Technische Universität Dresden
Institute for Theoretical Computer Science
Chair for Automata Theory

LTCS–Report

Beyond Optimal: Interactive Identification of Better-than-optimal Repairs (Extended Version)

Francesco Kriegel

LTCS-Report 24-05

This is an extended version of an article accepted at the KRR track of the 40th ACM/SIGAPP Symposium on Applied Computing (SAC 2025).

Postal Address:
Lehrstuhl für Automatentheorie
Institut für Theoretische Informatik
TU Dresden
01062 Dresden

http://lat.inf.tu-dresden.de

Visiting Address:
Nöthnitzer Str. 46
Dresden

Beyond Optimal: Interactive Identification of Better-than-optimal Repairs (Extended Version)

Francesco Kriegel
Theoretical Computer Science, Technische Universität Dresden
Dresden, Germany
Center for Scalable Data Analytics and Artificial Intelligence (ScaDS.AI)
Dresden/Leipzig, Germany
francesco.kriegel@tu-dresden.de

Abstract

We propose an interactive repair method for the description logic $\mathcal{EL}$ that is based on the optimal-repair framework. The obtained repair might not be optimal in the theoretical sense, i.e. more than a minimal amount of consequences might have been removed — but from a practical perspective it is superior to a theoretically optimal repair as the interaction strategy enables the users to identify further faulty consequences connected to the initially reported errors.

CCS Concepts

• Theory of computation → Description logics; **• Computing methodologies → Description logics**; **Ontology engineering**.

Keywords

Knowledge-base repair, Optimal repair, Disputable consequence, Interactive repair

1 Introduction

Knowledge-based systems (KBS) represent complex domains in an explicit, structured manner and comprise an inference engine that reasons about the domain to draw precise conclusions and to answer queries. Many KBS additionally include an explanation component that makes transparent the derivation of a conclusion by showing a human-readable proof to the user. A modern foundation of KBS is the family of Description Logics (DLs) [3]. Knowledge is represented as a DL knowledge base (KB) consisting of an assertion box (ABox) and a DL ontology, which is further subdivided into a terminology box (TBox) and a role box (RBox). Complete domain knowledge is not required to build a KB as DLs adopt the open-world assumption and so allow for statements of which the truth cannot be determined from the represented knowledge. Factual assertions concerning specific individuals of the domain are declared in the ABox, such as classifications of the individuals into concepts and connections between the individuals by roles. The ontology governs all individuals and defines the domain's terminology in terms of concepts and their hierarchy (in the TBox), roles and their characteristics (in the RBox), as well as further rules and constraints.

The separation of instance-level and schema-level knowledge facilitates using the same ontology across multiple ABoxes. This is valuable in applications where sensitive data is processed and privacy must be protected, e.g. in the medical domain. For example, The Systematized Nomenclature of Medicine – Clinical Terms (SNOMED CT)[L1] is a multilingual ontology that describes medical terms used in clinical documentation and reporting, and is the most comprehensive computer-processable clinical terminology in the world. SNOMED CT is used in clinical decision support systems to assist healthcare professionals in making accurate diagnoses, suggesting appropriate treatments, and predicting outcomes based on patient-specific information. Other domains in which DLs have been employed are e-commerce [16, 29], finance,[L2, L3, L4] biology,[L5] and car manufacturing and Industry 4.0 [30]. Moreover, DLs serve as the logical foundation of the Web Ontology Language (OWL) [18].[L6, L7]

Even though DL-based KBS produce logically correct and explainable inferences, faulty conclusions might be drawn if the KB itself contains errors. Of course, the KB should then be appropriately repaired. The classical method is to pinpoint the statements in the KB from which the incorrect conclusion was drawn, and then either delete a minimal number of them such that the observed error vanishes or present these statements to knowledge engineers and domain experts for rectification. However, often only parts of statements are erroneous and thus deletion of whole statements would erase too much. On the other hand, it might be difficult for the experts to correct the statements since they first need to understand how the faulty consequence is inferred from them.

The $\mathcal{EL}$ family [1, 2, 19, 25, 27, 28] stands out from the various DLs available. Since $\mathcal{EL}$ strikes a balance between expressivity and computational complexity, it offers short reasoning latency and scalability to even large ontologies, which makes it an ideal choice for many applications. For example, SNOMED CT is formulated in $\mathcal{EL}$ and OWL comprises the profile OWL 2 EL[L8] based on $\mathcal{EL}$. Currently the fastest $\mathcal{EL}$ reasoner is ELK [20],[L9] which is a highly optimized, multi-threaded implementation of the polynomial-time completion algorithm. It can classify SNOMED CT (with more than 360,000 concepts) in a few seconds on a modern laptop.

In this article, we propose an interactive repair method for $\mathcal{EL}$ that surmounts both practicability issues of the classical method. As underlying repair framework we employ the optimal repairs [6, 9], which resolve errors not by deleting a minimal number of statements but by modifying the KB such that only minimally many consequences are removed (including the observed faulty ones). Moreover, the best such repair is interactively identified in a top-down manner. Unlike the classical method, the experts do not need to consult, in a bottom-up manner, proofs of the unwanted consequence to appropriately correct the KB. Instead, they start with the reported error broken down into atomic statements and proceed towards logical causes of identified faulty statements, finally reaching statements in the KB. Through this guidance, the experts' workload is significantly lowered.

From a technical perspective, each optimal repair can be obtained from the input by saturation and then delete and copy operations. To ensure that no consequence is lost unnecessarily, the input ABox is initially saturated by adding statements implied by the TBox. While deletions are then necessary to remove the unwanted consequences, the copy mechanism ensures that not too many consequences get lost since each of the copies can be modified differently. The interaction process terminates in polynomial time and allows the experts to efficiently control these three operations and thus how the repair is constructed.

(1) *Control of the delete operation:* The causes of unwanted consequences are identified, which are afterwards used to compile a plan which statements need to be removed to obtain the repair.
(2) *Control of the copy operation:* When an object is split into copies that are modified in different ways, it is explored which copies actually exist in the underlying domain such that the others can be eliminated.
(3) *Control of the saturation:* A second phase is devoted to statements entailed by the input that are still undecided and of which all substantiations entail a rejected query. These disputable consequences have no substantiations in the repair anymore and should thus be investigated.

An implementation of the underlying repair construction as well as the smart interaction strategy is available.[L10] It comes in form of a plug-in for the KB editor Protégé.[L11] This extended version contains all technical details that could not be included in the conference article for space restrictions.

2 Preliminaries

First, we provide general definitions regarding optimal repairs. To this end, we assume an arbitrary model-based logic consisting of a set of all *statements*, a set of all *interpretations*, and a relation $\models$ between them such that $\mathcal{I} \models \alpha$ indicates that the interpretation $\mathcal{I}$ *satisfies* the statement α. A *knowledge base (KB)* $\mathcal{K}$ is a finite set of statements, and we say that $\mathcal{I}$ is a *model* of $\mathcal{K}$ and write $\mathcal{I} \models \mathcal{K}$ if $\mathcal{I}$ satisfies every statement in $\mathcal{K}$. Moreover, $\mathcal{K}$ is *consistent* if it has a model, and $\mathcal{K}$ *entails* another KB $\mathcal{K}'$, written $\mathcal{K} \models \mathcal{K}'$, if every model of $\mathcal{K}$ is also one of $\mathcal{K}'$. We further assume that the statements are subdivided into *assertions* and *ontological statements*, and so each KB is a disjoint union $\mathcal{A} \uplus \mathcal{O}$ of an *assertion box (ABox)* $\mathcal{A}$ consisting of assertions and an *ontology* $\mathcal{O}$ with ontological statements.[1]

Definition 2.1. A *repair request* is an assertion set $\mathcal{P} \coloneqq \mathcal{P}_+ \uplus \mathcal{P}_-$ partitioned into an *addition part* $\mathcal{P}_+$ and a *removal part* $\mathcal{P}_-$. Of a consistent KB $\mathcal{K} \coloneqq \mathcal{A} \uplus \mathcal{O}$, an *ABox repair* for $\mathcal{P}$ is an ABox $\mathcal{B}$ s.t.

- $\mathcal{B} \cup \mathcal{O}$ is consistent,
- $\mathcal{B} \cup \mathcal{O} \models \alpha$ for each $\alpha \in \mathcal{P}_+$, and
- $\mathcal{B} \cup \mathcal{O} \not\models \beta$ for each $\beta \in \mathcal{P}_-$.

If there is an ABox repair, then $\mathcal{P}$ is *feasible* w.r.t. $\mathcal{K}$.

In the above definition, we treat the ABox as refutable and the ontology as static. We may just use the denotation "repair" when no confusion can arise. Instead of $\alpha \in \mathcal{P}_+$ we may also write $+\alpha \in \mathcal{P}$, and likewise $-\beta \in \mathcal{P}$ for $\beta \in \mathcal{P}_-$. We observe that a repair request $\mathcal{P}$ is feasible iff. $\mathcal{P}_+$ is a repair iff. $\mathcal{P}_+ \cup \mathcal{O}$ is consistent and entails no statement in $\mathcal{P}_-$. Moreover, repairs as defined above have no connection to the input ABox, but the following order relation between repairs takes it into account.

Definition 2.2. Consider two repairs $\mathcal{B} \coloneqq \mathcal{B}_+ \uplus \mathcal{B}_-$ and $\mathcal{C} \coloneqq \mathcal{C}_+ \uplus \mathcal{C}_-$ of $\mathcal{K}$ for $\mathcal{P}$, where $\mathcal{B}_- \coloneqq \{ \gamma \mid \gamma \in \mathcal{B} \text{ and } \mathcal{K} \models \gamma \}$ and $\mathcal{C}_-$ is defined likewise. We write $\mathcal{B} \leq \mathcal{C}$ and say that $\mathcal{C}$ is *at least as good as* $\mathcal{B}$ if $\mathcal{B} \cup \mathcal{O} \models \mathcal{C}_+$ and $\mathcal{C} \cup \mathcal{O} \models \mathcal{B}_-$. Moreover, we write $\mathcal{B} < \mathcal{C}$ and say that $\mathcal{C}$ is *better than* $\mathcal{B}$ if $\mathcal{B} \leq \mathcal{C}$ but $\mathcal{C} \nleq \mathcal{B}$, i.e. either less knowledge is added, $\mathcal{C} \cup \mathcal{O} \not\models \mathcal{B}_+$, or less knowledge is removed, $\mathcal{B} \cup \mathcal{O} \not\models \mathcal{C}_-$. We call $\mathcal{B}$ *optimal* if there is no repair better than $\mathcal{B}$, and $\mathcal{P}$ is *optimally coverable* w.r.t. $\mathcal{K}$ if every repair of $\mathcal{K}$ for $\mathcal{P}$ is at most as good as some optimal one. If $\mathcal{P}$ is optimally coverable w.r.t. $\mathcal{K}$ and there is exactly one optimal repair of $\mathcal{K}$ w.r.t. $\mathcal{P}$ up to equivalence w.r.t. $\mathcal{O}$, then $\mathcal{P}$ is *deterministic* w.r.t. $\mathcal{K}$.

In order to comply with the repair request, optimal repairs preserve as much as possible knowledge entailed by the input KB while containing as little as possible new knowledge — yet not every assertion entailed by $\mathcal{K}$ should be preserved in the concrete application. The reason is that such an entailed assertion might only make sense in the application domain as long as it is substantiated. For instance, if we repair for an assertion stating that Bob has a particular disease, then we would not want to keep the consequence that Bob is ill, unless there is knowledge that he has another disease. In contrast, if it should be repaired that Alice is a celebrity, then we would still want to retain the consequence that Alice is a human. In order to formulate this precisely, we use substantiations. In the literature, justifications of a statement γ have been defined as *subsets* of the refutable part that together with the static part entail γ. In order to eliminate dependence on the syntax, our following definition instead defines substantiations as KBs *entailed* by the refutable part.[2] We further take the provided information in the repair request into account by treating its addition part like the ABox of the input KB, since both are the "positive knowledge" before the repair process.

Definition 2.3. W.r.t. a KB $\mathcal{K}$ and a repair request $\mathcal{P}$, a *substantiation* of an assertion γ is an ABox $\mathcal{J}$ s.t. $\mathcal{A} \cup \mathcal{P}_+ \models \mathcal{J}$ and $\mathcal{J} \cup \mathcal{O} \models \gamma$.

With that, we call a consequence of $\mathcal{K}$ disputable if a repair entails it while another not (i.e. it could be included in a repair or not), but none of its substantiations can be preserved in any repair (i.e. it is not justified anymore).

Definition 2.4. Given a consistent KB $\mathcal{K}$ and a feasible repair request $\mathcal{P}$, a *disputable consequence* of $\mathcal{K}$ w.r.t. $\mathcal{P}$ is an assertion γ s.t.

- $\mathcal{K} \cup \mathcal{P}_+ \models \gamma$,
- there is a repair $\mathcal{B}$ of $\mathcal{K}$ for $\mathcal{P}$ with $\mathcal{B} \cup \mathcal{O} \models \gamma$,
- there is a repair $\mathcal{B}$ of $\mathcal{K}$ for $\mathcal{P}$ with $\mathcal{B} \cup \mathcal{O} \not\models \gamma$, and
- for each repair $\mathcal{B}$ of $\mathcal{K}$ for $\mathcal{P}$, the KB $\mathcal{B} \cup \mathcal{O}$ does not entail any substantiation of γ w.r.t. $\mathcal{K}$ and $\mathcal{P}$.

In order to obtain an optimal repair that makes sense in the application domain, we recommend to decide each disputable consequence by hand and accordingly refine the repair request, i.e. add all accepted ones to $\mathcal{P}_+$ and all rejected ones to $\mathcal{P}_-$.

[1] We write $A = B \uplus C$ iff. $A = B \cup C$ and $B \cap C = \emptyset$.

[2] We chose another denotation to avoid confusion. Justifications are defined in the same way except that $\mathcal{A} \models \mathcal{J}$ is replaced by $\mathcal{J} \subseteq \mathcal{A}$.

The following example illustrates why the addition part $\mathcal{P}_+$ is taken into account in the above definitions.

Example 2.A. The KB $\mathcal{K}$ is formulated in the DL $\mathcal{EL}$ and consists of the ABox $\mathcal{A} \coloneqq \{\mathsf{bob} : \mathsf{HasCold},\ \mathsf{bob} : \mathsf{HasDiagnose1}\}$ and the ontology $\mathcal{O} \coloneqq \{\mathsf{HasCold} \sqsubseteq \mathsf{IsIll},\ \mathsf{HasDiagnose1} \sqcap \mathsf{HasDiagnose2} \sqsubseteq \mathsf{HasFlu},\ \mathsf{HasFlu} \sqsubseteq \mathsf{IsIll}\}$. The repair request $\mathcal{P}$ has addition part $\mathcal{P}_+ \coloneqq \{\mathsf{bob} : \mathsf{HasDiagnose2}\}$ and removal part $\mathcal{P}_- \coloneqq \{\mathsf{bob} : \mathsf{HasCold}\}$. Moreover, we consider the assertion $\gamma \coloneqq \mathsf{bob} : \mathsf{IsIll}$.

For this input, the unique optimal ABox repair is $\mathcal{B} \coloneqq \{\mathsf{bob} : \mathsf{HasDiagnose1},\ \mathsf{bob} : \mathsf{HasDiagnose2}\}$. This is because the first assertion $\mathsf{bob} : \mathsf{HasCold}$ must be deleted for $\mathcal{P}_-$. Furthermore, $\mathsf{bob} : \mathsf{HasDiagnose2}$ must be entailed since it is in $\mathcal{P}_+$ but it is not implied by other statements w.r.t. $\mathcal{O}$, and thus it can only be added to the repair as is. Then the assertions $\mathsf{bob} : \mathsf{HasFlu}$ and $\mathsf{bob} : \mathsf{IsIll}$ are entailed by this repair when taking $\mathcal{O}$ into account, i.e. adding them would only yield an equivalent repair. In the end, we cannot remove less knowledge or add less knowledge and so obtain a better repair, i.e. $\mathcal{B}$ is optimal.

Now, let's check if γ is disputable. We have $\mathcal{K} \cup \mathcal{P}_+ \models \gamma$. The above repair $\mathcal{B}$ satisfies that $\mathcal{B} \cup \mathcal{O} \models \gamma$. On the other hand, also $\{\mathsf{bob}{:}\mathsf{HasDiagnose2}\}$ is a repair, and $\{\mathsf{bob}{:}\mathsf{HasDiagnose2}\} \cup \mathcal{O} \not\models \gamma$. The substantiations of γ are $\mathcal{J}_1 \coloneqq \{\mathsf{bob} : \mathsf{HasCold}\}$ and $\mathcal{J}_2 \coloneqq \{\mathsf{bob} : \mathsf{HasDiagnose1},\ \mathsf{bob} : \mathsf{HasDiagnose2}\}$. Since for the above repair $\mathcal{B}$, the union $\mathcal{B} \cup \mathcal{O}$ entails $\mathcal{J}_2$, the consequence γ is not disputable.

