



HAL
open science

Querying and Repairing Inconsistent Prioritized Knowledge Bases: Complexity Analysis and Links with Abstract Argumentation

Meghyn Bienvenu, Camille Bourgaux

► **To cite this version:**

Meghyn Bienvenu, Camille Bourgaux. Querying and Repairing Inconsistent Prioritized Knowledge Bases: Complexity Analysis and Links with Abstract Argumentation. KR 2020 - 17th International Conference on Principles of Knowledge Representation and Reasoning, Sep 2020, Rhodes, Greece. pp.141-151. hal-02947251v1

HAL Id: hal-02947251

<https://inria.hal.science/hal-02947251v1>

Submitted on 24 Sep 2020 (v1), last revised 7 Jun 2024 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Querying and Repairing Inconsistent Prioritized Knowledge Bases: Complexity Analysis and Links with Abstract Argumentation

Meghyn Bienvenu¹, Camille Bourgaux²

¹ CNRS & University of Bordeaux, France

² DI ENS, ENS, CNRS, PSL University & Inria, Paris, France
meghyn.bienvenu@u-bordeaux.fr, camille.bourgaux@ens.fr

Abstract

In this paper, we explore the issue of inconsistency handling over prioritized knowledge bases (KBs), which consist of an ontology, a set of facts, and a priority relation between conflicting facts. In the database setting, a closely related scenario has been studied and led to the definition of three different notions of optimal repairs (global, Pareto, and completion) of a prioritized inconsistent database. After transferring the notions of globally-, Pareto- and completion-optimal repairs to our setting, we study the data complexity of the core reasoning tasks: query entailment under inconsistency-tolerant semantics based upon optimal repairs, existence of a unique optimal repair, and enumeration of all optimal repairs. Our results provide a nearly complete picture of the data complexity of these tasks for ontologies formulated in common DL-Lite dialects. The second contribution of our work is to clarify the relationship between optimal repairs and different notions of extensions for (set-based) argumentation frameworks. Among our results, we show that Pareto-optimal repairs correspond precisely to stable extensions (and often also to preferred extensions), and we propose a novel semantics for prioritized KBs which is inspired by grounded extensions and enjoys favourable computational properties. Our study also yields some results of independent interest concerning preference-based argumentation frameworks.

1 Introduction

Ontology-mediated query answering (OMQA) improves data access through the use of an ontology, which provides a convenient vocabulary for query formulation and captures domain knowledge that is exploited during query evaluation (Poggi et al. 2008; Bienvenu and Ortiz 2015; Xiao et al. 2018). There is now a large literature on OMQA, with much of the work adopting description logics (DLs) (Baader et al. 2017), or the closely related OWL (2) standard (OWL Working Group 2009), as the ontology specification language. In particular, the DL-Lite family of lightweight DLs (Calvanese et al. 2007; Artale et al. 2009) (which underpin the OWL 2 QL profile (Motik et al. 2012)) has been a main focus of both theoretical and practical research due to its favourable computational properties.

While it is often reasonable to assume that the ontology has been debugged and contains only trusted knowledge, real-world datasets are plagued by data quality issues,

which may render the KB inconsistent. This has led to a large body of work on how to handle data inconsistencies in OMQA, with the proposal of several inconsistency-tolerant semantics to provide meaningful answers to queries posed over inconsistent KBs (see e.g. the surveys (Bienvenu and Bourgaux 2016) and (Bienvenu 2019)). Many of these semantics are based upon some notion of repair, defined as an inclusion-maximal subset of the data that is consistent with the ontology. This is in particular the case for the AR, brave, and IAR semantics, which correspond respectively to a query answer holding w.r.t. all repairs, at least one repair, or the intersection of all repairs (Lembo et al. 2010; Bienvenu and Rosati 2013). The computational properties of these and other semantics are now quite well understood, and some practical algorithms and implementations have begun to be developed, see e.g. (Lembo et al. 2015; Bienvenu, Bourgaux, and Goasdoué 2019; Tsalapati et al. 2016; Trivela, Stoilos, and Vassalos 2018).

In many scenarios, there is some information about the dataset that can be used to select the most relevant repairs (e.g. we might know which relations or sources are most reliable, or the likelihood of certain kinds of facts being correct). To exploit such information, variants of the preceding semantics have been considered in which we restrict attention to the most preferred repairs based upon cardinality, weight, or a stratification of the dataset into priority levels (Bienvenu, Bourgaux, and Goasdoué 2014). While the latter forms of preferences are quite natural, there are other relevant ways of defining preferred repairs that are worth exploring. In particular, in the database area, the seminal work of Staworko, Chomicki, and Marcinkowski (2012) supposes that preferences are given in terms of a priority relation (i.e. acyclic binary relation) between conflicting facts. Such ‘fact-level’ preferences have been shown to naturally arise in applications like information extraction and can e.g. be declaratively specified using rules (Fagin et al. 2016). To lift a priority relation between facts to the level of repairs, three different methods were proposed by Staworko et al. Pareto-optimal and globally-optimal repairs are defined as those for which there is no possible substitution of facts that leads to a (Pareto / global) improvement, while completion-optimal repairs correspond to

greedily constructing a repair based upon some compatible total order (see Section 3 for formal definitions). The complexity of reasoning with these three kinds of optimal repair has been investigated, primarily focusing on constraints given as functionality dependencies (Fagin, Kimelfeld, and Kolaitis 2015; Kimelfeld, Livshits, and Peterfreund 2017; Livshits and Kimelfeld 2017).

In this paper, we explore the use of fact-level preferences and optimal repairs for inconsistency handling in OMQA. Our first contribution is a data complexity analysis of the central reasoning problems related to optimal repairs: conjunctive query entailment under inconsistency-tolerant semantics based upon optimal repairs, uniqueness of optimal repairs, and enumeration of all optimal repairs. Our results provide a nearly complete picture of the data complexity for the three types of optimal repair and for ontologies formulated in DL-Lite_{core} and DL-Lite_{horn}^ℓ; however, as we make precise later, many of our results transfer to other data-tractable ontology languages. Not surprisingly, we find that reasoning with optimal repairs is generally more challenging than for standard repairs. In particular, query entailment under variants of the AR, IAR, and brave semantics based upon any of the three notions of optimal repair is intractable in data complexity, whereas with standard repairs, the IAR and brave semantics allow for tractable query answering.

Our second contribution is to establish connections with abstract argumentation frameworks (AFs). More precisely, we show that every prioritized KB with only binary conflicts naturally corresponds to a preference-based AF (Kaci, van der Torre, and Villata 2018). To handle general prioritized KBs, we need preference-based set-based AFs (SETAFs), which we introduce as a natural extension of SETAFs (Flouris and Bikakis 2019). This correspondence enables us to compare repairs and extensions. We determine that Pareto-optimal repairs are precisely the stable extensions of the corresponding preference-based (SET)AF, and under reasonable assumptions, also coincide with the preferred extensions. To establish the latter result, we prove a technically challenging result about symmetric preference-based SETAFs that we believe is of independent interest. Globally-optimal and completion-optimal repairs correspond to proper subsets of the stable extensions, but do not at present have any analog in the argumentation setting.

The argumentation connection situates optimal repairs within a broader context and lays the foundations for importing ideas and results from the argumentation literature. Indeed, our third contribution is to propose a new notion of grounded repair, directly inspired by grounded extensions from argumentation. We show that the (unique) grounded repair is contained in the intersection of Pareto-optimal repairs. As the grounded repair can be computed in polynomial time from the dataset and conflicts, it yields a tractable approximation of the Pareto variant of the IAR semantics for the considered DLs. Moreover, we show that it is more productive than the recently proposed Elect semantics (Belabbes, Benferhat, and Chomicki 2019; Belabbes and Benferhat 2019). These advantages motivate us to take a closer look at the computational properties of the grounded semantics. We prove in particular a matching P lower bound

and show how the semantics can be computed via the well-founded semantics of logic programs.

Proofs are omitted due to lack of space but can be found in the long version (Bienvenu and Bourgaux 2020).

2 Preliminaries

Even if many of our results apply to more general settings, our focus is on description logic (DL) knowledge bases, and we will in particular consider the DL-Lite_{core} and DL-Lite_{horn}^ℓ dialects of the DL-Lite family.

Syntax. A DL *knowledge base* (KB) $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ consists of an ABox \mathcal{A} and a TBox \mathcal{T} , both constructed from a set N_C of *concept names* (unary predicates), a set of N_R of *role names* (binary predicates), and a set N_I of *individuals* (constants). The ABox (dataset) consists of a finite number of *concept assertions* of the form $A(a)$ and *role assertions* of the form $R(a, b)$, where $A \in N_C$, $R \in N_R$, $a, b \in N_I$. The TBox (ontology) consists of a set of axioms whose form depends on the DL in question. In DL-Lite_{core}, TBox axioms are *concept inclusions* $B \sqsubseteq C$ built according to the following grammar, where $A \in N_C$ and $R \in N_R$:

$$B := A \mid \exists S, \quad C := B \mid \neg B, \quad S := R \mid R^-.$$

DL-Lite_{horn}^ℓ extends DL-Lite_{core} with *role inclusions* of the form $S \sqsubseteq Q$ where $Q := S \mid \neg S$ and concept inclusions of the form $B_1 \sqcap \dots \sqcap B_n \sqsubseteq C$.

