

Towards Even More Irresistible Axiom Weakening*

Roberto Confalonieri¹, Pietro Galliani¹, Oliver Kutz¹, Daniele Porello²,
Guendalina Righetti¹, and Nicolas Troquard¹

¹ Free University of Bozen-Bolzano, Italy,

{firstname.lastname}@unibz.it

² ISTC-CNR

daniele.porello@cnr.it

Abstract. Axiom weakening is a technique that allows for a fine-grained repair of inconsistent ontologies. Its main advantage is that it repairs ontologies by making axioms less restrictive rather than by deleting them, employing the use of refinement operators. In this paper, we build on previously introduced axiom weakening for *ALC*, and make it much more irresistible by extending its definitions to deal with *SRIQ*, the expressive and decidable description logic underlying OWL 2 DL. We extend the definitions of refinement operator to deal with *SRIQ* constructs, in particular with role hierarchies, cardinality constraints and nominals, and illustrate its application. Finally, we discuss the problem of termination of an iterated weakening procedure.

1 Introduction

The traditional approach to repairing inconsistent ontologies amounts to identify problematic axioms and to remove them (e.g., [19, 11, 10, 4]). Whilst this approach is sufficient to guarantee that the obtained ontology is consistent, it tends to lead to information loss as a secondary effect. For instance, let us assume that our ontology contains the following axioms:

$$\text{Polygamist} \sqsubseteq \text{Person} \sqcap \geq 2 \text{ married.Person} \quad (1)$$

$$\text{Polygamist} \sqsubseteq \text{MarriedPerson} \quad (2)$$

$$\text{MarriedPerson} \sqsubseteq \text{Person} \sqcap \leq 1 \text{ married.Person} \sqcap \geq 1 \text{ married.Person} \quad (3)$$

$$\text{Polygamist}(\text{mary}) \quad (4)$$

According to a classical approach, repairing such an ontology can be done by removing any of the four axioms. However, by removing axiom (1) or (2), we would abandon the polygamist essence of the ontology, by removing axiom (3) we would abandon the more classical view about marriage, and removing axiom (4) trivialises the concept *Polygamist*. Ideally one wants to preserve as much

* Copyright © 2020 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

information as possible, and replace these axioms by weaker versions thereof instead of removing them.

Approaches to repair ontologies more gently via *axiom weakening* were proposed in the literature [8, 7, 20, 2]. In [7], concept refinement in \mathcal{EL}^{++} ontologies is introduced in the context of concept invention. A concept refinement operator to generalise \mathcal{EL}^{++} concepts is proposed and its properties are analysed. This line of work was continued in [20] where the authors define an abstract refinement operator for generalising and specialising \mathcal{ALC} concepts, and weakening \mathcal{ALC} axioms. Crucially, they propose a repairing ontology procedure that solves inconsistencies by weakening axioms and not by removing them. In [2], the authors present general theoretical results for axiom weakening in Description Logics (and instantiations of their approach for the case of \mathcal{EL}). In particular they show that the repairing procedure always yields a consistent ontology in, at most, exponential number of iterations. Practical applications are proposed for \mathcal{EL} ontologies.

Refinement operators in Description Logic have been studied with application to Machine Learning [5, 15, 14, 16]. Concept refinement operators [13] come in two flavours. A generalisation operator wrt. an ontology \mathcal{O} is a function γ that associates with a concept C a set $\gamma_{\mathcal{O}}(C)$ of concepts which are ‘super-concepts’ of C . Dually, a specialisation operator wrt. an ontology \mathcal{O} is a function ρ that associates with a concept C a set $\rho_{\mathcal{O}}(C)$ of concepts which are ‘sub-concepts’ of C . The notions of ‘super’, and ‘sub-concept’ are here implicitly defined by the respective functions, rather than by a purely syntactic procedure. Intuitively, a concept D is a generalised super-concept of concept C wrt. to ontology \mathcal{O} if in every model of the ontology the extension of D subsumes the extension of C . So for instance, the concept $\exists has_component.Carbon$ is a generalisation of $LivingBeing$ and $= 2 has_bodypart.Legs$ is a specialisation of $LivingBeing$ (assuming an appropriate background ontology \mathcal{O}).

In [20], the authors showed that refinement operators enjoy a few properties that make them suitable for implementation of axiom weakening. In particular, deciding whether a concept is a refinement of another concept has the same worst-case complexity as deciding concept subsumption in the underlying logic. Refinement operators are then used to weaken axioms, and to repair inconsistent ontologies. Experimentally, it is shown that repairing ontologies via axiom weakening maintains significantly more information than repairing ontologies via axiom deletion, using e.g., measures that evaluate preservation of taxonomic structure. In [18], ontology repairs via concept refinements and axiom weakening is used to merge two mutually inconsistent ontologies.

In this paper, we extend refinement operators and axiom weakening of [20] to deal with $SR\mathcal{OIQ}$, the logical underpinning of W3C OWL2, with the intention to make the approach more widely applicable. We also provide a proof of almost-certain termination of the ontology repairing procedure based on axiom weakening, originally proposed in [20], and extended here to deal with $SR\mathcal{OIQ}$.

2 Preliminaries

From a formal point of view, an ontology is a set of formulas in an appropriate logical language with the purpose of describing a particular domain of interest. We briefly introduce *SR_{OIQ}*; for full details see [1]. The syntax of *SR_{OIQ}* is based on three disjoint sets N_C, N_R, N_I of *concept names*, *role names*, and *individual names*, respectively. The set of *SR_{OIQ}* *roles* and *concepts* is generated by the grammar

$$\begin{aligned} R, S &::= U \mid E \mid r \mid r^- , \\ C &::= \perp \mid \top \mid A \mid \neg C \mid C \sqcap C \mid C \sqcup C \mid \forall R.C \mid \exists R.C \mid \\ &\geq n S.C \mid \leq n S.C \mid \exists S.Self \mid \{i\} , \end{aligned}$$

where U and E are the universal role and empty role respectively, $r \in N_R$, S is *simple* (see further) in the *RBox* \mathcal{R} , $A \in N_C$, n is a non-negative integer, and $i \in N_I$. In the following, $\mathcal{L}(N_C, N_R, N_I)$ and $\mathcal{L}_r(N_R)$, denote respectively the set of concepts and roles that can be built over N_C, N_R, N_I in *SR_{OIQ}*. We denote by $\text{nnf}(C)$ the negation normal form of the concept C .

The size of a concept in $\mathcal{L}(N_C, N_R, N_I)$ will be useful in the proofs. We define it precisely next.