Last, we check if γ is disputable when the addition part $\mathcal{P}_+$ is not taken into account, i.e. we replace $\mathcal{A} \cup \mathcal{P}_+ \models \mathcal{J}$ with $\mathcal{A} \models \mathcal{J}$ in Definition 2.3 and likewise $\mathcal{K} \cup \mathcal{P}_+ \models \gamma$ with $\mathcal{K} \models \gamma$ in Definition 2.4. We also have $\mathcal{K} \models \gamma$ and, as above, γ is entailed by a repair and not entailed by another. However, $\mathcal{J}_1$ is the only substantiation of γ and no repair entails it, i.e. γ would be disputable now. This is unexpected since the input gave no reason to doubt validity of the second assertion $\mathsf{bob} : \mathsf{HasDiagnose1}$ in $\mathcal{A}$.

Next, we formulate an equivalent characterization of disputable consequences.

Lemma 2.B. *γ is a disputable consequence of $\mathcal{K}$ w.r.t. $\mathcal{P}$ iff.*

- *$\mathcal{K} \cup \mathcal{P}_+ \models \gamma$,*
- *the ABox $\mathcal{P}_+ \cup \{\gamma\}$ is a repair of $\mathcal{K}$ for $\mathcal{P}$, i.e. $\mathcal{P}_+ \cup \{\gamma\} \cup \mathcal{O}$ is consistent and entails no statement in $\mathcal{P}_-$,*
- *$\mathcal{P}_+$ is no substantiation of γ w.r.t. $\mathcal{K}$ and $\mathcal{P}$, i.e. $\mathcal{P}_+ \cup \mathcal{O} \not\models \gamma$, and*
- *no substantiation $\mathcal{J}$ of γ w.r.t. $\mathcal{K}$ and $\mathcal{P}$ is a repair of $\mathcal{K}$ for $\mathcal{P}$, i.e. $\mathcal{J} \cup \mathcal{O}$ is inconsistent, does not entail all statements in $\mathcal{P}_+$, or entails some statement in $\mathcal{P}_-$.*

Proof.
- The first condition does not differ and so needs no treatment.
- Let $\mathcal{B}$ be a repair of $\mathcal{K}$ for $\mathcal{P}$ such that $\mathcal{B} \cup \mathcal{O}$ entails γ. Then $\mathcal{B} \cup \mathcal{O} \models \mathcal{P}_+ \cup \{\gamma\} \cup \mathcal{O}$. Since $\mathcal{B} \cup \mathcal{O}$ is consistent, also $\mathcal{P}_+ \cup \{\gamma\} \cup \mathcal{O}$ is consistent. Moreover, since $\mathcal{B} \cup \mathcal{O}$ does not entail any statement in $\mathcal{P}_-$, neither does $\mathcal{P}_+ \cup \{\gamma\} \cup \mathcal{O}$. It follows that $\mathcal{P}_+ \cup \{\gamma\}$ is a repair. The converse direction is obvious.
- Let $\mathcal{B}$ be a repair of $\mathcal{K}$ for $\mathcal{P}$ such that $\mathcal{B} \cup \mathcal{O}$ does not entail γ. Since $\mathcal{B} \cup \mathcal{O} \models \mathcal{P}_+ \cup \mathcal{O}$, it follows that $\mathcal{P}_+ \cup \mathcal{O}$ neither entails γ. The converse direction follows easily by feasibility of $\mathcal{P}$.
- Assume that, for each repair $\mathcal{B}$ of $\mathcal{K}$ for $\mathcal{P}$ and for each substantiation $\mathcal{J}$ of γ w.r.t. $\mathcal{K}$ and $\mathcal{P}$, we have $\mathcal{B} \cup \mathcal{O} \not\models \mathcal{J}$. Further let $\mathcal{J}$ be a substantiation of γ w.r.t. $\mathcal{K}$ and $\mathcal{P}$. Since $\mathcal{J} \cup \mathcal{O} \models \mathcal{J}$, it follows that $\mathcal{J}$ is no repair of $\mathcal{K}$ for $\mathcal{P}$.
  Regarding the converse direction, let $\mathcal{B}$ be a repair of $\mathcal{K}$ for $\mathcal{P}$ and let $\mathcal{J}$ be a substantiation of γ w.r.t. $\mathcal{K}$ and $\mathcal{P}$. Then $\mathcal{J} \cup \mathcal{P}_+$ is also a substantiation of γ, but by assumption no repair. If $\mathcal{J} \cup \mathcal{P}_+ \cup \mathcal{O}$ would be inconsistent, then consistency of $\mathcal{B} \cup \mathcal{O}$ implies that $\mathcal{B} \cup \mathcal{O} \not\models \mathcal{J} \cup \mathcal{P}_+ \cup \mathcal{O}$. Otherwise, $\mathcal{J} \cup \mathcal{P}_+ \cup \mathcal{O}$ would entail some statement in $\mathcal{P}_-$, but since $\mathcal{B} \cup \mathcal{O}$ does not it follows that $\mathcal{B} \cup \mathcal{O} \not\models \mathcal{J} \cup \mathcal{P}_+ \cup \mathcal{O}$ as well. In either case, since $\mathcal{B} \cup \mathcal{O} \models \mathcal{P}_+ \cup \mathcal{O}$, it follows that $\mathcal{B} \cup \mathcal{O} \not\models \mathcal{J}$. □

Furthermore, if the repair request is non-deterministic, then it should be further refined to eventually identify an optimal repair appropriate for the application. Formally, we say that a repair request $\mathcal{P}'$ is a *refinement* of $\mathcal{P}$ if $\mathcal{P}_+ \subseteq \mathcal{P}'_+$ and $\mathcal{P}_- \subseteq \mathcal{P}'_-$, and at least one of these inclusions is strict (i.e. does not hold in the converse direction). Then every repair of $\mathcal{K}$ for $\mathcal{P}'$ is also one for $\mathcal{P}$, but not vice versa. In Section 3 we will develop an interactive method for refining a given repair request to a deterministic one in the DL $\mathcal{EL}$.

The Description Logic $\mathcal{EL}$. We recall the DL $\mathcal{EL}$, on which all other DLs in the $\mathcal{EL}$ family are based. In order to structurally describe the domain of interest, we fix *individual names (INs)*, *concept names (CNs)*, and *role names (RNs)*. *Concept descriptions (CDs)* are built by $C ::= \top \mid A \mid C \sqcap C \mid \exists r.C$ where A ranges over all CNs and r over all RNs. We call $\top$ the *top CD*, $C \sqcap D$ the *conjunction* of C and D, and $\exists r.C$ the *existential requirement* on r with *intent* C. Since nestings of and order in conjunctions are irrelevant, we also use conjunctions $\bigsqcap \Phi$ of finite sets Φ of CDs. CNs and existential requirements are *atoms* and each CD C is a conjunction of atoms, called the *top-level conjuncts* of C and gathered in the set $\mathsf{Conj}(C)$. A *terminological box (TBox)* $\mathcal{T}$ is a finite set of *concept inclusions (CIs)* $C \sqsubseteq D$ involving CDs C, D, and an $\mathcal{EL}$ *ontology* is such a TBox without anything else. An *assertion box (ABox)* $\mathcal{A}$ is a finite set of *concept assertions (CAs)* $a : C$ and *role assertions (RAs)* $(a, b) : r$ involving INs a, b, CDs C, and RNs r. A KB consists of an ABox and a TBox.

The semantics of $\mathcal{EL}$ can be defined by translation into the two-variable fragment of first-order logic (FOL): $\tau_x(\top)$ is any tautology with one free variable x, $\tau_x(A) \coloneqq A(x)$, $\tau_x(C \sqcap D) \coloneqq \tau_x(C) \wedge \tau_x(D)$, $\tau_x(\exists r.C) \coloneqq \exists y.\,(r(x,y) \wedge \tau_y(C))$ where τ_y is obtained from τ_x by swapping x and y, $\tau(C \sqsubseteq D) \coloneqq \forall x.\,(\tau_x(C) \to \tau_x(D))$, $\tau(a : C) \coloneqq \tau_x(C)[x/a]$, $\tau((a,b) : r) \coloneqq r(x,y)[x/a, y/b]$, $\tau(\mathcal{K}) \coloneqq \bigwedge\{\, \tau(\alpha) \mid \alpha \in \mathcal{K} \,\}$. Thus, $\mathcal{EL}$ inherits the model-theoretic semantics of FOL, defined by means of interpretations $\mathcal{I}$ consisting of a non-empty set $\mathsf{Dom}(\mathcal{I})$, called the *domain*, and an interpretation function $\cdot^{\mathcal{I}}$ that gives meaning to the INs, CNs, and RNs by assigning them to elements, subsets, and binary relations of $\mathsf{Dom}(\mathcal{I})$, respectively. We say that α *entails* β and write $\alpha \models \beta$ if $\tau(\alpha) \models \tau(\beta)$, i.e. if every model of $\tau(\alpha)$ is a model of $\tau(\beta)$. Unlike FOL, entailment in $\mathcal{EL}$ is decidable, viz. in polynomial time. We say that a CD C is *subsumed by* a CD D w.r.t. a TBox $\mathcal{T}$, written $C \sqsubseteq^{\mathcal{T}} D$, if $\mathcal{T}$ entails $C \sqsubseteq D$.

Quantified ABoxes. A *quantified ABox (qABox)* $\exists X. \mathcal{A}$ consists of a finite set X of *variables* and a finite set $\mathcal{A}$, called *matrix*, that contains assertions $u : A$ and $(u, v) : r$ involving INs or variables u, v, CNs A, and RNs r. Each variable in X and each IN is an *object* of $\exists X. \mathcal{A}$. A KB can now also consist of a qABox and a TBox. QABoxes are ABoxes in which variables can be additionally used in place of INs. Since these variables are existentially quantified, they are "anonymous individuals" whose names are not exposed. Moreover, only CNs are allowed within qABoxes, but complex CDs can be represented by the use of variables — e.g. the ABox $\{a : (A \sqcap \exists r. B)\}$ is equivalent to the qABox $\exists \{x\}. \{a : A,\ (a, x) : r,\ x : B\}$. Conjunctive queries (CQs), primitive positive formulas (pp-formulas) in FOL, and qABoxes are syntactic variants of each other. The *union* of two qABoxes is $\exists X. \mathcal{A} \cup \exists Y. \mathcal{B} \coloneqq \exists (X \cup Y). (\mathcal{A} \cup \mathcal{B})$ where w.l.o.g. $X \cap Y = \emptyset$ (otherwise variables need to be renamed).

A qABox $\exists X. \mathcal{A}$ can be translated into FOL by $\tau(\exists X. \mathcal{A}) \coloneqq \exists x_1. \cdots \exists x_n. \bigwedge \{ \tau(\alpha) \mid \alpha \in \mathcal{A} \}$ where $X = \{x_1, \ldots, x_n\}$ is an arbitrary enumeration, and a KB $\mathcal{K}$ consisting of $\exists X. \mathcal{A}$ and a TBox $\mathcal{T}$ is translated to $\tau(\mathcal{K}) \coloneqq \tau(\exists X. \mathcal{A}) \wedge \tau(\mathcal{T})$. If $\mathcal{K} \models \beta$, then we also say that $\exists X. \mathcal{A}$ entails β w.r.t. $\mathcal{T}$ and write $\exists X. \mathcal{A} \models^{\mathcal{T}} \beta$.

Entailment between two qABoxes is an NP-complete problem, but whether a qABox entails an ABox can be decided in polynomial time. Without a TBox, $\exists X. \mathcal{A} \models \exists Y. \mathcal{B}$ iff. there is a *homomorphism* from $\exists Y. \mathcal{B}$ to $\exists X. \mathcal{A}$, which is a function h that sends each IN a to itself and each variable in Y to an object of $\exists X. \mathcal{A}$ such that applying h within any assertion in $\mathcal{B}$ yields an assertion in $\mathcal{A}$. With a TBox $\mathcal{T}$, entailment can be decided by first *saturating* $\exists X. \mathcal{A}$ by means of $\mathcal{T}$ (i.e. compute the chase or the universal model) and then checking for a homomorphism from $\exists Y. \mathcal{B}$ to the saturation.

For some applications model-based entailment is too strong and it suffices to compare qABoxes based on their consequences from a query language. In DL, important query languages are IQ and IRQ. The former consists of all CAs (sometimes also called *instance queries*, IQs), and the latter of all CAs and RAs. As further query languages, CQ consists of all qABoxes, and gloIRQ extends IRQ by all global IQs $\exists \{x\}. \{C(x)\}$ where C is a CD. Given a query language QL, we say that $\exists X. \mathcal{A}$ QL-*entails* $\exists Y. \mathcal{B}$ w.r.t. $\mathcal{T}$ and write $\exists X. \mathcal{A} \models^{\mathcal{T}}_{\mathsf{QL}} \exists Y. \mathcal{B}$ if $\exists Y. \mathcal{B} \models^{\mathcal{T}} \gamma$ only if $\exists X. \mathcal{A} \models^{\mathcal{T}} \gamma$ for each query $\gamma \in \mathsf{QL}$. Since the TBox is fixed, CQ-entailment coincides with model-based entailment. In contrast, IQ- and IRQ-entailment are decidable in polynomial time. Without a TBox, $\exists X. \mathcal{A} \models_{\mathsf{IQ}} \exists Y. \mathcal{B}$ iff. there is a *simulation* from $\exists Y. \mathcal{B}$ to $\exists X. \mathcal{A}$, which is a "non-functional homomorphism" that can relate each object of $\exists Y. \mathcal{B}$ to multiple objects of $\exists X. \mathcal{A}$. With a TBox $\mathcal{T}$, we check for a simulation from $\exists Y. \mathcal{B}$ to the IQ-saturation of $\exists X. \mathcal{A}$ w.r.t. $\mathcal{T}$, which is obtained by materializing all CAs implied by the TBox: while there is an object u and a CI $C \sqsubseteq D$ such that the matrix of the current qABox entails $u : C$ but not $u : D$, we extend the current qABox with $u : D$ but represented by the use of variables if D is complex. Unlike the above saturations, we can here reuse variables that have already been introduced for the same CD (w.l.o.g. denoted as x_E where E is the subconcept to be represented) and thus the IQ-saturation can be computed in polynomial time. Alternatively, the canonical model of the input KB computed by an IQ-complete calculus (such as the completion procedure [1, 20]) can be treated as a qABox to obtain the IQ-saturation, though might be larger than necessary. Furthermore, $\exists X. \mathcal{A} \models^{\mathcal{T}}_{\mathsf{IRQ}} \exists Y. \mathcal{B}$ iff. $\exists X. \mathcal{A} \models^{\mathcal{T}}_{\mathsf{IQ}} \exists Y. \mathcal{B}$ and each RA in $\mathcal{B}$ involving only INs is also contained in $\mathcal{A}$. Technical details on these model-based and consequence-based entailment relations can be found in [4, 6, 8, 9, 11, 13] and the accompanying extended versions.

The following lemma shows that IRQ-entailment is invariant under addition of ontological knowledge.

Lemma 2.C. *If* $\exists Y. \mathcal{B} \models_{\mathsf{IRQ}} \exists Z. C$, *then* $\exists Y. \mathcal{B} \models^{\mathcal{T}}_{\mathsf{IRQ}} \exists Z. C$.

Proof. Consider a query $\gamma \in \mathsf{IRQ}$ with $\exists Z. C \models^{\mathcal{T}} \gamma$. If $\gamma = (a, b) : r$ is a RA, then it is contained in C. The assumption yields $\gamma \in \mathcal{B}$, and so we have $\exists Y. \mathcal{B} \models^{\mathcal{T}} \gamma$.

Now let $\gamma = a : C$ be CA. According to Lemma 22 in [26] there is a CD D such that $\exists Z. C \models a : D$ and $D \sqsubseteq^{\mathcal{T}} C$. Then $\exists Y. \mathcal{B} \models a : D$ by assumption, and thus $\exists Y. \mathcal{B} \models^{\mathcal{T}} a : C$. □

Note, however, that IRQ-entailment is not invariant under addition of any knowledge, i.e. $\exists Y. \mathcal{B} \models_{\mathsf{IRQ}} \exists Z. C$ does not imply in general that $\exists X. \mathcal{A} \cup \exists Y. \mathcal{B} \models^{\mathcal{T}}_{\mathsf{IRQ}} \exists X. \mathcal{A} \cup \exists Z. C$. One might be tempted to use similar arguments as in the above proof, and additionally try to show that there is a simulation from $\exists X. \mathcal{A} \cup \exists Z. C$ to $\exists X. \mathcal{A} \cup \exists Y. \mathcal{B}$. The precondition $\exists Y. \mathcal{B} \models_{\mathsf{IRQ}} \exists Z. C$ would already yield a simulation $\mathfrak{S}$ from $\exists Z. C$ to $\exists Y. \mathcal{B}$, and at first one might believe that one only needs to extend $\mathfrak{S}$ by all reflexive pairs with objects of $\exists X. \mathcal{A}$, i.e. show that $\mathfrak{S} \cup \{ (x, x) \mid x \in X \}$ is the required simulation. But this does not work. A counterexample is as follows.

