngr}$ relating employees to their managers. The following mappings determine the extension of the predicates based on the data, where each att_i represents the attributes occurring only in the source:

$$\begin{array}{ll}
\mathsf{SalEmployee}(\mathsf{name}, att_1) & \to \mathsf{SalEmp}(\mathsf{name}), \\
\mathsf{SalEmployee}(\mathsf{name}, \mathsf{mngr}, att_2) & \to \mathsf{hasMngr}(\mathsf{name}, \mathsf{mngr}), \\
\mathsf{SalEmployee}(\mathsf{mngr}, att_3) & \to \mathsf{Mngr}(\mathsf{mngr}), \\
\mathsf{ITEmployee}(\mathsf{surname}, att_4) & \to \mathsf{ITEmp}(\mathsf{surname}).
\end{array}$$

TBox $\mathcal{T}_{ex}$ specifies the meaning of its vocabulary using inclusions (i) $\mathsf{SalEmp} \sqsubseteq \mathsf{Emp}$ and $\mathsf{ITEmp} \sqsubseteq \mathsf{Emp}$, which say that both sales and IT employees are company employees; (ii) $\exists\mathsf{hasMngr}^- \sqsubseteq \mathsf{Mngr}$, specifying the range of the $\mathsf{hasMngr}$ relation, and (iii) $\mathsf{Emp} \sqsubseteq \exists\mathsf{hasMngr}$, requiring that employees have a (maybe unspecified) manager. Such inclusions influence query answering: when asking for the names of all company employees, the system will retrieve all relevant sales and IT employees; this is achieved via query rewriting, where the query is reformulated as the union of queries over the sales and IT databases. ◊

OBDA has received a great deal of attention in recent years. Researchers have studied the limits of first-order rewritability in ontology languages [Calvanese *et al.*, 2007; Artale *et al.*, 2009], established bounds on the size of rewritings [Gottlob *et al.*, 2014; Kikot *et al.*, 2014], developed optimisation techniques [Kontchakov *et al.*, 2014], and implemented systems well-suited for real-world applications [Calvanese *et al.*, 2017; Calvanese *et al.*, 2011].

An important observation about the conventional semantics of OBDA is that it is set-based: the materialisation of the views defined by the mappings is formalised as a *virtual ABox* consisting of a set of facts over the ontology predicates. This treatment is, however, in contrast with the semantics of database views, which is based on bags (multisets) and where duplicate tuples are retained by default. The distinction between set and bag semantics in databases is very significant in practice; in particular, it influences the evaluation of aggre-

*This work was supported by the Royal Society under a University Research Fellowship, the EPSRC projects ED3 and DBOnto, and the Research Council of Norway via the Sirius SFI.

gate queries, which combine various aggregation functions such as Min, Max, Sum, Count or Avg with the grouping functionality provided in SQL by the GroupBy construct.

Example 2. Consider the query asking for the number of employees named Lee. Assume there are two different employees named Lee, which are represented as different tuples in the sales database (e.g., tuples with the same employee name, but different ID). Under the conventional semantics of OBDA, the virtual ABox would contain a single fact $\mathsf{SalEmp}(Lee)$; hence, the query would wrongly return one, even under the semantics for counting aggregate queries in [Calvanese *et al.*, 2008; Kostylev and Reutter, 2015]. The correct count can be obtained by considering the extension of SalEmp as a bag with multiple occurrences of Lee. $\diamondsuit$

The goal of this paper is to propose and study a bag semantics for OBDA which is compatible with the semantics of standard databases and can provide a suitable foundation for the future study of aggregate queries. We focus on conjunctive query (CQ) answering over *DL-Lite*$_\mathcal{R}$ ontologies under bag semantics, and our main contributions are as follows.

1. We propose the ontology language *DL-Lite*$^{bag}_\mathcal{R}$ and its restriction *DL-Lite*$^{bag}_{core}$, where ABoxes consist of a bag of facts, thus providing a faithful representation of the views defined by OBDA mappings. We define the semantics of query answering in this setting and show that it is compatible with the conventional set-based semantics.
2. We show that, in contrast to the set case, ontologies may not have a universal model (i.e., a single model over which all CQs can be correctly evaluated), and bag query answering becomes CONP-hard in data complexity even if we restrict ourselves to *DL-Lite*$^{bag}_{core}$ ontologies.
3. To regain tractability, we study the class of *rooted CQs* [Bienvenu *et al.*, 2012], where each connected component of the query graph is required to contain an individual or an answer variable. This is a very general class, which arguably captures most practical OBDA queries. We show that rooted CQs over *DL-Lite*$^{bag}_{core}$ ontologies not only admit a universal model and enjoy favourable computational properties, but also allow for rewritings that can be directly evaluated over the bag ABox of the ontology.

For the proofs of all results we refer to [Nikolaou *et al.*, 2017].

2 Preliminaries

Syntax of Ontologies We fix a vocabulary consisting of countably infinite and pairwise disjoint sets of *individuals* $\mathbf{I}$ (i.e., constants), *variables* $\mathbf{X}$, *atomic concepts* $\mathbf{C}$ (unary predicates) and *atomic roles* $\mathbf{R}$ (binary predicates). A *role* is an atomic role $P \in \mathbf{R}$ or its *inverse* P^-. A *concept* is an atomic concept in $\mathbf{C}$ or an expression $\exists R$, where R is a role. An *inclusion* is an expression of the form $S_1 \sqsubseteq S_2$ with S_1 and S_2 either both concepts or both roles. A *disjointness axiom* is an expression of the form $\mathsf{Disj}(S_1, S_2)$ with S_1 and S_2 either both concepts or both roles. A *concept assertion* is of the form $A(a)$ with $a \in \mathbf{I}$ and $A \in \mathbf{C}$. A *role assertion* is of the form $P(a, b)$ with $a, b \in \mathbf{I}$ and $P \in \mathbf{R}$. A *DL-Lite*$_\mathcal{R}$ *TBox* is a finite set of inclusions and disjointness axioms. An *ABox* is a finite set of concept and role assertions. A *DL-Lite*$_\mathcal{R}$ *ontology* is a pair $\langle \mathcal{T}, \mathcal{A}\rangle$ with $\mathcal{T}$ a *DL-Lite*$_\mathcal{R}$ TBox and $\mathcal{A}$ an ABox. The ontology language *DL-Lite*$_{core}$ restricts *DL-Lite*$_\mathcal{R}$ by disallowing inclusions and disjointness axioms for roles.

Semantics of Ontologies An *interpretation* $\mathcal{I}$ is a pair $\langle \Delta^\mathcal{I}, \cdot^\mathcal{I}\rangle$, where the *domain* $\Delta^\mathcal{I}$ is a non-empty set, and the *interpretation function* $\cdot^\mathcal{I}$ maps each $a \in \mathbf{I}$ to $a^\mathcal{I} \in \Delta^\mathcal{I}$ such that $a^\mathcal{I} \neq b^\mathcal{I}$ for all $a, b \in \mathbf{I}$,[1] each $A \in \mathbf{C}$ to a subset $A^\mathcal{I}$ of $\Delta^\mathcal{I}$ and each $P \in \mathbf{R}$ to a subset $P^\mathcal{I}$ of $\Delta^\mathcal{I} \times \Delta^\mathcal{I}$. The interpretation function extends to concepts and roles as follows: $(R^-)^\mathcal{I} = \{(u, v) \mid (v, u) \in R^\mathcal{I}\}$ and $(\exists R)^\mathcal{I} = \{u \in \Delta^\mathcal{I} \mid (u, v) \in R^\mathcal{I} \text{ for some } v \in \Delta^\mathcal{I}\}$.

An interpretation $\mathcal{I}$ *satisfies* ABox $\mathcal{A}$ if $a^\mathcal{I} \in A^\mathcal{I}$ for all $A(a) \in \mathcal{A}$ and $(a^\mathcal{I}, b^\mathcal{I}) \in P^\mathcal{I}$ for all $P(a, b) \in \mathcal{A}$; $\mathcal{I}$ *satisfies* TBox $\mathcal{T}$ if $S_1^\mathcal{I} \subseteq S_2^\mathcal{I}$ for all $S_1 \sqsubseteq S_2$ in $\mathcal{T}$ and $S_1^\mathcal{I} \cap S_2^\mathcal{I} = \emptyset$ for all $\mathsf{Disj}(S_1, S_2)$ in $\mathcal{T}$; $\mathcal{I}$ is a *model* of ontology $\langle \mathcal{T}, \mathcal{A}\rangle$ if it satisfies $\mathcal{T}$ and $\mathcal{A}$. An ontology is *satisfiable* if it has a model.