Semantics. An *interpretation* has the form $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where $\Delta^{\mathcal{I}}$ is a non-empty set and $\cdot^{\mathcal{I}}$ maps each $A \in N_C$ to $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$, each $R \in N_R$ to $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$, and each $a \in N_I$ to $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$. The function $\cdot^{\mathcal{I}}$ is straightforwardly extended to general concepts and roles, e.g. $(\neg B_1)^{\mathcal{I}} = \Delta^{\mathcal{I}} \setminus B_1^{\mathcal{I}}$, $(B_1 \sqcap B_2)^{\mathcal{I}} = B_1^{\mathcal{I}} \cap B_2^{\mathcal{I}}$, $(R^-)^{\mathcal{I}} = \{(c, d) \mid (d, c) \in R^{\mathcal{I}}\}$ and $(\exists Q)^{\mathcal{I}} = \{c \mid \exists d : (c, d) \in Q^{\mathcal{I}}\}$. An interpretation \mathcal{I} satisfies $G \sqsubseteq H$ if $G^{\mathcal{I}} \subseteq H^{\mathcal{I}}$; it satisfies $A(a)$ (resp. $R(a, b)$) if $a^{\mathcal{I}} \in A^{\mathcal{I}}$ (resp. $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}}$). We call \mathcal{I} a *model* of $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ if \mathcal{I} satisfies all axioms in \mathcal{T} and assertions in \mathcal{A} . A KB \mathcal{K} is *consistent* if it has a model; otherwise it is inconsistent. We say that an ABox \mathcal{A} is \mathcal{T} -*consistent* if the KB $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ is consistent.

Queries. A *first-order (FO) query* is a first-order logic formula whose atoms are built using the predicate symbols in $N_C \cup N_R$ and constants in N_I . We will focus on the subclass of *conjunctive queries* (CQs) which take the form $q(\vec{x}) = \exists \vec{y} \psi(\vec{x}, \vec{y})$, where ψ is a conjunction of atoms of the forms $A(t)$ or $R(t, t')$ whose terms are either variables from $\vec{x} \cup \vec{y}$ or individuals. When a CQ consists of a single atom, we called it an instance query (IQ). A query without free variables is called *Boolean*, and we refer to Boolean CQs as BCQs. A Boolean query q is *satisfied* by an interpretation \mathcal{I} , written $\mathcal{I} \models q$, iff q is satisfied in \mathcal{I} according to standard first-order logic semantics; q is *entailed* by a KB \mathcal{K} , written $\mathcal{K} \models q$, iff $\mathcal{I} \models q$ for every model \mathcal{I} of \mathcal{K} . Henceforth, when we speak of a (Boolean) query, without specifying the type of query, we mean a (Boolean) CQ.

3 Querying Inconsistent Prioritized KBs

An inconsistent KB entails every Boolean query under the standard semantics. Several inconsistency-tolerant semantics have been defined to obtain meaningful answers in this context (see (Bienvenu and Bourgaux 2016) for a survey). Most of these semantics are based on the notion of *repairs*, which correspond to the different ways of restoring consistency by minimally removing some assertions.

Definition 1 (Repair). An ABox repair of a KB $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$, or repair for short, is an inclusion-maximal \mathcal{T} -consistent subset of \mathcal{A} . The set of repairs of \mathcal{K} is denoted by $SRep(\mathcal{K})$.

Another central notion is that of *conflicts*, which are the minimal subsets of assertions that contradict the TBox.

Definition 2 (Conflict). A conflict of a KB $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$, is a minimal \mathcal{T} -inconsistent subset of \mathcal{A} . The set of all conflicts of \mathcal{K} is denoted by $Conf(\mathcal{K})$ and can be represented as a conflict hypergraph whose vertices are assertions and whose hyperedges are the conflicts.

To simplify the presentation, we assume throughout the paper that the ABox does not contain any assertion that is inconsistent with the TBox, i.e., every conflict contains at least two assertions. Note that self-contradictory assertions do not occur in any repair, and such assertions can be readily identified and removed using existing reasoning algorithms.

In the database setting, Staworko, Chomicki, and Marcinkowski (2012) defined three notions of *optimal repairs* based on a *priority relation* over facts, which expresses preferences between conflicting facts. Such a relation could e.g. represent expert knowledge of how to resolve certain kinds of conflicts or be learned from manual conflict resolution (Martinez et al. 2014; Tanon, Bourgaux, and Suchanek 2019). We recast these notions in the KB setting.

Definition 3 (Priority relation, completion). Given a KB $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$, a priority relation \succ for \mathcal{K} is an acyclic binary relation over the assertions of \mathcal{A} such that if $\alpha \succ \beta$, then there exists $C \in Conf(\mathcal{K})$ such that $\{\alpha, \beta\} \subseteq C$. A priority relation \succ is total iff for all $\alpha \neq \beta$, if there exists $C \in Conf(\mathcal{K})$ such that $\{\alpha, \beta\} \subseteq C$, then $\alpha \succ \beta$ or $\beta \succ \alpha$. A completion of \succ is a total priority relation $\succ' \supseteq \succ$.

Definition 4 (Prioritized KB). A prioritized KB is a pair (\mathcal{K}, \succ) consisting of a KB \mathcal{K} and priority relation \succ for \mathcal{K} . The notation \mathcal{K}_\succ will be used as shorthand for (\mathcal{K}, \succ) .

Definition 5 (Optimal repairs). Consider a prioritized KB \mathcal{K}_\succ with $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$, and let $\mathcal{A}' \in SRep(\mathcal{K})$.

- A Pareto improvement of \mathcal{A}' (w.r.t. \succ) is a \mathcal{T} -consistent $\mathcal{A}'' \subseteq \mathcal{A}$ such that there exists $\beta \in \mathcal{A}' \setminus \mathcal{A}''$ such that $\beta \succ \alpha$ for every $\alpha \in \mathcal{A}' \setminus \mathcal{A}''$.
- A global improvement of \mathcal{A}' (w.r.t. \succ) is a \mathcal{T} -consistent $\mathcal{A}'' \subseteq \mathcal{A}$ such that $\mathcal{A}'' \neq \mathcal{A}'$ and for every $\alpha \in \mathcal{A}' \setminus \mathcal{A}''$, there exists $\beta \in \mathcal{A}'' \setminus \mathcal{A}'$ such that $\beta \succ \alpha$.

The repair \mathcal{A}' is a:

- Pareto-optimal repair of \mathcal{K}_\succ if there is no Pareto improvement of \mathcal{A}' w.r.t. \succ .
- Globally-optimal repair of \mathcal{K}_\succ if there is no global improvement of \mathcal{A}' w.r.t. \succ .

- Completion-optimal repair of \mathcal{K}_\succ if \mathcal{A}' is a globally-optimal repair of $\mathcal{K}_{\succ'}$, for some completion \succ' of \succ .

We denote by $GRep(\mathcal{K}_\succ)$, $PRep(\mathcal{K}_\succ)$ and $CRep(\mathcal{K}_\succ)$ the sets of globally-, Pareto-, and completion-optimal repairs.

Staworko, Chomicki, and Marcinkowski (2012) showed the following relation between optimal repairs:

$$CRep(\mathcal{K}_\succ) \subseteq GRep(\mathcal{K}_\succ) \subseteq PRep(\mathcal{K}_\succ) \subseteq SRep(\mathcal{K}).$$

They also show that \mathcal{A}' is a completion-optimal repair iff it can be obtained by the following greedy procedure: while some assertion has not been considered, pick an assertion that is maximal w.r.t. \succ among those not yet considered, and add it to the current set if it does not lead to a contradiction.