- $\exists Y. \mathcal{B} \coloneqq \exists \{y_1, y_2, y_3\}. \{(a, y_1) : r,\ (a, y_2) : r,\ y_2 : C,\ (y_1, b) : s,\ b : D,\ (y_2, y_3) : s\}$
- $\exists Z. C \coloneqq \exists \{z_1, z_2, z_3\}. \{(a, z_1) : r,\ (a, z_2) : r,\ z_2 : C,\ (z_1, z_3) : s,\ z_3 : D,\ (z_2, b) : s\}$

It is easy to verify that $\exists Y. \mathcal{B} \models_{\mathsf{IRQ}} \exists Z. C$. Now with the qABox $\exists X. \mathcal{A} \coloneqq \exists \emptyset. \{b : E\}$, we have that $\exists X. \mathcal{A} \cup \exists Y. \mathcal{B} \not\models_{\mathsf{IRQ}} \exists X. \mathcal{A} \cup \exists Z. C$ since the latter entails $a : \exists r. (C \sqcap \exists s. E)$ but the former not.

Optimal Repairs in $\mathcal{EL}$. There is no general approach to computing optimal repairs, and we will now recall results obtained so far. Previous research focused on repair requests without an addition part. It seems that abduction methods [17, 21, 22] could be used to treat the addition part but it is still unclear how optimality could be achieved, and thus we leave this for future research. Instead, we just assume that the addition part $\mathcal{P}_+$ is already entailed by the input KB (which could be achieved by simply adding all statements in $\mathcal{P}_+$ to the KB), and so $\mathcal{P}_+$ is only to be preserved by every repair.

Optimal repairs need not exist in every setting [15]. $\mathcal{EL}$ TBoxes can be optimally repaired when the left-hand sides of CIs are fixed [23, 24]. Moreover, we can compute optimal repairs of KBs consisting of a refutable qABox and a static TBox [4, 6, 8, 9, 11, 12, 13]. In the following, we will reformulate definitions regarding optimal repairs for some of these settings and recall main results.

Assume that the input KB $\mathcal{K}$ consists of a refutable qABox $\exists X. \mathcal{A}$ and a static $\mathcal{EL}$ TBox $\mathcal{T}$, and further let QL be the query language used to access this KB. Since such KBs are always consistent, we do not need to explicitly require consistency. Now a *repair request* is a finite subset $\mathcal{P}$ of QL and, as explained above, such that the given KB $\mathcal{K}$ entails its addition part $\mathcal{P}_+$. Thus the optimal repairs are qABoxes entailed by $\mathcal{K}$ since no new knowledge must be added to make $\mathcal{P}_+$ entailed. In this sense, a QL-*repair* of $\mathcal{K}$ for $\mathcal{P}$ (also called a QL-*repair* of $\exists X.\mathcal{A}$ for $\mathcal{P}$ w.r.t. $\mathcal{T}$) is a qABox $\exists Y.\mathcal{B}$ such that

(R1) $\exists X.\mathcal{A} \models^{\mathcal{T}}_{\mathsf{QL}} \exists Y.\mathcal{B}$,
(R2) $\exists Y.\mathcal{B} \models^{\mathcal{T}} \alpha$ for each $+\alpha \in \mathcal{P}$, and
(R3) $\exists Y.\mathcal{B} \not\models^{\mathcal{T}} \beta$ for each $-\beta \in \mathcal{P}$.

$\exists Y.\mathcal{B}$ is *optimal* if there is no QL-repair $\exists Z.\mathcal{C}$ that strictly QL-entails $\exists Y.\mathcal{B}$ (i.e. $\exists Z.\mathcal{C} \models^{\mathcal{T}}_{\mathsf{QL}} \exists Y.\mathcal{B}$ but $\exists Y.\mathcal{B} \not\models^{\mathcal{T}}_{\mathsf{QL}} \exists Z.\mathcal{C}$).

The set of optimal QL-repairs can effectively be computed and every QL-repair is QL-entailed by an optimal one (i.e. each repair request is optimally coverable) in the following situations: in $\mathcal{EL}$ with QL = IQ and arbitrary TBoxes and with QL = CQ but $\mathcal{P} \subseteq$ IQ and cycle-restricted TBoxes [6, 13], in $\mathcal{EL}$ with QL = CQ but $\mathcal{P} \subseteq$ IQ and arbitrary TBoxes but infinite repairs [4], in $\mathcal{ELROI}(\bot)$[3] with QL = CQ but $\mathcal{P} \subseteq$ gloIRQ and terminating TBoxes [9], in $\mathcal{EL}$ with QL = IRQ and arbitrary TBoxes [8, 12], and in $\mathcal{EL}^\bot$ with QL = gloIRQ and arbitrary TBoxes [11]. In all aforementioned cases, a construction of canonical repairs is provided such that every optimal repair is entailed by a canonical one. We will recall the construction for $\mathcal{EL}$ and QL = IRQ, which is especially interesting since between $\mathcal{EL}$ ABoxes model-based entailment coincides with IRQ-entailment [8] and qABoxes can represent $\mathcal{EL}$ ABoxes.

Let $\mathcal{P} \subseteq$ IRQ be a repair request. We use nominals to express RAs $(a,b):r$ and replace them with CAs $a:\exists r.\{b\}$ [9]. Keep in mind that each of these CAs contributes the atoms $\exists r.\{b\}$ and $\{b\}$. Since entailment between qABoxes is characterized by a rewrite system that can copy objects into fresh variables and delete assertions [4], we can construct repairs by copying and deleting too. Moreover, since optimal repairs retain as many consequences as possible, we construct them from saturations. The canonical repairs have a closed-form representation involving copies of the form $\langle\!\langle u, \Phi \rangle\!\rangle$ where u is an object in the saturation and Φ is a repair type, which specifies what must be deleted for this copy.

More specifically, let $\exists X^{\mathcal{T}}.\mathcal{A}^{\mathcal{T}}$ be the IQ-saturation of the input KB. Each *repair type* Φ for u consists of atoms occurring in $\mathcal{P}$ or $\mathcal{T}$[4] and must satisfy the following three conditions:

(RT1) $\mathcal{A}^{\mathcal{T}} \models u : C$ for each atom C in Φ.
(RT2) $C \not\sqsubseteq^{\emptyset} D$ for each two atoms C, D in Φ.
(RT3) For each atom C in Φ and each CI $E \sqsubseteq F$ in $\mathcal{T}$ with $\mathcal{A}^{\mathcal{T}} \models u{:}E$[5] and $F \sqsubseteq^{\mathcal{T}} C$, there is an atom D in Φ with $E \sqsubseteq^{\emptyset} D$.[6]

In order to ensure that each copy $\langle\!\langle u, \Phi \rangle\!\rangle$ is no instance of any atom in Φ, the matrix of each canonical IRQ-repair consists of the following assertions:

(CR1) $\langle\!\langle u, \Phi \rangle\!\rangle : A$ if $u : A \in \mathcal{A}^{\mathcal{T}}$ and $A \notin \Phi$, and
(CR2) $(\langle\!\langle u, \Phi \rangle\!\rangle, \langle\!\langle v, \Psi \rangle\!\rangle) : r$ if $(u,v) : r \in \mathcal{A}^{\mathcal{T}}$ and, for each $\exists r.C \in \Phi$ with $\mathcal{A}^{\mathcal{T}} \models v : C$, there is an atom $D \in \Psi$ with $C \sqsubseteq^{\emptyset} D$.

We finally need to select which of the copies $\langle\!\langle a, \Phi \rangle\!\rangle$ is used as the IN a. For this purpose, a *repair seed* $\mathcal{S}$ maps each IN a to a repair type $\mathcal{S}_a$ for a such that:

(RS1) For each $+\,a : C$ in $\mathcal{P}$, there is no atom $D \in \mathcal{S}_a$ with $C \sqsubseteq^{\mathcal{T}} D$.
(RS2) For each $+\,(a,b) : r$ in $\mathcal{P}$ and for each $\exists r.C \in \mathcal{S}_a$ with $\exists X.\mathcal{A} \models^{\mathcal{T}} b : C$, there is an atom $D \in \mathcal{S}_b$ with $C \sqsubseteq^{\emptyset} D$.
(RS3) For each $-\,a : C$ in $\mathcal{P}$ with $\exists X.\mathcal{A} \models^{\mathcal{T}} a : C$,[7] there is an atom $D \in \mathcal{S}_a$ with $C \sqsubseteq^{\emptyset} D$.
(RS4) For each $-\,(a,b) : r$ in $\mathcal{P}$ with $\exists X.\mathcal{A} \models^{\mathcal{T}} (a,b) : r$, we have $\exists r.\{b\} \in \mathcal{S}_a$.
(RS5) $\{a\} \notin \mathcal{S}_a$

(RS1) is new since previous work on optimal repairs in $\mathcal{EL}$ has assumed $\mathcal{P}_+ = \emptyset$. Support can be added either through the static part of the input KB [9] or by means of Lemma 3 in [10], from which (RS1) is derived. Also Condition (RS2) is new and together with Instruction (CR2) ensures that every $+(a,b) : r$ is entailed, as observed in Lemma 4.5 in [12]. Moreover, the interplay of Conditions (RS4) and (RS5) and Instruction (CR2) makes sure that no $-\,(a,b) : r$ is entailed. Only (RS3) is from the original definition and guarantees together with Instructions (CR1) and (CR2) that no $-\,a : C$ is entailed [4, 6]. In the end, the *canonical* IRQ-*repair* induced by $\mathcal{S}$ is denoted as

$$\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S})$$

and its variable set consists of all copies $\langle\!\langle u, \Phi \rangle\!\rangle$ except those of the form $\langle\!\langle a, \mathcal{S}_a \rangle\!\rangle$, which rather are synonyms of the INs a. This repair is saturated, i.e. it entails a query $\gamma \in$ IRQ w.r.t. $\mathcal{T}$ iff. it entails γ w.r.t. the empty TBox.

Evidently, canonical repairs are computable in polynomial time w.r.t. data complexity, i.e. computation time is dominated by $\mathcal{T}$ and $\mathcal{P}$. In practice we should not compute the whole exponentially large canonical repair but only an equivalent sub-qABox, called *optimized repair* [6, 9]. Experiments have shown that such optimized repairs of real-world KBs can indeed be computed in practice [6]. With the revised definition of repair types used here a further speed-up is expected, but this would still need to be verified empirically.

Not every canonical repair is optimal, but every optimal repair is equivalent to a canonical one. Thus in order to get all optimal repairs in exponential time, we can enumerate all repair seeds, compute the induced repairs, and then filter out the non-optimal ones. Alternatively to filtering the repairs, we could also filter the repair seeds since it is decidable in polynomial time whether a repair seed induces an optimal repair [10, 12].

Comparison of Repair Seeds. IRQ-entailment between canonical repairs can be characterized with a relation between their seeds. Let Φ and Ψ be sets of CDs. We say that Φ is *covered* by Ψ and write $\Phi \leq \Psi$ if, for each $C \in \Phi$, there is $D \in \Psi$ with $C \sqsubseteq^{\emptyset} D$. Given repair seeds $\mathcal{S}$ and $\mathcal{S}'$, we say that $\mathcal{S}$ is IRQ-*covered* by $\mathcal{S}'$ and write $\mathcal{S} \leq_{\mathsf{IRQ}} \mathcal{S}'$ if $\mathcal{S}_a \leq \mathcal{S}'_a$ for each IN a and furthermore $\mathsf{Succ}(\mathcal{S}'_a, r, b) \leq \mathcal{S}'_b$ implies $\mathsf{Succ}(\mathcal{S}_a, r, b) \leq \mathcal{S}_b$ for each RA $(a,b){:}r$ in $\exists X.\mathcal{A}$, where

$$\mathsf{Succ}(\Phi, r, v) \coloneqq \{\, C \mid \exists r.C \in \Phi \text{ and } \mathcal{A}^{\mathcal{T}} \models v : C \,\}.$$

Note that, for each RA $(a,b){:}r$ in $\exists X.\mathcal{A}$, we have $\mathsf{Succ}(\mathcal{S}_a, r, b) \leq \mathcal{S}_b$ iff. the repair induced by $\mathcal{S}$ contains $(a,b) : r$, because $\langle\!\langle a, \mathcal{S}_a \rangle\!\rangle$ and a are synonyms, likewise for b, and Instruction (CR2) is equivalent to:

- $(\langle\!\langle u, \Phi \rangle\!\rangle, \langle\!\langle v, \Psi \rangle\!\rangle) : r$ if $(u,v) : r \in \mathcal{A}^{\mathcal{T}}$ and $\mathsf{Succ}(\Phi, r, v) \leq \Psi$.

[3] $\mathcal{ELROI}(\bot)$ is the normal form of Horn-$\mathcal{ALCROI}$ where, additionally, regular role expressions can be used in parts of negative polarity, e.g. in premises of CIs and in repair requests.
[4] $\mathsf{Atoms}(\mathcal{T}, \mathcal{P})$ denotes the set of all atoms occurring in $\mathcal{P}$ or $\mathcal{T}$.
[5] Here the quantifier "$\exists X^{\mathcal{T}}.$" is dropped to allow named access to the variables, which would otherwise be protected from outside access. This is only done in technical considerations and must not be allowed when users access qABoxes.
[6] This is the revised, more efficient definition as in [4] and differs from earlier articles.
[7] This is equivalent to $\mathcal{A}^{\mathcal{T}} \models a : C$ since $\exists X^{\mathcal{T}}.\mathcal{A}^{\mathcal{T}}$ is the saturation.

Lemma 2.D. *If* $\mathcal{S} \leq_{\mathsf{IRQ}} \mathcal{S}'$, *then* $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A},\mathcal{S}) \models_{\mathsf{IRQ}} \mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A},\mathcal{S}')$.

Proof. Analogously as in the proof of Proposition 27 in [14] one can show that the relation $\{\ (\langle\!\langle u,\Phi'\rangle\!\rangle,\langle\!\langle u,\Phi\rangle\!\rangle) \mid \Phi \leq \Phi'\ \}$ is a simulation from $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A},\mathcal{S}')$ to $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A},\mathcal{S})$, and thus each CA entailed by the repair induced by $\mathcal{S}'$ is also entailed by the repair induced by $\mathcal{S}$. Since $\mathsf{Succ}(\mathcal{S}'_a,r,b) \leq \mathcal{S}'_b$ implies $\mathsf{Succ}(\mathcal{S}_a,r,b) \leq \mathcal{S}_b$ for each RA $(a,b):r$ in $\exists X.\mathcal{A}$, we further conclude that each RA entailed by the repair induced by $\mathcal{S}'$ is entailed by the repair induced by $\mathcal{S}$ as well. Thus, $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A},\mathcal{S})$ IRQ-entails $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A},\mathcal{S}')$. □

The converse direction also holds w.r.t. the old definition of repair types used in [6, 8, 12, 13]. For the more efficient definition this is currently unclear, but we won't need it here.

Two Approaches to Repairing for Role Assertions. There are two ways to treat the RAs in the removal part $\mathcal{P}_-$. Here, we use nominals to replace each RA $(a,b):r$ with the equivalent CA $a:\exists r.\{b\}$ [9]. Alternatively, we could first repair for all RAs in $\mathcal{P}_-$ and then repair for all CAs in $\mathcal{P}_-$ [12].

With the first option (stripped down to $\mathcal{EL}$ as above) we have the additional atoms $\exists r.\{b\}$ and $\{b\}$ for each RA $(a,b):r$ in $\mathcal{P}_-$. Since they cannot occur in the $\mathcal{EL}$ TBox $\mathcal{T}$, they are not implied by other CDs and thus will not be treated by Condition (RT3) in any way. In order to obtain a repair seed $\mathcal{S}$, it is only necessary to include the atom $\exists r.\{b\}$ in the repair type $\mathcal{S}_a$ for every $(a,b):r$ in $\mathcal{P}_-$, and to require that $\mathcal{S}_b$ must not contain the atom $\{b\}$, see Conditions (RS4) and (RS5). With that, $\mathsf{Succ}(\mathcal{S}_a,r,b)$ cannot be covered by $\mathcal{S}_b$ and thus the induced repair does not entail $(a,b):r$. This first option was easier to implement for us.

In contrast, with the second option we would first compute the saturation, then copy every IN a into a fresh variable x_a, then delete all RAs in $\mathcal{P}_-$, and finally repair for the rest of $\mathcal{P}_-$. The connection to the first option is as follows. For each nominal $\{b\}$ coming from a RA $(a,b):r$ in $\mathcal{P}_-$, copies $\langle\!\langle b,\Phi\rangle\!\rangle$ in repairs by the first approach correspond to copies $\langle\!\langle b,\Phi\rangle\!\rangle$ in repairs by the second approach if $\{b\} \notin \Phi$, and to copies $\langle\!\langle x_b,\Phi\setminus\{\{b\}\}\rangle\!\rangle$ otherwise. This is because in the second approach b and x_b have the same repair types (since the RAs in $\mathcal{P}_-$ do not contribute any atoms), and in the first approach Φ is a repair type for b iff. $\Phi\cup\{\{b\}\}$ is a repair type for b, and the repair types for b not containing $\{b\}$ are the same in both approaches. That is, the duplication of INs is implicitly done by the first option employed here.