Definition 1. *The size $|C|$ of a concept C is inductively defined as follows. For $C \in N_C \cup \{\top, \perp\}$, $|C| = 1$. For $i \in N_I$, $|\{i\}| = 1$. For $R \in N_R$, $|\exists R.Self| = 2$. Then for $R \in N_R$ and an arbitrary C , $|\neg C| = 1 + |C|$; $|C \sqcap D| = |C \sqcup D| = 1 + |C| + |D|$; $|\exists R.C| = |\forall R.C| = 1 + |C|$, and for a non-negative integer n , $|\geq n R.C| = |\leq n R.C| = \log(n) + 1 + |C|$.*

A *TBox* \mathcal{T} is a finite set of concept inclusions (GCIs) of the form $C \sqsubseteq D$ where C and D are concepts. It is used to store terminological knowledge regarding the relationships between concepts. An *ABox* \mathcal{A} is a finite set of formulas of the form $C(a)$, $R(a, b)$, $\neg R(a, b)$, $a = b$, and $a \neq b$, which express knowledge about objects in the knowledge domain. An *RBox* \mathcal{R} is a finite set of role inclusions (RIA) $R \sqsubseteq S$, complex role inclusions $R_1 \circ R_2 \sqsubseteq S$, and *disjoint*(R, S) (R and S *simple*, see next), where R, R_1, R_2 , and S are roles. A role $R \in N_R$ is *simple* in \mathcal{R} if it is not implied by any composition of roles; it is non-simple otherwise. An inverse role R^- is simple if R is simple. We take U and E as simple. (Transitive, reflexive, irreflexive, symmetric, and asymmetric roles can be defined through appropriate *TBox* or *RBox* axioms.) A *SR_{OIQ}* ontology $\mathcal{O} = \mathcal{T} \cup \mathcal{R} \cup \mathcal{A}$ is defined by a *TBox* \mathcal{T} , an *RBox* \mathcal{R} , and an *ABox* \mathcal{A} , where the \mathcal{R} is assumed to be *regular*, cf. [9].

The semantics of *SR_{OIQ}* is defined through *interpretations* $I = (\Delta^I, \cdot^I)$, where Δ^I is a non-empty *domain*, and \cdot^I is a function mapping every individual name to an element of Δ^I , each concept name to a subset of the domain, and each role name to a binary relation on the domain; see [1] for details. The interpretation \mathcal{I} is a *model* of the ontology \mathcal{O} if it satisfies all the axioms in \mathcal{O} .

Given two concepts C and D , we say that C is *subsumed* by D wrt. the ontology \mathcal{O} ($C \sqsubseteq_{\mathcal{O}} D$) if $C^I \subseteq D^I$ for every model I of \mathcal{O} , where we write C^I for

the extension of the concept C according to I . We write $C \equiv_{\mathcal{O}} D$ when $C \sqsubseteq_{\mathcal{O}} D$ and $D \sqsubseteq_{\mathcal{O}} C$. C is *strictly subsumed by* D wrt. \mathcal{O} ($C \sqsubset_{\mathcal{O}} D$) if $C \sqsubseteq_{\mathcal{O}} D$ and $C \not\equiv_{\mathcal{O}} D$. Given two roles, R and S , R is subsumed by S wrt. the ontology \mathcal{O} ($R \sqsubseteq_{\mathcal{O}} S$) if $R^I \subseteq S^I$ for every model I of \mathcal{O} . We may use the shortcuts $R \equiv_{\mathcal{O}} S$ and $R \sqsubset_{\mathcal{O}} S$ with the obvious interpretation.

Definition 2. Let \mathcal{O} be a *SR \mathcal{O} I \mathcal{Q}* ontology. The set of subconcepts of \mathcal{O} is given by

$$\text{sub}(\mathcal{O}) = \{\top, \perp\} \cup \bigcup_{C(a) \in \mathcal{O}} \text{sub}(C) \cup \bigcup_{C \sqsubseteq D \in \mathcal{O}} \text{sub}(C) \cup \text{sub}(D),$$

where $\text{sub}(C)$ is the set of subconcepts in C .

We now define the upward and downward cover sets of concept names and atomic roles respectively. Intuitively, the upward cover of the concept C collects the most specific subconcepts of \mathcal{O} that subsume C ; conversely, the downward cover of C collects the most general subconcepts from \mathcal{O} subsumed by C . The interpretation of the upward and downcover covers of atomic roles is similar. The concepts in $\text{sub}(\mathcal{O})$ are *some* concepts that are relevant in the context of \mathcal{O} , and that are used as building blocks for generalisations and specialisations. The properties of $\text{sub}(\mathcal{O})$ guarantee that the upward and downward cover sets are finite.

Definition 3. Let $\mathcal{O} = \mathcal{T} \cup \mathcal{R} \cup \mathcal{A}$ be an ontology. Let C be a concept, the upward cover and downward cover of C wrt. \mathcal{O} are:

$$\begin{aligned} \text{UpCov}_{\mathcal{O}}(C) &:= \{D \in \text{sub}(\mathcal{O}) \mid C \sqsubseteq_{\mathcal{O}} D \text{ and} \\ &\quad \nexists D' \in \text{sub}(\mathcal{O}) \text{ with } C \sqsubset_{\mathcal{O}} D' \sqsubset_{\mathcal{O}} D\}, \\ \text{DownCov}_{\mathcal{O}}(C) &:= \{D \in \text{sub}(\mathcal{O}) \mid D \sqsubseteq_{\mathcal{O}} C \text{ and} \\ &\quad \nexists D' \in \text{sub}(\mathcal{O}) \text{ with } D \sqsubset_{\mathcal{O}} D' \sqsubset_{\mathcal{O}} C\}. \end{aligned}$$

Let r be a role name, the upward cover and downward cover of r wrt. \mathcal{O} (where $N_R^- = \{r^- \mid r \in N_R\}$):

$$\begin{aligned} \text{UpCov}_{\mathcal{O}}(r) &:= \{s \in N_R \cup N_R^- \cup \{E, U\} \mid r \sqsubseteq_{\mathcal{O}} s \text{ and} \\ &\quad \nexists s' \in N_R \cup N_R^- \cup \{E, U\} \text{ with } r \sqsubset_{\mathcal{O}} s' \sqsubset_{\mathcal{O}} s \text{ and} \\ &\quad s, s' \text{ are simple in } \mathcal{R}\}. \\ \text{DownCov}_{\mathcal{O}}(r) &:= \{s \in N_R \cup N_R^- \cup \{E, U\} \mid s \sqsubseteq_{\mathcal{O}} r \text{ and} \\ &\quad \nexists s' \in N_R \cup N_R^- \cup \{E, U\} \text{ with } r \sqsubset_{\mathcal{O}} s' \sqsubset_{\mathcal{O}} s \text{ and} \\ &\quad s, s' \text{ are simple in } \mathcal{R}\}. \end{aligned}$$

Let n be a non-negative integer:

$$\begin{aligned} \text{UpCov}_{\mathcal{O}}(n) &:= \{n, n+1\}, \\ \text{DownCov}_{\mathcal{O}}(n) &:= \begin{cases} \{n-1, n\} & \text{when } n > 1 \\ \{n\} & \text{when } n = 0. \end{cases} \end{aligned}$$

In and of themselves, $\text{UpCov}_{\mathcal{O}}$ and $\text{DownCov}_{\mathcal{O}}$ miss ‘interesting’ refinements.

