Almost certain termination for $\mathcal{ALC}$ weakening

Roberto Confalonieri[1][0000−0003−0936−2123], Pietro Galliani[1][0000−0003−2544−5332], Oliver Kutz[1][0000−0003−1517−7354], Daniele Porello[2][0000−0003−3655−0218], Guendalina Righetti[1][0000−0002−4027−5434], and Nicolas Troquard[1][0000−0002−5763−6080]

[1] Free University of Bozen-Bolzano, Faculty of Computer Science, Italy
name.surname@unibz.it
[2] University of Genoa, Department of Antiquity, Philosophy, and History, Italy
daniele.porello@unige.it

Abstract. Concept refinement operators have been introduced to describe and compute generalisations and specialisations of concepts, with, amongst others, applications in concept learning and ontology repair through axiom weakening. We here provide a probabilistic proof of almost-certain termination for iterated refinements, thus for an axiom weakening procedure for the fine-grained repair of $\mathcal{ALC}$ ontologies. We determine the computational complexity of refinement membership, and discuss performance aspects of a prototypical implementation, verifying that almost-certain termination means actual termination in practice.

Keywords: axiom weakening · refinement operator · ontology repair · almost-certain termination.

1 Introduction

The traditional approach to repairing inconsistent ontologies amounts to identifying problematic axioms and removing them (e.g., [18,10,9,4]). Whilst this approach is sufficient to guarantee the consistency of the resulting ontology, it often leads to unnecessary information loss.

Approaches to repairing ontologies more gently via *axiom weakening* were proposed in the literature [8,6,19,2]. In [6], 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 [19], where the authors define an abstract refinement operator for generalising and specialising $\mathcal{ALC}$ concepts and weakening $\mathcal{ALC}$ axioms. They propose an ontology repair procedure that solves inconsistencies by weakening axioms rather than by removing them. In [2], the authors present general theoretical results for axiom weakening in Description Logics (DLs) and $\mathcal{EL}$ in particular. Refinement operators in Description Logic have also been studied with applications to Machine Learning [5,13,12,14].

Concept refinement operators come in two flavours [11]. A generalisation operator w.r.t. 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 w.r.t. 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 w.r.t. 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 $\mathcal{O}$).

Refinement operators enjoy a few properties that render them suitable for an implementation of axiom weakening [19]. 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. Ontology repairs via concept refinements and axiom weakening have also been used to merge two mutually inconsistent ontologies [17].

In this paper, we fill a gap in the above sketched research landscape and provide a proof of almost-certain termination of the ontology repair procedure based on the axiom weakening proposed in [19]. Since infinite non-stabilising chains of refinements exist in principle, this is the best we could hope for. We also verify in an empirical study that this theoretical result implies actual and robust (that is, reproducible) termination in a number of test scenarios using real world as well as synthetic ontologies.

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 $\mathcal{ALC}$; for full details see [1]. The syntax of $\mathcal{ALC}$ is based on two disjoint sets N_C and N_R, *concept names* and *role names* respectively. The set of $\mathcal{ALC}$ *concepts* is generated by the grammar (where $R \in N_R$ and $A \in N_C$):

$$C ::= \bot \mid \top \mid A \mid \neg C \mid C \sqcap C \mid C \sqcup C \mid \forall R.C \mid \exists R.C$$

In the following, $\mathcal{L}(N_C, N_R)$ denotes the set of concepts and roles that can be built over N_C, N_R in $\mathcal{ALC}$. $\mathsf{nnf}(C)$ denotes the negation normal form of concept C. $|C|$ denotes the size of a concept, defined as:

Definition 1. *The* size $|C|$ *of a concept* C *is inductively defined as follows. For* $C \in N_C \cup \{\top, \bot\}$, $|C| = 1$. *Then,* $|\neg C| = 1 + |C|$; $|C \sqcap D| = |C \sqcup D| = 1 + |C| + |D|$; *and* $|\exists R.C| = |\forall R.C| = 1 + |C|$.

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 $C(a)$ and $R(a, b)$, which express knowledge about objects in the knowledge domain. An $\mathcal{ALC}$ ontology $\mathcal{O} = \mathcal{T} \cup \mathcal{A}$ is defined by a *TBox* $\mathcal{T}$ and an *ABox* $\mathcal{A}$.

