Repairing $\mathcal{EL}$ TBoxes by Means of Countermodels Obtained by Model Transformation*

Willi Hieke, Francesco Kriegel, and Adrian Nuradiansyah

Theoretical Computer Science, Technische Universität Dresden, Dresden, Germany
`firstname.lastname@tu-dresden.de`

Abstract. Knowledge engineers might face situations in which an unwanted consequence is derivable from an ontology. It is then desired to revise the ontology such that it no longer entails the consequence. For this purpose, we introduce a novel technique for repairing TBoxes formulated in the description logic $\mathcal{EL}$. Specifically, we first compute a canonical model of the TBox and then transform it into a countermodel to the unwanted consequence. As formalism for the model transformation we employ transductions. We then obtain a TBox repair as the axiomatization of the logical intersection of the original TBox and the theory of the countermodel. In fact, we construct a set of countermodels, each of which induces a TBox repair. For the actual computation of the repairs we use results from Formal Concept Analysis.

Keywords: Description logic · TBox repair · Countermodel · Model transformation · Canonical model · Logical intersection · Canonical base

1 Introduction

Description logics (DLs) are a family of logic-based knowledge representation languages, supporting terminological knowledge (a schema) as well as assertional knowledge (the data). Common reasoning services allow for deducing implicit consequences that logically follow from the explicitly stated knowledge. Sometimes, we encounter situations where the derivation of such a consequence must be made impossible — either because it is invalid in the underlying domain of interest, or since it is privacy-sensitive information that needs to be hidden. The classical approach to repairing a knowledge base in such situations is to remove axioms such that the remaining axioms do not entail the unwanted consequence anymore, i.e., a classical repair is a subset of the given knowledge base. As a downside, also other consequences might vanish that are actually wanted.

* Willi Hieke is supported by DFG Research Training Group 1763 (QuantLA), and Francesco Kriegel and Adrian Nuradiansyah are funded by DFG in project number 430150274.

Specifically, the computation of classical repairs is based on *justifications*, which are minimal subsets of the ontology that entail the unwanted consequence — as pointed out in [20], a repair can be obtained by deleting one axiom of each justification. This classical approach was first used to repair inconsistent ontologies [21, 22], which was later extended to a more expressive DL $\mathcal{SHOIN}$ in [16].

More fine-grained repairs — in the sense that not too many other consequences are affected — can be obtained if axioms are weakened instead of removed completely [3, 10, 14, 18, 23]. A repair obtained this way is not a subset of the given knowledge base but is logically entailed by it. While the approaches in [10, 14] first apply syntactic structural transformations to replace the axioms in an ontology with a set of weaker axioms before the modified ontology is repaired using the classical approach, the approaches in [18, 23] directly weaken those axioms that are responsible for the entailment of the unwanted consequence. In [3] a general framework for constructing so-called *gentle repairs* based on axiom weakening was developed, which can in principle be applied to every monotonic logic. Furthermore, conditions on the weakening relations are formulated that guarantee termination, and an instantiation of the framework for $\mathcal{EL}$ is provided.

However, weakening axioms is not the only way to obtain such a non-classical repair. In this document, we specifically consider the problem of repairing a TBox with respect to a given concept inclusion (the unwanted consequence), all within the light-weight description logic $\mathcal{EL}$ for which most common reasoning tasks can be solved in polynomial time [1, 2, 6, 7]. Our approach first constructs a countermodel to the unwanted consequence, and then produces a repair as the axiomatization of the logical intersection of the given TBox and the theory of the countermodel. Such a countermodel is simply an interpretation that contains an element being an instance of the premise but not of the conclusion of the concept inclusion to be removed.

To describe the construction of suitable countermodels, we utilize the formalism of transductions [8]. Such a transduction specifies how an input interpretation is transformed into the output interpretations, e.g., by means of logical formulae. Specifically, we adapt the idea that underlies an approach to computing (optimal) ABox repairs [4] in order to transform a canonical model into a countermodel. In order to actually compute a repair, we employ the results on axiomatizing concept inclusions from closure operators [17]. In particular, an axiomatization of the logical intersection is obtained as the canonical base of the infimum of the closure operator induced by the TBox to be repaired and the closure operator induced by a countermodel to the unwanted consequence.