Example 1. Let $N_C = \{A, B, C\}$ and $\mathcal{O} = \{A \sqsubseteq B\}$. Then $\text{sub}(\mathcal{O}) = \{A, B, \top, \perp\}$. Now $\text{UpCov}_{\mathcal{O}}(A \sqcap C) = \{A\}$. Iterating, we get $\text{UpCov}_{\mathcal{O}}(A) = \{A, B\}$ and $\text{UpCov}_{\mathcal{O}}(B) = \{B, \top\}$. We could reasonably expect $B \sqcap C$ to be also a generalisation of $A \sqcap C$ wrt. \mathcal{O} but it will be missed by the iterated application of $\text{UpCov}_{\mathcal{O}}$, because $B \sqcap C \notin \text{sub}(\mathcal{O})$. Similarly, $\text{UpCov}_{\mathcal{O}}(\exists R.A) = \{\top\}$, even though we could expect $\exists R.B$ to be a generalisation of $\exists R.A$.

To take care of these omissions, we introduce generalisation and specialisation operators that will recursively exploit the structural complexity of the concept being refined.

Let \uparrow and \downarrow be two functions with domain $\mathcal{L}(N_C, N_R, N_I) \cup \mathcal{L}_r(N_R) \cup \mathbb{N}$. They map every concept to an element of the powerset of $\mathcal{L}(N_C, N_R, N_I)$, every role to an element of the powerset of $\mathcal{L}_r(N_R)$, and every non-negative integers to the powerset of \mathbb{N} . We define $\zeta_{\uparrow, \downarrow}$, the *abstract refinement operator*, by induction on the structure of concept descriptions as shown in Table 1.

We now define concrete refinement operators from the abstract operator $\zeta_{\uparrow, \downarrow}$.

Definition 4. *The generalisation operator and specialisation operator are defined, respectively, as*

$$\gamma_{\mathcal{O}} = \zeta_{\text{UpCov}_{\mathcal{O}}, \text{DownCov}_{\mathcal{O}}} \quad \text{and} \quad \rho_{\mathcal{O}} = \zeta_{\text{DowCov}_{\mathcal{O}}, \text{UpCov}_{\mathcal{O}}} .$$

Returning to Example 1, notice that for $N_C = \{A, B, C\}$ and $\mathcal{O} = \{A \sqsubseteq B\}$, we now have $\gamma_{\mathcal{O}}(A \sqcap C) = \{A \sqcap C, B \sqcap C, A \sqcap \top, A\}$.

Some comments are in order about Table 1. As in [20] the domain of $\gamma_{\mathcal{O}}$ and $\rho_{\mathcal{O}}$ is the set of concepts in negative normal form. In practice it can be extended straightforwardly to all concepts by modifying the clause $\zeta_{\uparrow, \downarrow}(\neg A)$ with $\zeta_{\uparrow, \downarrow}(\neg C) = \{\text{nnf}(\neg C') \mid C' \in \downarrow(C)\} \cup \uparrow(\neg C)$. The cases of $\forall R.C$ and $\exists R.C$ were already present for \mathcal{ALC} in [20]. Here, they are amended with the possibility to refine the R -role. Specific cases have been added to deal with SROIQ concepts and roles constructs.

Definition 5. *Given a DL concept C , its i -th refinement iteration by means of $\zeta_{\uparrow, \downarrow}$ (viz., $\zeta_{\uparrow, \downarrow}^i(C)$) is inductively defined as follows:*

- $\zeta_{\uparrow, \downarrow}^0(C) = \{C\}$;
- $\zeta_{\uparrow, \downarrow}^{j+1}(C) = \zeta_{\uparrow, \downarrow}^j(C) \cup \bigcup_{C' \in \zeta_{\uparrow, \downarrow}^j(C)} \zeta_{\uparrow, \downarrow}(C')$, $j \geq 0$.

The set of all concepts reachable from C by means of $\zeta_{\uparrow, \downarrow}$ in a finite number of steps is $\zeta_{\uparrow, \downarrow}^(C) = \bigcup_{i \geq 0} \zeta_{\uparrow, \downarrow}^i(C)$.*

2.1 Basic properties

Some basic properties about $\gamma_{\mathcal{O}}$ and $\rho_{\mathcal{O}}$ will help to build intuition, and will be useful in the remainder of this paper.

Lemma 1. *For every ontology \mathcal{O} :*

Table 1. Abstract refinement operator.

$$\begin{aligned}
\zeta_{\uparrow,\downarrow}(A) &= \uparrow(A) && , A \in N_C \\
\zeta_{\uparrow,\downarrow}(\neg A) &= \{\text{nnf}(\neg C) \mid C \in \downarrow(A)\} \cup \uparrow(\neg A) && , A \in N_C \\
\zeta_{\uparrow,\downarrow}(\top) &= \uparrow(\top) \\
\zeta_{\uparrow,\downarrow}(\perp) &= \uparrow(\perp) \\
\zeta_{\uparrow,\downarrow}(C \sqcap D) &= \{C' \sqcap D' \mid C' \in \zeta_{\uparrow,\downarrow}(C)\} \cup \\
&\quad \{C \sqcap D' \mid D' \in \zeta_{\uparrow,\downarrow}(D)\} \cup \uparrow(C \sqcap D) \\
\zeta_{\uparrow,\downarrow}(C \sqcup D) &= \{C' \sqcup D' \mid C' \in \zeta_{\uparrow,\downarrow}(C)\} \cup \\
&\quad \{C \sqcup D' \mid D' \in \zeta_{\uparrow,\downarrow}(D)\} \cup \uparrow(C \sqcup D) \\
\zeta_{\uparrow,\downarrow}(\forall R.C) &= \{\forall R'.C \mid R' \in \downarrow(R)\} \cup \{\forall R.C' \mid C' \in \zeta_{\uparrow,\downarrow}(C)\} \cup \uparrow(\forall R.C) \\
\zeta_{\uparrow,\downarrow}(\exists R.C) &= \{\exists R'.C \mid R' \in \uparrow(R)\} \cup \{\exists R.C' \mid C' \in \zeta_{\uparrow,\downarrow}(C)\} \cup \uparrow(\exists R.C) \\
\end{aligned}$$