The semantics of $\mathcal{ALC}$ 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 w.r.t. 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 w.r.t. $\mathcal{O}$ ($C \sqsubset_{\mathcal{O}} D$) if $C \sqsubseteq_{\mathcal{O}} D$ and $C \not\equiv_{\mathcal{O}} D$.

We now define the upward and downward covers of concept names and atomic roles respectively. In this paper, their range will consist of the finite set of subconcepts of the ontology $\mathcal{O}$, which is defined as follows:

Definition 2. *For $\mathcal{O}$ an $\mathcal{ALC}$ ontology, the set of* subconcepts *of $\mathcal{O}$ is given by*

$$\mathsf{sub}(\mathcal{O}) = \{\top, \bot\} \cup \bigcup_{C(a)\in\mathcal{A}} \mathsf{sub}(C) \cup \bigcup_{C \sqsubseteq D \in \mathcal{T}} \big(\mathsf{sub}(C) \cup \mathsf{sub}(D)\big) \ ,$$

where $\mathcal{O} = \mathcal{T} \cup \mathcal{A}$. $\mathsf{sub}(C)$ is the set of subconcepts in C inductively defined over the structure of C.

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 concepts in $\mathsf{sub}(\mathcal{O})$ 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 $\mathsf{sub}(\mathcal{O})$ guarantee that the upward and downward cover sets are finite.

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

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

Note that the basic $\mathsf{UpCov}_{\mathcal{O}}$ and $\mathsf{DownCov}_{\mathcal{O}}$ 'miss' a mumber of relevant refinements, depending on the definition of $\mathsf{sub}(\mathcal{O})$. Consider the following example.

Example 1. Let $N_C = \{A, B, C\}$ and $\mathcal{O} = \{A \sqsubseteq B\}$. Then $\mathsf{sub}(\mathcal{O}) = \{A, B, \top, \bot\}$. According to Def. 3, $\mathsf{UpCov}_{\mathcal{O}}(A \sqcap C) = \{A\}$. Iterating, we get $\mathsf{UpCov}_{\mathcal{O}}(A) = \{A, B\}$ and $\mathsf{UpCov}_{\mathcal{O}}(B) = \{B, \top\}$. Semantically, $B \sqcap C$ is also a generalisation of $A \sqcap C$ w.r.t. $\mathcal{O}$. However, it is missed by the iterated application of $\mathsf{UpCov}_{\mathcal{O}}$, because $B \sqcap C \notin \mathsf{sub}(\mathcal{O})$. Similarly, $\mathsf{UpCov}_{\mathcal{O}}(\exists R.A) = \{\top\}$, even though one would expect semantically that also $\exists R.B$ is a generalisation of $\exists R.A$.

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

Table 1. Abstract refinement operator for $\mathcal{ALC}$.

To address this situation, we introduce generalisation and specialisation operators that recursively exploit the syntactic structure of the concept being refined. Let $\uparrow$ and $\downarrow$ be two functions with domain $\mathcal{L}(N_C, N_R)$ that map every concept to a set of concepts in $\mathcal{L}(N_C, N_R)$. 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*

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

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

Some comments are in order about Table 1. As in [19] the domain of $\gamma_{\mathcal{O}}$ and $\rho_{\mathcal{O}}$ is the set of concepts in negation normal form. In practice it can be extended straightforwardly to all concepts by modifying the clause $\zeta_{\uparrow,\downarrow}(\neg A)$ with $\zeta_{\uparrow,\downarrow}(\neg C) = \{\mathsf{nnf}(\neg C') \mid C' \in {\downarrow}(C)\} \cup {\uparrow}(\neg C)$.