3 The Smart Interaction Strategy

When a KB consisting of a qABox $\exists X.\mathcal{A}$ and a static $\mathcal{EL}$ TBox $\mathcal{T}$ should be repaired for a feasible repair request $\mathcal{P}$, it is not useful to compute all optimal repairs by enumerating all repair seeds and then let the experts choose among them. With that approach the workload of the experts would be too high since in the worst case there are exponentially many optimal repairs and each of them might be of exponential size (even their optimized variant).

From a practical perspective, not every optimal repair makes sense in the domain of interest. This is because the unwanted consequences in $\mathcal{P}_-$ as well as their logical causes w.r.t. $\mathcal{T}$ may contain conjunctions, and in order to prevent entailment of a conjunction it suffices to choose one of its conjuncts and make it not entailed. The construction of an optimal repair is thus non-deterministic and there are multiple repair seeds in general. Instead of making random choices, we recommend to interact with the experts to refine the given repair request $\mathcal{P}$ to a deterministic one and so identify a useful optimal repair. With our interactive approach the experts need to answer at most polynomially many questions, i.e. their workload is significantly lower.

This section presents the *smart interaction strategy*, which runs in two phases. By means of backward chaining, Phase 1 computes causes of identified errors and interacts with the experts when choices must be made. Phase 1 terminates with a deterministic refinement of the initially provided repair request $\mathcal{P}$. Already with Phase 1 every optimal repair can be reached. Phase 2 is concerned with disputable consequences: they first need to be decided by the experts, and then Phase 2 proceeds with the further refined repair request like Phase 1. In the end, a repair that is optimal w.r.t. the given repair request $\mathcal{P}$ and all experts answers has been identified.

3.1 Fundamentals of the Strategy and Phase 1

In the following, let $\mathcal{P}^0 \coloneqq \mathcal{P}$ be the initial repair request. The strategy maintains three sets. Undecided queries are held by the set $\mathcal{Q}$, which is initially empty, and all queries currently in $\mathcal{Q}$ are displayed to the experts for the purpose of decision making, e.g. in form of a list with action buttons for each entry (accept and reject). Queries accepted by the experts are added to $\mathcal{P}_+$, whereas rejected queries are added to $\mathcal{P}_-$. Initially all queries not entailed by the input KB are removed from $\mathcal{P}_-$ since for these no repair is necessary. The strategy evolves as follows, always ensuring that $\mathcal{P}_+ \uplus \mathcal{P}_-$ refines $\mathcal{P}^0$ and is feasible (i.e. $\mathcal{P}_+ \not\models^{\mathcal{T}} \beta$ for each $\beta \in \mathcal{P}_-$).

(SIS1) Whenever a non-atomic query α has been added to $\mathcal{P}_-$, it needs to be inspected to find out why it should not hold.
- If α has the form $a:C$ where C is a conjunction, then for each top-level conjunct $D \in \mathsf{Conj}(C)$, the query $a:D$ is added to $\mathcal{Q}$.
- If α has the form $a:\exists r.C$, then for each IN b where $\exists X.\mathcal{A}$ entails $(a,b):r$ and $b:C$ w.r.t. $\mathcal{T}$, the two queries $(a,b):r$ and $b:C$ are added to $\mathcal{Q}$.

Nothing needs to be done for atomic queries, i.e. CAs $a:A$ where A is a CN or RAs $(a,b):r$, since these can be directly deleted from the qABox.

(SIS2) Furthermore, we need to ensure that inference with the TBox cannot restore a rejected query: for each rejected query α added to $\mathcal{P}_-$, all implicant queries β with $\exists X.\mathcal{A} \models^{\mathcal{T}} \beta$ and $\beta \models^{\mathcal{T}} \alpha$ must be rejected too.
- RAs have no implicants w.r.t. an $\mathcal{EL}$ TBox and thus need no further treatment.
- For CAs, it suffices to restrict attention to the implicants as in Condition (RT3): when a CA $a:C$ has been added to $\mathcal{P}_-$, we add to $\mathcal{P}_-$ all CAs $a:D$ where $D \sqsubseteq E$ is a CI in $\mathcal{T}$ with $\exists X.\mathcal{A} \models^{\mathcal{T}} a:D$[8] and $E \sqsubseteq^{\mathcal{T}} C$.

[8] Since the IN a already occurs in the input qABox $\exists X.\mathcal{A}$, we do not need to use the saturation here but must thus take the TBox in the entailment into account.

(SIS3) Last, inherited answers are computed after every answer received from the experts. To this end, each currently undecided query α in $\mathcal{Q}$ is checked.
- If $\mathcal{P}_+$ entails α w.r.t. $\mathcal{T}$, then α inherits acceptance and is moved from $\mathcal{Q}$ to $\mathcal{P}_+$.
- If $\mathcal{P}_+ \cup \{\alpha\}$ entails w.r.t. $\mathcal{T}$ any assertion in $\mathcal{P}_-$, then α inherits rejection and is moved from $\mathcal{Q}$ to $\mathcal{P}_-$.

These instructions enable control of the delete operation. Phase 1 ends as soon as no undecided queries are contained in $\mathcal{Q}$ anymore. As all queries are built from the polynomially many sub-CDs of the input and $\mathcal{EL}$ allows for polynomial-time reasoning, Phase 1 terminates in polynomial time.

3.2 Induced Repairs after Phase 1

We denote by $\mathcal{P}^1_+$ and $\mathcal{P}^1_-$ the sets of accepted and, respectively, rejected queries at the end of Phase 1. Then $\mathcal{P}^1 \coloneqq \mathcal{P}^1_+ \uplus \mathcal{P}^1_-$ is a repair request that refines the initial repair request $\mathcal{P}^0$. Furthermore, we define the mapping $\mathcal{S}^1$ that sends each IN a to the set

$$\mathcal{S}^1_a \coloneqq \mathsf{Max}\{\, C \mid a : C \in \mathcal{P}^1_- \,\} \cup \{\, \exists r.\{b\} \mid (a,b) : r \in \mathcal{P}^1_- \,\},$$

where the operator Max selects the CDs that are maximal w.r.t. subsumption $\sqsubseteq^\emptyset$. We first show that this definition yields a repair seed for $\mathcal{P}^1$. It follows that its induced repair entails each accepted query but no rejected query, i.e. the identified repair actually reflects all decisions made by the experts. In particular, it is a repair for $\mathcal{P}^0$ since $\mathcal{P}^1$ refines $\mathcal{P}^0$.

Lemma 3.1. *$\mathcal{S}^1$ is a repair seed for $\mathcal{P}^1$ and thus also for $\mathcal{P}^0$.*

Proof. We first explain why each set $\mathcal{S}^1_a$ is a repair type for a.

(RT1) Condition (RT1) is satisfied since each query shown to the experts is entailed by the input qABox $\exists X.\mathcal{A}$ w.r.t. the given TBox $\mathcal{T}$.

(RT2) The operator Max in the above definition ensures that Condition (RT2) holds.

(RT3) Since Instruction (SIS2) ensures that causes of rejected queries are identified and treated as rejected too, and since Instruction (SIS1) ensures that, for each rejected query $a : C$, there is a rejected query $a : D$ with D an atom subsuming C, Condition (RT3) is satisfied as well.

Now we verify that $\mathcal{S}^1$ satisfies the five conditions of a repair seed.

(RS1) Let $+\,a : C \in \mathcal{P}^1$ and assume to the contrary that there was an atom $D \in \mathcal{S}^1_a$ with $C \sqsubseteq^\emptyset D$, i.e. $-\,a : D \in \mathcal{P}^1$. Then $\mathcal{P}^1_+$ would entail the rejected query $a : D$, which contradicts Instruction (SIS3): if $a : C$ appeared first, then $a : D$ would have inherited acceptance, and otherwise $a : C$ would have inherited rejection.

(RS2) by Lemma 3.A, see below.

(RS3) by the very definition of $\mathcal{S}^1$ since initially all CAs in $\mathcal{P}^0_-$ and entailed by the input KB are in $\mathcal{P}_-$. To see this, consider a CA $a : C$ in $\mathcal{P}^0_-$ with $\exists X.\mathcal{A} \models^\mathcal{T} a : C$. Then $\mathcal{P}_-$ contains $a : C$. If C is no atom, then Instruction (SIS1) adds $a : D$ for all $D \in \mathsf{Conj}(C)$ to the set $\mathcal{Q}$ of undecided queries. For Instruction (SIS3) at least one of them must be rejected: if the user has accepted all but a last one, say $a : E$, then $\mathcal{P}_+$ contains $a{:}D$ for all $D \in \mathsf{Conj}(C) \setminus \{E\}$ and thus $a{:}E$ inherits rejection since $\mathcal{P}_+ \cup \{a{:}E\} \models^\mathcal{T} a{:}C$. In any case, at the end of Phase 1 there is a top-level conjunct $D \in \mathsf{Conj}(C)$ with $a{:}D \in \mathcal{P}^1_-$. Then $C \sqsubseteq^\emptyset D$ and, moreover, $\mathcal{S}^1_a$ contains either D itself or, for the operator Max, an atom subsuming D. Thus $C \sqsubseteq^\emptyset F$ for some $F \in \mathcal{S}^1_a$.

(RS4) by the very definition of $\mathcal{S}^1$ since initially all RAs in $\mathcal{P}^0_-$ and entailed by the input KB are in $\mathcal{P}_-$.

(RS5) Every query $a : \{a\}$ is automatically accepted by Instruction (SIS3). None of them is therefore in $\mathcal{P}^1_-$ and thus $\{a\} \notin \mathcal{S}^1_a$. ◻

Lemma 3.A. *$\mathsf{Succ}(\mathcal{S}^1_a, r, b) \leq \mathcal{S}^1_b$ for each RA $(a,b) : r$ in $\exists X.\mathcal{A}$ and not rejected during Phase 1 (i.e. not in $\mathcal{P}^1_-$).*

Proof. First note that $\exists r.\{b\}$ is not in $\mathcal{S}^1_a$ since $(a,b) : r$ has not been rejected in Phase 1. Now consider an existential requirement $\exists r.C$ in $\mathcal{S}^1_a$ with $\exists X.\mathcal{A} \models^\mathcal{T} b : C$. This means that $a : \exists r.C$ is a rejected query in $\mathcal{P}^1_-$. After it has been rejected, the new queries $(a,b) : r$ and $b : C$ are generated by Instruction (SIS1). Since the first has not been rejected, the second must have been, i.e. $b : C$ is added to $\mathcal{P}_-$. Subsequently, also a query $b : D$ where $D \in \mathsf{Conj}(C)$ must have been rejected due to Instructions (SIS1) and (SIS3), and thus $\mathcal{S}^1_b$ contains an atom subsuming C. ◻

Next, we show that the strategy is fine-grained enough in the sense that a unique repair is identified.

Proposition 3.2. *$\mathcal{P}^1$ is a deterministic repair request, for which the only optimal IRQ-repair is the one induced by $\mathcal{S}^1$.*

Proof. We show that every IRQ-repair of the input KB for $\mathcal{P}^1$ is IRQ-entailed by $\mathsf{rep}^\mathcal{T}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S}^1)$. To this end, consider an IRQ-repair $\exists Y.\mathcal{B}$ for $\mathcal{P}^1$. Then there is a repair seed $\mathcal{S}^\mathcal{F}$ such that the induced canonical IRQ-repair $\mathsf{rep}^\mathcal{T}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S}^\mathcal{F})$ IRQ-entails $\exists Y.\mathcal{B}$ [6, 8]. This seed $\mathcal{S}^\mathcal{F}$ is defined by means of a mapping $\mathcal{F}$ with which the aforementioned IRQ-entailment was verified — here it suffices to know that the repair types $\mathcal{S}^\mathcal{F}_a$ have the following form:

$$\mathcal{S}^\mathcal{F}_a = \mathsf{Max}\{\, C \mid C \in \mathsf{Atoms}(\mathcal{T},\mathcal{P}),\ \exists X.\mathcal{A} \models^\mathcal{T} a : C, \text{ and } \exists Y.\mathcal{B} \not\models^\mathcal{T} a : C \,\}.$$

According to Lemma 2.D it suffices to verify the following two claims since then $\mathsf{rep}^\mathcal{T}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S}^1)$ IRQ-entails $\mathsf{rep}^\mathcal{T}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S}^\mathcal{F})$ and thus also $\exists Y.\mathcal{B}$.

(1) $\mathcal{S}^1_a \leq \mathcal{S}^\mathcal{F}_a$ for each IN a.
(2) $\mathsf{Succ}(\mathcal{S}^\mathcal{F}_a, r, b) \leq \mathcal{S}^\mathcal{F}_b$ implies $\mathsf{Succ}(\mathcal{S}^1_a, r, b) \leq \mathcal{S}^1_b$ for each RA $(a,b) : r$ in $\exists X.\mathcal{A}$.

We start with the first claim. To this end, consider an atom $C \in \mathcal{S}^1_a$, i.e. the assertion $a{:}C$ is in $\mathcal{P}^1_-$. Since $\exists Y.\mathcal{B}$ is a repair for $\mathcal{P}^1$, it does not entail this assertion w.r.t. $\mathcal{T}$. Furthermore, this assertion $a : C$ is entailed by the input qABox $\exists X.\mathcal{A}$ w.r.t. $\mathcal{T}$, since otherwise it would not have been considered as a query by the smart interaction strategy and could then not be in $\mathcal{P}^1_-$. Therefore $\mathcal{S}^\mathcal{F}_a$ contains C itself or another atom subsuming C. Since this holds for all C, we conclude that $\mathcal{S}^1_a \leq \mathcal{S}^\mathcal{F}_a$ for every IN a.

It remains to verify the second claim. Assume that $(a,b) : r$ is a RA in $\exists X.\mathcal{A}$ with $\mathsf{Succ}(\mathcal{S}^\mathcal{F}_a, r, b) \leq \mathcal{S}^\mathcal{F}_b$, i.e. $\mathsf{rep}^\mathcal{T}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S}^\mathcal{F})$ contains the RA $(a,b) : r$. Since $\mathsf{rep}^\mathcal{T}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S}^\mathcal{F})$ is a repair for

$\mathcal{P}^1$, this RA is not contained in $\mathcal{P}^1_-$. By Lemma 3.A we infer that $\mathsf{Succ}(\mathcal{S}^1_a, r, b) \leq \mathcal{S}^1_b$. ◻

As first main result, we verify that every optimal repair can really be reached with the strategy.

Theorem 3.3. *For every optimal* IRQ*-repair for* $\mathcal{P}^0$*, an inducing repair seed can be identified with Phase 1.*

Proof. Consider an optimal IRQ-repair for $\mathcal{P}^0$. Since every repair is entailed by a canonical repair, every optimal repair is equivalent to a canonical repair. Thus, we can assume w.l.o.g. that the considered optimal repair is of the form $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S})$ for a repair seed $\mathcal{S}$.

During Phase 1, we accept an undecided query iff. it follows from $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S})$. We will show that $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S})$ is IRQ-entailed by $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S}^1)$, where $\mathcal{S}^1$ is the repair seed obtained at the end of Phase 1. Optimality then yields that both are IRQ-equivalent.

First, we show two claims.

(1) $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S})$ entails each accepted query in $\mathcal{P}^1_+$.

We show the claim by induction. Initially, $\mathcal{P}_+ = \mathcal{P}^0_+$ and each query in this set is entailed by $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S})$ by assumption. Next, a query that is manually added to $\mathcal{P}_+$ follows from $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S})$ according to the above specification. Last, a query added to $\mathcal{P}_+$ by inheritance in Instruction (SIS3) follows from the set of all previously accepted queries. By induction hypothesis, all these previously accepted queries follow from $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S})$, and thus also the newly added one.

(2) $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S})$ entails no rejected query in $\mathcal{P}^1_-$.

Again, we use induction. In the beginning we have $\mathcal{P}_- = \mathcal{P}^0_-$ and since $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S})$ is a repair for $\mathcal{P}^0$ it does not entail any query in $\mathcal{P}_-$. Next, each query manually added to $\mathcal{P}_-$ according the above specification is not entailed by $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S})$. Last, consider a query α added to $\mathcal{P}_-$ by inheritance in Instruction (SIS3), i.e. there is a previously rejected query β in $\mathcal{P}_-$ with $\mathcal{P}_+ \cup \{\alpha\} \models^{\mathcal{T}} \beta$. By induction hypothesis, β is not entailed by $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S})$, and the first claim above yields that $\mathcal{P}_+$ is entailed by $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S})$. It follows that α cannot be entailed by $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S})$.