Queries A *conjunctive query* (*CQ*) $q(\mathbf{x})$ with *answer* variables $\mathbf{x}$ is a formula $\exists \mathbf{y}.\, \phi(\mathbf{x}, \mathbf{y})$, where $\mathbf{x}$, $\mathbf{y}$ are (possibly empty) repetition-free tuples of variables and $\phi(\mathbf{x}, \mathbf{y})$ is a conjunction of atoms of the form $A(t)$, $P(t_1, t_2)$ or $z = t$, where $A \in \mathbf{C}$, $P \in \mathbf{R}$, $z \in \mathbf{x} \cup \mathbf{y}$, and $t, t_1, t_2 \in \mathbf{x} \cup \mathbf{y} \cup \mathbf{I}$. If $\mathbf{x}$ is inessential, then we write q instead of $q(\mathbf{x})$. If $\mathbf{x}$ is the empty tuple $\langle\rangle$, then q is *Boolean*. A *union* of CQs (*UCQ*) is a disjunction of CQs with the same answer variables.

The equality atoms in a CQ $q(\mathbf{x}) = \exists \mathbf{y}.\, \phi(\mathbf{x}, \mathbf{y})$ yield an equivalence relation $\sim$ on terms $\mathbf{x} \cup \mathbf{y} \cup \mathbf{I}$, and we write $\tilde{t}$ for the equivalence class of a term t. The *Gaifman graph* of $q(\mathbf{x})$ has a node $\tilde{t}$ for each $t \in \mathbf{x} \cup \mathbf{y} \cup \mathbf{I}$ in ϕ, and an edge $\{\tilde{t}_1, \tilde{t}_2\}$ for each atom in ϕ over t_1 and t_2. We assume that all CQs are *safe*: for each $z \in \mathbf{x} \cup \mathbf{y}$, the class $\tilde{z}$ contains a term mentioned in an atom of $\phi(\mathbf{x}, \mathbf{y})$ that is not an equality.

The *certain answers* $q^\mathcal{K}$ to a (U)CQ $q(\mathbf{x})$ over a *DL-Lite*$_\mathcal{R}$ ontology $\mathcal{K}$ are the set of all tuples $\mathbf{a}$ of individuals such that $q(\mathbf{a})$ holds in every model of $\mathcal{K}$. A class of queries $\mathcal{Q}_1$ is *rewritable* to a class $\mathcal{Q}_2$ for an ontology language $\mathcal{O}$ if for any $q_1 \in \mathcal{Q}_1$ and TBox $\mathcal{T}$ in $\mathcal{O}$, there is $q_2 \in \mathcal{Q}_2$ such that, for any ABox $\mathcal{A}$ in $\mathcal{O}$ with $\langle \mathcal{T}, \mathcal{A}\rangle$ satisfiable, $q_1^{\langle \mathcal{T}, \mathcal{A}\rangle}$ equals the answers to q_2 in (the least model of) $\mathcal{A}$. Checking $\mathbf{a} \in q^{\langle \mathcal{T}, \mathcal{A}\rangle}$ for a tuple $\mathbf{a}$, (U)CQ q, and *DL-Lite*$_\mathcal{R}$ ontology $\langle \mathcal{T}, \mathcal{A}\rangle$ is an NP-complete problem with AC^0 data complexity (i.e., when $\mathcal{T}$ and q are fixed) [Calvanese *et al.*, 2007]. The latter follows from the rewritability of UCQs to themselves for *DL-Lite*$_\mathcal{R}$.

Bags A *bag* over a set M is a function $\Omega : M \to \mathbb{N}_0^\infty$, where $\mathbb{N}_0^\infty$ is the set of nonnegative integers and infinity. The value $\Omega(c)$ is the *multiplicity* of c in M. A bag Ω is *finite* if there are finitely many $c \in M$ with $\Omega(c) > 0$ and there is no c with $\Omega(c) = \infty$. The *empty bag* $\emptyset$ over M is the bag such that $\emptyset(c) = 0$ for all $c \in M$. Given bags Ω_1 and Ω_2 over M, let $\Omega_1 \subseteq \Omega_2$ if $\Omega_1(c) \leq \Omega_2(c)$ for each $c \in M$.

The *intersection* $\cap$, *max union* $\cup$, *arithmetic union* $\uplus$, and *difference* $-$ are the binary operations defined for bags Ω_1 and Ω_2 over the same set M as follows: for every $c \in M$, $(\Omega_1 \cap \Omega_2)(c) = \min\{\Omega_1(c), \Omega_2(c)\}$, $(\Omega_1 \cup \Omega_2)(c) = \max\{\Omega_1(c), \Omega_2(c)\}$, $(\Omega_1 \uplus \Omega_2)(c) = \Omega_1(c) + \Omega_2(c)$, and $(\Omega_1 - \Omega_2)(c) = \max\{0, \Omega_1(c) - \Omega_2(c)\}$; difference is well-defined only when Ω_2 is finite.

[1]We adopt the unique name assumption for convenience; dropping it does not affect results (modulo minor changes of definitions).

3 *DL-Lite*$_\mathcal{R}$ with Bag Semantics

In this section we present a bag semantics for *DL-Lite*$_\mathcal{R}$ ontologies, define the associated query answering problem, and establish its intractability in data complexity.

We formalise ABoxes as bags of facts (rather than sets) in order to faithfully represent the materialised views over source data defined by OBDA mappings.

Definition 3. *A* bag ABox *is a finite bag over the set of concept and role assertions. A DL-Lite*$_\mathcal{R}^{bag}$ ontology *is a pair* $\langle\mathcal{T},\mathcal{A}\rangle$ *of a DL-Lite*$_\mathcal{R}$ *TBox* $\mathcal{T}$ *and a bag ABox* $\mathcal{A}$*; the ontology is DL-Lite*$_{core}^{bag}$ *if* $\mathcal{T}$ *is a DL-Lite*$_{core}$ *TBox.*

The semantics of *DL-Lite*$_\mathcal{R}^{bag}$ is based on *bag interpretations* $\mathcal{I}$, with atomic concepts and roles mapped to bags of domain elements and pairs of elements, respectively, and where the interpretation function is extended to complex concepts and roles in the natural way; in particular, a concept $\exists P$ is interpreted as the bag projection of $P^\mathcal{I}$ to the first component, where each occurrence of a pair (u, v) in $P^\mathcal{I}$ contributes to the multiplicity of domain element u in $(\exists P)^\mathcal{I}$.

Definition 4. *A* bag interpretation $\mathcal{I}$ *is a pair* $\langle\Delta^\mathcal{I}, \cdot^\mathcal{I}\rangle$ *defined the same as in the set case with the exception that* $A^\mathcal{I}$ *and* $P^\mathcal{I}$ *are bags (not sets) over* $\Delta^\mathcal{I}$ *and* $\Delta^\mathcal{I}\times\Delta^\mathcal{I}$*, respectively. The interpretation function extends to concepts and roles as follows:* $(P^-)^\mathcal{I}$ *maps each* $(u, v) \in \Delta^\mathcal{I}\times\Delta^\mathcal{I}$ *to* $P^\mathcal{I}(v, u)$*, and* $(\exists R)^\mathcal{I}$ *maps each* $u \in \Delta^\mathcal{I}$ *to* $\sum_{v\in\Delta^\mathcal{I}} R^\mathcal{I}(u, v)$*.*

The definition of semantics of ontologies is as expected.