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

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

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

Lemma 1. *$\mathcal{L}(N_C, N_R)$ is closed under $\gamma_{\mathcal{O}}$ and $\rho_{\mathcal{O}}$. If $C \in \mathcal{L}(N_C, N_R)$ then every refinement in $\gamma_{\mathcal{O}}(C)$ and $\rho_{\mathcal{O}}(C)$ is also in $\mathcal{L}(N_C, N_R)$.*

Algorithm 1 RepairOntologyWeaken($\mathcal{O}$)

1: $O^{\text{ref}} \leftarrow \mathsf{FindMaximallyConsistentSet}(\mathcal{O})$
2: **while** $\mathcal{O}$ is inconsistent **do**
3: $\quad \phi_{\mathsf{bad}} \leftarrow \mathsf{FindBadAxiom}(\mathcal{O})$
4: $\quad \Phi_{\mathsf{weaker}} \leftarrow \mathsf{WeakenAxiom}(\phi_{\mathsf{bad}}, O^{\text{ref}})$
5: $\quad \mathcal{O} \leftarrow \mathcal{O} \setminus \{\phi_{\mathsf{bad}}\} \cup \Phi_{\mathsf{weaker}}$
6: **end while**
7: Return $\mathcal{O}$

3 Repairing Ontologies

The refinement operators can be used as components of a method for repairing inconsistent $\mathcal{ALC}$ ontologies by weakening, instead of removing, problematic axioms. Given an inconsistent ontology $\mathcal{O} = \mathcal{T} \cup \mathcal{A}$, we proceed as described in Algorithm 1.

We first need to find a consistent subontology 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 $\mathcal{O}$ (line 1), and choose it as reference ontology O^{ref}; another option is to choose the intersection of all maximally consistent subsets of $\mathcal{O}$ (e.g., [15]). Once a reference ontology O^{ref} has been chosen, and as long as $\mathcal{O}$ is inconsistent, we select a 'bad axiom' (line 3) in $\mathcal{T} \cup \mathcal{A}$) and replace it with a random weakening of it w.r.t. O^{ref} (lines 4 and 5). For added flexibility, a *weakening of an axiom* is a set of axioms.

Definition 6 (Axiom weakening). *For $\mathcal{O} = \mathcal{T} \cup \mathcal{A}$ an $\mathcal{ALC}$ ontology and ϕ an axiom in $\mathcal{T} \cup \mathcal{A}$, the set of (least)* weakenings *of ϕ is the set $g_{\mathcal{O}}(\phi)$ such that:*

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

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

Lemma 2. *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 the definitions of $g_{\mathcal{O}}$, $\gamma_{\mathcal{O}}$, and $\rho_{\mathcal{O}}$, it clearly follows that $C' \sqsubseteq C$ 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 class assertions, the result follows in a similar way. For the weakening of role assertions, the result simply follows from the definition of $\top$. □

The cases of GCIs and class assertions axioms are rather straightforward. The weakening of a GCI $C \sqsubseteq D$ is obtained by either generalising D or specialising C.

The weakening of the class assertion $C(a)$ is obtained by generalising C. As the refinement operators are reflexive, the choice here is also to make $g_{\mathcal{O}}$ reflexive, so an axiom may be weakened into itself. For the case of the role assertion axiom $R(a,b)$, one should observe that in absence of role hierarchies, nominals, and set constructors, there is nothing weaker apart from a trivial statement. Weakening $R(a,b)$ into the two axioms $\top(a)$ and $\top(b)$ allows us to preserve the signature. To keep $g_{\mathcal{O}}$ reflexive, we also allow $R(a,b)$ to be weakened into itself.

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 an ontology before such a replacement, I satisfies the same axioms even after it. Since any concept can be generalised to the $\top$ concept or specialised to the $\bot$ concept (in finitely many steps), any subsumption axiom is a finite number of weakenings away from the trivial axiom $\bot \sqsubseteq \top$. Any assertional axiom $C(a)$ is also a finite number of generalisations away from the trivial assertion $\top(a)$. Similarly, every assertional axiom of type $R(a,b)$ is one step of generalisation away from the set of trivial assertions $\top(a)$ and $\top(b)$ (whilst maintaining the signature of the Abox).