We proceed with proving that $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S}^1) \models_{\mathsf{IRQ}} \mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S})$. To this end, consider a CA $a : C$ not entailed by $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S}^1)$. If this CA is not entailed by the input KB, then it cannot be entailed by $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S})$ either. Otherwise, Lemma XII in the erratum to [4, 5] yields an atom $D \in \mathcal{S}^1_a$ with $C \sqsubseteq^{\mathcal{T}} D$, and thus $a : D$ is a rejected query in $\mathcal{P}^1_-$. By Claim 2 above we infer that $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S})$ does not entail $a : D$, and thus neither $a : C$.

It remains to consider the RAs. Let $(a, b) : r$ be not entailed by $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S}^1)$. Since in $\mathcal{EL}$ entailment of a RA is the same as containment of it, we infer that $(a, b) : r$ is not contained in $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S}^1)$ either. If the input KB does not contain this RA, then neither does $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S})$. Otherwise, we have $\mathsf{Succ}(\mathcal{S}^1_a, r, b) \not\leq \mathcal{S}^1_b$ by definition. Lemma 3.A yields $(a, b) : r \in \mathcal{P}^1_-$ and thus $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S})$ does not entail $(a, b) : r$ by Claim 2 above. ◻

3.3 Control of the Copy Operation

So far, Phase 1 only allows to control the delete operation, i.e. the repair might contain undesired copies. On the one hand, such a situation can only occur when the input is insufficiently specified: the repair request does not preclude these copies and thus an optimal repair will contain them. On the other hand, these copies might not be immediately problematic since they could only later be revealed when the repair is queried — then one can simply repair again.[9]

In order to also control the copy operation and thus the creation of copies, we can employ the following additional instruction.

(SIS4) When a query $a : \exists r_1. \cdots \exists r_n.C$ has been added to $\mathcal{P}_-$, the experts needs to specify which copies of objects linked to a by an $r_1 \cdots r_n$-chain exist.

- If C is a conjunction, then the query $a : \exists r_1. \cdots \exists r_n.C \setminus D$ is added to $\mathcal{Q}$ for each top-level conjunct $D \in \mathsf{Conj}(C)$, where $C \setminus D \coloneqq \bigsqcap \mathsf{Conj}(C) \setminus \{D\}$.
- The query $a : \exists r_1. \cdots \exists r_n.E$ is added to $\mathcal{Q}$ for each CI $E \sqsubseteq F \in \mathcal{T}$ with $\exists X.\mathcal{A} \models^{\mathcal{T}} a : \exists r_1. \cdots \exists r_n.E$ and $F \sqsubseteq^{\mathcal{T}} C$.

The number of queries to be decided by the experts is polynomial with Instructions (SIS1), (SIS2), and (SIS3), but can be worst-case exponential when the additional Instruction (SIS4) is employed.

Another alternative would be to wait until Phase 1 without the above additional instruction has finished, and then determine superfluous copies as follows. For each object u reachable from an IN a by a role path $r_1, \ldots, r_n$ in the input KB, determine all copies $\langle\!\langle u, \Phi_1 \rangle\!\rangle, \ldots, \langle\!\langle u, \Phi_n \rangle\!\rangle$ in the repair induced by $\mathcal{S}^1$ that are still reachable from a by the role path $r_1, \ldots, r_n$. Then compute characteristic CDs $C_1, \ldots, C_n$ such that C_i only has copy $\langle\!\langle u, \Phi_i \rangle\!\rangle$ as instance and present the queries $a : \exists r_1. \cdots \exists r_n.C_i$ to the experts. If such a query is rejected, then the corresponding copy $\langle\!\langle u, \Phi_i \rangle\!\rangle$ is deleted. With that, still all causes need to be considered by means of Instruction (SIS2), viz. to avoid re-introduction of the unwanted copies through the TBox. Since there might be exponentially many copies in the worst case, this alternative approach could also need exponentially many additional queries (at least one for each reachable copy).

3.4 Disputable Consequences and Phase 2

We now consider the disputable consequences of the input KB $\mathcal{T} \uplus \exists X.\mathcal{A}$ w.r.t. the refined repair request $\mathcal{P}^1$ obtained from Phase 1, which we also call *disputable consequence at the end of Phase 1.* In this specific setting, Definitions 2.3 and 2.4 read as follows.

Definition 3.4. A *substantiation* of a query $\gamma \in \mathsf{IRQ}$ is a qABox $\exists Y.\mathcal{B}$ with $\exists X.\mathcal{A} \cup \mathcal{P}^1_+ \models \exists Y.\mathcal{B}$ and $\exists Y.\mathcal{B} \models^{\mathcal{T}} \gamma$.

We obtain the following corollary to Lemma 2.B. Regarding the first statement, recall that we assume $\exists X.\mathcal{A} \models^{\mathcal{T}} \mathcal{P}_+$ and thus $\exists X.\mathcal{A} \cup \mathcal{P}_+ \models^{\mathcal{T}} \gamma$ iff. $\exists X.\mathcal{A} \models^{\mathcal{T}} \gamma$.

[9] The author assumes that, in practice, nobody would read a large real-world KB statement by statement.

Corollary 3.5. *A query $\gamma \in \mathsf{IRQ}$ is a disputable consequence at the end of Phase 1 iff. it fulfills all of the following conditions:*

(DC1) *γ is entailed by $\exists X.\mathcal{A}$ w.r.t. $\mathcal{T}$.*

(DC2) *γ has not been decided by the experts or by inheritance, i.e.*
- *$\mathcal{P}^1_+ \cup \{\gamma\}$ does not entail w.r.t. $\mathcal{T}$ any query in $\mathcal{P}^1_-$, and*
- *$\mathcal{P}^1_+$ does not entail γ w.r.t. $\mathcal{T}$.*

(DC3) *For every substantiation $\mathcal{J}$ of γ, we have that $\mathcal{J}$ does not entail w.r.t. $\mathcal{T}$ all accepted queries in $\mathcal{P}^1_+$, or $\mathcal{J}$ entails w.r.t. $\mathcal{T}$ some rejected query in $\mathcal{P}^1_-$.*

Recall from Lemma 3.1 that the repair induced by the repair seed identified in Phase 1 does not entail any rejected query in $\mathcal{P}^1_-$. No substantiations for a disputable consequence γ can thus be retained in the repair, i.e. all bare causes for γ in the data must be removed. Since validity of γ cannot be decided by the information collected from the experts in Phase 1, and keeping γ in the repair might not be reasonable, such disputable consequences γ are rather shown to the experts for examination, viz. right in the beginning of Phase 2.

Phase 2 controls the saturation and runs as follows. First we compute all disputable consequences. Since we consider the query language IRQ, we restrict attention to disputable RAs and disputable CAs involving sub-CDs of the input, i.e. of the form $a : C$ where C occurs in $\mathcal{P}^0$ or $\mathcal{T}$. All these queries are then presented to the experts by adding them to $\mathcal{Q}$. Afterwards, the strategy proceeds as in Phase 1. Like the first phase, Phase 2 ends when $\mathcal{Q}$ contains no undecided query anymore. Then all disputable consequences have been processed, and the final repair is computed from the so identified repair seed $\mathcal{S}^2$. Lemma 3.1 and Proposition 3.2 hold analogously.

4 Computing Disputable Consequences

Next, we will develop a practical method for computing the disputable consequences at the end of Phase 1. Conditions (DC1) and (DC2) in Corollary 3.5 can be checked in polynomial time, but it is not obvious how Condition (DC3) can be checked in an efficient way. Naïvely following the very definition will not work in practice since we would need to go through all queries in IRQ and all substantiations. In order to find a more efficient approach, we transform Condition (DC3) into the following equivalent condition:

- For every qABox $\exists Y.\mathcal{B}$, if $\exists X.\mathcal{A} \cup \mathcal{P}^1_+ \models \exists Y.\mathcal{B}$, and $\exists Y.\mathcal{B} \models^{\mathcal{T}} \alpha$ for every $\alpha \in \mathcal{P}^1_+$, and $\exists Y.\mathcal{B} \not\models^{\mathcal{T}} \beta$ for each $\beta \in \mathcal{P}^1_-$, then $\exists Y.\mathcal{B} \not\models^{\mathcal{T}} \gamma$. (see proof of Proposition 4.8)

Now the qABoxes $\exists Y.\mathcal{B}$ almost satisfy the definition of a repair of $\exists X.\mathcal{A}$ for $\mathcal{P}^1$, the only exceptions are that the entailment of $\exists Y.\mathcal{B}$ by the input KB does not involve the TBox $\mathcal{T}$ but additionally $\mathcal{P}^1_+$. Such repairs are obtained directly from $\exists X.\mathcal{A}$ and not from its saturation, and are thus called "unsaturated."

Definition 4.1. An *unsaturated repair* of a qABox $\exists X.\mathcal{A}$ for a repair request $\mathcal{P}$ w.r.t. a TBox $\mathcal{T}$ is a qABox $\exists Y.\mathcal{B}$ such that

(UR1) $\exists X.\mathcal{A} \cup \mathcal{P}_+ \models \exists Y.\mathcal{B}$,

(UR2) $\exists Y.\mathcal{B} \models^{\mathcal{T}} \alpha$ for each $\alpha \in \mathcal{P}_+$, and

(UR3) $\exists Y.\mathcal{B} \not\models^{\mathcal{T}} \beta$ for each $\beta \in \mathcal{P}_-$.

Moreover, we call $\exists Y.\mathcal{B}$ QL-*optimal* if it is not strictly QL-entailed by another unsaturated repair.

With this definition in place, Condition (DC3) can be rewritten to:

- no unsat. repair of $\exists X.\mathcal{A}$ for $\mathcal{P}^1$ w.r.t. $\mathcal{T}$ entails γ w.r.t. $\mathcal{T}$

4.1 Unsaturated Repairs

Unsaturated repairs can be constructed from the input qABox and not from its saturation, i.e. they are independent from the employed query language. Only optimality must refer to a particular entailment relation. In order to define a canonical form of unsaturated repairs, we keep the notion of repair types as it is and only include copies of objects of the input qABox, whereas objects and assertions produced by the saturation are ignored.

Since we also need to take the addition part $\mathcal{P}_+$ into account, we first define the qABox $\exists X'.\mathcal{A}'$ as the union of $\exists X.\mathcal{A}$ and a qABox equivalent to $\mathcal{P}_+$, which can be obtained from $\mathcal{P}_+$ by replacing each CA with a tree-shaped sub-qABox (e.g. obtained by introducing a variable for each intent C of an existential requirement $\exists r.C$ occurring in $\mathcal{P}_+$, as explained on Page 5).

Lemma 4.2. *There is a homomorphism h from $\exists X'.\mathcal{A}'$ to the IQ-saturation $\exists X^{\mathcal{T}}.\mathcal{A}^{\mathcal{T}}$ such that $h(u) = u$ for each object u of $\exists X.\mathcal{A}$.*

Proof. We build the required homomorphism h in two steps. First, since $\exists X.\mathcal{A}$ is a sub-qABox of its IQ-saturation $\exists X^{\mathcal{T}}.\mathcal{A}^{\mathcal{T}}$, we can define $h(u) \coloneqq u$ for each object u of $\exists X.\mathcal{A}$.

Next, recall the assumption that $\exists X.\mathcal{A} \models^{\mathcal{T}} \mathcal{P}_+$. This implies that the matrix $\mathcal{A}$ contains all RAs in $\mathcal{P}_+$ and further that the IQ-saturation $\exists X^{\mathcal{T}}.\mathcal{A}^{\mathcal{T}}$ entails all CAs in $\mathcal{P}_+$ (w.r.t. the empty TBox). The latter means that, for each $a{:}C$ in $\mathcal{P}_+$, there is a homomorphism $h_{a:C}$ from a qABox equivalent to $\{a{:}C\}$ to $\exists X^{\mathcal{T}}.\mathcal{A}^{\mathcal{T}}$. Now $\exists X'.\mathcal{A}'$ is (equivalent to) the union of $\exists X.\mathcal{A}$ with all the latter qABoxes, where w.l.o.g. all these qABoxes have pairwise disjoint sets of variables. We thus further define $h(u) \coloneqq h_{a:C}(u)$ for each object u of the qABox equivalent to $\{a : C\}$. The so obtained mapping is well-defined since, for each $a : C$ in $\mathcal{P}_+$, we have $h_{a:C}(a) = a$ and a is the only IN in the qABox equivalent to $\{a : C\}$. □

Definition 4.3. The *canonical unsaturated repair* induced by a repair seed $\mathcal{S}$ is the qABox $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S})$ of which the objects are all pairs $\langle\!\langle u, \Phi \rangle\!\rangle$ with an object u of $\exists X'.\mathcal{A}'$ and a repair type Φ for $h(u)$, and the matrix consists of the following assertions:

(CUR1) $\langle\!\langle u, \Phi \rangle\!\rangle : A$ if $u : A \in \mathcal{A}'$ and $A \notin \Phi$, and

(CUR2) $(\langle\!\langle u, \Phi \rangle\!\rangle, \langle\!\langle v, \Psi \rangle\!\rangle){:}r$ if $(u, v){:}r \in \mathcal{A}'$ and $\mathsf{Succ}(\Phi, r, h(v)) \leq \Psi$.

Moreover, an object $\langle\!\langle u, \Phi \rangle\!\rangle$ is a variable if u is a variable (i.e. $u \in X$) or $\Phi \neq \mathcal{S}_u$, and each IN a and $\langle\!\langle a, \mathcal{S}_a \rangle\!\rangle$ are treated as synonyms.

Lemma 4.A. *The mapping $\langle\!\langle u, \Phi \rangle\!\rangle \mapsto u$ is a homomorphism from $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S})$ to $\exists X'.\mathcal{A}'$, and $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S})$ satisfies Condition (UR1).*

Proof. We first verify that the mapping is a homomorphism.

- For each IN a, we have $a = \langle\!\langle a, \mathcal{S}_a \rangle\!\rangle$ by Definition 4.3, and thus a is mapped to a.
- If the matrix of $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S})$ contains $\langle\!\langle u, \Phi \rangle\!\rangle{:}A$, then by Instruction (CUR1) the matrix of $\exists X'.\mathcal{A}'$ contains $u : A$.
- If $(\langle\!\langle u, \Phi \rangle\!\rangle, \langle\!\langle v, \Psi \rangle\!\rangle) : r$ is in the matrix of $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S})$, then by Instruction (CUR1) the matrix of $\exists X'.\mathcal{A}'$ contains $(u, v) : r$.

Since entailment between qABoxes coincides with existence of a homomorphism in the converse direction, it follows that $\exists X'.\mathcal{A}'$ entails $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S})$. □

Lemma 4.B. *The mapping* $\langle\!\langle u, \Phi\rangle\!\rangle \mapsto \langle\!\langle h(u), \Phi\rangle\!\rangle$ *is a homomorphism from* $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S})$ *to* $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S})$*, and* $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S})$ *satisfies Condition (UR3).*

Proof. We begin with showing that the mapping is a homomorphism.

- Consider an IN a. Then $h(a) = a$ by definition of a homomorphism, $a = \langle\!\langle a, \mathcal{S}_a\rangle\!\rangle$ by Definition 4.3, and also $a = \langle\!\langle a, \mathcal{S}_a\rangle\!\rangle$ by definition of a canonical (saturated) repair. Thus, a is mapped to a.
- Let $\langle\!\langle u, \Phi\rangle\!\rangle : A$ be in the matrix of $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S})$, i.e. $u : A \in \mathcal{A}'$ and $A \notin \Phi$ by Instruction (CUR1). Then the matrix of the saturation of $\exists X.\mathcal{A}$ contains $h(u) : A$, and thus the matrix of $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S})$ contains $\langle\!\langle h(u), \Phi\rangle\!\rangle : A$.
- Last, consider $(\langle\!\langle u, \Phi\rangle\!\rangle, \langle\!\langle v, \Psi\rangle\!\rangle) : r$ in the matrix of $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S})$. For Instruction (CUR2) we thus have $(u, v) : r \in \mathcal{A}'$ and $\mathsf{Succ}(\Phi, r, h(v)) \leq \Psi$. It follows that the matrix of the saturation of $\exists X.\mathcal{A}$ contains $(h(u), h(v)) : r$, which implies that the matrix of $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S})$ contains $(\langle\!\langle h(u), \Phi\rangle\!\rangle, \langle\!\langle h(v), \Psi\rangle\!\rangle) : r$.

We conclude that $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S})$ entails $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S})$. Since $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S})$ entails w.r.t. $\mathcal{T}$ no assertions in $\mathcal{P}_-$, neither does $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S})$. ◻

Lemma 4.C. *The matrix of* $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S})$ *entails* $\langle\!\langle u, \Phi\rangle\!\rangle : C$ *iff. the matrix of* $\exists X'.\mathcal{A}'$ *entails* $u : C$ *and* C *is not subsumed w.r.t.* $\mathcal{T}$ *by any atom in* Φ.