Definition 5. *A bag interpretation* $\mathcal{I} = \langle\Delta^\mathcal{I}, \cdot^\mathcal{I}\rangle$ satisfies *a bag ABox* $\mathcal{A}$ *if* $\mathcal{A}(A(a)) \leq A^\mathcal{I}(a^\mathcal{I})$ *for each concept assertion* $A(a)$ *in* $\mathcal{A}$ *and* $\mathcal{A}(P(a, b)) \leq P^\mathcal{I}(a^\mathcal{I}, b^\mathcal{I})$ *for each role assertion* $P(a, b)$. Satisfaction *of* $\mathcal{T}$ *is defined as in the set case, except that* $\subseteq$ *and* $\cap$ *are applied to bags instead of sets. Bag interpretation* $\mathcal{I}$ *is a* bag model *of the DL-Lite*$_\mathcal{R}^{bag}$ *ontology* $\langle\mathcal{T},\mathcal{A}\rangle$*, written* $\mathcal{I} \models^\mathsf{b} \langle\mathcal{T},\mathcal{A}\rangle$*, if it satisfies both* $\mathcal{T}$ *and* $\mathcal{A}$*. The ontology is* satisfiable *if it has a bag model.*

Example 6. Let $\mathcal{K}_{ex} = \langle\mathcal{T}_{ex},\mathcal{A}_{ex}\rangle$ be a *DL-Lite*$_\mathcal{R}^{bag}$ ontology with $\mathcal{T}_{ex}$ as in Example 1 and $\mathcal{A}_{ex}$ has $\mathsf{SalEmp}(\mathit{Lee})$ with multiplicity 3, $\mathsf{ITEmp}(\mathit{Lee})$ and $\mathsf{hasMngr}(\mathit{Lee}, \mathit{Hill})$ both with multiplicity 2 (and all other assertions with multiplicity 0). Let $\mathcal{I}_{ex}$ be the bag interpretation mapping individuals to themselves and with the following non-zero values:

$$\mathsf{SalEmp}^{\mathcal{I}_{ex}}(\mathit{Lee}) = \mathsf{Emp}^{\mathcal{I}_{ex}}(\mathit{Lee}) = 3,\ \mathsf{ITEmp}^{\mathcal{I}_{ex}}(\mathit{Lee}) = 2,$$
$$\mathsf{hasMngr}^{\mathcal{I}_{ex}}(\mathit{Lee}, \mathit{Hill}) = 2,\ \mathsf{hasMngr}^{\mathcal{I}_{ex}}(\mathit{Lee}, w) = 1,$$
$$\mathsf{Mngr}^{\mathcal{I}_{ex}}(\mathit{Hill}) = 2,\ \mathsf{Mngr}^{\mathcal{I}_{ex}}(w) = 1,$$

where w is a fresh element. We can check that $\mathcal{I}_{ex} \models^\mathsf{b} \mathcal{K}_{ex}$. $\diamondsuit$

We now define the notion of query answering under bag semantics. We first define the answers $q^\mathcal{I}$ of a CQ $q(\mathbf{x})$ over a bag interpretation $\mathcal{I}$. Intuitively, $q^\mathcal{I}$ is a bag of tuples of individuals such that each valid embedding λ of the body of q into $\mathcal{I}$ contributes separately to the multiplicity of the tuple $\lambda(\mathbf{x})$ in $q^\mathcal{I}$; in turn, the contribution of each specific λ is the product of the multiplicities of the images of the query atoms under λ. The latter is in accordance with the interpretation of joins in the bag relational algebra and SQL, where the multiplicity of a tuple in a join is the product of the multiplicities of the joined tuples (e.g., see [García-Molina *et al.*, 2009]).

Definition 7. *Let* $q(\mathbf{x}) = \exists\mathbf{y}.\,\phi(\mathbf{x},\mathbf{y})$ *be a CQ. The* bag answers $q^\mathcal{I}$ *to* q *over a bag interpretation* $\mathcal{I} = \langle\Delta^\mathcal{I}, \cdot^\mathcal{I}\rangle$ *are defined as the bag over tuples of individuals from* $\mathbf{I}$ *of the same size as* $\mathbf{x}$ *such that, for every such tuple* $\mathbf{a}$*,*

$$q^\mathcal{I}(\mathbf{a}) \;=\; \sum\nolimits_{\lambda\in\Lambda} \prod\nolimits_{S(\mathbf{t}) \text{ in } \phi(\mathbf{x},\mathbf{y})} S^\mathcal{I}(\lambda(\mathbf{t})),$$

where Λ *is the set of all* valuations $\lambda : \mathbf{x}\cup\mathbf{y}\cup\mathbf{I} \to \Delta^\mathcal{I}$ *such that* $\lambda(\mathbf{x}) = \mathbf{a}^\mathcal{I}$*,* $\lambda(a) = a^\mathcal{I}$ *for each* $a \in \mathbf{I}$*, and* $\lambda(z) = \lambda(t)$ *for each* $z = t$ *in* $\phi(\mathbf{x},\mathbf{y})$*.*

If q is Boolean then $q^\mathcal{I}$ are defined only for the empty tuple $\langle\rangle$. Also, conjunction $\phi(\mathbf{x},\mathbf{y})$ may contain repeated atoms, and hence can be seen as a bag of atoms; while repeated atoms are redundant in the set case, they are essential in the bag setting [Chaudhuri and Vardi, 1993] and thus the definition of $q^\mathcal{I}(\mathbf{a})$ treats each copy of a query atom $S(\mathbf{t})$ separately.

The following definition of certain answers, capturing open-world query answering, is a reformulation of the definition in [Kostylev and Reutter, 2015] for counting queries. It is a natural extension of the set notion to bags: a query answer is certain for a given multiplicity if it occurs with at least that multiplicity in every bag model of the ontology.

Definition 8. *The* bag certain answers $q^\mathcal{K}$ *to a query* q *over a DL-Lite*$_\mathcal{R}^{bag}$ *ontology* $\mathcal{K}$ *are the bag* $\bigcap_{\mathcal{I}\models^\mathsf{b}\mathcal{K}} q^\mathcal{I}$*.*

We study the problem $\textsc{BagCert}[\mathcal{Q},\mathcal{O}]$ of checking, given a query q from a class of CQs $\mathcal{Q}$, ontology $\mathcal{K} = \langle\mathcal{T},\mathcal{A}\rangle$ from an ontology language $\mathcal{O}$, tuple $\mathbf{a}$ over $\mathbf{I}$, and number $k \in \mathbb{N}_0^\infty$, whether $q^\mathcal{K}(\mathbf{a}) \geq k$; *data complexity* of $\textsc{BagCert}$ is studied under the assumption that $\mathcal{T}$ and q are fixed. Following [Grumbach and Milo, 1996], we assume that the multiplicities of assertions in $\mathcal{A}$ and k (if not infinity) are given in unary.

Example 9. Let $q_{ex}(x) = \exists y.\,\mathsf{hasMngr}(x, y)$ and $\mathcal{K}_{ex}$ be as in Example 6. Then $q_{ex}^{\mathcal{K}_{ex}}(\mathit{Lee}) = 3$. Indeed, on the one hand, $q_{ex}^{\mathcal{I}_{ex}}(\mathit{Lee}) = 3$ for $\mathcal{I}_{ex}$ in Example 6. On the other, for any bag model $\mathcal{I}$ of $\mathcal{K}_{ex}$, $q_{ex}^\mathcal{I}(\mathit{Lee}) = \Sigma_{u\in\Delta^\mathcal{I}}\mathsf{hasMngr}^\mathcal{I}(\mathit{Lee}^\mathcal{I}, u) \geq 3$, because $\mathcal{A}_{ex}(\mathsf{SalEmp}(\mathit{Lee})) = 3$ and $\mathcal{T}_{ex}$ contains inclusions $\mathsf{SalEmp} \sqsubseteq \mathsf{Emp}$ and $\mathsf{Emp} \sqsubseteq \exists\mathsf{hasMngr}$. $\diamondsuit$

The bag semantics can be seen as a generalisation of the set semantics of *DL-Lite*: first, satisfiability under bag semantics reduces to the set case; second, certain answers under bag and set semantics coincide if multiplicities are ignored.

Proposition 10. *Let* $\langle\mathcal{T},\mathcal{A}\rangle$ *be a DL-Lite*$_\mathcal{R}$ *ontology and* $\langle\mathcal{T},\mathcal{A}'\rangle$ *be a DL-Lite*$_\mathcal{R}^{bag}$ *ontology with the same TBox such that* $\{S(\mathbf{t}) \mid \mathcal{A}'(S(\mathbf{t})) \geq 1\} = \mathcal{A}$*. Then, the following holds:*

1. $\langle\mathcal{T},\mathcal{A}\rangle$ *is satisfiable if and only if* $\langle\mathcal{T},\mathcal{A}'\rangle$ *is satisfiable;*
2. *for each CQ* q *and tuple* $\mathbf{a}$ *of individuals from* $\mathbf{I}$*,* $\mathbf{a} \in q^{\langle\mathcal{T},\mathcal{A}\rangle}$ *if and only if* $q^{\langle\mathcal{T},\mathcal{A}'\rangle}(\mathbf{a}) \geq 1$*.*

An important property of satisfiable *DL-Lite*$_\mathcal{R}$ ontologies $\mathcal{K}$ is the existence of so called universal models for CQs, that is, models $\mathcal{I}$ such that the certain answers to every CQ q over $\mathcal{K}$ can be obtained by evaluating q over $\mathcal{I}$ [Calvanese *et al.*, 2007]. This notion extends naturally to bags.