The structure of this document is as follows. In the next Section 2 we briefly recall important notions of the description logic $\mathcal{EL}$. Section 3 describes transductions that can be utilized to construct countermodels, and Section 4 explains how the logical intersection (of a TBox and a countermodel) can be axiomatized. We close this document with some concluding remarks in Section 5.

2 Preliminaries

We presume familiarity with basic notions of the description logic $\mathcal{EL}$. Given a *signature* $\Sigma \coloneqq \Sigma_{\mathsf{C}} \cup \Sigma_{\mathsf{R}}$ consisting of concept names and role names, $\mathcal{EL}$ *concept descriptions* are built from Σ using the constructors $\top$, $\sqcap$, and $\exists$. Throughout this document we assume that all signatures are finite. (Nested) conjunctions are treated like sets, i.e., nestings, repetitions, and order are irrelevant. An *atom* is either a concept name or an existential restriction. Every concept description C is a conjunction of atoms, the *top-level conjunction* of C, and the set of all these atoms is denoted as $\mathsf{Conj}(C)$. Specifically, $\top$ is the empty conjunction. Given a concept C, we denote by $\mathsf{Sub}(C)$ the set of all subconcepts of C (including C itself) and $\mathsf{Atoms}(C)$ is the subset containing all atoms occurring as subconcepts in C. Likewise, $\mathsf{Sub}(\mathcal{T})$ consists of all subconcepts occurring in $\mathcal{T}$. We say that an interpretation $\mathcal{I}$ *satisfies* a concept inclusion (CI) $C \sqsubseteq D$, written $\mathcal{I} \models C \sqsubseteq D$, if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$. Given a concept inclusion $C \sqsubseteq D$ and a TBox $\mathcal{T}$, we say that $\mathcal{T}$ *entails* $C \sqsubseteq D$ and that C *is subsumed by* D w.r.t. $\mathcal{T}$, written $\mathcal{T} \models C \sqsubseteq D$ and $C \sqsubseteq_{\mathcal{T}} D$, respectively, if each model of $\mathcal{T}$ satisfies $C \sqsubseteq D$.

3 Constructing Countermodels by Model Transformation

Within this section, we are going to develop a method that produces countermodels to a concept inclusion entailed by a TBox. For this purpose, assume that $\mathcal{T}$ is a TBox and further that $C \sqsubseteq D$ is a concept inclusion, both formulated in the description logic $\mathcal{EL}$, such that $\mathcal{T}$ entails $C \sqsubseteq D$. We start with formally defining the notion of a countermodel.

Definition 1. *Let $C \sqsubseteq D$ be a concept inclusion. A* countermodel *to $C \sqsubseteq D$ is an interpretation that does not satisfy $C \sqsubseteq D$.*

Note that an interpretation $\mathcal{I}$ is a countermodel to $C \sqsubseteq D$ if and only if $\mathcal{I}$ contains a domain element d such that $d \in C^{\mathcal{I}}$ and $d \notin D^{\mathcal{I}}$. Of course, such a countermodel can only exist if $C \sqsubseteq D$ is *no tautology*, i.e., if it is not valid in all interpretations — we thus assume further that $C \sqsubseteq D$ satisfies this condition.

3.1 The Underlying Idea of the Countermodel Construction

As a starting point for constructing suitable countermodels we use the canonical model $\mathcal{I}_{C,\mathcal{T}}$ induced by the premise C and the TBox $\mathcal{T}$, because it contains an element d_C that is already an instance of C and, since it is a model of $\mathcal{T}$, this element d_C is also an instance of D. In order to obtain a countermodel we will modify $\mathcal{I}_{C,\mathcal{T}}$ in a way such that the distinguished element d_C is still an instance of C but not an instance of D anymore.

Definition 2. **[19]** *Let C be an $\mathcal{EL}$ concept and $\mathcal{T}$ be a TBox. The* canonical model *$\mathcal{I}_{C,\mathcal{T}}$ of C w.r.t. $\mathcal{T}$ is defined as follows:*

$$\Delta^{\mathcal{I}_{C,\mathcal{T}}} \coloneqq \{d_C\} \cup \{d_{C'} \mid \exists r.C' \in \mathsf{Sub}(C) \cup \mathsf{Sub}(\mathcal{T})\}$$

$$A^{\mathcal{I}_{C,\mathcal{T}}} := \{\, d_D \mid \mathcal{T} \models D \sqsubseteq A \,\}$$
$$r^{\mathcal{I}_{C,\mathcal{T}}} := \{\, (d_D, d_{D'}) \mid \mathcal{T} \models D \sqsubseteq \exists r.D' \text{ and } \exists r.D' \in \mathsf{Sub}(\mathcal{T}) \cup \mathsf{Conj}(D) \,\}$$