SROIQ concepts:

$$\begin{aligned}
\zeta_{\uparrow,\downarrow}(\exists R.\text{Self}) &= \{\exists R'.\text{Self} \mid R' \in \uparrow(R)\} \cup \uparrow(\exists R.\text{Self}) \\
\zeta_{\uparrow,\downarrow}(\{i\}) &= \uparrow(\{i\}) \\
\zeta_{\uparrow,\downarrow}(\leq n R.C) &= \{\leq m R.C \mid m \in \uparrow(n)\} \cup \{\leq n R'.C \mid R' \in \downarrow(R)\} \cup \\
&\quad \{\leq n R.C' \mid C' \in \zeta_{\uparrow,\downarrow}(C)\} \cup \uparrow(\leq n R.C) \\
\zeta_{\uparrow,\downarrow}(\geq n R.C) &= \{\geq m R.C \mid m \in \downarrow(n)\} \cup \{\geq n R'.C \mid R' \in \uparrow(R)\} \cup \\
&\quad \{\geq n R.C' \mid C' \in \zeta_{\uparrow,\downarrow}(C)\} \cup \uparrow(\geq n R.C) \\
\end{aligned}$$

SROIQ roles:

$$\begin{aligned}
\zeta_{\uparrow,\downarrow}(r) &= \uparrow(r) \\
\zeta_{\uparrow,\downarrow}(r^-) &= \{s^- \mid s \in \uparrow(r), s \in N_R\} \cup \{s \mid s^- \in \uparrow(r), s^- \in N_R^-\} \\
\zeta_{\uparrow,\downarrow}(U) &= \uparrow(U) \\
\zeta_{\uparrow,\downarrow}(E) &= \uparrow(E) \\
\end{aligned}$$

1. **generalisation:** if $X \in \gamma_{\mathcal{O}}(Y)$ then $Y \sqsubseteq_{\mathcal{O}} X$
specialisation: if $X \in \rho_{\mathcal{O}}(Y)$ then $X \sqsubseteq_{\mathcal{O}} Y$
2. **cover nontriviality:** if $C \not\equiv_{\mathcal{O}} \top$ then there exists some $D \in \text{UpCov}_{\mathcal{O}}(C)$ such that $C \sqsubset_{\mathcal{O}} D$, and if $C \not\equiv_{\mathcal{O}} \perp$ then there exists some $D \in \text{DownCov}_{\mathcal{O}}(C)$ such that $D \sqsubset_{\mathcal{O}} C$
3. **cover triviality:** if $C \equiv_{\mathcal{O}} \top$ then $\top \in \text{UpCov}_{\mathcal{O}}(C)$, and if $C \equiv_{\mathcal{O}} \perp$ then $\perp \in \text{DownCov}_{\mathcal{O}}(C)$
4. **trivial generalisability:** $\top \in \gamma_{\mathcal{O}}^*(C)$, $U \in \gamma_{\mathcal{O}}^*(r)$
falsehood specialisability: $\perp \in \rho_{\mathcal{O}}^*(C)$, $E \in \rho_{\mathcal{O}}^*(r)$
5. **generalisation finiteness:** $\gamma_{\mathcal{O}}(C)$ is finite
specialisation finiteness: $\rho_{\mathcal{O}}(C)$ is finite

Lemma 2. $\mathcal{L}(N_C, N_R, N_I)$ is closed under $\gamma_{\mathcal{O}}$ and $\rho_{\mathcal{O}}$. If $C \in \mathcal{L}(N_C, N_R, N_I)$ then every refinement in $\gamma_{\mathcal{O}}(C)$ and $\rho_{\mathcal{O}}(C)$ is also in $\mathcal{L}(N_C, N_R, N_I)$.

Proof. (Sketch) The possibly delicate cases involve the refinements of roles. The condition on the upcover and downcover of a role R to contain only simple roles (cf. Definition 3) forces that every refinement of a role is simple. So, the restriction to simple roles guarantees that, e.g., $\zeta_{\uparrow,\downarrow}(\leq n R.C)$ and $\zeta_{\uparrow,\downarrow}(\geq n R.C)$ are in \mathcal{SROIQ} , as no complex role may appear in the scope of the cardinality restriction.

Notice that, by dropping that condition, we may have that, e.g., for roles p, q, r, s and axioms $r \sqsubseteq s, p \circ q \sqsubseteq s$ in \mathcal{O} , when generalising from $(\geq n r.C)$ to $(\geq n s.C)$ we violate the simplicity condition for roles in cardinality restrictions. \square

2.2 Complexity

We now analyse the computational aspects of the refinement operators.

Definition 6. *Given a \mathcal{SROIQ} ontology \mathcal{O} and concepts C, D , the problems $\gamma_{\mathcal{O}}$ -MEMBERSHIP and $\rho_{\mathcal{O}}$ -MEMBERSHIP ask whether $D \in \gamma_{\mathcal{O}}(C)$ and $D \in \rho_{\mathcal{O}}(C)$, respectively.*

The refinement operators $\gamma_{\mathcal{O}}$ and $\rho_{\mathcal{O}}$ are efficient, in the sense that deciding $\gamma_{\mathcal{O}}$ -MEMBERSHIP and $\rho_{\mathcal{O}}$ -MEMBERSHIP is not harder than deciding concept subsumption in \mathcal{SROIQ} . Recall that subsumption in \mathcal{SROIQ} is N2ExpTime-complete [12]. We can show that $\gamma_{\mathcal{O}}$ -MEMBERSHIP and $\rho_{\mathcal{O}}$ -MEMBERSHIP belong to the same complexity class.

Theorem 1. *$\gamma_{\mathcal{O}}$ -MEMBERSHIP and $\rho_{\mathcal{O}}$ -MEMBERSHIP are N2ExpTime-complete.*

Proof. (Sketch) We sketch the case of $\gamma_{\mathcal{O}}$ -MEMBERSHIP. The case of $\rho_{\mathcal{O}}$ -MEMBERSHIP is analogous.