Example 6. Consider the prioritized KB \mathcal{K}_\succ given on Figure 1. The sets of optimal repairs of \mathcal{K}_\succ are as follows.

$$\begin{aligned} CRep(\mathcal{K}_\succ) &= \{ \{ \text{Rattler}(a), \text{Carnivorous}(a), \text{EatMeat}(a, b) \}, \\ &\quad \{ \text{Boa}(a), \text{Carnivorous}(a), \text{EatMeat}(a, b) \} \} \\ GRep(\mathcal{K}_\succ) &= CRep(\mathcal{K}_\succ) \cup \{ \{ \text{Mammal}(a), \text{Oviparous}(a), \\ &\quad \text{Carnivorous}(a), \text{EatMeat}(a, b) \} \} \\ PRep(\mathcal{K}_\succ) &= GRep(\mathcal{K}_\succ) \cup \{ \\ &\quad \{ \text{Rattler}(a), \text{Herbivorous}(a), \text{EatPlant}(a, b) \}, \\ &\quad \{ \text{Boa}(a), \text{Herbivorous}(a), \text{EatPlant}(a, b) \}, \\ &\quad \{ \text{Mammal}(a), \text{Oviparous}(a), \\ &\quad \text{Herbivorous}(a), \text{EatPlant}(a, b) \} \} \end{aligned}$$

$SRep(\mathcal{K})$ additionally contains repairs with $\text{Stone}(b)$ instead of the $\text{EatPlant}(a, b)$ or $\text{EatMeat}(a, b)$ assertions.

We next introduce the three natural inconsistency-tolerant semantics based upon repairs. The AR semantics is arguably the most natural and well-known semantics and requires that a query be entailed from *every* repair of the KB. The IAR semantics is a more cautious semantics that evaluates queries over the *intersection* of all repairs. Finally, the brave semantics is the most adventurous semantics, returning yes to queries that are entailed by *some* repair.

Definition 7 (AR, IAR and brave semantics). For $X \in \{C, G, P, S\}$, a prioritized KB \mathcal{K}_\succ entails a query q under

- X-AR semantics, written $\mathcal{K}_\succ \models_{AR}^X q$,
iff $\langle \mathcal{T}, \mathcal{A}' \rangle \models q$ for every $\mathcal{A}' \in XRep(\mathcal{K}_\succ)$;
- X-IAR semantics, written $\mathcal{K}_\succ \models_{IAR}^X q$,
iff $\langle \mathcal{T}, \mathcal{A}^\cap \rangle \models q$ where $\mathcal{A}^\cap = \bigcap_{\mathcal{A}' \in XRep(\mathcal{K}_\succ)} \mathcal{A}'$;
- X-brave semantics, written $\mathcal{K}_\succ \models_{brave}^X q$,
iff $\langle \mathcal{T}, \mathcal{A}' \rangle \models q$ for some $\mathcal{A}' \in XRep(\mathcal{K}_\succ)$.

We remark that these semantics are related as follows:

$$\mathcal{K}_\succ \models_{IAR}^X q \Rightarrow \mathcal{K}_\succ \models_{AR}^X q \Rightarrow \mathcal{K}_\succ \models_{brave}^X q.$$

Example 8 (Ex. 6 cont'd). Considering the different semantics based upon completion-optimal repairs, we observe that

- $\mathcal{K}_\succ \models_{IAR}^C \text{Carnivorous}(a)$,
- $\mathcal{K}_\succ \models_{AR}^C \text{Snake}(a)$ but $\mathcal{K}_\succ \not\models_{IAR}^C \text{Snake}(a)$,
- $\mathcal{K}_\succ \models_{brave}^C \text{Boa}(a)$ but $\mathcal{K}_\succ \not\models_{AR}^C \text{Boa}(a)$.

$\mathcal{T} = \{ \text{Boa} \sqsubseteq \text{Snake}, \text{Boa} \sqsubseteq \text{Ovoviviparous},$
 $\text{Rattler} \sqsubseteq \text{Snake}, \text{Rattler} \sqsubseteq \text{Ovoviviparous},$
 $\text{EatPlant} \sqsubseteq \text{Eat}, \text{EatMeat} \sqsubseteq \text{Eat},$
 $\text{Mammal} \sqsubseteq \neg \text{Snake}, \text{Oviparous} \sqsubseteq \neg \text{Ovoviviparous},$
 $\text{Boa} \sqsubseteq \neg \text{Rattler}, \text{Carnivorous} \sqsubseteq \neg \text{Herbivorous},$
 $\text{Carnivorous} \sqsubseteq \neg \exists \text{EatPlant}, \text{Herbivorous} \sqsubseteq \neg \exists \text{EatMeat},$
 $\exists \text{EatPlant}^- \sqsubseteq \neg \exists \text{EatMeat}^-, \exists \text{Eat}^- \sqsubseteq \neg \text{Stone} \}$

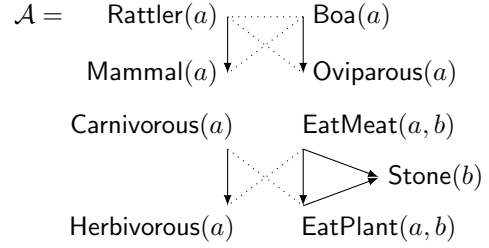


Figure 1: Prioritized KB example. \mathcal{T} formalizes knowledge about animal groups and \mathcal{A} gives information about individuals a and b . An arrow from α to β indicates that $\alpha \succ \beta$ while a dotted line indicates a conflict without priority between assertions.

If we consider now AR semantics for the different kinds of optimal repairs, we find that

- $\mathcal{K}_{\succ} \models_{AR}^C \text{Snake}(a)$ but $\mathcal{K}_{\succ} \not\models_{AR}^C \text{Snake}(a)$,
- $\mathcal{K}_{\succ} \models_{AR}^G \text{Carnivorous}(a)$ but $\mathcal{K}_{\succ} \not\models_{AR}^P \text{Carnivorous}(a)$,
- $\mathcal{K}_{\succ} \models_{AR}^P \text{Eat}(a, b)$ but $\mathcal{K}_{\succ} \not\models_{AR}^S \text{Eat}(a, b)$.

4 Complexity Analysis

In this section, we analyze the data complexity of the central reasoning tasks related to optimal repairs.

4.1 Overview

We define the problems and complexity classes we consider as well as some relevant properties of DL languages.

Problems. Given a prioritized KB \mathcal{K}_{\succ} and a type of repairs $X \in \{C, G, P, S\}$, *repair checking* (ISREP) is deciding whether a set of assertions belongs to $XRep(\mathcal{K}_{\succ})$; *AR (resp. IAR, brave) entailment* (AR, IAR, BRAVE) is deciding whether \mathcal{K}_{\succ} entails a given BCQ under X-AR (resp. X-IAR, X-brave) semantics; *uniqueness* (UNIQUE) is deciding whether $|XRep(\mathcal{K}_{\succ})| = 1$; and *enumeration* (ENUM) is enumerating all elements of $XRep(\mathcal{K}_{\succ})$.

Complexity classes. All of the complexity results stated in this paper are for the *data complexity* measure, i.e., the complexity is with respect to the size of \mathcal{A} , with \mathcal{T} and q treated as fixed (hence of constant size).

For decision problems, we will refer to the following complexity classes: AC^0 is the class of problems that can be solved by a uniform family of circuits of constant depth and polynomial-size, with unbounded-fanin AND and OR gates; P is the class of problems solvable in polynomial time; NP is the class of problems solvable in non-deterministic polynomial time; Σ_2^P is the class of problems solvable in non-deterministic polynomial time with access to an NP oracle; coNP (resp. Π_2^P) is the class of problems whose complement is in NP (resp. Σ_2^P). These classes are related as follow: $AC^0 \subset P \subseteq NP \subseteq \Sigma_2^P$ and $AC^0 \subset P \subseteq coNP \subseteq \Pi_2^P$. It is widely believed that all of the inclusions are proper.

For enumeration problems, we will use the following classes: TOTALP is the class of enumeration problems that can be solved in polynomial time in $n + m$ where n is the input size and m the output size; DELAYP is the class of enumeration problems that can be solved by an algorithm

	$SRep(\mathcal{K})$	$PRep(\mathcal{K}_{\succ})$	$GRep(\mathcal{K}_{\succ})$	$CRep(\mathcal{K}_{\succ})$
ISREP	in P	in P	coNP-c	in P
AR	coNP-c	coNP-c	Π_2^P -c	coNP-c
IAR	in AC^0	coNP-c	Π_2^P -c	coNP-c
BRAVE	in AC^0	NP-c	Σ_2^P -c	NP-c
UNIQUE	in P	coNP-c	Π_2^P -c [‡]	in P
ENUM	DELAYP* not	TOTALP [†] not	TOTALP [†]	DELAYP*

Table 1: Data complexity overview. Lower bounds hold for DL-Lite_{core}. Upper bounds hold for DL-Lite_{horn}[‡] except those noted with * which hold only for DL-Lite_{core}. Results for the $SRep(\mathcal{K})$ case are stated (or are straightforward consequences of results) in (Bienvenu and Bourgaux 2016; Bienvenu and Rosati 2013; Livshits and Kimelfeld 2017). [‡] in P if \succ is transitive [†] INCP if \succ is score-structured * INCP for DL-Lite_{horn}[‡] KBs with a score-structured priority relation \succ (see Section 4.4 for the definition)

such that the delay between the k^{th} and the $k + 1^{th}$ solution is bounded by a polynomial in the input size; INCP is the class of enumeration problems that can be solved by an algorithm such that the delay between the k^{th} and the $k + 1^{th}$ solution is bounded by a polynomial in the size of the input and k . These classes are related as follow: $DELAYP \subseteq INCP \subseteq TOTALP$. If the decision problem associated to an enumeration problem is NP-hard, then the enumeration problem is not in TOTALP.