Proof. We start with the only-if direction and assume that the matrix of the unsaturated repair $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S})$ entails $\langle\!\langle u, \Phi\rangle\!\rangle : C$. Then, since $\langle\!\langle u, \Phi\rangle\!\rangle \mapsto \langle\!\langle h(u), \Phi\rangle\!\rangle$ is a homomorphism from $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S})$ to $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S})$, the matrix of the saturated repair $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S})$ entails $\langle\!\langle h(u), \Phi\rangle\!\rangle : C$. Lemma XII in the erratum to [4, 5] implies that no atom in Φ subsumes C w.r.t. $\emptyset$. Furthermore, since $\langle\!\langle u, \Phi\rangle\!\rangle \mapsto u$ is a homomorphism from $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S})$ to $\exists X'.\mathcal{A}'$, the matrix of $\exists X'.\mathcal{A}'$ entails $u : C$.

Now assume that an atom D in Φ subsumed C w.r.t. $\mathcal{T}$. Since $D \in \Phi$, the matrix of the saturated repair $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S})$ would not entail $\langle\!\langle h(u), \Phi\rangle\!\rangle : D$ by Lemma XII in the erratum to [4, 5]. Since $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S})$ is saturated by Lemma XIII in the erratum to [4, 5] and $C \sqsubseteq^{\mathcal{T}} D$, the matrix of $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S})$ would neither entail $\langle\!\langle h(u), \Phi\rangle\!\rangle : C$, a contradiction. Consequently, C is not subsumed w.r.t. $\mathcal{T}$ by any atom in Φ.

We proceed with the if direction, assuming that the matrix of $\exists X'.\mathcal{A}'$ entails $u{:}C$ and C is not subsumed w.r.t. $\mathcal{T}$ by any atom in Φ. Actually, the proof is almost the same as for the second statement of Lemma XII in the erratum to [4, 5], but we include it here for being complete. We make an induction on C.

- The case where $C = \top$ is trivial.
- Assume that $C = A$ for a CN A. By Lemma II in [7], the matrix $\mathcal{A}'$ contains the CA $u : A$. Since no atom in Φ subsumes A w.r.t. $\mathcal{T}$, we infer that $A \notin \Phi$. According to Definition 4.3 the CA $\langle\!\langle u, \Phi\rangle\!\rangle : A$ is contained and thus entailed by the matrix of $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S})$.
- Let $C = C_1 \sqcap \cdots \sqcap C_n$ be a conjunction of atoms $C_1, \ldots, C_n$ where $n \geq 2$. The preconditions immediately imply that, for each index i, the matrix $\mathcal{A}'$ entails $u{:}C_i$ and Φ does not contain an atom subsuming C_i w.r.t. $\mathcal{T}$ (otherwise there would be an atom subsuming C since $C \sqsubseteq^{\emptyset} C_i$). The induction hypothesis yields that the matrix of $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S})$ entails $\langle\!\langle u, \Phi\rangle\!\rangle : C_i$ for each i, and thus also $\langle\!\langle u, \Phi\rangle\!\rangle : C$.
- Consider the last case where $C = \exists r.D$ is an existential requirement. According to Lemma II in [7], it follows from the preconditions that there exists some object v such that the matrix $\mathcal{A}'$ contains $(u, v){:}r$ and entails $v{:}D$. Since $\exists r.D$ is not subsumed by an atom in Φ w.r.t. $\mathcal{T}$, it follows that $D \not\sqsubseteq^{\mathcal{T}} E$ for each $\exists r.E \in \Phi$. Thus for each $\exists r.E \in \Phi$, there is some atom $F_E \in \mathsf{Conj}(E)$ such that $D \not\sqsubseteq^{\mathcal{T}} F_E$. According to Lemma XI in the erratum to [4, 5] there exists a repair type Ψ for $h(v)$ that covers the repair pre-type $\mathsf{Max}\{\, F_E \mid \exists r.E \in \Phi \text{ and } \mathcal{A}^{\mathcal{T}} \models h(v) : E \,\}$ and that does not contain an atom subsuming D w.r.t. $\mathcal{T}$. Applying the induction hypothesis then yields that the matrix of $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S})$ entails $\langle\!\langle v, \Psi\rangle\!\rangle : D$. By the very construction of Ψ, it follows that the matrix of $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S})$ contains the RA $(\langle\!\langle u, \Phi\rangle\!\rangle, \langle\!\langle v, \Psi\rangle\!\rangle) : r$, hence it entails $\langle\!\langle u, \Phi\rangle\!\rangle : C$.
  Note that, in the above repair pre-type it would be wrong to only include those maximal atoms F_E for which the matrix $\mathcal{A}'$ entails $v : E$ since it would then not be guaranteed that $\mathsf{Succ}(\Phi, r, h(v)) \leq \Psi$, i.e. the unsaturated repair might not contain the necessary RA. ◻

Lemma 4.D. $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S})$ *satisfies Condition (UR2).*

Proof. Consider a CA $a{:}C$ in $\mathcal{P}_+$. Then $\exists X'.\mathcal{A}'$ entails $a{:}C$ by its very definition. Moreover by Condition (RS1), C is not subsumed w.r.t. $\mathcal{T}$ by any atom in $\mathcal{S}_a$, and thus Lemma 4.C ensures that $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S})$ entails $a : C$ since a and $\langle\!\langle a, \mathcal{S}_a\rangle\!\rangle$ are synonyms.

Now let $(a, b) : r$ be a RA in $\mathcal{P}_+$. Then $\exists X'.\mathcal{A}'$ contains this RA, and for Condition (RS2) we have $\mathsf{Succ}(\mathcal{S}_a, r, b) \leq \mathcal{S}_b$. Since $h(b) = b$, it follows that $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S})$ contains the RA $(\langle\!\langle a, \mathcal{S}_a\rangle\!\rangle, \langle\!\langle b, \mathcal{S}_b\rangle\!\rangle) : r$, which equals $(a, b) : r$. ◻

The above lemmas yield that every canonical unsaturated repair satisfies Definition 4.3 and thus the part "unsaturated repair" of their denotation is justified.

Proposition 4.4. $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S})$ *is an unsaturated repair of* $\exists X.\mathcal{A}$ *for* $\mathcal{P}$ *w.r.t.* $\mathcal{T}$.

The canonical unsaturated repairs are complete in the sense that they cover all unsaturated repairs, similarly to the saturated repairs.

Proposition 4.5. *Each unsat. repair is entailed by a canonical one.*

Proof. Consider an unsaturated repair $\exists Y.\mathcal{B}$ of $\exists X.\mathcal{A}$ for $\mathcal{P}$ w.r.t. $\mathcal{T}$. Then, according to Condition (UR1), $\exists Y.\mathcal{B}$ is entailed by $\exists X.\mathcal{A} \cup \mathcal{P}_+$ and so there is a homomorphism k from $\exists Y.\mathcal{B}$ to $\exists X'.\mathcal{A}'$. Similarly to the proof of Proposition 8 in [6], we define a mapping $\mathcal{F}$ from objects of $\exists Y.\mathcal{B}$ to subsets of $\mathsf{Atoms}(\mathcal{T}, \mathcal{P})$ by

$$\mathcal{F}(u) \coloneqq \mathsf{Max}\{\, C \mid C \in \mathsf{Atoms}(\mathcal{T}, \mathcal{P}),\ \mathcal{A}^{\mathcal{T}} \models h(k(u)) : C, \text{ and } \mathcal{B} \not\models^{\mathcal{T}} u : C \,\}.$$

Recall that $\exists X^{\mathcal{T}}.\mathcal{A}^{\mathcal{T}}$ is the IQ-saturation of $\exists X.\mathcal{A}$ w.r.t. $\mathcal{T}$.

We now verify that each set $\mathcal{F}(u)$ is a repair type for $h(k(u))$.

(RT1) By the very definition of $\mathcal{F}$ the object $h(k(u))$ is an instance of each atom in $\mathcal{F}(u)$.

(RT2) For the Max operator each two atoms in $\mathcal{F}(u)$ are subsumption-incomparable.

(RT3) Let $C \in \mathcal{F}(u)$ and $E \sqsubseteq F \in \mathcal{T}$ with $\mathcal{A}^{\mathcal{T}} \models h(k(u)) : E$ and $F \sqsubseteq^{\mathcal{T}} C$. By definition of $\mathcal{F}$ we infer from $C \in \mathcal{F}(u)$ that $\mathcal{B} \not\models^{\mathcal{T}} u : C$. With $F \sqsubseteq^{\mathcal{T}} C$ and $E \sqsubseteq F \in \mathcal{T}$ it further follows that $\mathcal{B} \not\models^{\mathcal{T}} u : E$. So there is a top-level conjunct $E' \in \mathsf{Conj}(E)$ such that $\mathcal{B} \not\models^{\mathcal{T}} u : E'$. The assumption $\mathcal{A}^{\mathcal{T}} \models h(k(u)) : E$ implies that also $\mathcal{A}^{\mathcal{T}} \models h(k(u)) : E'$. We conclude that $\mathcal{F}(u)$ contains either E' itself or an atom subsuming E', and thus E is subsumed by an atom in $\mathcal{F}(u)$.

Restricting the mapping $\mathcal{F}$ to all INs yields a repair seed $\mathcal{S}^{\mathcal{F}}$, i.e. where $\mathcal{S}^{\mathcal{F}}_a \coloneqq \mathcal{F}(a)$ for each IN a, since the following conditions are satisfied.

(RS1) Let $+\, a : C \in \mathcal{P}$. Then $\exists Y.\mathcal{B} \models^{\mathcal{T}} a : C$, and so $\exists Y.\mathcal{B} \models^{\mathcal{T}} a : D$ for each atom D with $C \sqsubseteq^{\mathcal{T}} D$. It follows that $\mathcal{F}(a)$ does not contain any atom D with $C \sqsubseteq^{\mathcal{T}} D$.

(RS2) Further let $+\, (a, b) : r \in \mathcal{P}$. Then $\exists Y.\mathcal{B}$ entails $(a, b) : r$ and thus also $a : \exists r.\{b\}$. It follows that $\mathcal{F}(a)$ does not contain $\exists r.\{b\}$.
Moreover, consider $\exists r.C \in \mathcal{F}(a)$ with $\exists X.\mathcal{A} \models^{\mathcal{T}} b : C$. Then $\exists Y.\mathcal{B} \not\models^{\mathcal{T}} a : \exists r.C$. Since $\exists Y.\mathcal{B}$ entails $(a, b) : r$, we infer that $\exists Y.\mathcal{B} \not\models^{\mathcal{T}} b : C$. With $h(k(b)) = b$ it follows that $\mathcal{F}(b)$ contains an atom D with $C \sqsubseteq^{\emptyset} D$.

(RS3) Now consider $-\, a : C \in \mathcal{P}$ with $\exists X.\mathcal{A} \models^{\mathcal{T}} a : C$. By assumption, $\exists Y.\mathcal{B} \not\models^{\mathcal{T}} a : C$. So there must be a top-level conjunct $C' \in \mathsf{Conj}(C)$ with $\exists Y.\mathcal{B} \not\models^{\mathcal{T}} a : C'$ but $\exists X.\mathcal{A} \models^{\mathcal{T}} a : C'$. Further recall that $h(k(a)) = a$. It follows that $\mathcal{F}(a)$ contains C' itself or an atom subsuming C', and thus $\mathcal{F}(a)$ contains an atom subsuming C.

(RS4) Let $-\, (a, b) : r \in \mathcal{P}$ with $\exists X.\mathcal{A} \models^{\mathcal{T}} (a, b) : r$. By assumption, $\exists Y.\mathcal{B} \not\models^{\mathcal{T}} (a, b) : r$. Since $h(k(a)) = a$ and $(a, b) : r$ is equivalent to $a : \exists r.\{b\}$, we conclude that $\mathcal{F}(a)$ contains $\exists r.\{b\}$.

(RS5) If the nominal $\{a\}$ occurs in $\mathcal{P}$, then it cannot be in $\mathcal{S}^{\mathcal{F}}_a$ since $a : \{a\}$ is a tautology, which is entailed by $\mathcal{B}$ w.r.t. $\mathcal{T}$.

Last, we will verify that the mapping

$$u \mapsto \langle\!\langle k(u), \mathcal{F}(u) \rangle\!\rangle$$

is a homomorphism from $\exists Y.\mathcal{B}$ to $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S}^{\mathcal{F}})$.

- Consider a CA $u : A \in \mathcal{B}$. Then $k(u) : A \in \mathcal{A}'$ and $A \notin \mathcal{F}(u)$. It follows that $\langle\!\langle k(u), \mathcal{F}(u) \rangle\!\rangle : A$ is contained in the matrix of $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S}^{\mathcal{F}})$.
- Consider a RA $(u, v) : r \in \mathcal{B}$. Then $(k(u), k(v)) : r \in \mathcal{A}'$. To show that $(\langle\!\langle k(u), \mathcal{F}(u) \rangle\!\rangle, \langle\!\langle k(v), \mathcal{F}(v) \rangle\!\rangle) : r$ is contained in the matrix of $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S}^{\mathcal{F}})$, we verify that $\mathsf{Succ}(\mathcal{F}(u), r, h(k(v)))$ is covered by $\mathcal{F}(v)$. To this end, let $\exists r.C \in \mathcal{F}(u)$ with $\mathcal{A}^{\mathcal{T}} \models h(k(v)) : C$. Then $\mathcal{B} \not\models^{\mathcal{T}} u : \exists r.C$ and thus $\mathcal{B} \not\models^{\mathcal{T}} v : C$. It follows that C is subsumed by some atom in $\mathcal{F}(v)$.

We conclude that $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S}^{\mathcal{F}})$ entails $\exists Y.\mathcal{B}$. ◻

As next step, we determine the computational complexity of query answering w.r.t. an unsaturated repair.

Theorem 4.6. *W.r.t. the size of the input KB and repair request only, deciding entailment of queries in* IRQ *by canonical unsaturated repairs is in* P, *but is* NP*-complete if the TBox is taken into account in the entailment. Hardness already holds without an addition part and even if the TBox is cycle-restricted.*

Proof. The first statement for CAs is a corollary to Lemma 4.C. Moreover, deciding if a qABox entails a RA is trivial since one only needs to look if the RA is contained in the matrix, even when the TBox is taken into account. Thus, to determine if a RA is entailed, one only needs to check Instruction (CUR2). Regarding the second statement, we show that satisfiability of a 3-CNF can be reduced to entailment of a CA by an unsaturated repair and a TBox in polynomial time. NP-hardness is then inherited from the former decision problem.

Let α be a propositional formula in 3-CNF, i.e. α is a conjunction of disjunctions over clauses that consist of three literals each, say $\alpha = \beta_1 \land \cdots \land \beta_n$ where $\beta_i = \lambda_{i,1} \lor \lambda_{i,2} \lor \lambda_{i,3}$ for each index i and each $\lambda_{i,j}$ is either a propositional variable or a negation of a propositional variable. We define the TBox $\mathcal{T}_\alpha$ consisting of the following CIs:

- $T_{\beta_1} \sqcap \cdots \sqcap T_{\beta_n} \sqsubseteq T_\alpha$
- $T_{\lambda_{i,j}} \sqsubseteq T_{\beta_i}$ for all i, j
- $F_p \sqsubseteq T_{\neg p}$ for each propositional variable p occurring in α
- $F_p \sqcap T_p \sqsubseteq E$ for each propositional variable p occurring in α.

We further take the qABox $\exists\{x\}.\{\, (a, x) : r,\ x : T_p,\ x : F_p \mid p \in \mathsf{Var}(\alpha) \,\}$, the repair request $\{a : \exists r.E\}$ without addition part, and the query $a : \exists r.T_\alpha$. For this input, there is only the repair seed $\mathcal{S}$ where $\mathcal{S}_a \coloneqq \{\exists r.E\}$.[10] We will show that the induced unsaturated repair entails the query w.r.t. the TBox iff. the formula α is satisfiable.

First let α be satisfiable, certified by the variable assignment $g\colon \mathsf{Var}(\alpha) \to \{T, F\}$ under which α evaluates to T. By flipping the values we obtain a repair type for x, namely $\{\, T_p \mid g(p) = F \,\} \cup \{\, F_p \mid g(p) = T \,\} \cup \{E\}$. In the unsaturated repair the copy of x annotated with this repair type is an instance of T_p iff. $g(p) = T$ and of F_p iff. $g(p) = F$, and this copy is an r-successor of a since the repair type contains E. Inference with the TBox yields that this copy is an instance of T_α, and thus the query is entailed.

It remains to prove the only-if direction. To this end, assume that the unsaturated repair induced by $\mathcal{S}$ entails $a : \exists r.T_\alpha$ w.r.t. $\mathcal{T}$ and is denoted by $\exists Y.\mathcal{B}$. Since the TBox does not introduce new r-successors, the matrix $\mathcal{B}$ must contain a RA $(a, y) : r$ for which y is an instance of T_α w.r.t. $\mathcal{T}$. The repair type that annotates y must contain E since $\mathcal{S}_a$ contains $\exists r.E$, and thus this repair type must contain F_p or T_p for each variable $p \in \mathsf{Var}(\alpha)$. It follows that $\mathcal{B}$ does not contain both $y : F_p$ and $y : T_p$ for any variable p. We define a variable assignment $g\colon \mathsf{Var}(\alpha) \to \{F, T\}$ by $g(p) \coloneqq F$ if $\mathcal{B}$ contains $y : F_p$, and otherwise by $g(p) \coloneqq T$. We show that α evaluates to T under g, which shows that α is satisfiable.