To appropriately modify the canonical model, we adopt a technique for repairing DL ABoxes that was introduced in [4] and aims at getting rid of unwanted information about individuals a represented as a concept assertion $P(a)$. If the given ABox $\mathcal{A}$ entails this assertion, then one can compute an ABox entailed by $\mathcal{A}$ from which $P(a)$ is no longer derivable. This computation is based on the negation of a recursive characterization of the instance problem (see Lemma 9 of [4]): $\mathcal{A}$ does not entail $P(a)$ if and only if there is an atom A or $\exists r.E$ occurring as a top-level conjunct of P such that $A(a) \notin \mathcal{A}$ or each r-successor of a is no instance of E w.r.t. $\mathcal{A}$, respectively. It follows that removing $P(a)$ from the consequences of $\mathcal{A}$ can be done by simply choosing a top-level conjunct of P and then, if the top-level conjunct is a concept name A, removing $A(a)$ from $\mathcal{A}$ or, if the top-level conjunct is an existential restriction $\exists r.E$, recursively modifying the role successors of a such that none of them is an instance of E anymore.

To minimize the amount of information lost by such a repairing process, this technique does not only remove assertions from $\mathcal{A}$, but also splits objects by introducing copies of them, which are created based on sets of atoms occurring in P. In particular, for each object u from the input ABox, exponentially many copies $y_{u,\mathcal{K}}$ are introduced, where each $\mathcal{K}$ is a subset of $\mathsf{Atoms}(P)$. These copies are then used as objects in the ABox repair, and the assertions in the repair are computed in a way such that each $y_{u,\mathcal{K}}$ is not an instance of every atom in $\mathcal{K}$.

In order to adapt the aforementioned repair technique to an approach to constructing countermodels, we view the canonical model $\mathcal{I}_{C,\mathcal{T}}$ as the ABox $\mathcal{A}$, the right-hand side D of the concept inclusion $C \sqsubseteq D$ as the concept P, and the distinguished element d_C as the individual name a. In particular, each domain element of the countermodel will be a copy of a domain element of $\mathcal{I}_{C,\mathcal{T}}$ indexed by a set $\mathcal{K} \subseteq \mathsf{Atoms}(D)$, and the interpretation function is defined in a way such that each copy is not in the extension of each atom in $\mathcal{K}$. We use the formalism of transductions for precisely describing the transformation of the canonical model.

As preparation, we first define a transduction in its most basic setting.

Definition 3. [8] *A* transduction τ *is a binary relation on interpretations. The* image *of an interpretation $\mathcal{I}$ under τ is the set $\tau(\mathcal{I}) := \{\, \mathcal{J} \mid (\mathcal{I}, \mathcal{J}) \in \tau \,\}$. If τ is functional, then we instead identify $\tau(\mathcal{I})$ with the unique $\mathcal{J}$ where $(\mathcal{I}, \mathcal{J}) \in \tau$.*

Informally, the image $\tau(\mathcal{I})$ consists of all interpretations obtained by transforming $\mathcal{I}$ according to τ. There are several classes of transductions; most prominently the *monadic second-order (MSO) transductions*, which are *copying transductions* that can be described by a set of MSO formulae. A transduction τ is called *copying* if there is an *index set* $\mathbb{I}$ such that, for each interpretation $\mathcal{I}$ and for each $\mathcal{J} \in \tau(\mathcal{I})$, the domain $\Delta^{\mathcal{J}}$ consists of copies of elements from the domain $\Delta^{\mathcal{I}}$ that are indexed with $\mathbb{I}$, i.e., $\Delta^{\mathcal{J}} \subseteq \{\, d_i \mid d \in \Delta^{\mathcal{I}} \text{ and } i \in \mathbb{I} \,\}$.

We now introduce a functional transduction $\tau_{\mathsf{repair},D}$ that resembles the transformation of an ABox into compliant anonymizations as in Definition 11 of [4], which is briefly described above.