Results. Table 1 presents an overview of the complexity results for DL-Lite. However, since most of the results hold for some DLs outside the DL-Lite family, we will state them for DLs that satisfy some of the following properties:

- BOUNDCONF: the size of conflicts is bounded;
- BINCONF: conflicts have size at most two;
- POLYBCQ: polynomial time BCQ entailment;
- POLYCONS: polynomial time consistency checking.

The most commonly considered DL-Lite dialects, such as DL-Lite_{core} and DL-Lite_R (the basis for OWL 2 QL), satisfy all of these properties, while DL-Lite_{horn}[‡] satisfies all except BINCONF (the size of conflicts is bounded by the number of concept and role names occurring in the TBox).

4.2 Query Entailment

We start by establishing the complexity of repair checking and BCQ entailment under the various semantics.

The upper bounds are stated in the following theorem and can be proven by adapting existing results from databases and inconsistency-tolerant OMQA (recall that proofs of all results are provided in (Bienvenu and Bourgaux 2020)). Observe that this result applies to all DLs satisfying POLYCONS and POLYBCQ, which includes prominent Horn DLs like \mathcal{EL} and Horn- \mathcal{SHIQ} .

Theorem 9. *For DLs satisfying POLYCONS, repair checking is in coNP for globally-optimal repairs, in P for Pareto or completion-optimal repairs.*

For DLs satisfying POLYCONS and POLYBCQ, G-AR and G-IAR entailment are in Π_2^p , G-brave entailment in Σ_2^p , P-AR, P-IAR, C-AR and C-IAR entailment are in coNP, and P-brave and C-brave entailment in NP.

For Pareto-optimal and completion-optimal repairs, coNP-hardness (resp. NP-hardness) of AR and IAR (resp. brave) IQ entailment in DL-Lite_{core} follow from the special case where the priority relation is given by priority levels, for which it is known the three families of optimal repairs we consider coincide (Bourgaux 2016). The following theorem establishes the remaining lower bounds.

Theorem 10. *In DL-Lite_{core}, repair checking is coNP-hard for globally-optimal repairs, G-AR and G-IAR entailment are Π_2^p -hard, and G-brave entailment Σ_2^p -hard, even for IQs.*

4.3 Uniqueness and Enumeration

We now turn our attention to the problems of deciding uniqueness and enumerating repairs. Note that in the case of classical repairs, deciding whether $|SRep(\mathcal{K})| = 1$ amounts to checking whether \mathcal{K} is consistent (recall our assumption that all self-contradictory assertions have been removed).

As observed by Livshits and Kimelfeld (2017) in the database setting, classical repairs correspond to the maximal independent sets (MISs) of the conflict hypergraph. It is known that MIS enumeration for graphs is in DELAYP (Johnson, Papadimitriou, and Yannakakis 1988), while for hypergraphs with bounded-size hyperedges, the problem is in INCP (Eiter and Gottlob 1995; Boros et al. 2000). Since the conflict hypergraph can be tractably computed for DLs with bounded-size conflicts, enumeration of $SRep(\mathcal{K})$ is in INCP for DLs with bounded conflicts, and in DELAYP for DLs with binary conflicts.

Kimelfeld, Livshits, and Peterfreund (2017) and Livshits and Kimelfeld (2017) provide intractability results for UNIQUE and ENUM in the case of globally- and Pareto-optimal repairs of databases with functional dependencies. Their proofs do not transfer to DL-Lite KBs, but via different proofs, we can establish analogous results in our context.

Theorem 11. *Uniqueness is coNP-complete for Pareto-optimal repairs, Π_2^p -complete for globally-optimal repairs. Upper bounds hold for DLs satisfying POLYCONS. Lower bounds hold for DL-Lite_{core}.*

Corollary 12. *Enumeration of Pareto-optimal or globally-optimal repairs is not in TOTALP for DL-Lite_{core}.*

Kimelfeld, Livshits, and Peterfreund (2017) provide an algorithm which, given an arbitrary conflict hypergraph and a priority relation, decides uniqueness for completion-optimal repairs in polynomial time. Hence deciding whether $|CRep(\mathcal{K}_\succ)| = 1$ is in P for DLs with bounded conflicts, since in this case the conflict hypergraph can be computed in polynomial time. Livshits and Kimelfeld (2017) describe an algorithm that enumerates completion-optimal repairs with polynomial delay for *conflict graphs*. It follows that for DLs with binary conflicts, enumeration of completion-optimal repairs is in DELAYP. The case of completion-optimal repairs with non-binary conflicts remains open.

4.4 Case of Transitive Priority Relations

A natural case to consider is when the priority relation is *transitive*. This occurs in particular when the priority relation captures the relative reliability of the facts.

Definition 13. *A priority relation is transitive if $\alpha_1 \succ \dots \succ \alpha_n$ and $\{\alpha_1, \alpha_n\} \subseteq C \in Conf(\mathcal{K})$ implies $\alpha_1 \succ \alpha_n$.*

Kimelfeld, Livshits, and Peterfreund (2017) proved that when the priority relation is transitive, deciding uniqueness for globally-optimal repairs can be done in polynomial time when the conflict hypergraph is given. This result is applicable to all DLs for which the conflict hypergraph is computable in polynomial time. A natural question is whether any of our other problems become easier if we assume \succ is transitive. We found that this is not the case, although the transitivity requirement makes the lower bounds proofs more involved. In particular, we can no longer use the coNP-hardness proof for uniqueness to derive that the enumeration of globally-optimal repairs is not in TOTALP, but instead show this via the coNP-hardness of deciding whether a given set of ABoxes is exactly $GRep(\mathcal{K}_\succ)$.

Theorem 14. *When \succ is transitive, uniqueness is in P for globally-optimal repairs and DLs satisfying BOUNDCONF. All other lower bounds in Table 1 still hold for DL-Lite_{core}.*

Score-structured priority relations. An interesting special case of transitive priority relation arises when every assertion α is assigned a natural number called its *score* (or *priority level*) $s(\alpha)$, which induces a priority relation as follows: for every pair of assertions α and β that belongs to a conflict, $\alpha \succ \beta$ if and only if $s(\alpha) > s(\beta)$. Livshits and Kimelfeld (2017) call such a priority relation *score-structured* and show that deciding whether a priority is score-structured is in P and that if \succ is score-structured, $CRep(\mathcal{K}_\succ) = GRep(\mathcal{K}_\succ) = PRep(\mathcal{K}_\succ)$. This relation was also given in (Bienvenu, Bourgaux, and Goasdoué 2014; Bourgaux 2016) where it is shown that they further coincide with another notion of preferred repairs directly defined from ABoxes partitioned into priority levels.

For score-structured priorities, the complexity of all decision problems is the same as in the case of completion-optimal repairs, with the lower bounds shown in (Bourgaux 2016). By contrast, for enumeration, it follows from the algorithm for enumerating score-structured priority repairs of Livshits and Kimelfeld that enumeration is in INCP for DLs with bounded conflicts and consistency checking in P.

5 Preference-Based Argumentation

In preparation for the following section, we recall the basics of argumentation frameworks and previously studied extensions with preferences and collective attacks. We also prove some new results (Theorems 22, 23, 28, and 32).

5.1 Argumentation Frameworks

We consider *finite argumentation frameworks*. Argumentation frameworks over an infinite set of arguments have also been studied in the literature but finiteness is an ordinary assumption and is made for all results we recall or extend here.

Definition 15. An argumentation framework (AF) is a pair $(Args, \rightsquigarrow)$ where $Args$ is a finite set of arguments and $\rightsquigarrow \subseteq Args \times Args$ is the attack relation. When $(\alpha, \beta) \in \rightsquigarrow$, we say that α attacks β , alternatively denoted by $\alpha \rightsquigarrow \beta$.

We recall some commonly used notation and terminology, letting $(Args, \rightsquigarrow)$ be the considered AF and $A \subseteq Args$. We use $A^+ = \{\beta \mid \alpha \rightsquigarrow \beta \text{ for some } \alpha \in A\}$ to denote the set of arguments attacked by arguments from A . We say that A defends $\gamma \in Args$ (or, γ is defended by A) iff $\{\beta \mid \beta \rightsquigarrow \gamma\} \subseteq A^+$. A set $A \subseteq Args$ is *conflict-free* if $A \cap A^+ = \emptyset$. The characteristic function $\Gamma_F : 2^{Args} \mapsto 2^{Args}$ of $F = (Args, \rightsquigarrow)$ is defined as follows: $\Gamma_F(A) = \{\alpha \mid \alpha \text{ is defended by } A\}$. A set $A \subseteq Args$ is *admissible* if it is conflict-free and $A \subseteq \Gamma_F(A)$.