Recall that α is a conjunction of clauses $\beta_i = \lambda_{i,1} \lor \lambda_{i,2} \lor \lambda_{i,3}$. By the very definition of the TBox $\mathcal{T}_\alpha$, and since $\mathcal{B} \models^{\mathcal{T}} y : T_\alpha$, at

[10] This is also the reason why the RN r is used, namely to ensure that there is only one repair seed. Had we instead assigned the truth values to the IN a, and used the repair request $\{a : E\}$ and the query $a : T_\alpha$, then there would have been multiple repair seeds.

least one of the assertions $y : T_{\lambda_{i,j}}$ with $j \in \{1, 2, 3\}$ must be entailed by $\mathcal{B}$ for each clause β_i.

- If the literal $\lambda_{i,j}$ is a variable p, then $\mathcal{B} \models^{\mathcal{T}} y : T_p$ requires that the matrix $\mathcal{B}$ already contains this assertion $y : T_p$. Since $\mathcal{B} \not\models^{\mathcal{T}} y : E$, it cannot contain the assertion $y : F_p$ as well, and thus $g(p) = T$.
- Otherwise, the literal $\lambda_{i,j}$ is a negated variable $\neg p$. Since the matrix $\mathcal{B}$ can only contain CAs involving the CNs F_q and T_q for variables $q \in \mathsf{Var}(\alpha)$, the consequence $y : T_{\neg p}$ can only be produced by means of the CI $F_p \sqsubseteq T_{\neg p}$ and therefore the matrix $\mathcal{B}$ must contain the assertion $y : F_p$. It follows that $g(p) = F$, and so $\neg p$ evaluates to T under g.

It now easily follows that each clause β_i evaluates to T under g, and thus the whole formula α as well.

Last, we show that IRQ-query answering is in NP. According to Theorem 2 in [7], the given query is entailed iff. there is a homomorphism from the query (seen as qABox) to the saturation of the unsaturated repair. Since the unsaturated repair has exponential size in the worst case, it should not be fully computed. Instead, such a homomorphism has polynomial size since it sends the polynomially many objects in the qABox representation of the query to the objects in the saturation, which are either of the form $\langle\!\langle u, \Phi \rangle\!\rangle$ (in the unsaturated repair) or of the form x_C for a CD C that occurs in the TBox $\mathcal{T}$ (added by saturation), and thus these objects need polynomial size only. It thus suffices to guess a polynomial-size mapping from the objects in the qABox representation of the query to such objects, and then check whether it is a homomorphism, which would certify the entailment. For the latter, note that to check if an assertion $(\langle\!\langle u, \Phi \rangle\!\rangle, x_C) : r$ is present boils down to checking if the saturation rule can be applied at $\langle\!\langle u, \Phi \rangle\!\rangle$ in order to generate this successor x_C. To this end, it is enough to guess polynomially many assertions in the neighborhood of $\langle\!\langle u, \Phi \rangle\!\rangle$ since the TBox consists of polynomially many CIs, each of which has a polynomial-size premise. Checking for presence of the other assertions can be done in polynomial time in the obvious way. ◻

Like with the saturated repairs, the covers relation $\leq_{\mathsf{IRQ}}$ is connected to the entailment relation $\models_{\mathsf{IRQ}}$.

Lemma 4.E. *If $\mathcal{S} \leq_{\mathsf{IRQ}} \mathcal{S}'$, then* $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S}) \models_{\mathsf{IRQ}} \mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S}')$.

Proof. The proof of Lemma 2.D works here similarly. ◻

4.2 Computing Disputable Consequences from Unsaturated Repairs

As last step we show how the disputable consequences can be efficiently computed with one particular unsaturated repair.

Lemma 4.7. *The canonical unsaturated repair induced by $\mathcal{S}^1$ is the only* IRQ*-optimal unsaturated repair for $\mathcal{P}^1$.*

Proof. The proof is similar to Proposition 3.2. Consider an unsaturated repair $\exists Y.\mathcal{B}$ for $\mathcal{P}^1$. According to Proposition 4.5 it is entailed by the canonical unsaturated repair induced by some repair seed $\mathcal{S}^{\mathcal{F}}$. In order to show that $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S}^1)$ IRQ-entails $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S}^{\mathcal{F}})$ and thus also $\exists Y.\mathcal{B}$, we use Lemma 4.E and will verify the following two claims:

(1) $\mathcal{S}^1_a \leq \mathcal{S}^{\mathcal{F}}_a$ for each IN a.
(2) $\mathsf{Succ}(\mathcal{S}^{\mathcal{F}}_a, r, b) \leq \mathcal{S}^{\mathcal{F}}_b$ implies $\mathsf{Succ}(\mathcal{S}^1_a, r, b) \leq \mathcal{S}^1_b$ for each RA $(a, b) : r$ in $\exists X.\mathcal{A}$.

We start with the first claim. To this end, consider an atom $C \in \mathcal{S}^1_a$, i.e. the assertion $a : C$ is in $\mathcal{P}^1_-$. Since $\exists Y.\mathcal{B}$ is a repair for $\mathcal{P}^1$, it does not entail this assertion w.r.t. $\mathcal{T}$. Furthermore, this assertion $a : C$ is entailed by the input qABox $\exists X.\mathcal{A}$ w.r.t. $\mathcal{T}$, since otherwise it would not have been considered as a query by the smart interaction strategy and could then not be in $\mathcal{P}^1_-$. According to the definition of $\mathcal{F}$ in the proof of Proposition 4.5, $\mathcal{F}(a)$ contains C itself or an atom subsuming C. Since this holds for all C and since $\mathcal{F}(a) = \mathcal{S}^{\mathcal{F}}_a$, we conclude that $\mathcal{S}^1_a \leq \mathcal{S}^{\mathcal{F}}_a$ for every IN a.

It remains to verify the second claim. Assume that $(a, b) : r$ is a RA in $\exists X.\mathcal{A}$ with $\mathsf{Succ}(\mathcal{S}^{\mathcal{F}}_a, r, b) \leq \mathcal{S}^{\mathcal{F}}_b$. According to Definition 4.3, $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S}^{\mathcal{F}})$ contains the RA $(a, b) : r$ (recall that $\langle\!\langle a, \mathcal{S}^{\mathcal{F}}_a \rangle\!\rangle$ and a are synonyms, and likewise for b). Since $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S}^{\mathcal{F}})$ is a repair for $\mathcal{P}^1$, this RA is not contained in $\mathcal{P}^1_-$. By Lemma 3.A we infer that $\mathsf{Succ}(\mathcal{S}^1_a, r, b) \leq \mathcal{S}^1_b$. ◻

The above lemma now allows us to formulate a more efficient variant of Condition (DC3).

Proposition 4.8. *For each query $\gamma \in$ IRQ, Condition (DC3) in Corollary 3.5 is equivalent to the following:*

(DC3*) *γ is not entailed w.r.t. $\mathcal{T}$ by the unsat. repair induced by $\mathcal{S}^1$.*

Proof. Among the following statements, the first two are equivalent by Definition 3.4, the next two are equivalent by standard equivalence-preserving transformations of logical formulas, and the last two by Definition 4.1.

- For every substantiation $\mathcal{J}$ of γ, we have that $\mathcal{J}$ does not entail w.r.t. $\mathcal{T}$ all accepted queries in $\mathcal{P}^1_+$, or $\mathcal{J}$ entails w.r.t. $\mathcal{T}$ some rejected query in $\mathcal{P}^1_-$.
- For each qABox $\exists Y.\mathcal{B}$, if $\exists X.\mathcal{A} \cup \mathcal{P}^1_+ \models \exists Y.\mathcal{B}$ and $\exists Y.\mathcal{B} \models^{\mathcal{T}} \gamma$, then $\exists Y.\mathcal{B} \not\models^{\mathcal{T}} \alpha$ for some $\alpha \in \mathcal{P}^1_+$ or $\exists Y.\mathcal{B} \models^{\mathcal{T}} \beta$ for some $\beta \in \mathcal{P}^1_-$.
- For every qABox $\exists Y.\mathcal{B}$, if $\exists X.\mathcal{A} \cup \mathcal{P}^1_+ \models \exists Y.\mathcal{B}$, and $\exists Y.\mathcal{B} \models^{\mathcal{T}} \alpha$ for every $\alpha \in \mathcal{P}^1_+$, and $\exists Y.\mathcal{B} \not\models^{\mathcal{T}} \beta$ for each $\beta \in \mathcal{P}^1_-$, then $\exists Y.\mathcal{B} \not\models^{\mathcal{T}} \gamma$.
- γ is not entailed w.r.t. $\mathcal{T}$ by any unsaturated repair of $\exists X.\mathcal{A}$ for $\mathcal{P}^1$ w.r.t. $\mathcal{T}$.

It remains to show that the last statement is equivalent to (DC3*). Consider a query $\gamma \in$ IRQ and an unsaturated repair $\exists Y.\mathcal{B}$ for $\mathcal{P}^1$ such that $\exists Y.\mathcal{B} \models^{\mathcal{T}} \gamma$. Since $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S}^1)$ is the only IRQ-optimal unsaturated repair for $\mathcal{P}^1$ by Lemma 4.7, $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S}^1) \models_{\mathsf{IRQ}} \exists Y.\mathcal{B}$. By Lemma 2.C we obtain $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S}^1) \models^{\mathcal{T}}_{\mathsf{IRQ}} \exists Y.\mathcal{B}$ and it thus follows that $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S}^1) \models^{\mathcal{T}} \gamma$. The converse direction is trivial. ◻

This Condition (DC3*) is much easier to check than (DC3) as we only need to compute (the optimized variant of) a single unsaturated repair and then determine which queries it entails. Unfortunately, this cannot be done by only looking at the repair seed as for saturated repairs, and we will prove that deciding disputable consequences is co NP-complete. Even though such a complexity result seems to prohibit a practical use, our implementation works surprisingly fast even with large TBoxes such as (the $\mathcal{EL}$ fragment of) SNOMED CT. Since saturated repairs can be computed in practice [6], the same holds for unsaturated repairs as they are constructed similarly but directly from the input qABox. Therefore disputable consequences can be computed in practice as well.

Theorem 4.9. *Deciding disputable consequences at the end of Phase 1 is* co NP*-complete.*

Proof. We use the same input as in the proof of Theorem 4.6, without any modifications. No queries need to be decided by the experts in Phase 1 and we obtain $\mathcal{P}^1_+ = \emptyset$ and $\mathcal{P}^1_- = \{a : \exists r.E\}$, which is the given repair request. Further recall that there is only one repair seed, namely with $\mathcal{S}_a = \{\exists r.E\}$, and that the formula α is satisfiable iff. the induced unsat. repair entails the query $a : \exists r.T_\alpha$.

Now, since the query $a : \exists r.T_\alpha$ is entailed by the input qABox and TBox and since it has not been decided by the experts nor by inheritance within Phase 1, we conclude by Proposition 4.8 that the formula α is satisfiable iff. the query $a : \exists r.T_\alpha$ is not disputable. It follows that recognizing disputable consequences is co NP-hard.

It remains to prove containment in co NP. Conditions (DC1) and (DC2) can be checked in polynomial time since, more generally, entailment of queries in IRQ by ABoxes and qABoxes has this complexity. Furthermore, Conditions (DC3) and (DC3*) are equivalent by Proposition 4.8 and the negation of the latter can be decided in non-deterministic polynomial time, see Theorem 4.6. □

Surprisingly, it suffices to restrict attention to disputable CAs.

Proposition 4.10. *There are no disputable RAs at the end of Phase 1.*

Proof. Let $\gamma = (a, b) : r$ be a RA that satisfies Conditions (DC1) and (DC2). We will show that it cannot satisfy Condition (DC3*) and thus is no disputable consequence. We first observe that:

γ does not fulfill (DC3*)
iff. $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S}^1) \models^{\mathcal{T}} \gamma$
iff. the matrix of $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S}^1)$ contains γ

By (DC1) the input KB entails γ and thus its matrix $\mathcal{A}$ contains γ. We conclude that:

the matrix of $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S}^1)$ contains γ
iff. $\mathsf{Succ}(\mathcal{S}^1_a, r, b) \leq \mathcal{S}^1_b$
iff. for every $-a : \exists r.C \in \mathcal{P}^1$ with $\exists X.\mathcal{A} \models^{\mathcal{T}} b : C$, there is $-b : D \in \mathcal{P}^1$ with $C \sqsubseteq^{\emptyset} D$.

The last statement is satisfied. To see this, assume that $a : \exists r.C$ was a rejected query with $\exists X.\mathcal{A} \models^{\mathcal{T}} b : C$. Then Instruction (SIS1) would have added the two queries γ and $b : C$ to $\mathcal{Q}$, but this is a contradiction since, on the one hand, Phase 1 does not end until all queries in $\mathcal{Q}$ have been decided and, on the other hand, γ is undecided at the end of Phase 1 by (DC2) (i.e. it could not have been added to $\mathcal{Q}$). Thus, there is no rejected query $a : \exists r.C$ with $\exists X.\mathcal{A} \models^{\mathcal{T}} b : C$, and so the last statement in the above chain of equivalences is vacuously satisfied. We conclude that γ does not fulfill (DC3*). □

The disputable consequences are exactly the CAs in the "entailment difference" between the saturated and the unsaturated repair induced by $\mathcal{S}^1$, i.e. those CAs entailed by the saturated repair but not by the unsaturated repair. Our implementation is based on this.

Lemma 4.11. *γ is a disputable consequence at the end of Phase 1 iff.*
(DC4) $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S}^1) \models^{\mathcal{T}} \gamma$ *and*
(DC3*) $\mathsf{rep}^{\mathcal{T}}_{\mathsf{unsat}}(\exists X.\mathcal{A}, \mathcal{S}^1) \not\models^{\mathcal{T}} \gamma$.

Proof. Regarding the only-if direction, let $a : C$ be disputable. Proposition 4.8 yields (DC3*), and we now show that (DC1) and (DC2) yield (DC4). We have $\exists X.\mathcal{A} \models^{\mathcal{T}} a : C$ by (DC1). Since according to (DC2) $\mathcal{P}^1_+ \cup \{a : C\}$ does not entail w.r.t. $\mathcal{T}$ any query in $\mathcal{P}^1_-$, there is no $a : D \in \mathcal{P}^1_-$ with $C \sqsubseteq^{\mathcal{T}} D$. It follows that there is no $D \in \mathcal{S}^1_a$ with $C \sqsubseteq^{\mathcal{T}} D$, and thus $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S}^1) \models^{\mathcal{T}} a : C$ by Lemma XII in the erratum to [4].

In the converse direction, (DC3*) is already there. By (DC4) and since $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S}^1)$ is entailed by $\exists X.\mathcal{A}$, we infer that $\exists X.\mathcal{A} \models^{\mathcal{T}} a : C$, which is (DC1). By Lemma 3.1, $\mathcal{S}^1$ is a repair seed for $\mathcal{P}^1$ and so it induces a repair for $\mathcal{P}^1$. Thus $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S}^1)$ entails $\mathcal{P}^1_+$ w.r.t. $\mathcal{T}$, and also $a : C$ by (DC4). We conclude that $\mathcal{P}^1_+ \cup \{a : C\}$ does not entail any query in $\mathcal{P}^1_-$ since neither does $\mathsf{rep}^{\mathcal{T}}_{\mathsf{IRQ}}(\exists X.\mathcal{A}, \mathcal{S}^1)$. Furthermore, (DC3*) implies that $\mathcal{P}^1_+ \not\models^{\mathcal{T}} a : C$. So (DC2) is fulfilled as well. □

A Small Example

For instance, the ABox $\{\mathsf{mike} : \exists \mathsf{drives}.\mathsf{Porsche}\}$ expresses that Mike drives a Porsche. The TBox $\{\mathsf{Porsche} \sqsubseteq \mathsf{SportsCar}, \mathsf{SportsCar} \equiv \mathsf{Car} \sqcap \mathsf{Fast} \sqcap \mathsf{Loud}\}$ states that every Porsche is a sports car, and that every sports car is a fast and loud car and vice versa. When asked how fast his car can go, Mike answers that he actually does not drive a fast car. Thus, the ontology should be repaired for the faulty consequence $\mathsf{mike} : \exists \mathsf{drives}.(\mathsf{Car} \sqcap \mathsf{Fast})$. With the classical repair method we could only delete the single statement from the ABox, and no knowledge is preserved at all. Our interactive method would instead identify an ideal repair as follows. First, the saturation is determined: $\exists \{x\}.\{(\mathsf{mike}, x) : \mathsf{drives},\ x : \mathsf{Porsche},\ x : \mathsf{SportsCar},\ x : \mathsf{Car},\ x : \mathsf{Fast},\ x : \mathsf{Loud}\}$.