Theorem 1. *If Algorithm 1 returns $\mathcal{O}$, then $\mathcal{O}$ is a consistent $\mathcal{ALC}$ ontology.*

Example 2. Consider the ontology $\mathcal{O}$ containing the following inconsistent set of axioms:

(1) $\mathsf{Vehicle} \sqsubseteq \exists \mathit{has}.\mathsf{Motor}$; (2) $\mathsf{Bike} \sqsubseteq \mathsf{Vehicle}$;
(3) $\mathsf{Bike} \sqsubseteq \neg\exists \mathit{has}.\mathsf{Motor}$; (4) $\mathsf{Motor} \sqsubseteq \mathsf{MeansOfPropulsion}$.

Suppose that $\mathsf{FindBadAxiom}(\mathcal{O})$ returns axiom (1) as the most problematic one. According to our definitions, a possible weakening of the axiom returned by $\mathsf{WeakenAxiom}((1), O^{\mathrm{ref}})$ may be $(1)^*$ $\mathsf{Vehicle} \sqsubseteq \exists \mathit{has}.\mathsf{MeansOfPropulsion}$. Replacing axiom (1) with its weakening $(1)^*$, the resulting ontology becomes consistent.

4 Iterated refinements and termination

Clearly, the set of "one-step" refinements of a concept is always finite, given the finiteness of $\mathsf{sub}(\mathcal{O})$. Moreover, every concept can be refined in a finite number of iterations to $\top$ (or $\bot$). 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 w.r.t. $\mathcal{O}$ it is easy to see that we can obtain an infinite chain of generalisations that never reaches $\top$, i.e., $A \sqsubseteq_{\mathcal{O}} \exists R.A \sqsubseteq_{\mathcal{O}} \exists R.\exists R.A \ldots$. 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 $\bot$ are taken as generalisation and specialisation of the given concept respectively.

Another possibility is to abandon certain termination and adopt *almost-certain* (or *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 specialisation out of its set of specialisations: results and proofs are entirely symmetrical) we will almost surely reach $\top$ ($\bot$) 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 3, which establishes an upper-bound on the rate of growth of the set of possible generalisations (specialisations) along a chain.

Definition 7. *Let $\mathcal{O}$ be an $\mathcal{ALC}$ ontology and let $C \in \mathsf{sub}(\mathcal{O})$. 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.*

The upper bound to the size of $\gamma(C)$ and $\rho(C)$ provided in part 1 of the following lemma gives a uniform upper bound to the size of generalisation/specialisation sets, for which we so far only knew that they are always finite.

Lemma 3. *Let $\mathcal{O}$ be an $\mathcal{ALC}$ ontology and let C be any concept (not necessarily in $\mathsf{sub}(\mathcal{O})$). Furthermore, let $k = |\mathsf{sub}(\mathcal{O})|$ and $q = \max(\{|C|, |\mathsf{nnf}(\neg C)| \mid C \in \mathsf{sub}(\mathcal{O}\})$. Then the following two properties hold:*

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

Proof. The proof is by structural induction. The intuition behind it is the following: by our definitions, in a generalisation/specialisation step we essentially select a single subcomponent C' of the current expression C and we replace it with some element of $\mathsf{sub}(\mathcal{O})$. But this set is 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. We next present the main parts of the structural induction on C (leaving out the analogous cases for $\exists R.C$ and $C \sqcup D$), which we assume is in Negation Normal Form.

1. If C is an atomic concept A, $\top$, or $\bot$ then $\gamma_{\mathcal{O}}(C) = \mathsf{UpCov}_{\mathcal{O}}(C)$, $\rho_{\mathcal{O}}(C) = \mathsf{DownCov}_{\mathcal{O}}(C)$ and $|C| = 1$. Thus, $F(C), F'(C) \leq |\mathsf{sub}(\mathcal{O})| \leq k \leq 3k|C|$ and $G(C) \leq q < q + 1$, as required.
2. If C is a negated atomic concept $\neg A$, $\gamma_{\mathcal{O}}(C) = \{\mathsf{nnf}(\neg C) | C \in \mathsf{DownCov}_{\mathcal{O}}(A)\} \cup \mathsf{UpCov}_{\mathcal{O}}(\neg A)$, $\rho_{\mathcal{O}}(C) = \{\mathsf{nnf}(\neg C) | C \in \mathsf{UpCov}_{\mathcal{O}}(A)\} \cup \mathsf{DownCov}_{\mathcal{O}}(\neg A)$, and $|C| = 2$. Thus, $F(C), F'(C) \leq 2|\mathsf{sub}(\mathcal{O})| \leq 2k \leq 3k|C|$ and $G(C) \leq q < q + 2$.