For proving hardness, we first show that deciding whether $C' \in \text{UpCover}_{\mathcal{O}}(C)$ is as hard as concept subsumption. Then we show that $\gamma_{\mathcal{O}}$ -MEMBERSHIP is just as hard. The proofs are identical to the ones in [20].

For the upper bound, we first establish the complexity of computing the sets $\text{UpCover}_{\mathcal{O}}(C)$, $\text{DownCover}_{\mathcal{O}}(C)$, $\text{UpCover}_{\mathcal{O}}(r)$, $\text{DownCover}_{\mathcal{O}}(r)$. It suffices to check for every $D \in \text{sub}(\mathcal{O})$ whether $D \in \text{UpCov}_{\mathcal{O}}(C)$, or $D \in \text{DownCov}_{\mathcal{O}}(C)$, and collect those concepts for which the answer is positive. Each check can be done with at most $1 + 4|\text{sub}(\mathcal{O})|$ calls to a \mathcal{SROIQ} subsumption sub-routine in non-deterministic double exponential time. The size of $\text{sub}(\mathcal{O})$ is linear in the size of $\mathcal{T} \cup \mathcal{A}$. An analogous exhaustive check of all roles in $N_R \cup \{U, E\}$ permits to compute $\text{UpCov}_{\mathcal{O}}(r)$, and $\text{DownCov}_{\mathcal{O}}(r)$. Checking whether a role s is simple in \mathcal{R} can be done efficiently.

Finally, we can decide $\gamma_{\mathcal{O}}$ -MEMBERSHIP resorting to at most $|C|$ (one for every subconcept and role in C) computations of the sets $\text{UpCov}_{\mathcal{O}}(r)$, $\text{DownCov}_{\mathcal{O}}(r)$, $\text{UpCover}_{\mathcal{O}}(C')$ and $\text{DownCover}_{\mathcal{O}}(C')$, where $|C'| \leq |C|$.

Overall, the problem of deciding whether $D \in \gamma_{\mathcal{O}}(C)$ can be simulated by a deterministic oracle Turing machine with a polynomial number of calls to an N2ExpTime oracle. \square

3 Repairing Ontologies

Our refinement operators can be used as components of a method for repairing inconsistent \mathcal{SROIQ} ontologies by weakening, instead of removing, problematic axioms. In this paper, we do not refine the $RBox$, so the regularity of the role hierarchy remains intact in weakened ontologies.

Given an inconsistent ontology $\mathcal{O} = \mathcal{T} \cup \mathcal{A} \cup \mathcal{R}$, we proceed as described in Algorithm 1. We first need to find a consistent subontology \mathcal{O}^{ref} of \mathcal{O} to serve as *reference ontology* to be able to compute a non-trivial upcover and downcover. One approach is to pick a random maximally consistent subset of \mathcal{O} , $\text{FindMaximallyConsistentSet}(\mathcal{O})$, and choose it as reference ontology \mathcal{O}^{ref} ; another is to choose the intersection of all maximally consistent subsets of \mathcal{O} (e.g., [17]).

Once a reference ontology \mathcal{O}^{ref} has been chosen, and as long as \mathcal{O} is inconsistent, we select a “bad axiom” $\text{FindBadAxiom}(\mathcal{O})$ (in $\mathcal{T} \cup \mathcal{A}$) and replace it with a random weakening of it with respect to \mathcal{O}^{ref} . We can randomly sample a number of (or all the) minimally inconsistent subsets of axioms $I_1, I_2, \dots, I_k \subseteq \mathcal{O}$ and return one axiom in $\mathcal{T} \cup \mathcal{A}$ from the ones occurring the most often.

The set of all weakenings of an axiom with respect to a reference ontology is defined as follows:

Definition 7 (Axiom weakening). *Let $\mathcal{O} = \mathcal{T} \cup \mathcal{A} \cup \mathcal{R}$ be a \mathcal{SROIQ} ontology and ϕ be an axiom in $\mathcal{T} \cup \mathcal{A}$. The set of (least) weakenings of ϕ is the set $g_{\mathcal{O}}(\phi)$, such that:*

- $g_{\mathcal{O}}(C \sqsubseteq D) = \{C' \sqsubseteq D \mid C' \in \rho_{\mathcal{O}}(C)\} \cup \{C \sqsubseteq D' \mid D' \in \gamma_{\mathcal{O}}(D)\}$;
- $g_{\mathcal{O}}(C(a)) = \{C'(a) \mid C' \in \gamma_{\mathcal{O}}(C)\}$;
- $g_{\mathcal{O}}(R(a, b)) = \{R'(a, b) \mid R' \in \gamma_{\mathcal{O}}(R)\}$;
- $g_{\mathcal{O}}(\neg R(a, b)) = \{\neg R'(a, b) \mid R' \in \rho_{\mathcal{O}}(R)\}$;
- $g_{\mathcal{O}}(a = b) = \{a = b, \perp \sqsubseteq \top\}$;
- $g_{\mathcal{O}}(a \neq b) = \{a \neq b, \perp \sqsubseteq \top\}$.

The subprocedure $\text{WeakenAxiom}(\phi, \mathcal{O}^{\text{ref}})$ randomly returns one axiom in $g_{\mathcal{O}^{\text{ref}}}(\phi)$. For every subsumption or assertional axiom ϕ , the axioms in the set $g_{\mathcal{O}^{\text{ref}}}(\phi)$ are indeed weaker than ϕ , in the sense that – given the reference ontology $\mathcal{O}^{\text{ref}} - \phi$ entails them (and the opposite is not guaranteed).