Definition 4. *For each interpretation $\mathcal{I}$, we define $\tau_{\mathsf{repair},D}(\mathcal{I})$ as follows:*

$$\begin{aligned}\Delta^{\tau_{\mathsf{repair},D}(\mathcal{I})} &:= \{d_{\mathcal{K}} \mid d \in \Delta^{\mathcal{I}},\ \mathcal{K} \subseteq \mathsf{Atoms}(D),\ d \in F^{\mathcal{I}} \text{ for each } F \in \mathcal{K}, \\ &\qquad \text{and } \mathcal{K} \text{ does not contain } \sqsubseteq_{\emptyset}\text{-comparable atoms}\} \\ A^{\tau_{\mathsf{repair},D}(\mathcal{I})} &:= \{d_{\mathcal{K}} \mid d \in A^{\mathcal{I}} \text{ and } A \notin \mathcal{K}\} \\ r^{\tau_{\mathsf{repair},D}(\mathcal{I})} &:= \{(d_{\mathcal{K}}, e_{\mathcal{L}}) \mid (d,e) \in r^{\mathcal{I}} \text{ and for each } \exists r.Q \in \mathcal{K} \text{ with } e \in Q^{\mathcal{I}}, \\ &\qquad \text{there is } F \in \mathcal{L} \text{ such that } Q \sqsubseteq_{\emptyset} F\}\end{aligned}$$

In particular, $\tau_{\mathsf{repair},D}$ is a copying transduction with index set $\mathfrak{P}(\mathsf{Atoms}(D))$.

Example 5. Consider the TBox $\mathcal{T} = \{A \sqsubseteq A_1, B \sqsubseteq B_1 \sqcap B_2\}$ and the concept inclusion $C \sqsubseteq D := A \sqcap \exists r.B \sqsubseteq A_1 \sqcap \exists r.(B_1 \sqcap B_2)$, which is entailed by $\mathcal{T}$ but is no tautology. The canonical model $\mathcal{I}_{C,\mathcal{T}}$ of C w.r.t. $\mathcal{T}$ is illustrated in Figure 1.

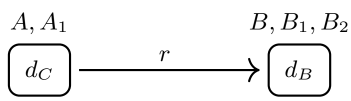

Fig. 1. The canonical model

Applying the transduction $\tau_{\mathsf{repair},D}$ to the canonical model $\mathcal{I}_{C,\mathcal{T}}$ yields a countermodel to $C \sqsubseteq D$ that is depicted in Figure 2. The gray nodes represent the elements that are an instance of C but not an instance of D.

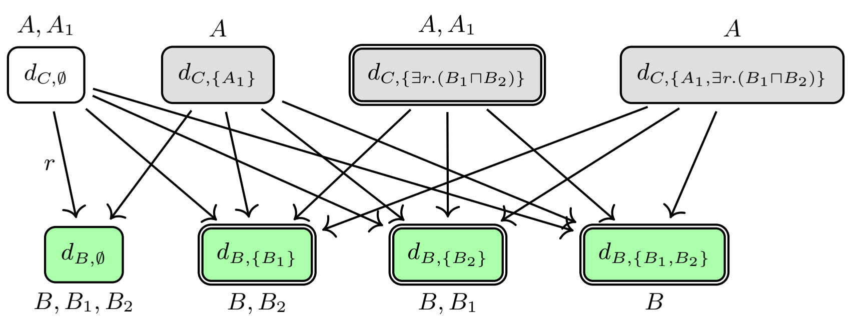

Fig. 2. The countermodel obtained by applying $\tau_{\mathsf{repair},D}$ to the canonical model

Proposition 6. *For each $\mathcal{EL}$ TBox $\mathcal{T}$ and each non-tautological concept inclusion $C \sqsubseteq D$ entailed by $\mathcal{T}$, the interpretation $\tau_{\mathsf{repair},D}(\mathcal{I}_{C,\mathcal{T}})$ is a countermodel to $C \sqsubseteq D$.*