3. $|C \sqcap D| = |C| + |D| + 1$ and $\gamma_{\mathcal{O}}(C \sqcap D) = \{C' \sqcap D \mid C' \in \gamma_{\mathcal{O}}(C)\} \cup \{C \sqcap D' \mid D' \in \gamma_{\mathcal{O}}(D)\} \cup \mathsf{UpCov}_{\mathcal{O}}(C \sqcap D)$.
By induction hypothesis, $|\{C' \sqcap D \mid C' \in \gamma_{\mathcal{O}}(C)\}| \leq 3k|C|$ and $|\{C \sqcap D' \mid D' \in \gamma_{\mathcal{O}}(D)| \leq 3k|D|$ and furthermore $|\mathsf{UpCov}_{\mathcal{O}}(C \sqcap D)| \leq k$, and so
$$F(C \sqcap D) \leq 3k|C| + 3k|D| + k \leq 3k|C \sqcap D|.$$
Moreover, by induction hypothesis if $C' \in \gamma_{\mathcal{O}}(C)$ then $|C'| \leq G(C) \leq |C| + q$, and so $|C' \sqcap D| \leq |C| + |D| + 1 + q = |C \sqcap D| + q$; if $D' \in \gamma_{\mathcal{O}}(D)$ then $|D| \leq G(D) \leq |D| + q$ and so $C \sqcap D' \leq |C| + |D| + 1 + q = |C \sqcap D| + q$; and $|C''| \leq q \leq |C \sqcap D| + q$ for all $C'' \in \mathsf{UpCov}_{\mathcal{O}}(C \sqcap D)$. Thus, for all $C'' \in \gamma(C \sqcap D)$ we have that $|C''| \leq |C \sqcap D| + q$.
Similarly, $\rho_{\mathcal{O}}(C \sqcap D) = \{C' \sqcap D \mid C' \in \rho_{\mathcal{O}}(C)\} \cup \{C \sqcap D' \mid D' \in \rho_{\mathcal{O}}(D)\} \cup \mathsf{DownCov}_{\mathcal{O}}(C \sqcap D)$, and a completely analogous argument applies.
4. $|\forall R.C| = |C| + 1$ and $\gamma_{\mathcal{O}}(\forall R.C) = \{\forall R.C' \mid C' \in \gamma_{\mathcal{O}}(C)\} \cup \mathsf{UpCov}_{\mathcal{O}}(\forall R.C)$. By induction hypothesis, $|\{\forall R.C' \mid C' \in \gamma_{\mathcal{O}}(C)\}| \leq 3k|C|$; moreover, $|\mathsf{UpCov}_{\mathcal{O}}(\forall R.C)| \leq k$. Thus, $F(\forall R.C) \leq 3k|C| + k \leq 3k(|C| + 1) = 3k|\forall R.C|$. Furthermore, if $C' \in \gamma_{\mathcal{O}}(C)$, by induction hypothesis $|C'| \leq |C| + q$ and hence $|\forall R.C'| \leq |C| + 1 + q = |\forall R.C| + q$; and if $C'' \in \mathsf{UpCov}_{\mathcal{O}}(\forall R.C)$ then $|C''| \leq q \leq |\forall R.C| + q$; and so whenever $C'' \in \gamma_{\mathcal{O}}(\forall R.C)$ we have that $|C''| \leq |\forall R.C| + q$, as required.
Similarly, $\rho_{\mathcal{O}}(\forall R.C) = \{\forall R.C' \mid C' \in \rho_{\mathcal{O}}(C)\} \cup \mathsf{DownCov}_{\mathcal{O}}(\forall R.C)$, and a completely analogous argument shows that $F'(\forall R.C) \leq 3k|\forall R.C|$ and that $|C''| \leq |\forall R.C| + q$ for all $C'' \in \rho_{\mathcal{O}}(\forall R.C)$. □