Since the consequence $\mathsf{mike} : \exists \mathsf{drives}.(\mathsf{Car} \sqcap \mathsf{Fast})$ is rejected, the variable x representing Mike's Porsche is split into two copies: the first copy is fast and loud but no car anymore, and the second copy is a loud car but not fast. Both copies are not unrealistic since he might drive a sports bike instead, and his car might only be loud for a manipulated exhaust system and not for a powerful engine. To find out which copy indeed exists and should occur in the repair, the two subsequent queries $\mathsf{mike} : \exists \mathsf{drives}.\mathsf{Fast}$ and $\mathsf{mike} : \exists \mathsf{drives}.\mathsf{Car}$ are generated. The first copy is eliminated iff. the first query is rejected, and similarly for the second copy. Obviously, such a copy must not be a Porsche nor a sports car anymore as both would restore the removed information.

With the additional CI $\exists \mathsf{drives}.\mathsf{SportsCar} \sqsubseteq \mathsf{CoolGuy}$ in the TBox the consequence $\mathsf{mike} : \mathsf{CoolGuy}$ is disputable. It could be that Mike is a cool guy for another reason and then we would like to keep this information. Assuming that only the second query above was accepted, we obtain the repair $\exists \{y\}.\{(\mathsf{mike}, y) : \mathsf{drives},\ x : \mathsf{Car},\ x : \mathsf{Loud},\ \mathsf{mike} : \mathsf{CoolGuy}\}$, or equivalently $\{\mathsf{mike} : \exists \mathsf{drives}.(\mathsf{Car} \sqcap \mathsf{Loud}),\ \mathsf{mike} : \mathsf{CoolGuy}\}$. Compared to the classical repair much more consequences are preserved.

The Protégé Plugin

An implementation of the underlying repair construction as well as the smart interaction strategy is available.[L10] It comes in form of a plug-in for the KB editor Protégé.[L11] As programming language we chose Scala.[L12] It can thus be used on any operating system for which a Java virtual machine (JVM) is available, which includes the major systems Unix, Linux, MacOS, and Windows. It is recommended to use a modern JVM like GraalVM[L13] since it offers faster code execution (often approximately twice as fast as a standard JVM). Furthermore, the implementation employs the currently fastest $\mathcal{EL}$ reasoner: ELK.[L9]

On MacOS, the easiest way to try out the implementation is to use the installer script.[L14] For other operating systems this script can be easily adapted. More details are explained on the start page of the GitHub repository.[L10] Once installed, the plug-in can be activated in Protégé from the menu: Window ⟶ Tabs ⟶ Interactive Optimal Repair. A new tab is then opened. After loading an $\mathcal{EL}$ KB, just switch to this tab and then click the "Start" button.

When started, the plug-in first determines if the loaded KB is within $\mathcal{EL}$ and thus supported. It then initializes the reasoner. After clicking the "Next" button the repair request containing the unwanted consequences to be repaired for as well as the wanted consequences to be retained must be specified. After another click on the "Next" button the smart interaction strategy starts to run. At any time point, the shown list contains all currently undecided questions. For each of them, the expert can accept or reject them, depending on the validity in the domain of interest, by clicking the respective button. As soon as there are no undecided queries anymore, the "Next" button becomes available again. Clicking it allows to specify for which query language the repair is to be computed (IRQ or CQ). A last click on the "Compute" button triggers the actual computation of the identified repair, by which the loaded KB is finally overwritten.

Even though Theorem 4.9 seems to indicate that computing the disputable consequences for Phase 2 is intractable, the implementation works sufficiently fast even with large TBoxes such as (the $\mathcal{EL}$ fragment of) SNOMED CT,[11] which contains more than 360,000 concept names. Interactively identifying a repair of an ABox representing data on a patient having a common cold and then computing this repair completes within about four to five minutes. More specifically, in this experiment we used an ABox containing a single CA stating that a particular person has a common cold; the user interaction then amounted to about 30 questions only (as the interaction process is local to the statements to be repaired for). This computation time can surely be further improved with optimizations or faster programming languages such as C++.

5 Summary and Future Prospects

We have delved into the topic of identifying and computing a practically relevant repair of a given knowledge base consisting of a quantified ABox and a static $\mathcal{EL}$ TBox, where the consequences repaired for are concept and role assertions. To this end, we introduced the *smart interaction strategy* to the optimal-repair framework, with which experts can interactively determine a suitable repair in polynomial time. Each such repair can be constructed by means of three operations (copying, deleting, and saturation) and the strategy allows us to efficiently control them. Moreover, we considered *disputable consequences*. Since the optimal repairs retain all disputable consequences but none of their substantiations within the input knowledge base, it might not be desirable to keep each of them but these decisions need to be made by the experts. An implementation in form of a plug-in for the knowledge-base editor Protégé is provided.[L10]

As future work, we want to extend the strategy to optimal repairs in more expressive DLs, e.g. Horn-$\mathcal{ALCROI}$ [9]. In order to increase support for wanted consequences in the repair request, we also want to combine it with interaction strategies for existing or novel abduction methods.

[11]To obtain the $\mathcal{EL}$ fragment of SNOMED CT one merely needs to delete a few role inclusions.

Acknowledgements

This work was supported by Deutsche Forschungsgemeinschaft (DFG) in Projects 430150274 (Repairing Description Logic Ontologies) and 389792660 (TRR 248: Foundations of Perspicuous Software Systems), and has further been supported by the Saxon State Ministry for Science, Culture, and Tourism (SMWK) by funding the Center for Scalable Data Analytics and Artificial Intelligence (ScaDS.AI).

References

[1] Franz Baader, Sebastian Brandt, and Carsten Lutz. 2005. Pushing the $\mathcal{EL}$ envelope. In *Proceedings of the 19th International Joint Conference on Artificial Intelligence (IJCAI)*. Professional Book Center, 364–369. http://ijcai.org/Proceedings/05/Papers/0372.pdf.

[2] Franz Baader, Sebastian Brandt, and Carsten Lutz. 2008. Pushing the $\mathcal{EL}$ envelope further. In *Proceedings of the 4th OWLED Workshop on OWL: Experiences and Directions* (CEUR Workshop Proceedings). Vol. 496. CEUR-WS.org. http://ceur-ws.org/Vol-496/owled2008dc%5C_paper%5C_3.pdf.

[3] Franz Baader, Ian Horrocks, Carsten Lutz, and Ulrike Sattler. 2017. *An Introduction to Description Logic*. Cambridge University Press. DOI: 10.1017/9781139025355.

[4] Franz Baader, Patrick Koopmann, and Francesco Kriegel. 2023. Optimal repairs in the description logic $\mathcal{EL}$ revisited. In *Proceedings of the 18th European Conference on Logics in Artificial Intelligence (JELIA)* (Lecture Notes in Computer Science). Vol. 14281. Springer, 11–34. DOI: 10.1007/978-3-031-43619-2_2. See the erratum https://doi.org/10.5281/zenodo.10276270.

[5] Franz Baader, Patrick Koopmann, and Francesco Kriegel. 2023. Optimal Repairs in the Description Logic $\mathcal{EL}$ Revisited (Extended Version). LTCS-Report 23-03. Chair of Automata Theory, Institute of Theoretical Computer Science, Technische Universität Dresden, Dresden, Germany. DOI: 10.25368/2023.121.

[6] Franz Baader, Patrick Koopmann, Francesco Kriegel, and Adrian Nuradiansyah. 2021. Computing optimal repairs of quantified ABoxes w.r.t. static $\mathcal{EL}$ TBoxes. In *Proceedings of the 28th International Conference on Automated Deduction (CADE)* (Lecture Notes in Computer Science). Vol. 12699. Springer, 309–326. DOI: 10.1007/978-3-030-79876-5_18.

[7] Franz Baader, Patrick Koopmann, Francesco Kriegel, and Adrian Nuradiansyah. 2021. Computing Optimal Repairs of Quantified ABoxes w.r.t. Static $\mathcal{EL}$ TBoxes (Extended Version). LTCS-Report 21-01. Chair of Automata Theory, Institute of Theoretical Computer Science, Technische Universität Dresden, Dresden, Germany. DOI: 10.25368/2022.64.

[8] Franz Baader, Patrick Koopmann, Francesco Kriegel, and Adrian Nuradiansyah. 2022. Optimal ABox repair w.r.t. static $\mathcal{EL}$ TBoxes: from quantified ABoxes back to ABoxes. In *Proceedings of the 19th Extended Semantic Web Conference (ESWC)* (Lecture Notes in Computer Science). Vol. 13261. Springer, 130–146. DOI: 10.1007/978-3-031-06981-9_8.

[9] Franz Baader and Francesco Kriegel. 2022. Pushing optimal ABox repair from $\mathcal{EL}$ towards more expressive Horn-DLs. In *Proceedings of the 19th International Conference on Principles of Knowledge Representation and Reasoning (KR)*, 22–32. DOI: 10.24963/kr.2022/3. See the addendum https://doi.org/10.5281/zenodo.8060198.

[10] Franz Baader, Francesco Kriegel, and Adrian Nuradiansyah. 2022. Error-tolerant reasoning in the description logic $\mathcal{EL}$ based on optimal repairs. In *Proceedings of the 6th International Joint Conference on Rules and Reasoning (RuleML+RR)* (Lecture Notes in Computer Science). Vol. 13752. Springer, 227–243. DOI: 10.1007/978-3-031-21541-4_15.

[11] Franz Baader, Francesco Kriegel, and Adrian Nuradiansyah. 2024. Inconsistency- and error-tolerant reasoning w.r.t. optimal repairs of $\mathcal{EL}^\perp$ ontologies. In *Proceedings of the 13th International Symposium on Foundations of Information and Knowledge Systems (FoIKS)* (Lecture Notes in Computer Science). Vol. 14589. Springer, 3–22. DOI: 10.1007/978-3-031-56940-1_1.

[12] Franz Baader, Francesco Kriegel, and Adrian Nuradiansyah. 2023. Treating role assertions as first-class citizens in repair and error-tolerant reasoning. In *Proceedings of the 38th ACM/SIGAPP Symposium on Applied Computing (SAC)*. Association for Computing Machinery, 974–982. DOI: 10.1145/3555776.3577630.

[13] Franz Baader, Francesco Kriegel, Adrian Nuradiansyah, and Rafael Peñaloza. 2020. Computing compliant anonymisations of quantified ABoxes w.r.t. $\mathcal{EL}$ policies. In *Proceedings of the 19th International Semantic Web Conference (ISWC)* (Lecture Notes in Computer Science). Vol. 12506. Springer, 3–20. DOI: 10.1007/978-3-030-62419-4_1.

[14] Franz Baader, Francesco Kriegel, Adrian Nuradiansyah, and Rafael Peñaloza. 2020. Computing Compliant Anonymisations of Quantified ABoxes w.r.t. $\mathcal{EL}$ Policies (Extended Version). LTCS-Report 20-08. Chair of Automata Theory, Institute of Theoretical Computer Science, Technische Universität Dresden, Dresden, Germany. DOI: 10.25368/2022.263.

[15] Franz Baader, Francesco Kriegel, Adrian Nuradiansyah, and Rafael Peñaloza. 2018. Making repairs in description logics more gentle. In *Proceedings of the 16th International Conference on Principles of Knowledge Representation and Reasoning (KR)*. AAAI Press, 319–328. https://aaai.org/ocs/index.php/KR/KR18/paper/view/18056.

[16] E. Di Sciascio, F.M. Donini, M. Mongiello, and G. Piscitelli. 2002. Meeting in the agora: a description logic approach to peer-to-peer e-commerce. In *Proceedings of the 24th International Conference on Information Technology Interfaces (ITI)*, 319–324 vol.1. DOI: 10.1109/ITI.2002.1024693.

[17] Fajar Haifani, Patrick Koopmann, Sophie Tourret, and Christoph Weidenbach. 2022. Connection-minimal abduction in $\mathcal{EL}$ via translation to FOL. In *Proceedings of the 11th International Joint Conference on Automated Reasoning (IJCAR)* (Lecture Notes in Computer Science). Vol. 13385. Springer, 188–207. DOI: 10.1007/978-3-031-10769-6_12.

[18] Pascal Hitzler, Markus Krötzsch, and Sebastian Rudolph. 2010. *Foundations of Semantic Web Technologies*. Chapman and Hall/CRC Press. http://www.semantic-web-book.org/.

[19] Yevgeny Kazakov, Markus Krötzsch, and František Simančík. 2012. Practical reasoning with nominals in the $\mathcal{EL}$ family of description logics. In *Proceedings of the 13th International Conference on Principles of Knowledge Representation and Reasoning (KR)*. AAAI Press. http://www.aaai.org/ocs/index.php/KR/KR12/paper/view/4540.

[20] Yevgeny Kazakov, Markus Krötzsch, and František Simančík. 2014. The incredible ELK – from polynomial procedures to efficient reasoning with $\mathcal{EL}$ ontologies. *J. Autom. Reason.*, 53, 1, 1–61. DOI: 10.1007/s10817-013-9296-3.

[21] Patrick Koopmann. 2021. Signature-based abduction with fresh individuals and complex concepts for description logics. In *Proceedings of the 30th International Joint Conference on Artificial Intelligence (IJCAI)*. ijcai.org, 1929–1935. DOI: 10.24963/IJCAI.2021/266.

[22] Patrick Koopmann, Warren Del-Pinto, Sophie Tourret, and Renate A. Schmidt. 2020. Signature-based abduction for expressive description logics. In *Proceedings of the 17th International Conference on Principles of Knowledge Representation and Reasoning (KR)*, 592–602. DOI: 10.24963/KR.2020/59.

[23] Francesco Kriegel. 2022. Optimal fixed-premise repairs of $\mathcal{EL}$ TBoxes. In *Proceedings of the 45th German Conference on Artificial Intelligence (KI)* (Lecture Notes in Computer Science). Vol. 13404. Springer, 115–130. DOI: 10.1007/978-3-031-15791-2_11.

[24] Francesco Kriegel. 2023. Optimal fixed-premise repairs of $\mathcal{EL}$ TBoxes (extended abstract). In *Recently Published Research (RPR) track of the 20th International Conference on Principles of Knowledge Representation and Reasoning (KR)*. DOI: 10.5281/zenodo.8341194.

[25] Markus Krötzsch. 2011. Efficient rule-based inferencing for OWL EL. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI)*. IJCAI/AAAI, 2668–2673. DOI: 10.5591/978-1-57735-516-8/IJCAI11-444.

[26] Carsten Lutz and Frank Wolter. 2010. Deciding inseparability and conservative extensions in the description logic $\mathcal{EL}$. *J. Symb. Comput.*, 45, 2, 194–228. DOI: 10.1016/j.jsc.2008.10.007.

[27] Sebastian Rudolph, Markus Krötzsch, and Pascal Hitzler. 2008. All elephants are bigger than all mice. In *Proceedings of the 21st International Workshop on Description Logics (DL)* (CEUR Workshop Proceedings). Vol. 353. CEUR-WS.org. https://ceur-ws.org/Vol-353/RudolphKraetzschHitzler.pdf.

[28] Sebastian Rudolph, Markus Krötzsch, and Pascal Hitzler. 2008. Cheap boolean role constructors for description logics. In *Proceedings of the 11th European Conference on Logics in Artificial Intelligence (JELIA)* (Lecture Notes in Computer Science). Vol. 5293. Springer, 362–374. DOI: 10.1007/978-3-540-87803-2_30.

[29] Jingchuan Shi, Jiaoyan Chen, Hang Dong, Ishita Khan, Lizzie Liang, Qunzhi Zhou, Zhe Wu, and Ian Horrocks. 2023. Subsumption prediction for e-commerce taxonomies. In *Proceedings of the 20th Extended Semantic Web Conference (ESWC)* (Lecture Notes in Computer Science). Vol. 13870. Springer, 244–261. DOI: 10.1007/978-3-031-33455-9_15.

[30] Dongzhuoran Zhou, Baifan Zhou, Zhuoxun Zheng, Egor V. Kostylev, Gong Cheng, Ernesto Jiménez-Ruiz, Ahmet Soylu, and Evgeny Kharlamov. 2022. Enhancing knowledge graph generation with ontology reshaping - Bosch case. In *Satellite Events of the 19th Extended Semantic Web Conference (ESWC)* (Lecture Notes in Computer Science). Vol. 13384. Springer, 299–302. DOI: 10.1007/978-3-031-11609-4_45.

Links

L1 https://www.snomed.org
L2 https://finregont.com
L3 https://spec.edmcouncil.org/fibo
L4 https://github.com/edmcouncil/fibo
L5 https://geneontology.org
L6 https://www.w3.org/standards/semanticweb
L7 https://www.w3.org/TR/owl2-overview
L8 https://www.w3.org/TR/owl2-profiles
L9 https://github.com/liveontologies/elk-reasoner
L10 https://github.com/francesco-kriegel/interactive-optimal-repairs
L11 https://protege.stanford.edu
L12 https://www.scala-lang.org
L13 https://www.graalvm.org
L14 https://raw.githubusercontent.com/francesco-kriegel/interactive-optimal-repairs/main/install-macos.sh