Lemma 3. *For every subsumption or assertional axiom ϕ , if $\phi' \in g_{\mathcal{O}}(\phi)$, then $\phi \models_{\mathcal{O}} \phi'$.*

Proof. Suppose $C' \sqsubseteq D' \in g_{\mathcal{O}}(C \sqsubseteq D)$. Then, by definition of $g_{\mathcal{O}}$ and Lemma 1.1, $C' \sqsubseteq C$ and $D \sqsubseteq D'$ are inferred from \mathcal{O} . Thus, by transitivity of subsumption, we obtain that $C \sqsubseteq D \models_{\mathcal{O}} C' \sqsubseteq D'$. For the weakening of assertions, the result follows immediately from Lemma 1.1 again. \square

Clearly, substituting an axiom ϕ with one axiom from $g_{\mathcal{O}}(\phi)$ cannot diminish the set of interpretations of an ontology: if I is an interpretation that satisfies the axioms of ontology before such a replacement, I satisfies the same axioms even after it. By Lemma 1.4, any subsumption axiom is a finite number of

Algorithm 1 RepairOntologyWeaken(\mathcal{O})

```
 $O^{\text{ref}} \leftarrow \text{FindMaximallyConsistentSet}(\mathcal{O})$   
while  $\mathcal{O}$  is inconsistent do  
   $\phi_{\text{bad}} \leftarrow \text{FindBadAxiom}(\mathcal{O})$   
   $\phi_{\text{weaker}} \leftarrow \text{WeakenAxiom}(\phi_{\text{bad}}, O^{\text{ref}})$   
   $\mathcal{O} \leftarrow \mathcal{O} \setminus \{\phi_{\text{bad}}\} \cup \{\phi_{\text{weaker}}\}$   
end while  
Return  $\mathcal{O}$ 
```

weakenings away from the trivial axiom $\perp \sqsubseteq \top$. This holds also for equality and inequality *ABox* axioms. Any assertional axiom $C(a)$ is also a finite number of generalisations away from the trivial assertion $\top(a)$. Similarly, every assertional axioms $R(a, b)$ and $\neg R(a, b)$ are a finite number of generalisations away from the trivial assertions $U(a, b)$ and $\neg E(a, b)$, respectively.

Example 2. Consider the ontology \mathcal{O} containing the inconsistent set of axioms 1-4 in the introduction. Suppose that $\text{FindBadAxiom}(\mathcal{O})$ returns axiom 3 as the most problematic one. According to our definitions, a possible weakening of the axiom returned by $\text{WeakenAxiom}(3, O^{\text{ref}})$ may be $\text{MarriedPerson} \sqsubseteq \text{Person} \sqcap \leq 2 \text{ married.Person} \sqcap \geq 1 \text{ married.Person}$. Replacing axiom 3 with its weakening, the resulting ontology is consistent.

Example 3. Imagine that the *RBox* of both our reference ontology O^{ref} and the ontology \mathcal{O} contains the following axioms: *parent_of* \sqsubseteq *ancestor_of*; *father_of* \sqsubseteq *parent_of*; *mother_of* \sqsubseteq *parent_of*; *parent_of* \circ *parent_of* \sqsubseteq *grandparent*; *disjoint*(*parent_of*, *grandparent*); *parent_of*⁻ \equiv *child_of*.

And imagine that \mathcal{O} includes also the following statements, which cause inconsistency: *father_of*(*bob*, *mary*) (1); *mother_of*(*mary*, *alice*) (2); *child_of*(*alice*, *bob*) (3). A possible resolution of the inconsistency could be the weakening of axiom (3): $\text{WeakenAxiom}((3), O^{\text{ref}}) : \text{ancestor_of}^-(\text{alice}, \text{bob})$.

4 Iterated refinements and termination

As noted by Lemma 1.5 the set of “one-step” refinements of a concept is always finite. Moreover, Lemma 1.4 indicates that every concept can be refined in a finite number of iterations to \top (or \perp). Nonetheless, an iterated application of the refinement operator can lead to cases of non-termination. For instance, given an ontology defined as $\mathcal{O} = \{A \sqsubseteq \exists r.A\}$, if we generalise the concept A wrt. \mathcal{O} it is easy to see that we can obtain an infinite chain of generalisations that never reaches \top , i.e., $A \sqsubseteq_{\mathcal{O}} \exists r.A \sqsubseteq_{\mathcal{O}} \exists r.\exists r.A \dots$. For practical reasons this may need to be avoided, or mitigated. Running into this non-termination ‘problem’ is not new in the DL literature. In [3], the problem occurs in the context of finding a least common subsumer of DL concepts. Different solutions have been proposed to avoid this situation. Typically, some assumptions are made over the structure of the *TBox*, or a fixed role depth of concepts is considered. In the latter view, it

is possible to restrict the number of nested quantifiers in a concept description to a fixed constant k , to forbid generalisations/specialisations already picked along a chain from being picked again, and to introduce the definition of role depth of a concept to prevent infinite refinements. If this role depth upper bound is reached in the refinement of a concept, then \top and \perp are taken as generalisation and specialisation of the given concept respectively.

Another possibility is to abandon certain termination and adopt *almost-sure* termination, that is, termination with probability 1. The idea is to associate probabilities to the refinement ‘branches’ available at each refinement step.

In what follows we will show that, indeed, if we start from any concept C and choose uniformly at random a generalisation out of its set of possible generalisations (or a refinement out of its set of refinements: the proofs are entirely symmetrical) we will almost surely reach \top (\perp) within a finite number of steps. This implies at once that an axiom will almost surely not be indefinitely weakened by our procedure, and that Algorithm 1 will almost surely terminate.

The key ingredient of the proof is Lemma 4, which establishes an upper-bound on the rate of growth of the set of possible generalisations (refinements) along a chain.

Definition 8. *Let \mathcal{O} be a $SR\mathcal{O}IQ$ ontology and let $C \in \text{sub}(\mathcal{O})$. Then let $F(C) = |\gamma_{\mathcal{O}}(C)|$ be the number of generalisations of C , let $F'(C) = |\rho_{\mathcal{O}}(C)|$ be the number of specialisations of C and let $G(C) = \max(\{|C'| \mid C' \in \gamma_{\mathcal{O}}(C) \cup \rho_{\mathcal{O}}(C)\})$ be the maximum size of any generalisation or specialisation of C .*

By setting an upper bound to the size of $\gamma(C)$ and $\rho(C)$, part 1 of the following lemma may in particular be seen as a quantitative version of Lemma 1.5.

Lemma 4. *Let \mathcal{O} be a $SR\mathcal{O}IQ$ ontology and let C be any concept (not necessarily in $\text{sub}(\mathcal{O})$). Furthermore, let $k = |\text{sub}(\mathcal{O})| + 2|N_R| + 2$ and $q = \max(\{|C|, |\text{nnf}(\neg C)| \mid C \in \text{sub}(\mathcal{O})\})$. Then*

1. $F(C), F'(C) \leq 3k|C|$;
2. $G(C) \leq |C| + q$.