Argumentation semantics are usually based upon *extensions*, intended to capture coherent sets of arguments. We recall some prominent notions of extension:

Definition 16. Let $F = (Args, \rightsquigarrow)$. Then $E \subseteq Args$ is a:

- complete extension iff E is conflict-free and $E = \Gamma_F(E)$;
- grounded extension iff E is the \subseteq -minimal complete extension, or equivalently, the least fixpoint of Γ_F ;
- preferred extension iff E is a \subseteq -maximal admissible set;
- stable extension iff $E^+ = Args \setminus E$.

While every stable extension is a preferred extension, the converse does not hold in general (Dung 1995). The term *coherent* designates AFs for which these two notions coincide (i.e. every preferred extension is stable). Coherence is viewed as a desirable property, and several sufficient conditions for coherence have been identified. In particular:

Definition 17. An AF $(Args, \rightsquigarrow)$ is symmetric iff \rightsquigarrow is symmetric and irreflexive.

Theorem 18. (Coste-Marquis, Devred, and Marquis 2005) Every symmetric AF is coherent.

5.2 Enriching AFs with Preferences

There has been significant interest in extending AFs to allow for preferences between arguments (Amgoud and Cayrol 2002; Kaci, van der Torre, and Villata 2018).

Definition 19. A preference-based argumentation framework (PAF) is a triple $(Args, \rightsquigarrow, \succ)$, where $Args$ and \rightsquigarrow are as before, and \succ is an acyclic binary relation over $Args$ (called the preference relation). Symmetric PAFs are obtained by requiring $(Args, \rightsquigarrow)$ to be symmetric.

It is typical to assume that the preference relation \succ is transitive. This is a reasonable assumption, but for the sake of generality, the preceding definition does not impose this.

The standard way of defining the semantics of PAFs is via reduction to (plain) AFs:

Definition 20. Given a PAF $(Args, \rightsquigarrow, \succ)$, the corresponding AF is $(Args, \rightsquigarrow_{\succ})$, where $\alpha \rightsquigarrow_{\succ} \beta$ iff $\alpha \rightsquigarrow \beta$ and $\beta \not\succ \alpha$. A subset $E \subseteq Args$ is a stable (resp. preferred, grounded, complete) extension of a PAF $(Args, \rightsquigarrow, \succ)$ iff it is a stable (resp. preferred, grounded, complete) extension of the corresponding AF.

While alternative reductions have been proposed in (Amgoud and Vesic 2014) and (Kaci, van der Torre, and Villata 2018), all of these reductions coincide for symmetric PAFs, which is the case that will interest us here.

Theorem 21. (Amgoud and Vesic 2014) Every symmetric PAF with a transitive preference relation is coherent.

We generalize the preceding result by showing that transitivity is not required:

Theorem 22. Every symmetric PAF is coherent.

Theorem 23 gives a characterization of AFs that correspond to symmetric PAFs. It generalizes the *strict acyclicity* condition characterizing the so-called *conflict+preference* AFs (Kaci, van der Torre, and Weydert 2006) that requires for any cycle $\alpha_1 \rightsquigarrow \alpha_2 \rightsquigarrow \dots \rightsquigarrow \alpha_n \rightsquigarrow \alpha_1$ that $\alpha_1 \rightsquigarrow \alpha_n \rightsquigarrow \dots \rightsquigarrow \alpha_2 \rightsquigarrow \alpha_1$.

Theorem 23. For every AF $F = (Args, \rightsquigarrow)$, the following statements are equivalent.

1. F is the corresponding AF of some symmetric PAF.
2. For any cycle $\alpha_1 \rightsquigarrow \alpha_2 \rightsquigarrow \dots \rightsquigarrow \alpha_n \rightsquigarrow \alpha_1$, there exists $(j, i) \in \{(1, 2), \dots, (n-1, n), (n, 1)\}$ with $\alpha_i \rightsquigarrow \alpha_j$.

5.3 Set-Based Argumentation Frameworks

Another well-studied extension of AFs is to allow collective attacks, in which a set of arguments together form an attack (Nielsen and Parsons 2006; Flouris and Bikakis 2019).

Definition 24. A set-based argumentation framework (SETAF) is a pair $(Args, \rightsquigarrow)$ where $Args$ is a finite set of arguments, and $\rightsquigarrow \subseteq (2^{Args} \setminus \{\emptyset\}) \times Args$ is the attack relation. We'll write $S \rightsquigarrow \alpha$ to mean $(S, \alpha) \in \rightsquigarrow$.

When working with SETAFs, we define A^+ as $\{\beta \mid S \rightsquigarrow \beta \text{ for some } S \subseteq A\}$ and say A defends β iff $A^+ \cap S \neq \emptyset$ whenever $S \rightsquigarrow \beta$. The definitions of characteristic function, conflict-free, admissibility, different types of extensions, and coherence for SETAFs are the same as for AFs but using these modified notions of defeated and defended arguments.

Symmetric SETAFs were recently defined as follows:

Definition 25. (Diller et al. 2020) A SETAF $(Args, \rightsquigarrow)$ is symmetric if the following two conditions hold: (**Symm-1**) if $S \rightsquigarrow \beta$ and $\alpha \in S$, then $S' \cup \{\beta\} \rightsquigarrow \alpha$ for some S' , and (**Irr**) there is no $S \rightsquigarrow \alpha$ with $\alpha \in S$.

Unfortunately, Diller et al. show their definition does not preserve the nice properties of symmetric AFs. In particular:

Theorem 26. Some symmetric SETAFs are not coherent.

We propose an alternative, and we believe quite natural, notion of symmetric SETAF, which we term ‘strongly symmetric’ to distinguish it from the preceding notion:

Definition 27. A SETAF $(Args, \rightsquigarrow)$ is strongly symmetric if it satisfies **(Irr)** and **(Symm-2)**: for every attack $S \rightsquigarrow \beta$ and every $\alpha \in S$, we have $S \setminus \{\alpha\} \cup \{\beta\} \rightsquigarrow \alpha$.

With our definition, symmetry implies coherence:

Theorem 28. Strongly symmetric SETAFs are coherent.

5.4 Adding Preferences to SETAFs

It is natural to combine the preceding two extensions, but to the best of our knowledge, this combination has not yet been considered. We propose the following definition:

Definition 29. A preference-based set-based argumentation framework (PSETAF) is a triple $(Args, \rightsquigarrow, \succ)$, where $(Args, \rightsquigarrow)$ is a SETAF, and \succ is an acyclic binary relation over $Args$, called the preference relation.

To define the semantics of PSETAFs, we give a reduction to SETAFs, which generalizes the one for PAFs:

Definition 30. Given a PSETAF $(Args, \rightsquigarrow, \succ)$, its corresponding SETAF is $(Args, \rightsquigarrow_{\succ})$, where the relation $\rightsquigarrow_{\succ} \subseteq (2^{Args} \setminus \{\emptyset\}) \times Args$ is defined as follows: $S \rightsquigarrow_{\succ} \alpha$ iff $S \rightsquigarrow \alpha$ and $\alpha \not\succeq \beta$ for every $\beta \in S$. A subset $E \subseteq Args$ is a stable (resp. preferred, grounded, complete) extension of a PSETAF $(Args, \rightsquigarrow, \succ)$ iff it is a stable (resp. preferred, grounded, complete) extension of the corresponding SETAF.

We will focus on strongly symmetric PSETAFs:

Definition 31. A PSETAF $(Args, \rightsquigarrow, \succ)$ is strongly symmetric if the SETAF $(Args, \rightsquigarrow)$ is strongly symmetric.

The following result lifts Theorem 21 to PSETAFs. The proof is quite intricate, and we leave open whether the same result holds without the transitivity assumption:

Theorem 32. Every strongly symmetric PSETAF with a transitive preference relation is coherent.

6 Linking Prioritized KBs and PSETAFs

This section places optimal repairs into a broader context by exhibiting a tight relationship between optimal repairs and extensions of argumentation frameworks. The argumentation connection is then exploited to define a new kind of prioritized repair with favourable computational properties.

Each prioritized KB naturally gives rise to a PSETAF in which the ABox assertions play the role of arguments, and the attack relation captures the conflicts.

Definition 33. Given a prioritized KB \mathcal{K}_{\succ} with $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$, the associated PSETAF is $F_{\mathcal{K}, \succ} = (\mathcal{A}, \rightsquigarrow_{\mathcal{K}}, \succ)$, where $\rightsquigarrow_{\mathcal{K}} = \{(C \setminus \{\alpha\}, \alpha) \mid C \in \text{Conf}(\mathcal{K}), \alpha \in C\}$.