Definition 11. *A bag model* $\mathcal{I}$ *of a DL-Lite*$_\mathcal{R}^{bag}$ *ontology* $\mathcal{K}$ *is* universal *for a class of queries* $\mathcal{Q}$ *if* $q^\mathcal{K} = q^\mathcal{I}$ *for any* $q \in \mathcal{Q}$*.*

Unfortunately, in contrast to the set case, even *DL-Lite*$_{core}^{bag}$ ontologies may not admit a universal bag model for all CQs.

Proposition 12. *There exists a satisfiable DL-Lite$^{bag}_{core}$ ontology that has no universal bag model for the class of all CQs.*

The lack of a universal model suggests that CQ answering under bag semantics is harder than in the set case. Indeed, this problem is coNP-hard in data complexity, which is in stark contrast to the AC^0 upper bound in the set case.

Theorem 13. BAGCERT[CQs, *DL-Lite$^{bag}_{core}$*] *is* coNP-*hard in data complexity.*

4 Universal Models for Rooted Queries

Theorem 13 suggests that bag semantics is generally not well-suited for OBDA. Our approach to overcome this negative result is to consider a restricted class of CQs, introduced in the context of query optimisation in DLs [Bienvenu *et al.*, 2012], called *rooted*: in a rooted CQ, each existential variable is connected in the Gaifman graph to an individual or an answer variable. Rooted CQs capture most practical queries; for example, they include all connected non-Boolean CQs.

Definition 14. *A CQ $q(\mathbf{x})$ is* rooted *if each connected component of its Gaifman graph has a node with a term in $\mathbf{x} \cup \mathbf{I}$.*

In contrast to arbitrary CQs, any satisfiable *DL-Lite$^{bag}_{core}$* ontology admits a universal bag model for rooted CQs. Although we define such a model, called *canonical*, in a fully declarative way, it can be intuitively seen as the result of applying a variant of the restricted chase procedure [Calì *et al.*, 2013] extended to bags. Starting from the ABox, the procedure successively "repairs" violations of $\mathcal{T}$ by extending the interpretation of concepts and roles in a minimal way.

To formalise canonical models, we need two auxiliary notions. First, the *concept closure* $\mathsf{ccl}_{\mathcal{T}}[u,\mathcal{I}]$ of an element $u \in \Delta^{\mathcal{I}}$ in a bag interpretation $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$ over a TBox $\mathcal{T}$ is the bag of concepts such that, for any concept C, $\mathsf{ccl}_{\mathcal{T}}[u,\mathcal{I}](C)$ is the maximum value of $C_0^{\mathcal{I}}(u)$ amongst all concepts C_0 satisfying $\mathcal{T} \models C_0 \sqsubseteq C$. Second, the *union* $\mathcal{I} \cup \mathcal{J}$ of bag interpretations $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$ and $\mathcal{J} = \langle \Delta^{\mathcal{J}}, \cdot^{\mathcal{J}} \rangle$ with $a^{\mathcal{I}} = a^{\mathcal{J}}$ for all $a \in \mathbf{I}$ is the bag interpretation $\langle \Delta^{\mathcal{I}} \cup \Delta^{\mathcal{J}}, \cdot^{\mathcal{I}\cup\mathcal{J}} \rangle$ with $a^{\mathcal{I}\cup\mathcal{J}} = a^{\mathcal{I}}$ for $a \in \mathbf{I}$ and $S^{\mathcal{I}\cup\mathcal{J}} = S^{\mathcal{I}} \cup S^{\mathcal{J}}$ for $S \in \mathbf{C} \cup \mathbf{R}$.

Definition 15. *The* canonical bag model $\mathcal{C}(\mathcal{K})$ *of a DL-Lite$^{bag}_{core}$ ontology $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ is the bag interpretation $\bigcup_{i \geq 0} \mathcal{C}_i(\mathcal{K})$ with the bag interpretations $\mathcal{C}_i(\mathcal{K}) = \langle \Delta^{\mathcal{C}_i(\mathcal{K})}, \cdot^{\mathcal{C}_i(\mathcal{K})} \rangle$ defined as follows:*
- $\Delta^{\mathcal{C}_0(\mathcal{K})} = \mathbf{I}$, $a^{\mathcal{C}_0(\mathcal{K})} = a$ *for each* $a \in \mathbf{I}$, *and* $S^{\mathcal{C}_0(\mathcal{K})}(\mathbf{a}) = \mathcal{A}(S(\mathbf{a}))$ *for each* $S \in \mathbf{C} \cup \mathbf{R}$ *and individuals* $\mathbf{a}$;
- *for each* $i > 0$, $\Delta^{\mathcal{C}_i(\mathcal{K})}$ *is*

$$\Delta^{\mathcal{C}_{i-1}(\mathcal{K})} \cup \{w^1_{u,R}, \dots, w^{\delta}_{u,R} \mid u \in \Delta^{\mathcal{C}_{i-1}(\mathcal{K})}, R \text{ a role}, \delta = \mathsf{ccl}_{\mathcal{T}}[u, \mathcal{C}_{i-1}(\mathcal{K})](\exists R) - (\exists R)^{\mathcal{C}_{i-1}(\mathcal{K})}(u)\},$$

where $w^j_{u,R}$ are fresh domain elements, called anonymous, *$a^{\mathcal{C}_i(\mathcal{K})} = a$ for all $a \in \mathbf{I}$, and, for all $A \in \mathbf{C}$, $P \in \mathbf{R}$, and elements u, v,*

$$A^{\mathcal{C}_i(\mathcal{K})}(u) = \begin{cases} \mathsf{ccl}_{\mathcal{T}}[u, \mathcal{C}_{i-1}(\mathcal{K})](A), & \text{if } u \in \Delta^{\mathcal{C}_{i-1}(\mathcal{K})}, \\ 0, & \text{otherwise}, \end{cases}$$

$$P^{\mathcal{C}_i(\mathcal{K})}(u,v) = \begin{cases} P^{\mathcal{C}_{i-1}(\mathcal{K})}(u,v), & \text{if } u, v \in \Delta^{\mathcal{C}_{i-1}(\mathcal{K})}, \\ 1, & \text{if } v = w^j_{u,P} \text{ or } u = w^j_{v,P^-}, \\ 0, & \text{otherwise}. \end{cases}$$

It is easily seen that $\mathcal{C}(\mathcal{K})$ satisfies $\mathcal{K}$ whenever $\mathcal{K}$ is satisfiable. We next show that it is universal for rooted CQs.

Theorem 16. *The canonical bag model $\mathcal{C}(\mathcal{K})$ of a satisfiable DL-Lite$^{bag}_{core}$ ontology $\mathcal{K}$ is universal for rooted CQs.*

Example 17. Consider an ontology $\mathcal{K}_r = \langle \mathcal{T}_r, \mathcal{A}_r \rangle$ with

$$\mathcal{T}_r = \{\mathsf{Emp} \sqsubseteq \exists \mathsf{hasMngr}, \exists \mathsf{hasMngr}^- \sqsubseteq \mathsf{Mngr}\},$$
$$\mathcal{A}_r(\mathsf{Emp}(\mathit{Lee})) = \mathcal{A}_r(\mathsf{Mngr}(\mathit{Hill})) = 1.$$

The canonical model $\mathcal{C}(\mathcal{K}_r)$ interprets (all with multiplicity 1) Emp by Lee, Mngr by Hill and $w^1_{\mathit{Lee},\mathsf{hasMngr}}$, and $\mathsf{hasMngr}$ by $(\mathit{Lee}, w^1_{\mathit{Lee},\mathsf{hasMngr}})$. Note that $\mathcal{C}(\mathcal{K}_r)$ is not universal for all CQs: for instance, $q_{nr}^{\mathcal{C}(\mathcal{K}_r)}(\langle\rangle) = 2$ for non-rooted $q_{nr} = \exists y.\, \mathsf{Mngr}(y)$, but $q_{nr}^{\mathcal{I}_{nr}}(\langle\rangle) = 1$ for the model $\mathcal{I}_{nr}$ interpreting Emp by Lee, $\mathsf{hasMngr}$ by $(\mathit{Lee}, \mathit{Hill})$, and Mngr by Hill. $\Diamond$

We conclude this section by showing an important property of rooted CQs, which justifies their favourable computational properties. As in the set case for arbitrary CQs, given a satisfiable *DL-Lite$^{bag}_{core}$* ontology $\mathcal{K}$ and a rooted CQ q, $q^{\mathcal{K}}$ can be computed over a small sub-interpretation of $\mathcal{C}(\mathcal{K})$.