Proof. The full proof via structural induction is lengthy but presents no particular difficulties. The intuition behind it is the following: by our definitions, in a generalisation/specialisation step we essentially select a single subcomponent C' (or r) of the current expression C and we replace it with some element of $\text{sub}(\mathcal{O})$ (or of $N_R \cup N_R^-$). But these two sets are finite, and the number of subcomponents of an expression C increases linearly with the size of C . Thus, the number of possible generalisations/specialisations of C increases linearly with the size of C , and every generalisation/specialisation step increases the size of the resulting expression by some at most constant amount. \square

We can now prove our required result by showing that, even though the size of the concept expression – and, therefore, the number of possible generalizations – grows with every generalization step, it grows slowly enough that the generalization chain will almost surely eventually pick an element in the upcover of the current concept which is strictly more general than it. Thus, \top will be almost surely reached in a finite number of steps.

Theorem 2. *Let \mathcal{O} be a $SR\mathcal{O}I\mathcal{Q}$ ontology, let C be any $SR\mathcal{O}I\mathcal{Q}$ concept, and let $(C_i)_{i \in \mathbb{N}}$ be a sequence of concepts such that $C_0 = C$ and each C_{i+1} is chosen randomly in $\gamma_{\mathcal{O}}(C_i)$ according to the uniform distribution.*

Then, with probability 1, there exists some $i \in \mathbb{N}$ such that $C_i = \top$.

Proof. Let us first prove that, if $C \not\equiv_{\mathcal{O}} \top$, there is almost surely some C_i in the chain such that $C_i \equiv_{\mathcal{O}} \top$ (and, therefore, such that $C_{i'} \equiv_{\mathcal{O}} \top$ for all $i' > i$).

By the previous lemma, we know that $\gamma_{\mathcal{O}}(C_i)$ contains at most $3k|C_i|$ concepts. Furthermore, for every concept C_i such that $C_i \not\equiv_{\mathcal{O}} \top$ there exists at least one $C' \in \text{UpCov}_{\mathcal{O}}(C_i) \subseteq \gamma_{\mathcal{O}}(C_i)$ such that $C \sqsubset_{\mathcal{O}} C'$ (c.f., Lemma 1.2): therefore, the probability that the successor of C_i will be some $C_{i+1} \in \text{UpCov}_{\mathcal{O}}(C_i)$ such that $C_i \sqsubset_{\mathcal{O}} C_{i+1}$ is at least $1/(3k|C_i|)$. Now let $|C_0| = N$: since $C_{i+1} \in \gamma_{\mathcal{O}}(C_i)$, we then have that $|C_i| \leq qi + N$. Therefore, the probability that at step i we do *not* select randomly an element of $\text{UpCov}_{\mathcal{O}}(C_i)$ that is strictly more general than C_i will be at most $\frac{3k(qi+N)-1}{3k(qi+N)} = \frac{i+\ell-\epsilon}{i+\ell}$ for $\ell = N/q$ and $\epsilon = 1/(3kq)$. But then the probability that we *never* select a strictly more general element from the upcover will be at most $\prod_{i=0}^{\infty} \frac{i+\ell-\epsilon}{i+\ell} = 0$,³ and thus our generalisation sequence $C = C_0 \sqsubseteq_{\mathcal{O}} C_1 \sqsubseteq_{\mathcal{O}} C_2 \dots$ will almost surely contain some C_i such that $C_{i+1} \in \text{UpCov}_{\mathcal{O}}(C_i) \subseteq \text{sub}(\mathcal{O})$ and $C \sqsubseteq_{\mathcal{O}} C_i \sqsubset_{\mathcal{O}} C_{i+1}$. By the same argument, the generalisation sequence starting from C_{i+1} will almost surely eventually reach some $C_{j+1} \in \text{sub}(\mathcal{O})$ with $C \sqsubset_{\mathcal{O}} C_{i+1} \sqsubset_{\mathcal{O}} C_{j+1}$, and so forth; and by applying this line of reasoning $|\text{sub}(\mathcal{O})|$ times, we will almost surely eventually reach some concept $D \equiv_{\mathcal{O}} \top$, as required.

Now let us consider a generalisation chain $D = D_0 \sqsubseteq_{\mathcal{O}} D_1 \sqsubseteq_{\mathcal{O}} D_2 \dots$, where as before every D_{i+1} is chosen randomly among $\gamma(D)$, starting from some concept $D \equiv_{\mathcal{O}} \top$. Now, since $D \equiv_{\mathcal{O}} \top$ and $D \sqsubseteq_{\mathcal{O}} D_i$ for all i , $\top \in \text{UpCov}_{\mathcal{O}}(D_i) \subseteq \gamma_{\mathcal{O}}(D_i)$ for all i (c.f., Lemma 1.3). Thus, at every iteration step i we have a probability of at least $1/|\gamma(D_i)|$ that $D_{i+1} = \top$; and if we let $N' = |D|$, by the previous results we obtain at once that $|\gamma(D_i)| \leq iq + N'$, and hence that the probability that we do not end up generalising D_i to \top is at most $(3k(iq + N') - 1)/(3k(iq + N'))$, and finally that the probability that we *never* reach \top is $\prod_{i=0}^{\infty} \frac{3k(iq+N')-1}{3k(iq+N')} = \prod_{i=0}^{\infty} \frac{i+\ell'-\epsilon'}{i+\ell'} = 0$ where $\ell' = N'/q$ and $\epsilon' = 1/3kq$. \square

Note that, by our definitions, \top can be further generalized to all elements of its upcover (that is, all concepts of $\text{sub}(\mathcal{O})$ which are equivalent to \top with respect to \mathcal{O}), and similarly \perp can be further specialized to other concepts in its downcover. If this behaviour is unwanted, it is easy to force the upcover of \top to contain only \top , and likewise for \perp ; but regardless of this, our results imply

³ One way to verify this is to observe that the series $\sum_{i=0}^{\infty} (\log(i + \ell - \epsilon) - \log(i + \ell))$ diverges to minus infinity. This in turn may be verified by noting that $\sum_{i=0}^{\infty} (\log(i + \ell - \epsilon) - \log(i + \ell)) \leq \sum_{i=0}^{\infty} (\log(i + \lceil \ell \rceil - \epsilon) - \log(i + \lceil \ell \rceil)) = \sum_{i=\lceil \ell \rceil}^{\infty} (\log(i - \epsilon) - \log(i))$, because $\log(i + \ell - \epsilon) - \log(i + \ell) \leq \log(i + \lceil \ell \rceil - \epsilon) - \log(i + \lceil \ell \rceil)$, and then showing that $-\sum_{i=\lceil \ell \rceil}^{\infty} (\log(i - \epsilon) - \log(i)) = \sum_{i=\lceil \ell \rceil}^{\infty} \log(i) - \log(i - \epsilon)$ diverges to plus infinity by means of the integral method: the terms of the series are all positive, and $\int_{\lceil \ell \rceil}^U \log(x) - \log(x - \epsilon) dx$ goes to infinity when U goes to infinity. Since the integral diverges, so does the series, which gives us our conclusion.

at once that every axiom will almost surely be weakened into $\top \sqsubseteq \perp$ in a finite number of weakening steps.