Remark 34. Recall that we assume the ABoxes do not contain self-conflicting assertions, hence $C \setminus \{\alpha\} \neq \emptyset$. If we choose not to make this assumption, we should omit such assertions when constructing the PSETAF, i.e., using $\mathcal{A} \setminus \{\alpha \mid \langle \mathcal{T}, \{\alpha\} \rangle \models \perp\}$ for the set of arguments and only considering $C \in \text{Conf}(\mathcal{K})$ with $|C| \geq 2$ to construct $\rightsquigarrow_{\mathcal{K}}$.

In this section, results are formulated for arbitrary KBs, except where otherwise noted.

6.1 Optimal Repairs vs. PSETAF Extensions

Now that we have translated prioritized KBs into argumentation frameworks, it is possible to compare preferred repairs and extensions. The following theorem shows that Pareto-optimal repairs correspond precisely to stable extensions.

Theorem 35. $\mathcal{A}' \subseteq \mathcal{A}$ is a Pareto-optimal repair of $\mathcal{K}_{\succ} = \langle \langle \mathcal{T}, \mathcal{A} \rangle, \succ \rangle$ iff \mathcal{A}' is a stable extension of $F_{\mathcal{K}, \succ}$.

Observe that our translation always produces strongly symmetric PSETAFs, which makes it possible to transfer Theorem 32 when the priority relation is transitive.

Theorem 36. If \succ is a transitive priority relation, then the PSETAF $F_{\mathcal{K}, \succ}$ is coherent.

Theorems 35 and 36 together show that for transitive \succ , Pareto-repairs coincide also with preferred extensions.

For KBs which contain only binary conflicts (e.g. core DL-Lite dialects), the resulting PSETAF is actually a PAF, so we can drop the transitivity requirement:

Theorem 37. For every prioritized KB \mathcal{K}_{\succ} such that $\mathcal{C} \in \text{Conf}(\mathcal{K})$ implies $|\mathcal{C}| = 2$, $F_{\mathcal{K}, \succ}$ is coherent. Thus, $\mathcal{A}' \in \text{PRep}(\mathcal{K}_{\succ})$ iff \mathcal{A}' is a preferred extension of $F_{\mathcal{K}, \succ}$.

Note that it follows from Theorem 35 that globally-optimal and completion-optimal repairs correspond to proper subsets of the stable extensions of $F_{\mathcal{K}, \succ}$, but they do not at present have any analog in the argumentation setting. For a (SET)AF F that corresponds to a (strongly) symmetric P(SET)AF P , it is possible to define new types of extensions that will correspond to globally- and completion-optimal repairs of a KB having P for associated P(SET)AF and to import results from the prioritized KB setting. In particular, when F is an AF that respects the conditions of Theorem 23, we can adapt the algorithms for completion-optimal repairs to tractably enumerate a non-empty subset of the stable/preferred extensions of F , while enumerating its stable/preferred extensions is not in TOTALP in general.

6.2 Grounded Semantics for Prioritized KBs

We next turn to the relationship between grounded extensions and optimal repair semantics. We start by clarifying the situation for the simplest case, in which there are no preferences (cf. (Croitoru and Vesic 2013) for a similar result in a different but related setting).

Theorem 38. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a KB, and let $F_{\mathcal{K}}$ be the SETAF corresponding to $F_{\mathcal{K}, \succ_{\emptyset}}$ (with \succ_{\emptyset} the empty relation). Then the grounded extension of $F_{\mathcal{K}}$ coincides with the intersection of the repairs of \mathcal{K} .

For prioritized KBs, we can use Theorem 35 and the fact that every stable extension is a complete extension to relate grounded extensions with Pareto-optimal repairs:

Theorem 39. If \mathcal{G} is the grounded extension of $F_{\mathcal{K}, \succ}$, then $\mathcal{G} \subseteq \mathcal{B}$ for every $\mathcal{B} \in \text{PRep}(\mathcal{K}_{\succ})$.

Thus, the grounded extension contains only assertions common to all Pareto-optimal repairs. It may not contain all such assertions, as the following example illustrates:

Example 40. Consider a prioritized KB $\mathcal{K}_>$ in which $\text{Conf}(\mathcal{K}) = \{\{\alpha, \beta\}, \{\alpha, \gamma\}, \{\beta, \gamma\}, \{\gamma, \delta\}\}$ and where $>$ contains $\alpha > \gamma$, $\beta > \gamma$, and $\gamma > \delta$. Then δ appears in the two Pareto-optimal repairs of $\mathcal{K}_>$, which are $\{\alpha, \delta\}$ and $\{\beta, \delta\}$, while the grounded extension of $F_{\mathcal{K}_>}$ is \emptyset .

We propose to use the grounded extension to define a new inconsistency-tolerant semantics for prioritized KBs:

Definition 41. A query q is entailed from $\mathcal{K}_>$ under grounded semantics, denoted $\mathcal{K}_> \models_{GR} q$, iff $\langle \mathcal{T}, \mathcal{G} \rangle \models q$, where \mathcal{G} is the grounded extension of $F_{\mathcal{K}_>}$.

As follows from Theorem 39, our new semantics provides an under-approximation of the P-IAR semantics:

Theorem 42. For every prioritized KB $\mathcal{K}_>$ and query q : $\mathcal{K}_> \models_{GR} q$ implies $\mathcal{K}_> \models_{IAR}^P q$.

Importantly, since the grounded extension of a (SET)AF can be computed in polynomial time, this semantics allows for tractable query answering in some relevant settings.

Theorem 43. Let \mathcal{L} be a DL satisfying POLYBCQ and POLYCONS. Then the problem of BCQ entailment over \mathcal{L} -KBs under grounded semantics is in P w.r.t. data complexity.

Corollary 44. For DL-Lite $_{\text{horn}}^{\mathcal{H}}$ KBs, BCQ entailment under grounded semantics is in P w.r.t. data complexity.

The grounded semantics can be compared to another recently proposed semantics for prioritized KBs, the *Elect semantics*, defined for KBs with binary conflicts by Belabbes, Benferhat, and Chomicki (2019) then generalized for non-binary conflicts (Belabbes and Benferhat 2019). Elect evaluates queries over a set of *elected assertions* and is tractable for DLs for which computing the conflicts and BCQ entailment can be done in polynomial time w.r.t. data complexity.

Definition 45. (Belabbes and Benferhat 2019) An assertion $\alpha \in \mathcal{A}$ is elected iff for every $C \in \text{Conf}(\mathcal{K})$, if $\alpha \in C$, then there exists $\beta \in C$ such that $\alpha > \beta$. The set of elected assertions is denoted by $\text{Elect}(\mathcal{K}_>)$.

The grounded semantics is more productive than Elect, i.e., is a more precise under-approximation of P-IAR.

Theorem 46. $\text{Elect}(\mathcal{K}_>) \subseteq \mathcal{G}$ where \mathcal{G} is the grounded extension of $F_{\mathcal{K}_>}$.

Example 47. Consider a prioritized KB $\mathcal{K}_>$ in which $\text{Conf}(\mathcal{K}) = \{\{\alpha, \beta\}, \{\beta, \gamma\}\}$ and where $>$ contains $\alpha > \beta$ and $\beta > \gamma$. Then $\text{Elect}(\mathcal{K}_>) = \{\alpha\}$ while $\mathcal{G} = \{\alpha, \gamma\}$. Indeed, $\alpha \in \mathcal{G}$ and γ is defended by $\{\alpha\}$ in the AF corresponding to $F_{\mathcal{K}_>}$, so $\gamma \in \mathcal{G}$.

Our next result establishes that reasoning under grounded semantics is P-complete in the settings of Theorem 43.

Theorem 48. IQ entailment under grounded semantics is P-hard w.r.t. data complexity for DL-Lite $_{\text{core}}$.

Dung (1995) observed that the grounded extension can be computed via the well-founded semantics of a simple logic program. The following lemma generalizes this result to k -SETAFs, i.e., SETAFs for which $S \rightsquigarrow \beta$ implies $|S| \leq k$.

Theorem 49. Let $F = (\text{Args}, \rightsquigarrow)$ be a k -SETAF. Then α is in the grounded extension of F iff $\text{acc}(\alpha)$ belongs to the well-founded model of the following normal logic program:

$$\begin{aligned} & \{\text{def}(x) \leftarrow \text{att}_i(y_1, \dots, y_i, x), \text{acc}(y_1), \dots, \text{acc}(y_i) \mid 1 \leq i \leq k\} \\ & \cup \{\text{acc}(x) \leftarrow \text{arg}(x), \neg \text{def}(x)\} \cup \{\text{arg}(\alpha) \mid \alpha \in \text{Args}\} \\ & \cup \{\text{att}_i(\alpha_1, \dots, \alpha_i, \beta) \leftarrow \mid \{\alpha_1, \dots, \alpha_i\} \rightsquigarrow \beta\} \end{aligned}$$

Intuitively, the preceding logic program computes the sets of defeated (*def*) and accepted (*acc*) arguments starting from the original sets of arguments (*arg*) and attacks (*att_i* encodes i -ary attacks). By adding rules to compute conflicts and populate the attack relations *att_i*, we can show that the grounded extension of $F_{\mathcal{K}_>}$ can be computed via logic programming. While stated for DL-Lite $_{\text{horn}}^{\mathcal{H}}$, the next theorem holds for any ontology (or constraint) language for which inconsistency can be characterized by a finite set of BCQs.