Theorem 18. *Let $\mathcal{K}$ be a satisfiable DL-Lite$^{bag}_{core}$ ontology with $\mathcal{C}(\mathcal{K}) = \bigcup_{i\geq 0} \mathcal{C}_i(\mathcal{K})$ and q be a rooted CQ having n atoms. Then, $q^{\mathcal{C}(\mathcal{K})} = q^{\mathcal{C}_n(\mathcal{K})}$.*

5 Rewritability of Rooted Queries

Rewritability is key for OBDA, and we next establish to what extent rooted CQs over bag semantics are rewritable.

The first idea would be to use the analogy with the set case and rewrite to unions of CQs. There are two corresponding operations for bags: max union $\cup$ and arithmetic union $\uplus$. So we may consider *max unions* $q_{max} = q_1(\mathbf{x}) \vee \cdots \vee q_n(\mathbf{x})$ or *arithmetic unions* $q_{ar} = q_1(\mathbf{x}) \veebar \cdots \veebar q_n(\mathbf{x})$ of CQs $q_i(\mathbf{x})$, $1 \leq i \leq n$, with the following semantics, for any interpretation $\mathcal{I}$: $q^{\mathcal{I}}_{max} = q^{\mathcal{I}}_1 \cup \cdots \cup q^{\mathcal{I}}_n$ and $q^{\mathcal{I}}_{ar} = q^{\mathcal{I}}_1 \uplus \cdots \uplus q^{\mathcal{I}}_n$, respectively. Our first result is negative: rewriting to either of these classes is not possible even for *DL-Lite$^{bag}_{core}$*.

Proposition 19. *The class of rooted CQs is rewritable neither to max nor to arithmetic unions of CQs for DL-Lite$^{bag}_{core}$.*

Next we show that rooted queries are rewritable to $\mathrm{BALG}^1_{\varepsilon}$-queries: the class directly corresponding to the algebra $\mathrm{BALG}^1_{\varepsilon}$ for bags [Grumbach *et al.*, 1996; Grumbach and Milo, 1996; Libkin and Wong, 1997]. Since $\mathrm{BALG}^1_{\varepsilon} \subset$ LOGSPACE [Grumbach and Milo, 1996], where $\mathrm{BALG}^1_{\varepsilon}$ is the complexity class for $\mathrm{BALG}^1_{\varepsilon}$ algebra evaluation, rewritability to $\mathrm{BALG}^1_{\varepsilon}$-queries is highly desirable.

Intuitively, in addition to projection $\exists$, join $\wedge$, and unions $\vee$ and $\veebar$, $\mathrm{BALG}^1_{\varepsilon}$ also allows for difference $\setminus$. Domain-dependent queries, inexpressible in algebraic query languages, are precluded by restrictions on the use of variables.

Definition 20. *A* $\mathrm{BALG}^1_{\varepsilon}$-query $q(\mathbf{x})$ *with* answer *variables* $\mathbf{x}$ *is one of the following, where q_i are $\mathrm{BALG}^1_{\varepsilon}$-queries:*
- $S(\mathbf{t})$, *for* $S \in \mathbf{C} \cup \mathbf{R}$, $\mathbf{t}$ *tuple over* $\mathbf{x} \cup \mathbf{I}$ *mentioning all* $\mathbf{x}$;
- $q_1(\mathbf{x}_1) \wedge q_2(\mathbf{x}_2)$, *for* $\mathbf{x} = \mathbf{x}_1 \cup \mathbf{x}_2$;
- $q_0(\mathbf{x}_0) \wedge (x = t)$, *for* $x \in \mathbf{x}_0$, $t \in \mathbf{X} \cup \mathbf{I}$, $\mathbf{x} = \mathbf{x}_0 \cup (\{t\} \setminus \mathbf{I})$;
- $\exists \mathbf{y}.\, q_0(\mathbf{x}, \mathbf{y})$; $q_1(\mathbf{x}) \vee q_2(\mathbf{x})$; $q_1(\mathbf{x}) \veebar q_2(\mathbf{x})$; $q_1(\mathbf{x}) \setminus q_2(\mathbf{x})$.

The semantics of $\mathrm{BALG}^1_\varepsilon$-queries is defined as follows.

Definition 21. *The* bag answers $q^{\mathcal{I}}$ *to a* $\mathrm{BALG}^1_\varepsilon$*-query* $q(\mathbf{x})$ *over a bag interpretation* $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$ *is the bag of tuples over* $\mathbf{I}$ *of the same size as* $\mathbf{x}$ *inductively defined as follows, for each tuple* $\mathbf{a}$ *and the corresponding mapping* λ *such that* $\lambda(\mathbf{x}) = \mathbf{a}^{\mathcal{I}}$ *and* $\lambda(a) = a^{\mathcal{I}}$ *for all* $a \in \mathbf{I}$*:*

- $S^{\mathcal{I}}(\lambda(\mathbf{t}))$*, if* $q(\mathbf{x}) = S(\mathbf{t})$*;*
- $q_1^{\mathcal{I}}(\lambda(\mathbf{x}_1)) \times q_2^{\mathcal{I}}(\lambda(\mathbf{x}_2))$*, if* $q(\mathbf{x}) = q_1(\mathbf{x}_1) \wedge q_2(\mathbf{x}_2)$*;*
- $q_0^{\mathcal{I}}(\lambda(\mathbf{x}_0))$*, if* $q(\mathbf{x}) = q_0(\mathbf{x}_0) \wedge (x = t)$ *and* $\lambda(x) = \lambda(t)$*;*
- 0*, if* $q(\mathbf{x}) = q_0(\mathbf{x}_0) \wedge (x = t)$ *and* $\lambda(x) \neq \lambda(t)$*;*
- $\sum_{\lambda' : \mathbf{y} \to \Delta^{\mathcal{I}}} q_0^{\mathcal{I}}(\mathbf{a}^{\mathcal{I}}, \lambda'(\mathbf{y}))$*, if* $q(\mathbf{x}) = \exists \mathbf{y}.\, q_0(\mathbf{x}, \mathbf{y})$*;*
- $(q_1^{\mathcal{I}} \mathbin{\mathsf{op}} q_2^{\mathcal{I}})(\mathbf{a}^{\mathcal{I}})$ *if* $q(\mathbf{x}) = q_1(\mathbf{x}) \mathbin{\mathsf{op}'} q_2(\mathbf{x})$*, where* op *is* $\cup$*,* $\uplus$*, or* $-$ *and* op' *is* $\vee$*,* $\veebar$*, or* $\setminus$*, respectively.*

The data complexity of $\mathrm{BALG}^1_\varepsilon$-query evaluation is obtained by showing that $\mathrm{BALG}^1_\varepsilon$-queries can be be mapped to the $\mathrm{BALG}^1_\varepsilon$ algebra of [Grumbach and Milo, 1996].

Proposition 22. *Given a fixed* $\mathrm{BALG}^1_\varepsilon$*-query* $q(\mathbf{x})$*, the problem of checking whether* $q^{\mathcal{C}(\langle \emptyset, \mathcal{A} \rangle)}(\mathbf{a}) \geq k$ *for a bag ABox* $\mathcal{A}$*, tuple* $\mathbf{a}$*, and* $k \in \mathbb{N}_0^\infty$ *is* AC^0 *reducible to* $\mathrm{BALG}^1_\varepsilon$*.*

Our rewriting algorithm is inspired by the algorithm in [Kikot *et al.*, 2012] for the set case of *DL-Lite*$_{\mathcal{R}}$. Before going into details, we provide a high-level description.

The key observation is that the set of valuations of a CQ $q(\mathbf{x}) = \exists \mathbf{y}.\, \phi(\mathbf{x}, \mathbf{y})$ over the bag canonical model $\mathcal{C}(\mathcal{K})$ can be partitioned into subsets, each of which is characterised by variables $\mathbf{z} \subseteq \mathbf{y}$ that are sent to anonymous elements of $\mathcal{C}(\mathcal{K})$. Hence, we can rewrite $q(\mathbf{x})$ for each of these subsets separately and then take an arithmetic union of the resulting queries, provided these queries are guaranteed to give the same answers as the corresponding subsets of valuations.

Our rewriting proceeds along the following steps.

Step 1. First, each $\mathbf{z}$ is checked for *realisability*, that is, whether the subquery induced by $\mathbf{z}$ can indeed be folded into the anonymous forest-shaped part of $\mathcal{C}(\mathcal{K})$. This can be done without the ABox, looking only at the atoms of q that link $\mathbf{z}$ to other terms of q (these linking atoms exist because q is rooted). Non-realisable $\mathbf{z}$ can be disregarded.