We can now prove our required result by showing that, even though the size of the concept expression – and, therefore, the number of possible generalisations – grows with every generalisation step, it grows slowly enough such that the generalisation 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 an $\mathcal{ALC}$ ontology, let C be any $\mathcal{ALC}$ 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 $C' \in \mathsf{UpCov}_{\mathcal{O}}(C_i) \subseteq \gamma_{\mathcal{O}}(C_i)$ such that $C \sqsubset_{\mathcal{O}} C'$ (for instance, the $\top$ concept itself): therefore, the probability that the successor of C_i will be some $C_{i+1} \in \mathsf{UpCov}_{\mathcal{O}}(C_i)$ 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 $\mathsf{UpCov}_{\mathcal{O}}(C_i)$ 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]

[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 $\sum_{i=0}^{\infty}(\log(i+$

and thus our generalisation sequence $C = C_0 \sqsubseteq_{\mathcal{O}} C_1 \sqsubseteq_{\mathcal{O}} C_2 \ldots$ will almost surely contain some C_i such that $C_{i+1} \in \mathsf{UpCov}_{\mathcal{O}}(C_i) \subseteq \mathsf{sub}(\mathcal{O})$ 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 $C_{j+1} \in \mathsf{sub}(\mathcal{O})$ with $C \sqsubset_{\mathcal{O}} C_{i+1} \sqsubset_{\mathcal{O}} C_{j+1}$, and so forth; and by applying this line of reasoning $|\mathsf{sub}(\mathcal{O})|$ times, we will almost surely eventually reach some concept $D \equiv_{\mathcal{O}} \top$, as required.

Now let us consider a generalisation chain $D = D_0 \sqsubseteq_{\mathcal{O}} D_1 \sqsubseteq_{\mathcal{O}} D_2 \ldots$, where as before every D_{i+1} is chosen randomly among $\gamma(D)$, starting from some concept $D \equiv_{\mathcal{O}} \top$. Now, since $D \equiv_{\mathcal{O}} \top$ we have $\top \in \mathsf{UpCov}_{\mathcal{O}}(D)$, and since $D \sqsubseteq_{\mathcal{O}} D_i$ for all i, $\top \in \mathsf{UpCov}_{\mathcal{O}}(D_i) \subseteq \gamma_{\mathcal{O}}(D_i)$ for all i. 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 $\epsilon' = 1/3kq$. □

Note that, by our definitions, $\top$ can be further generalized to all elements of its upcover (that is, all concepts of $\mathsf{sub}(\mathcal{O})$ which are equivalent to $\top$ with respect to $\mathcal{O}$), and similarly $\bot$ 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 $\bot$.

Corollary 1. *Algorithm 1 almost surely terminates.*

Proof. As long as the ontology $\mathcal{O}$ is inconsistent, Algorithm 1 will select one axiom that appears in some minimally inconsistent subset of atoms and attempt to weaken it. Since the ontology $\mathcal{O}$ contains a finite number of axioms, if the algorithm never terminates then at least one of these axioms must be weakened an infinite number of times without being ever turned into the trivial axiom $\bot \sqsubseteq \top$, or an axiom $\top(a)$ (or a trivial set of axioms $\{\top(a), \top(b)\}$). But this is impossible because of Theorem 2. □

It is worth remarking that this proof of almost-sure termination does not imply anything about the *expected time* of Algorithm 1 as a function of ontology size, not even that this expected time is finite. Indeed, note that it is possible for a randomised algorithm to almost surely terminate in finite time and yet have an infinite expected runtime.[4]

$\ell - \epsilon) - \log(i + \ell)) \leq \sum_{i=0}^{\infty}(\log(i + \lceil\ell\rceil - \epsilon) - \log(i + \lceil\ell\rceil)) = \sum_{i=\lceil\ell\rceil}^{\infty}(\log(i - \epsilon) - \log(i))$, 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.

[4] For example, suppose that the algorithm terminates in exactly n steps with probability $6/\pi^2 \cdot n^{-2}$. Using the fact that $\sum_{n=1}^{\infty} n^{-2} = \pi^2/6$, we have at once that the algorithm terminates in finite time with probability 1. However, the expectation of its runtime would be $6/\pi^2 \sum_{n=1}^{\infty} n^{-1}$, which diverges to infinity.