Theorem 50. For every DL-Lite $_{\text{horn}}^{\mathcal{H}}$ TBox \mathcal{T} , there exists a normal logic program $\Pi_{\mathcal{T}}$ such that for every ABox \mathcal{A} , priority relation $>$ for $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$, and assertion α , the following are equivalent:

- α belongs to the grounded extension of the PSETAF $F_{\mathcal{K}_>}$
- $\text{acc}(\text{id}(\alpha))$ belongs to the well-founded model of $\Pi_{\mathcal{T}} \cup \{\gamma \leftarrow \mid \gamma \in \mathcal{A}_{\text{id}}\} \cup \{\text{pref}(\text{id}(\alpha), \text{id}(\beta)) \leftarrow \mid \alpha > \beta\}$,

where \mathcal{A}_{id} is obtained from \mathcal{A} by adding an extra argument to every assertion β containing a unique *id*, denoted $\text{id}(\beta)$.

As the well-founded semantics is implemented in logic programming systems like XSB (Rao et al. 1997), the preceding theorem presents a method for implementing grounded semantics. Alternatively, the grounded extension can be naturally under-approximated by fixing $d > 0$ and considering $\Gamma_F^d(\emptyset)$ rather than the least fixpoint of Γ_F . The logic program from Theorem 50 can be modified to yield a non-recursive stratified program that computes $\Gamma_{F_{\mathcal{K}_>}}^d(\emptyset)$, which in turn can be expressed as a first-order (\sim SQL) query and evaluated using a relational database system.

7 Related Work

Our complexity results are similar to those obtained for prioritized databases with functional dependencies (Staworko, Chomicki, and Marcinkowski 2012; Kimelfeld, Livshits, and Peterfreund 2017; Livshits and Kimelfeld 2017). Compared to this work, we additionally consider IAR and brave semantics, study the impact of transitivity on all reasoning tasks, and establish connections with argumentation.

Inconsistency-tolerant semantics based on other kinds of preferred repairs have been investigated both in the database and DL or Datalog $^{\pm}$ contexts, often with a focus on repairs that have maximal cardinality or weight (Lopatenko and Bertossi 2007; Du, Qi, and Shen 2013; Baget et al. 2016; Lukaszewicz, Malizia, and Vaicenavicius 2019). However, such global optimality criteria lead to a higher computational complexity (typically $\Delta_2^P[O(\log n)]$ -hard). By contrast, Bienvenu, Bourgaux, and Goasdoué (2014) show that for preferred repairs based on priority levels and set-inclusion (corresponding to optimal repairs in the score-structured case), BCQ entailment remains in the first level of

the polynomial hierarchy for DL-Lite. Our results show that even if the priority relation over facts is not score-structured, and not even transitive, these computational properties are retained for Pareto- and completion-optimal repairs.

When \succ is score-structured, we can compare the grounded extension and the intersection of optimal repairs to the proposals of Benferhat, Bouraoui, and Tabia (2015) for selecting a single preferred \mathcal{T} -consistent subset of the ABox. In particular, the *non-defeated repair* is the union of the intersections of the repairs of $S_1, S_1 \cup S_2, \dots, S_1 \cup \dots \cup S_n$, where S_1, \dots, S_n is a partition of \mathcal{A} into priority levels. Since Elect coincides with non-defeated semantics for score-structured priorities (Belabbes and Benferhat 2019), the non-defeated repair is included in the grounded extension by Theorem 46; as the priority relation in Example 47 is score-structured, the inclusion may be strict. Another proposal is the *prioritized inclusion-based non-defeated repair*, which coincides with the intersection of the optimal repairs. The grounded extension thus lies between the non-defeated and prioritized inclusion-based non-defeated repairs. Other proposals are either included in non-defeated, or are not sound approximations of the intersection of the optimal repairs. For partially preordered ABoxes, Belabbes and Benferhat (2019) propose to go beyond Elect by intersecting the optimal repairs w.r.t. the score-structured priorities obtained by extending the original partial preorder into a total preorder. Interestingly, the obtained set of assertions corresponds exactly to the intersection of the completion-optimal repairs of the prioritized KB underpinned by the partially preordered KB (cf. appendix of (Bienvenu and Bourgaux 2020)).

Argumentation frameworks derived from inconsistent KBs have been considered before. Arguments are generally defined as pairs of a support (subset of the KB) and a conclusion (consequence of the support), with various attack relations (see e.g., (Amgoud and Cayrol 2002; Gorogiannis and Hunter 2011)). A series of papers starting with (Croitoru and Vesic 2013) links argumentation and inconsistency-tolerant querying of KBs, where the support of the argument is a subset of the ABox, the conclusion is a conjunction of facts entailed from the support and \mathcal{T} , and α attacks β iff the conclusion of α is \mathcal{T} -inconsistent with the support of β (in some papers, the argument consists of a whole derivation sequence, not just the support and conclusion). Stable and preferred extensions of such AFs correspond to repairs in the following sense: every such extension contains all those arguments whose supports are included in some particular repair. The work by Croitoru, Thomopoulos, and Vesic (2015) shows that it is possible to use a preference relation over facts to define notions of optimal extensions for AFs induced by inconsistent KBs in this fashion, and observe that such extensions correspond to optimal repairs. While touching on similar topics, an essential difference between this line of work and our own lies in the definition of the AF. Indeed, in the AFs of Croitoru et al., even if we group together arguments with the same supports, we would still have an exponential number of arguments. By contrast, our translation can be carried out in polynomial time under reasonable assumptions, enabling us to import tractability results from ar-

gumentation to OMQA. Moreover, since we use assertions as arguments, each priority relation yields a preference relation over arguments, enabling a transparent reduction to preference-based (SET)AFs. Croitoru, Thomopoulos, and Vesic do not discuss any connections to preference-based, nor set-based AFs, and our results on the equivalence between Pareto-optimal repairs and stable and preferred extensions of P(SET)AFs do not follow from their results.

8 Conclusion and Future Work

We have explored the problem of how to repair and query inconsistent KBs while taking into account a priority relation over facts. By leveraging connections to database repairs and abstract argumentation, we obtained a number of novel results for our setting, while at the same time contributing new results and research questions to these two areas.

After importing the notions of Pareto-, globally-, and completion-optimal repairs into the OMQA setting, our first contribution was a data complexity study that showed, unsurprisingly, that reasoning with optimal repairs is typically intractable and more challenging than for classical repairs. Nevertheless, there are several cases which are ‘only’ NP/coNP, which suggests that it may be interesting to devise practical SAT-based procedures, as has been successfully done for some other forms of repair (Bienvenu, Bourgaux, and Goasdoué 2014). It would also be relevant to implement and experiment our proposed grounded semantics as well as the first-order approximations we suggested. Our complexity study could be expanded to include further ontology languages as well as the combined complexity measure.

In order to provide an elegant translation of prioritized KBs to argumentation frameworks, we were naturally led to consider preference-based (SET)AFs. We believe that the new results we established in Section 5, in particular, Theorem 32, should be of interest to the argumentation community. Moreover, the connections we established between extensions of PSETAFs and repairs of prioritized KBs motivate a more detailed study of PSETAFs (a concrete open question is whether Theorem 32 holds in the absence of transitivity). Furthermore, this correspondence could be leveraged to explore new notions of extension for (strongly) symmetric P(SET)AFs inspired by completion- or globally-optimal repairs, or to design benchmarks for (P)(SET)AFs via a translation from inconsistent KBs.

An important question that has not yet been satisfactorily answered in the database literature is which of the three forms of optimal repair is most natural, independently of their computational costs. While we do not claim to provide a definitive answer to this question, we believe that our result showing that Pareto-optimal repairs coincide with stable extensions (and often also preferred extensions) speaks to the interest of adopting Pareto-optimal repairs. We should emphasize that while phrased for KBs, the connection between Pareto-optimal repairs and stable / preferred extensions holds equally well for databases with denial constraints (such as functional dependencies). Moreover, the tractable grounded semantics we proposed can be applied not just to KBs but also to prioritized databases.

Acknowledgements

This work was supported by Camille Bourgaux's CNRS PEPS grant and the ANR AI Chair INTENDED.