Step 2. For every realisable $\mathbf{z}$, CQ $q(\mathbf{x})$ is replaced (for this $\mathbf{z}$ in the arithmetic union) by a CQ $q_{\mathbf{z}}(\mathbf{x})$ obtained from q by replacing each maximal connected component of the subquery induced by $\mathbf{z}$ by just one linking atom. This transformation is equivalence-preserving, because the anonymous part of $\mathcal{C}(\mathcal{K})$ does not involve multiplicities other than 0 and 1.

Step 3. Finally, each resulting $q_{\mathbf{z}}(\mathbf{x})$ is rewritten to a $\mathrm{BALG}^1_\varepsilon$-query $\bar{q}_{\mathbf{z}}(\mathbf{x})$ by "chasing back" each unary atom and each binary atom mentioning a variable in $\mathbf{z}$ with the TBox; for the binary atoms it is also guaranteed, by means of difference, that the variable in $\mathbf{z}$ is indeed mapped to the anonymous part, thus avoiding double-counting in the arithmetic union.

For the rest of this section, let us fix a rooted CQ $q(\mathbf{x}) = \exists \mathbf{y}.\, \phi(\mathbf{x}, \mathbf{y})$ and a *DL-Lite*$^{bag}_{core}$ TBox $\mathcal{T}$. We start by formalising Step 1.

Definition 23. *Given an ontology* $\mathcal{K}$ *with a TBox* $\mathcal{T}$ *and variables* $\mathbf{z} \subseteq \mathbf{y}$*, let* $[q, \mathbf{z}]^{\mathcal{C}(\mathcal{K})}$ *be the bag of tuples over* $\mathbf{I}$ *such that, for each tuple* $\mathbf{a}$ *of individuals,*

$$[q, \mathbf{z}]^{\mathcal{C}(\mathcal{K})}(\mathbf{a}) \;=\; \sum\nolimits_{\lambda \in \Lambda_{\mathbf{z}}} \prod\nolimits_{S(\mathbf{t}) \text{ in } \phi(\mathbf{x}, \mathbf{y})} S^{\mathcal{C}(\mathcal{K})}(\lambda(\mathbf{t})),$$

where $\Lambda_{\mathbf{z}}$ *is the set of valuations* $\lambda : \mathbf{x} \cup \mathbf{y} \cup \mathbf{I} \to \Delta^{\mathcal{C}(\mathcal{K})}$ *such that* $\lambda(\mathbf{x}) = \mathbf{a}$*,* $\lambda(a) = a$ *for each* $a \in \mathbf{I}$*,* $\lambda(x) = \lambda(t)$ *for each* $x = t$ *in* $\phi(\mathbf{x}, \mathbf{y})$*,* $\lambda(z)$ *is an anonymous element for each* $z \in \mathbf{z}$*, and* $\lambda(y) \in \mathbf{I}$ *for each* $y \in \mathbf{y} \setminus \mathbf{z}$*.*

Hence, the bag answers to q can be partitioned as follows:

$$q^{\mathcal{C}(\mathcal{K})} \;=\; \biguplus\nolimits_{\mathbf{z} \subseteq \mathbf{y}} [q, \mathbf{z}]^{\mathcal{C}(\mathcal{K})}. \tag{1}$$

Variables $\mathbf{z} \subseteq \mathbf{y}$ are *equality-consistent* if $\phi(\mathbf{x}, \mathbf{y})$ has no equality $z = t$ with $z \in \mathbf{z}$ and $t \notin \mathbf{z}$. If $\mathbf{z}$ is not equality-consistent, then $[q, \mathbf{z}]^{\mathcal{C}(\mathcal{K})} = \emptyset$ and these $\mathbf{z}$ can be disregarded in (1). Next, we show which other $\mathbf{z}$ can be ignored.

Definition 24. *Given equality-consistent* $\mathbf{z} \subseteq \mathbf{y}$*, variables* $\mathbf{z}' \subseteq \mathbf{z}$ *are* maximally connected in the anonymous part *(*ma-connected*) if* $\tilde{z} \subseteq \mathbf{z}'$ *for the equivalence class* $\tilde{z}$ *of any* $z \in \mathbf{z}'$ *and the equivalence classes* $\tilde{\mathbf{z}}'$ *are a maximal subset of* $\tilde{\mathbf{z}}$ *connected in the Gaifman graph of* q *via nodes in* $\tilde{\mathbf{z}}$*.*

Next we introduce several notations for ma-connected $\mathbf{z}' \subseteq \mathbf{z}$ with equality-consistent $\mathbf{z} \subseteq \mathbf{y}$. First, let $\phi_{\mathbf{z}'}$ be the sub-conjunction of $\phi(\mathbf{x}, \mathbf{y})$ that consists of all atoms mentioning at least one variable in $\mathbf{z}'$ (these sub-conjunctions are disjoint for different $\mathbf{z}'$). Second, since q is rooted, $\phi_{\mathbf{z}'}$ contains an atom $\alpha_{\mathbf{z}'}$ of the form $P(t, z)$ or $P(z, t)$ with $z \in \mathbf{z}'$ and $t \notin \mathbf{z}$ (note that this definition may be non-deterministic). Third, let

$$q^a_{\mathbf{z}'}() = \exists \mathbf{x}'.\, \exists \mathbf{z}'.\, \phi_{\mathbf{z}'} \wedge \bigwedge\nolimits_{t \in \mathbf{t}_{\mathbf{z}'}} (t = a) \;\wedge\; \bigwedge\nolimits_{z \in \mathbf{z}'} (z \neq a),$$

where $\mathbf{t}_{\mathbf{z}'}$ are all such terms t, a is an individual in $\mathbf{t}_{\mathbf{z}'}$ if it exists or a fresh individual otherwise, and $\mathbf{x}' = \mathbf{t}_{\mathbf{z}'} \cap \mathbf{X}$, (this definition may also be non-deterministic because of a). Notice that $q^a_{\mathbf{z}'}$ is a Boolean CQ with possible equalities of individuals and inequalities, and we can define the bag answers of such a query q' over a bag interpretation $\mathcal{I}$ in the same way as for usual CQs in Definition 7 with the extra requirement that each contributing valuation λ should satisfy $\lambda(x) \neq \lambda(t)$ for each inequality $x \neq t$ of q' (and equalities of individuals are handled as usual equalities).

Definition 25. *Given equality-consistent variables* $\mathbf{z} \subseteq \mathbf{y}$*, ma-connected* $\mathbf{z}' \subseteq \mathbf{z}$ *are* realisable *by TBox* $\mathcal{T}$ *if*

$$(q^a_{\mathbf{z}'})^{\mathcal{C}(\langle \mathcal{T}, \mathcal{A}' \rangle)}(\langle \rangle) \geq 1,$$

where, for a fresh individual b*,* $\mathcal{A}'$ *is the bag ABox having either only the assertion* $P(a, b)$ *(with multiplicity 1), when* $\alpha_{\mathbf{z}'} = P(t, z)$*, or only* $P(b, a)$*, when* $\alpha_{\mathbf{z}'} = P(z, t)$*.*

This definition does not depend on the choice of $\alpha_{\mathbf{z}'}$ and a. Indeed, if there are two atoms $P_1(t_1, z_1)$ and $P_2(t_2, z_2)$ satisfying the definition of $\alpha_{\mathbf{z}'}$, then either $P_1 = P_2$ and both pairs (t_1, z_1) and (t_2, z_2) are mapped by a valuation of $q^a_{\mathbf{z}'}$ to the same tuple, or $\mathbf{z}'$ are not realisable regardless of the choice of $\alpha_{\mathbf{z}'}$. Similarly, if $\mathbf{t}_{\mathbf{z}'}$ contains two individuals a, a', then $q^a_{\mathbf{z}'}$ has the equality $a = a'$, and hence $\mathbf{z}'$ are not realisable regardless of this choice.

Intuitively, $\mathbf{z}'$ are realisable if their corresponding subquery $q^a_{\mathbf{z}'}$ is satisfied by the tree-shaped model induced by the TBox from a connection $\alpha_{\mathbf{z}'}$ of $\mathbf{z}'$ and the rest of the query. This definition does not essentially involve multiplicities, because all tuples of anonymous elements in the canonical model have multiplicity at most 1, and, hence, if $q^a_{\mathbf{z}'}$ matches a part of the canonical model, it does so in a unique way. Thus, checking realisability is decidable using standard set-based techniques.

Definition 26. *Variables* $\mathbf{z} \subseteq \mathbf{y}$ *are* realisable *by TBox* $\mathcal{T}$ *if they are equality-consistent and each non-empty ma-connected subset of* $\mathbf{z}$ *is realisable by* $\mathcal{T}$.