Tighter upper bounds than those of Lemma 3 may allow us to make such an estimate; we leave this question to future work.

5 Length of refinement chains and tractability in practice

We showed that the iterated weakening of concepts almost surely reaches $\top$ and that Algorithm 1 almost surely terminates. But this does not tell the whole story and we next discuss this from a more practical perspective.

The study of [19] provides empirical evidence that axiom weakening is significantly better than axiom removal for ontology repair (gentle vs. coarse repair). Here instead, we focus on the experimental evaluation of the almost-sure termination of our algorithm. For simplicity of exposition we focus on the evaluation of the problem of reaching the top concept by the iterated weakening of a concept. (Axiom weakening and ontology repair tasks are barely more than many repeated iterated weakenings.) The experiments exploit our implementation[5] of the refinement operator w.r.t. various reference (consistent) ontologies.

Ontologies. To better understand the practical aspects of our refinement operators, we performed experiments on real-world ontologies from the Gene Ontology knowledge base[6], and also using synthetic randomly generated ontologies.[7] About the latter, an ontology named C[num-c]_R[num-r]_[pconnect-ratio]_[cconnect-ratio]_[existconnect-ratio] is an $\mathcal{ALC}$ ontology, with a signature of [num-c] atomic concepts and [num-r] roles. Given two atomic concepts C_1 and C_2, some subset relations $C_1 \sqsubseteq C_2$ and $C_1 \sqsubseteq \neg C_2$ are randomly generated with roughly the probability [pconnect-ratio] and [cconnect-ratio], respectively. Given two atomic concepts C_1 and C_2, and a role R, subset relations $C_1 \sqsubseteq \exists R.C_2$ and $C_1 \sqsubseteq \neg\exists R.C_2$ are generated with probability roughly [existconnect-ratio]. All atomic classes are populated with one individual. For a set of parameters, we generated ontologies until a consistent one was found. Finally, we ran the experiments also on a hand-crafted ontology. a-and-b is a very small manually designed ontology.

$$A \sqsubseteq A \sqcap B$$
$$B \sqsubseteq A \sqcap B$$
$$A \sqsubseteq \exists R.(A \sqcap B)$$

Clearly, A, B, and $A \sqcap B$ are logically equivalent. It will help to put light on the limitations of the syntactic approach. It also serves to illustrate that despite the likelihood to obtain very large concepts during the iterated refinement of a concept, the process robustly terminates.

Setting and results. For each ontology, we ran a series of refinements. In each case, we ran 1,000 times the iterated random weakening of the concept $\bot$ until

[5] The implementation is available at https://bitbucket.org/troquard/ontologyutils.
[6] http://geneontology.org/docs/download-ontology/
[7] These ontologies can be found in the directory ontologyutils/src/master/resources/Random/ of the implementation.

ontology	axioms	$\|N_C\|$	$\|N_R\|$	min	max	average	median
goslim_mouse	13	44	9	2	14	4.948	5
goslim_plant	49	174	9	3	18	6.385	6
goslim_generic	72	143	9	2	17	5.170	5
goslim_drosophila	160	97	9	2	15	4.628	4
goslim_metagenomics	172	114	9	3	26	6.541	6
goslim_yeast	266	164	9	3	18	5.325	5
goslim_pir	670	514	9	3	22	7.422	7
C50_R10_0.001_0.001_0.001	164	51	10	2	17	4.648	4
C100_R10_0.001_0.001_0.001	413	101	10	2	17	4.925	4
C150_R10_0.001_0.001_0.001	803	151	10	2	17	5.043	5
C300_R10_0.0001_0.0001_0.0001	811	301	10	2	14	4.057	4
a-and-b	3	2	1	3	72	16.142	14

Table 2. Characteristics of ontologies, and sizes of the chains of generalisation over 1,000 runs of iterated generalisations from $\bot$ to $\top$. (Axioms, $|N_C|$ and $|N_R|$ are the number of logical axioms count, class count, and object property count as given by the metrics of the ontology in Protégé [16].)

the concept $\top$ was reached. In each case, we recorded the minimum length of the chain of refinements, the maximum length of the chain of refinements, the average length, and the median length. The results are reported in Table 2, alongside the characteristics of the ontologies (number of logical axioms, number of atomic concepts, and number of atomic roles) used in the experiments.