Proof. It is a finger exercise to adapt Lemmas 13 and 18 in [4] in order to obtain the following important property of $\tau_{\mathsf{repair},D}(\mathcal{I})$: for each copy $d_{\mathcal{K}} \in \Delta^{\tau_{\mathsf{repair},D}(\mathcal{I})}$ and for each atom $F \in \mathsf{Atoms}(D)$, we have $d_{\mathcal{K}} \in F^{\tau_{\mathsf{repair},D}(\mathcal{I})}$ if and only if $d \in F^{\mathcal{I}}$ and $\mathcal{K}$ does not contain an atom subsuming F. Since $C \sqsubseteq D$ is no tautology, there is a top-level conjunct E of D such that $C \not\sqsubseteq_{\emptyset} E$. It follows that applying $\tau_{\mathsf{repair},D}$ to the canonical model $\mathcal{I}_{C,\mathcal{T}}$ yields an interpretation in which $d_{C,\{E\}}$[1] is still an instance of C but not an instance of D anymore, i.e., $\tau_{\mathsf{repair},D}$ indeed transforms the canonical model into a countermodel to $C \sqsubseteq D$. □

3.2 Obtaining Countermodels Using MSO Transductions

As seen above, transductions can be defined using a *mathematical meta-language* — it is only necessary to describe all interpretations in the image, including their domains and their interpretation functions. A more formal approach to describing transductions is by using well-formed logical formulae. In the following, we introduce monadic second-order (MSO) transductions as formalism to describe model transformations, which is generally defined in [8], and tailored to description logic interpretations in [12]. Before introducing the formal definitions, we will explain the idea of a transduction.

An MSO transduction maps an input interpretation to output interpretations according to a tuple of MSO formulae called *definition scheme*. A definition scheme consists of formulae that define the domain, and the concept and role extensions. The *domain formula* δ of a definition scheme contains one free variable — and every element of the input interpretation that satisfies this formula is then an element of the output interpretation. The concept and role extensions of the output interpretation are defined accordingly by *concept formulae* θ and *role formulae* η — there is one concept formula for each concept symbol and role formula for each role symbol of the signature. For copying transductions, a definition scheme contains families of these formulae for the elements of the respective (finite) index set.

By $\mathrm{MSO}(\Sigma, \mathcal{W})$, we denote the set of MSO formulae built over a signature Σ and a finite set $\mathcal{W}$ of first-order and set variables, which are free in the respective formula. For a detailed introduction to first-order logic (FOL) and MSO, we refer the reader to [11].

Definition 7. [12] *Let Σ be a signature, let $\mathcal{W}$ be a finite set of first-order or monadic second-order variables called* parameters[2]*, and let $\mathbb{I}$ be a finite set called* index set. *A* monadic second-order definition scheme *is a tuple*

$$\mathsf{D} = (\chi, (\delta_i)_{i \in \mathbb{I}}, (\theta_{A,i})_{(A,i) \in \Sigma_{\mathsf{C}} \times \mathbb{I}}, (\eta_{r,i,j})_{(r,i,j) \in \Sigma_{\mathsf{R}} \times \mathbb{I} \times \mathbb{I}})$$

consisting of
- *a* precondition formula $\chi \in \mathrm{MSO}(\Sigma, \mathcal{W})$,

[1] We write $d_{C,\mathcal{K}}$ instead of $(d_C)_{\mathcal{K}}$.
[2] In general, MSO transductions allow for monadic second-order variables (often called set variables). However, we will only make use of first-order variables as parameters.

– domain formulae $\delta_i \in \text{MSO}(\Sigma, \mathcal{W} \cup \{x\})$ *for each* $i \in \mathbb{I}$,
– concept formulae $\theta_{A,i} \in \text{MSO}(\Sigma, \mathcal{W} \cup \{x\})$ *for each* $(A, i) \in \Sigma_\mathsf{C} \times \mathbb{I}$, *and*
– role formulae $\eta_{r,i,j} \in \text{MSO}(\Sigma, \mathcal{W} \cup \{x, y\})$ *for each* $(r, i, j) \in \Sigma_\mathsf{R} \times \mathbb{I} \times \mathbb{I}$.