We proceed to Step 2. For realisable $\mathbf{z} \subseteq \mathbf{y}$, let $q_{\mathbf{z}}(\mathbf{x})$ be the CQ $\exists \mathbf{y}'. \phi_{\mathbf{z}}(\mathbf{x}, \mathbf{y}')$ such that $\phi_{\mathbf{z}}(\mathbf{x}, \mathbf{y}')$ is obtained from $\phi(\mathbf{x}, \mathbf{y})$ by replacing $\phi_{\mathbf{z}'}$, for each ma-connected $\mathbf{z}' \subseteq \mathbf{z}$, with

$$\alpha_{\mathbf{z}'} \quad \wedge \quad \bigwedge_{y \in \mathbf{t}_{\mathbf{z}'} \cap \mathbf{X},\, t \in \mathbf{t}_{\mathbf{z}'}} (y = t),$$

where $\mathbf{t}_{\mathbf{z}'}$ is as in $q^a_{\mathbf{z}'}$, and $\mathbf{y}'$ is the subset of $\mathbf{y}$ remaining in $\phi_{\mathbf{z}}$. In other words, $q_{\mathbf{z}}$ contains, for each $\mathbf{z}'$, just one atom $\alpha_{\mathbf{z}'}$ and equalities identifying $t_{\mathbf{z}'}$ instead of conjunction $\phi_{\mathbf{z}'}$ in q.

The following lemma justifies Steps 1 and 2. It says that in partitioning (1) we only need to iterate over tuples $\mathbf{z}$ that are realisable by $\mathcal{T}$ and can also replace q with $q_{\mathbf{z}}$ for each $\mathbf{z}$.

Lemma 27. *For any ontology* $\mathcal{K}$ *with TBox* $\mathcal{T}$ *and* $\mathbf{z} \subseteq \mathbf{y}$ *with* $q_{\mathbf{z}}(\mathbf{x}) = \exists \mathbf{y}'. \phi_{\mathbf{z}}(\mathbf{x}, \mathbf{y}')$,

1. *if* $\mathbf{z}$ *is realisable by* $\mathcal{T}$ *then* $[q, \mathbf{z}]^{\mathcal{C}(\mathcal{K})} = [q_{\mathbf{z}}, \mathbf{z} \cap \mathbf{y}']^{\mathcal{C}(\mathcal{K})}$;
2. *if* $\mathbf{z}$ *is not realisable by* $\mathcal{T}$ *then* $[q, \mathbf{z}]^{\mathcal{C}(\mathcal{K})} = \emptyset$.

For Step 3, it suffices to rewrite each CQ $q_{\mathbf{z}}(\mathbf{x}) = \exists \mathbf{y}'. \phi_{\mathbf{z}}(\mathbf{x}, \mathbf{y}')$ to a $\mathrm{BALG}^1_\varepsilon$-query $\bar{q}_{\mathbf{z}}(\mathbf{x}) = \exists \mathbf{y}_{\mathbf{z}}. \psi_{\mathbf{z}}(\mathbf{x}, \mathbf{y}_{\mathbf{z}})$, for $\mathbf{y}_{\mathbf{z}} = \mathbf{y}' \setminus \mathbf{z}$, which is guaranteed to give $[q_{\mathbf{z}}, \mathbf{z} \cap \mathbf{y}']^{\mathcal{C}(\mathcal{K})}$ as the bag answers on the ABox in any ontology $\mathcal{K}$ with TBox $\mathcal{T}$. To this end, we use the following notation: for $t \in \mathbf{X} \cup \mathbf{I}$, let $\zeta_A(t) = A(t)$ for $A \in \mathbf{C}$, while $\zeta_{\exists P}(t) = \exists y. P(t, y)$ and $\zeta_{\exists P^-}(t) = \exists y. P(y, t)$ for $P \in \mathbf{R}$, where y is a variable different from t. Then, formula $\psi_{\mathbf{z}}(\mathbf{x}, \mathbf{y}_{\mathbf{z}})$ is obtained from $\phi_{\mathbf{z}}(\mathbf{x}, \mathbf{y}')$ by replacing all atoms mentioning a term $t \in \mathbf{I} \cup \mathbf{x} \cup \mathbf{y}_{\mathbf{z}}$ or a variable $z \in \mathbf{z}$ as follows:

- each $A(t)$ with $\bigvee_{\mathcal{T} \models C \sqsubseteq A} \zeta_C(t)$;
- each $P(t, z)$ with $\big(\bigvee_{\mathcal{T} \models C \sqsubseteq \exists P} \zeta_C(t)\big) \setminus \zeta_{\exists P}(t)$;
- each $P(z, t)$ with $\big(\bigvee_{\mathcal{T} \models C \sqsubseteq \exists P^-} \zeta_C(t)\big) \setminus \zeta_{\exists P^-}(t)$.

Note that $\phi_{\mathbf{z}}(\mathbf{x}, \mathbf{y}')$ does not contain any atoms of the form $A(z)$ for $z \in \mathbf{z}$, so $\psi_{\mathbf{z}}(\mathbf{x}, \mathbf{y}_{\mathbf{z}})$ does not mention variables $\mathbf{z}$. Also, atoms over roles without variables $\mathbf{z}$ stay intact, because $\mathcal{T}$ contains no role inclusions.

Finally, the rewriting of $q(\mathbf{x})$ over $\mathcal{T}$ is the $\mathrm{BALG}^1_\varepsilon$-query

$$\bar{q}(\mathbf{x}) = \textstyle\bigvee_{\mathbf{z} \text{ realisable by } \mathcal{T}} \bar{q}_{\mathbf{z}}(\mathbf{x}).$$

Example 28. Consider TBox $\mathcal{T}_r$ from Example 17 and the rooted CQ $q^r(x) = \exists y. \mathsf{hasMngr}(x, y) \wedge \mathsf{Mngr}(y)$. The query $\bar{q}^r(x) = \bar{q}^r_{\langle\rangle}(x) \vee \bar{q}^r_y(x)$, where $\bar{q}^r_{\langle\rangle}(x)$ and $\bar{q}^r_y(x)$ are

$$\exists y. \mathsf{hasMngr}(x, y) \wedge \big(\mathsf{Mngr}(y) \vee \exists z. \mathsf{hasMngr}(z, y)\big) \text{ and}$$
$$\big(\mathsf{Emp}(x) \vee \exists y. \mathsf{hasMngr}(x, y)\big) \setminus \exists y. \mathsf{hasMngr}(x, y),$$

is a rewriting of q^r over $\mathcal{T}_r$, since $\langle\rangle$ and y are realisable. $\diamondsuit$

The following theorem establishes the correctness of our approach and leads to the main rewritability result.

Theorem 29. *For any rooted CQ* q *and* $\textit{DL-Lite}^{bag}_{core}$ *ontology* $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ *we have that* $q^{\mathcal{C}(\mathcal{K})} = \bar{q}^{\mathcal{C}(\langle \emptyset, \mathcal{A} \rangle)}$.

Corollary 30. *The class of rooted CQs is rewritable to* $\mathrm{BALG}^1_\varepsilon$*-queries for* $\textit{DL-Lite}^{bag}_{core}$.

We conclude this section by establishing the complexity of rooted query answering. The bounds follow as an easy consequence of Theorem 18, Proposition 22, and Corollary 30.

Theorem 31. $\textsc{BagCert}[\text{rooted CQs}, \textit{DL-Lite}^{bag}_{core}]$ *is* NP*-complete and in* LOGSPACE *in data complexity.*

However, the next theorem implies that rooted queries are not $\mathrm{BALG}^1_\varepsilon$-rewritable for unrestricted $\textit{DL-Lite}^{bag}_{\mathcal{R}}$ TBoxes.

Theorem 32. $\textsc{BagCert}[\text{rooted CQs}, \textit{DL-Lite}^{bag}_{\mathcal{R}}]$ *is* coNP*-hard in data complexity.*

6 Related work

Query answering under bag semantics has received significant attention in the database literature [Libkin and Wong, 1994; Grumbach *et al.*, 1996; Grumbach and Milo, 1996; Libkin and Wong, 1997]. These works study the relative expressive power of bag algebra primitives, the relationship with set-based algebras, and establish the data complexity of query answering. Such problems have also been recently studied in the setting of Semantic Web and SPARQL 1.1 in [Kaminski *et al.*, 2016; Angles and Gutierrez, 2016].

Bag semantics in the context of Description Logics has been studied in [Jiang, 2010], where the author proposes a bag semantics for $\mathcal{ALC}$ and provides a tableaux algorithm. In contrast to our work, their results are restricted to ontology satisfiability and do not encompass CQ answering.