The case of a-and-b. The ontology is specifically written to trip the almost surely terminating iterated refinement procedure with our operator. When trying to reach $\top$ from $\bot$ in the ontology a-and-b by iterated weakening, first $\top$ is weakened into $\top$, A, B, or $A \sqcap B$. Then at each step, an instance of A or B may be replaced with A, B, or $A \sqcap B$, or with $\exists R.(A \sqcap B)$; and $\exists R.(\ldots)$ can be weakened, possibly replacing the instances A or B as before, or into $\top$.

During the iterated refinement of $\bot$ with the weakening operator, we observed (min, max, average, median): $(3, 42, 17.05, 15)$ with 100 runs, $(3, 72, 16.142, 14)$ with 1,000 runs, $(3, 76, 16.0958, 14)$ with 10,000 runs, $(3, 92, 16.01316, 14)$ with 100,000 runs. We see that long chains of refinements occur, but the procedure robustly terminates.

6 Outlook

We presented a set of refinement operators for $\mathcal{ALC}$, proving the almost-certain termination of their iterated application and verifying their practical tractability via experimental evaluation. Further additions to the general rules of refinements need to be studied, e.g., to deal with more general logics such as $\mathcal{SROIQ}$, as was initiated in [7]. Further directions of research include the study of high-level heuristics and closure conditions (e.g., for $\mathsf{sub}(\mathcal{O})$) to steer, enrich, and accelerate the refinement process.

References

1. Baader, F., Calvanese, D., McGuinness, D.L., Nardi, D., Patel-Schneider, P.F. (eds.): The Description Logic Handbook: Theory, Implementation, and Applications. Cambridge University Press, New York, NY, USA (2003)
2. Baader, F., Kriegel, F., Nuradiansyah, A., Peñaloza, R.: Making repairs in description logics more gentle. In: Proc. of KR 2018. pp. 319–328 (2018)
3. Baader, F., Küsters, R.: Nonstandard inferences in description logics: The story so far. In: Mathematical Problems from Applied Logic I: Logics for the XXIst Century. pp. 1–75. Springer (2006)
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.) Proc. of 10th ILP Conference. LNCS, vol. 1866, pp. 40–59. Springer (2000)
6. 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)
7. Confalonieri, R., Galliani, P., Kutz, O., Porello, D., Righetti, G., Troquard, N.: Towards even more irresistible axiom weakening. In: Borgwardt, S., Meyer, T. (eds.) Proceedings of the 33rd International Workshop on Description Logics (DL 2020) co-located with the 17th International Conference on Principles of Knowledge Representation and Reasoning (KR 2020), Online Event [Rhodes, Greece], September 12th to 14th, 2020. CEUR Workshop Proceedings, vol. 2663. CEUR-WS.org (2020), http://ceur-ws.org/Vol-2663/paper-8.pdf
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. Kalyanpur, A., Parsia, B., Sirin, E., Grau, B.C.: Repairing unsatisfiable concepts in OWL ontologies. In: ESWC. vol. 6, pp. 170–184. Springer (2006)
10. 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)
11. 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)
12. Lehmann, J., Hitzler, P.: Foundations of refinement operators for description logics. In: Proc. of the 17th ILP Conference. pp. 161–174 (2007)
13. Lehmann, J., Hitzler, P.: A refinement operator based learning algorithm for the ALC description logic. In: Proc. of the 17th ILP Conference. pp. 147–160 (2007)
14. Lehmann, J., Hitzler, P.: Concept learning in description logics using refinement operators. Machine Learning **78**(1-2), 203–250 (2010)
15. 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)

16. Musen, M.A.: The Protégé Project: A Look Back and a Look Forward. AI Matters **1**(4), 4–12 (2015)
17. 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.) Proc. of the 27th International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden. pp. 1942–1948 (2018)
18. 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)
19. 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). pp. 1981–1988. AAAI Press (2018)