References

- Amgoud, L., and Cayrol, C. 2002. Inferring from inconsistency in preference-based argumentation frameworks. *J. Autom. Reasoning* 29(2):125–169.
- Amgoud, L., and Vesic, S. 2014. Rich preference-based argumentation frameworks. *Int. J. Approx. Reasoning* 55(2):585–606.
- Artale, A.; Calvanese, D.; Kontchakov, R.; and Zakharyashev, M. 2009. The DL-Lite family and relations. *J. Artif. Intell. Res.* 36:1–69.
- Baader, F.; Horrocks, I.; Lutz, C.; and Sattler, U. 2017. *An Introduction to Description Logic*. Cambridge University Press.
- Baget, J.; Benferhat, S.; Bouraoui, Z.; Croitoru, M.; Mugnier, M.; Papini, O.; Rocher, S.; and Tabia, K. 2016. Inconsistency-tolerant query answering: Rationality properties and computational complexity analysis. In *JELIA*.
- Belabbes, S., and Benferhat, S. 2019. Inconsistency handling for partially preordered ontologies: Going beyond Elect. In *KSEM*.
- Belabbes, S.; Benferhat, S.; and Chomicki, J. 2019. Elect: An inconsistency handling approach for partially preordered lightweight ontologies. In *LPNMR*.
- Benferhat, S.; Bouraoui, Z.; and Tabia, K. 2015. How to select one preferred assertional-based repair from inconsistent and prioritized DL-Lite knowledge bases? In *IJCAI*.
- Bienvenu, M., and Bourgaux, C. 2016. Inconsistency-tolerant querying of description logic knowledge bases. In *Reasoning Web*.
- Bienvenu, M., and Bourgaux, C. 2020. Querying and repairing inconsistent prioritized knowledge bases: Complexity analysis and links with abstract argumentation. arXiv:2003.05746 [cs.LO].
- Bienvenu, M., and Ortiz, M. 2015. Ontology-mediated query answering with data-tractable description logics. In *Reasoning Web*.
- Bienvenu, M., and Rosati, R. 2013. Tractable approximations of consistent query answering for robust ontology-based data access. In *IJCAI*.
- Bienvenu, M.; Bourgaux, C.; and Goasdoué, F. 2014. Querying inconsistent description logic knowledge bases under preferred repair semantics. In *AAAI*.
- Bienvenu, M.; Bourgaux, C.; and Goasdoué, F. 2019. Computing and explaining query answers over inconsistent DL-Lite knowledge bases. *J. Artif. Intell. Res.* 64:563–644.
- Bienvenu, M. 2019. Inconsistency handling in ontology-mediated query answering: A progress report. In *DL*.
- Boros, E.; Elbassioni, K. M.; Gurvich, V.; and Khachiyan, L. 2000. An efficient incremental algorithm for generating all maximal independent sets in hypergraphs of bounded dimension. *Parallel Processing Letters* 10(4):253–266.
- Bourgaux, C. 2016. *Inconsistency Handling in Ontology-Mediated Query Answering*. Ph.D. Dissertation, University of Paris-Saclay, France.
- Calvanese, D.; De Giacomo, G.; Lembo, D.; Lenzerini, M.; and Rosati, R. 2007. Tractable reasoning and efficient query answering in description logics: The DL-Lite family. *J. Autom. Reasoning* 39(3):385–429.
- Coste-Marquis, S.; Devred, C.; and Marquis, P. 2005. Symmetric argumentation frameworks. In *ECSQARU*.
- Croitoru, M., and Vesic, S. 2013. What can argumentation do for inconsistent ontology query answering? In *SUM*.
- Croitoru, M.; Thomopoulos, R.; and Vesic, S. 2015. Introducing preference-based argumentation to inconsistent ontological knowledge bases. In *PRIMA*.
- Diller, M.; Keshavarzi Zafarghandi, A.; Linsbichler, T.; and Woltran, S. 2020. Investigating subclasses of abstract dialectical frameworks. *Argument & Computation* 11(1-2):191–219.
- Du, J.; Qi, G.; and Shen, Y. 2013. Weight-based consistent query answering over inconsistent SHIQ knowledge bases. *Knowl. Inf. Syst.* 34(2):335–371.
- Dung, P. M. 1995. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artif. Intell.* 77(2):321–358.
- Eiter, T., and Gottlob, G. 1995. Identifying the minimal transversals of a hypergraph and related problems. *SIAM J. Comput.* 24(6):1278–1304.
- Fagin, R.; Kimelfeld, B.; Reiss, F.; and Vansummeren, S. 2016. Declarative cleaning of inconsistencies in information extraction. *ACM Trans. Database Syst.* 41(1):6:1–6:44.
- Fagin, R.; Kimelfeld, B.; and Kolaitis, P. G. 2015. Dichotomies in the complexity of preferred repairs. In *PODS*.
- Flouris, G., and Bikakis, A. 2019. A comprehensive study of argumentation frameworks with sets of attacking arguments. *Int. J. Approx. Reasoning* 109:55–86.
- Gorogiannis, N., and Hunter, A. 2011. Instantiating abstract argumentation with classical logic arguments: Postulates and properties. *Artif. Intell.* 175(9-10):1479–1497.
- Johnson, D. S.; Papadimitriou, C. H.; and Yannakakis, M. 1988. On generating all maximal independent sets. *Inf. Process. Lett.* 27(3):119–123.
- Kaci, S.; van der Torre, L. W. N.; and Villata, S. 2018. Preference in abstract argumentation. In *COMMA*.
- Kaci, S.; van der Torre, L. W. N.; and Weydert, E. 2006. Acyclic argumentation: Attack = conflict + preference. In *ECAI*.
- Kimelfeld, B.; Livshits, E.; and Peterfreund, L. 2017. Detecting ambiguity in prioritized database repairing. In *ICDT*.
- Lembo, D.; Lenzerini, M.; Rosati, R.; Ruzzi, M.; and Savo, D. F. 2010. Inconsistency-tolerant semantics for description logics. In *RR*.
- Lembo, D.; Lenzerini, M.; Rosati, R.; Ruzzi, M.; and Savo, D. F. 2015. Inconsistency-tolerant query answering in ontology-based data access. *Journal Web Sem.* 33:3–29.

Livshits, E., and Kimelfeld, B. 2017. Counting and enumerating (preferred) database repairs. In *PODS*.

Lopatenko, A., and Bertossi, L. E. 2007. Complexity of consistent query answering in databases under cardinality-based and incremental repair semantics. In *ICDT*.

Lukasiewicz, T.; Malizia, E.; and Vaicnavicius, A. 2019. Complexity of inconsistency-tolerant query answering in Datalog+/- under cardinality-based repairs. In *AAAI*.

Martinez, M. V.; Parisi, F.; Pugliese, A.; Simari, G. I.; and Subrahmanian, V. S. 2014. Policy-based inconsistency management in relational databases. *Int. J. Approx. Reason.* 55(2):501–528.

Motik, B.; Cuenca Grau, B.; Horrocks, I.; Wu, Z.; Fokoue, A.; and Lutz, C. 2012. OWL 2 Web Ontology Language profiles. W3C Recommendation. Available at <http://www.w3.org/TR/owl2-profiles/>.

Nielsen, S. H., and Parsons, S. 2006. A generalization of Dung’s abstract framework for argumentation: Arguing with sets of attacking arguments. In *ArgMAS*.

OWL Working Group, W. 2009. *OWL 2 Web Ontology Language: Document overview*. W3C Recommendation. Available at <https://www.w3.org/TR/owl2-overview/>, accessed on July 3rd, 2020.

Poggi, A.; Lembo, D.; Calvanese, D.; De Giacomo, G.; Lenzerini, M.; and Rosati, R. 2008. Linking data to ontologies. *Journal of Data Semantics* 10:133–173.

Rao, P.; Sagonas, K.; Swift, T.; Warren, D. S.; and Freire, J. 1997. XSB: A system for efficiently computing WFS. In *LPNMR*.

Staworko, S.; Chomicki, J.; and Marcinkowski, J. 2012. Prioritized repairing and consistent query answering in relational databases. *Ann. Math. Artif. Intell.* 64(2-3):209–246.

Tanon, T. P.; Bourgaux, C.; and Suchanek, F. M. 2019. Learning how to correct a knowledge base from the edit history. In *WWW*.

Trivela, D.; Stoilos, G.; and Vassalos, V. 2018. A framework and positive results for IAR-answering. In *AAAI*.

Tsalapati, E.; Stoilos, G.; Stamou, G. B.; and Koletsos, G. 2016. Efficient query answering over expressive inconsistent description logics. In *IJCAI*.

Xiao, G.; Calvanese, D.; Kontchakov, R.; Lembo, D.; Poggi, A.; Rosati, R.; and Zakharyashev, M. 2018. Ontology-based data access: A survey. In *IJCAI*.