CQ answering under bag semantics is closely related to answering Count aggregate queries. The semantics of aggregate queries for database settings with incomplete information, such as inconsistent databases and data exchange, have been studied in [Arenas *et al.*, 2003; Libkin, 2006; Afrati and Kolaitis, 2008]. As pointed out in [Kostylev and Reutter, 2015], these techniques are not directly applicable to ontologies. The practical solution in [Calvanese *et al.*, 2008] is to give epistemic semantics to aggregate queries, where the query is evaluated over ABox facts entailed by the ontology; thus, the anonymous part of the ontology models is essentially ignored, and the semantics easily leads to counter-intuitive answers. To remedy these issues, [Kostylev and Reutter, 2015] propose a certain answer semantics for Count aggregate queries over ontologies and prove tight complexity bounds for $\textit{DL-Lite}_{\mathcal{R}}$ and $\textit{DL-Lite}_{core}$. Similarly to our work, their semantics is open-world and considers all models of the ontology for query evaluation, which leads to more intuitive answers. The main difference resides in the definition of the ontology language, where they consider set ABoxes and adopt conventional set-based semantics for TBox axioms. Although $\textit{DL-Lite}^{bag}_{\mathcal{R}}$ is closely related to the logic in [Kostylev and Reutter, 2015], the two settings do not coincide even for set ABoxes. For example, if $\mathcal{A}$ comprises only assertions $R(a, b)$ and $R(a, c)$ and $\mathcal{T}$ comprises axiom $\exists R \sqsubseteq B$, then the query over $\langle \mathcal{T}, \mathcal{A} \rangle$ that counts the number of individuals a in concept B returns 1 in the setting of [Kostylev and Reutter, 2015], while the corresponding $\textit{DL-Lite}^{bag}_{\mathcal{R}}$ query returns 2.

7 Conclusion and Future Work

We have studied OBDA under bag semantics and identified a general class of rewritable queries over $\textit{DL-Lite}^{bag}_{core}$ ontologies. As our framework covers already the class of Count aggregate queries, in future work we plan to extend it to capture further aggregate functions and more expressive ontologies.

References

[Afrati and Kolaitis, 2008] Foto N. Afrati and Phokion G. Kolaitis. Answering aggregate queries in data exchange. In *PODS*, 2008.

[Angles and Gutierrez, 2016] Renzo Angles and Claudio Gutierrez. The multiset semantics of SPARQL patterns. In *ISWC*, 2016.

[Arenas *et al.*, 2003] Marcelo Arenas, Leopoldo E. Bertossi, Jan Chomicki, Xin He, Vijay Raghavan, and Jeremy P. Spinrad. Scalar aggregation in inconsistent databases. *Theor. Comput. Sci.*, 296(3):405–434, 2003.

[Artale *et al.*, 2009] Alessandro Artale, Diego Calvanese, Roman Kontchakov, and Michael Zakharyaschev. The DL-Lite family and relations. *J. Artif. Intell. Res. (JAIR)*, 36:1–69, 2009.

[Bienvenu *et al.*, 2012] Meghyn Bienvenu, Carsten Lutz, and Frank Wolter. Query containment in description logics reconsidered. In *KR*, 2012.

[Calì *et al.*, 2013] Andrea Calì, Georg Gottlob, and Michael Kifer. Taming the infinite chase: Query answering under expressive relational constraints. *J. Artif. Intell. Res. (JAIR)*, 48:115–174, 2013.

[Calvanese *et al.*, 2007] Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati. Tractable reasoning and efficient query answering in description logics: The *DL-Lite* family. *J. Autom. Reasoning*, 39(3):385–429, 2007.

[Calvanese *et al.*, 2008] Diego Calvanese, Evgeny Kharlamov, Werner Nutt, and Camilo Thorne. Aggregate queries over ontologies. In *Proceedings of the 2nd International Workshop on Ontologies and Information Systems for the Semantic Web, ONISW*, 2008.

[Calvanese *et al.*, 2011] Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, Antonella Poggi, Mariano Rodriguez-Muro, Riccardo Rosati, Marco Ruzzi, and Domenico Fabio Savo. The MASTRO system for ontology-based data access. *Semantic Web*, 2(1):43–53, 2011.

[Calvanese *et al.*, 2017] Diego Calvanese, Benjamin Cogrel, Sarah Komla-Ebri, Roman Kontchakov, Davide Lanti, Martin Rezk, Mariano Rodriguez-Muro, and Guohui Xiao. Ontop: Answering SPARQL queries over relational databases. *Semantic Web*, 8(3):471–487, 2017.

[Chaudhuri and Vardi, 1993] Surajit Chaudhuri and Moshe Y. Vardi. Optimization of *Real* conjunctive queries. In *PODS*, 1993.

[García-Molina *et al.*, 2009] Hector García-Molina, Jeffrey D. Ullman, and Jennifer Widom. *Database Systems: The Complete Book*. Pearson Education, 2nd edition, 2009.

[Gottlob *et al.*, 2014] Georg Gottlob, Stanislav Kikot, Roman Kontchakov, Vladimir V. Podolskii, Thomas Schwentick, and Michael Zakharyaschev. The price of query rewriting in ontology-based data access. *Artif. Intell.*, 213:42–59, 2014.

[Grumbach and Milo, 1996] Stéphane Grumbach and Tova Milo. Towards tractable algebras for bags. *J. Comput. Syst. Sci.*, 52(3):570–588, 1996.

[Grumbach *et al.*, 1996] Stéphane Grumbach, Leonid Libkin, Tova Milo, and Limsoon Wong. Query languages for bags: expressive power and complexity. *SIGACT News*, 27(2):30–44, 1996.

[Jiang, 2010] Yuncheng Jiang. Description logics over multisets. In *6th International Workshop on Uncertainty Reasoning for the Semantic Web (URSW)*, 2010.

[Kaminski *et al.*, 2016] Mark Kaminski, Egor V. Kostylev, and Bernardo Cuenca Grau. Semantics and expressive power of subqueries and aggregates in SPARQL 1.1. In *WWW*, 2016.

[Kikot *et al.*, 2012] Stanislav Kikot, Roman Kontchakov, and Michael Zakharyaschev. Conjunctive query answering with OWL 2 QL. In *Principles of Knowledge Representation and Reasoning: Proceedings of the Thirteenth International Conference, KR*, 2012.

[Kikot *et al.*, 2014] Stanislav Kikot, Roman Kontchakov, Vladimir V. Podolskii, and Michael Zakharyaschev. On the succinctness of query rewriting over shallow ontologies. In *LICS*, 2014.

[Kontchakov *et al.*, 2014] Roman Kontchakov, Martin Rezk, Mariano Rodriguez-Muro, Guohui Xiao, and Michael Zakharyaschev. Answering SPARQL queries over databases under OWL 2 QL entailment regime. In *ISWC*, 2014.

[Kostylev and Reutter, 2015] Egor V. Kostylev and Juan L. Reutter. Complexity of answering counting aggregate queries over DL-Lite. *J. Web Sem.*, 33:94–111, 2015.

[Lenzerini, 2002] Maurizio Lenzerini. Data integration: A theoretical perspective. In *PODS*, 2002.

[Libkin and Wong, 1994] Leonid Libkin and Limsoon Wong. New techniques for studying set languages, bag languages and aggregate functions. In *PODS*, 1994.

[Libkin and Wong, 1997] Leonid Libkin and Limsoon Wong. Query languages for bags and aggregate functions. *JCSS*, 55(2):241–272, 1997.

[Libkin, 2006] Leonid Libkin. Data exchange and incomplete information. In *PODS*, 2006.

[Motik *et al.*, 2012] Boris Motik, Bernardo Cuenca Grau, Ian Horrocks, Zhe Wu, Achille Fokoue, and Carsten Lutz. OWL 2 Web Ontology Language Profiles (Second Edition). *W3C Recommendation*, 2012.

[Nikolaou *et al.*, 2017] Charalampos Nikolaou, Egor V. Kostylev, George Konstantinidis, Mark Kaminski, Bernardo Cuenca Grau, and Ian Horrocks. The Bag Semantics of Ontology-Based Data Access. *CoRR*, abs/1705.07105, 2017.

[Poggi *et al.*, 2008] Antonella Poggi, Domenico Lembo, Diego Calvanese, Giuseppe De Giacomo, Maurizio Lenzerini, and Riccardo Rosati. Linking data to ontologies. *J. Data Semantics*, 10:133–173, 2008.