5 Outlook

We presented a set of refinement operators covering most aspects of the full *SROIQ* language. Further additions to the general rules of refinements may be studied, e.g. those governing refinements of roles that can be obtained by considering complex roles in the up and down covers. Refinement operators have so far been implemented for the *ALC* fragment. This implementation will need to be extended to deal with the *SROIQ* refinement operators presented here. As done in [20] for *ALC* ontologies, this will allow us to run experiments to evaluate the usefulness of *SROIQ* ontology fine repairs via axiom weakening.

References

1. Baader, F., Horrocks, I., Lutz, C., Sattler, U.: An Introduction to Description Logic. Cambridge University Press (2017)
2. Baader, F., Kriegel, F., Nuradiansyah, A., Peñaloza, R.: Making repairs in description logics more gentle. In: Principles of Knowledge Representation and Reasoning: Proceedings of the Sixteenth International Conference, KR 2018, Tempe, Arizona, 30 October - 2 November 2018. pp. 319–328 (2018), <https://aaai.org/ocs/index.php/KR/KR18/paper/view/18056>
3. Baader, F., Küsters, R.: Nonstandard Inferences in Description Logics: The Story So Far, pp. 1–75. Springer New York, New York, NY (2006), https://doi.org/10.1007/0-387-31072-X_1
4. Baader, F., Peñaloza, R., Suntisrivaraporn, B.: Pinpointing in the description logic \mathcal{EL}^+ . In: Proc. of KI 2007. LNCS, vol. 4667, pp. 52–67. Springer (2007)
5. Badea, L., Nienhuys-Cheng, S.: A refinement operator for description logics. In: Cussens, J., Frisch, A.M. (eds.) Inductive Logic Programming, 10th International Conference, ILP 2000, London, UK, July 24–27, 2000, Proceedings. Lecture Notes in Computer Science, vol. 1866, pp. 40–59. Springer (2000), https://doi.org/10.1007/3-540-44960-4_3
6. Blockeel, H., Ramon, J., Shavlik, J.W., Tadepalli, P. (eds.): Inductive Logic Programming, 17th International Conference, ILP 2007, Corvallis, OR, USA, June 19–21, 2007, Revised Selected Papers, Lecture Notes in Computer Science, vol. 4894. Springer (2008), <https://doi.org/10.1007/978-3-540-78469-2>
7. Confalonieri, R., Eppe, M., Schorlemmer, M., Kutz, O., Peñaloza, R., Plaza, E.: Upward Refinement Operators for Conceptual Blending in the Description Logic \mathcal{EL}^{++} . Annals of Mathematics and Artificial Intelligence 82(1–3), 69–99 (2018), <https://www.sciencedirect.com/science/article/abs/pii/S0952197615002006>
8. Du, J., Qi, G., Fu, X.: A practical fine-grained approach to resolving incoherent owl 2 dl terminologies. In: Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management. pp. 919–928 (2014)
9. Horrocks, I., Kutz, O., Sattler, U.: The even more irresistible SROIQ. In: Doherty, P., Mylopoulos, J., Welty, C.A. (eds.) Proceedings, Tenth International Conference on Principles of Knowledge Representation and Reasoning, Lake District of the

- United Kingdom, June 2-5, 2006. pp. 57–67. AAAI Press (2006), <http://www.aaai.org/Library/KR/2006/kr06-009.php>
10. Kalyanpur, A., Parsia, B., Sirin, E., Grau, B.C.: Repairing unsatisfiable concepts in OWL ontologies. In: ESWC. vol. 6, pp. 170–184. Springer (2006)
 11. Kalyanpur, A., Parsia, B., Sirin, E., Hendler, J.: Debugging unsatisfiable classes in OWL ontologies. *Web Semantics: Science, Services and Agents on the World Wide Web* 3(4), 268–293 (2005)
 12. Kazakov, Y.: RIQ and SROIQ are harder than SHOIQ. In: *Principles of Knowledge Representation and Reasoning: Proceedings of the Eleventh International Conference, KR 2008, Sydney, Australia, September 16-19, 2008*. pp. 274–284 (2008), <http://www.aaai.org/Library/KR/2008/kr08-027.php>
 13. van der Laag, P.R., Nienhuys-Cheng, S.H.: Completeness and properness of refinement operators in inductive logic programming. *The Journal of Logic Programming* 34(3), 201 – 225 (1998), <http://www.sciencedirect.com/science/article/pii/S0743106697000770>
 14. Lehmann, J., Hitzler, P.: Foundations of refinement operators for description logics. In: Blockeel et al. [6], pp. 161–174, https://doi.org/10.1007/978-3-540-78469-2_18
 15. Lehmann, J., Hitzler, P.: A refinement operator based learning algorithm for the *ALC* description logic. In: Blockeel et al. [6], pp. 147–160, https://doi.org/10.1007/978-3-540-78469-2_17
 16. Lehmann, J., Hitzler, P.: Concept learning in description logics using refinement operators. *Machine Learning* 78(1-2), 203–250 (2010)
 17. Lembo, D., Lenzerini, M., Rosati, R., Ruzzi, M., Savo, D.F.: Inconsistency-tolerant semantics for description logics. In: *Proc. of RR 2010*. LNCS, vol. 6333, pp. 103–117. Springer (2010)
 18. Porello, D., Troquard, N., Peñaloza, R., Confalonieri, R., Galliani, P., Kutz, O.: Two Approaches to Ontology Aggregation Based on Axiom Weakening. In: Lang, J. (ed.) *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*. pp. 1942–1948. ijcai.org (2018), <https://doi.org/10.24963/ijcai.2018/268>
 19. Schlobach, S., Cornet, R.: Non-standard reasoning services for the debugging of description logic terminologies. In: *Proc. of IJCAI-03*. pp. 355–362. Morgan Kaufmann (2003)
 20. Troquard, N., Confalonieri, R., Galliani, P., Peñaloza, R., Porello, D., Kutz, O.: Repairing Ontologies via Axiom Weakening. In: McIlraith, S.A., Weinberger, K.Q. (eds.) *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*. pp. 1981–1988. AAAI Press (2018), <https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/17189>