Definition 8. [12] *Consider a definition scheme* D *with index set* $\mathbb{I}$ *and parameter set* $\mathcal{W}$. *Further let* $\mathcal{I}$ *be an interpretation and let* λ *be a* $\mathcal{W}$*-assignment in* $\mathcal{I}$. *If* $(\mathcal{I}, \lambda) \models \chi$, *then* D defines *the interpretation* $\hat{\mathsf{D}}(\mathcal{I}, \lambda)$ *as follows:*

$$\begin{aligned} \Delta^{\hat{\mathsf{D}}(\mathcal{I},\lambda)} &\coloneqq \{d_i \mid d \in \Delta^{\mathcal{I}},\ i \in \mathbb{I},\ \text{and } (\mathcal{I}, \lambda) \models \delta_i(d)\} \\ A^{\hat{\mathsf{D}}(\mathcal{I},\lambda)} &\coloneqq \{d_i \mid (\mathcal{I}, \lambda) \models \theta_{A,i}(d)\} \\ r^{\hat{\mathsf{D}}(\mathcal{I},\lambda)} &\coloneqq \{(d_i, e_j) \mid (\mathcal{I}, \lambda) \models \eta_{r,i,j}(d, e)\} \end{aligned}$$

$(\mathcal{I}, \lambda) \models \delta_i(d)$ *means* $(\mathcal{I}, \lambda') \models \delta_i(x)$ *where* λ' *extends* λ *by* $\lambda'(x) \coloneqq d$ *(and accordingly for* θ *and* η*).*[3] *The transduction* τ_D *induced by* D *is defined as*

$$\tau_\mathsf{D} \coloneqq \{(\mathcal{I}, \hat{\mathsf{D}}(\mathcal{I}, \lambda)) \mid \lambda \text{ is a } \mathcal{W}\text{-assignment in } \mathcal{I} \text{ with } (\mathcal{I}, \lambda) \models \chi\}.$$

As already mentioned, we consider MSO transductions as means of model transformation to ensure that our model transformations are well defined, and to only use logical notions. Another important benefit is that MSO transductions are always computable since MSO model checking is decidable — in fact, it is PSpace-complete [24]. However, as a side remark, we will point out a weakness of MSO transductions as formalism for model transformations.

In order to define an MSO transduction that is equivalent to $\tau_{\mathsf{repair},D}$, we need to devise a definition scheme $\mathsf{D}_{\mathsf{repair},D}$ such that its induced transduction $\tau_{\mathsf{D}_{\mathsf{repair},D}}$ equals $\tau_{\mathsf{repair},D}$. We have already seen above that the index set $\mathbb{I} = \mathfrak{P}(\mathsf{Atoms}(D))$ is used. Further recall that $\tau_{\mathsf{repair},D}$ needs to check the condition "for each $\exists r.Q \in \mathcal{K}$ with $e \in Q^{\mathcal{I}}$, there is $F \in \mathcal{L}$ such that $Q \sqsubseteq_\emptyset F$" during the creation of the interpretation function. In particular, it must be determined whether a concept inclusion is a tautology, i.e., is satisfied in *all* interpretations. One might be tempted to translate this condition into a conjunct of the formulae $\eta_{r,\mathcal{K},\mathcal{L}}$ as follows, where we denote by $C^\#$ the FOL-translation of a concept C.

$$\eta_{r,\mathcal{K},\mathcal{L}}(x, y) \coloneqq r(x, y) \wedge \bigwedge_{\exists r.Q \in \mathcal{K}} \Big(Q^\#(y) \to \bigvee_{F \in \mathcal{L}} \forall z : Q^\#(z) \to F^\#(z) \Big)$$

However, during the computation of the image of an interpretation $\mathcal{I}$ it will only be checked whether this conjunct is satisfied in $\mathcal{I}$ (and not in *all* interpretations).

There are two ways to resolve this issue. Firstly, instead of defining the formula $\eta_{r,\mathcal{K},\mathcal{L}}$ directly, we could provide a construction specification in which we externalize the problematic condition, e.g., as follows.

$$\eta_{r,\mathcal{K},\mathcal{L}}(x, y) \coloneqq \begin{cases} r(x, y) & \text{if, for each } \exists r.Q \in \mathcal{K} \text{ with } y \in Q^{\mathcal{I}}, \\ & \text{there is an atom } F \in \mathcal{L} \text{ s.t. } Q \sqsubseteq_\emptyset F \\ \bot & \text{otherwise} \end{cases}$$

[3] We will later write $\delta(\mathcal{W}, x)$ instead of $\delta(z_1, \ldots, z_n, x)$ for $\mathcal{W} = \{z_1, \ldots, z_n\}$.

Secondly, we could generalize the formalism of MSO transductions in that the formulae of the definition scheme are not evaluated in the input interpretation $\mathcal{I}$ but instead it is checked whether they are tautologies. If some subformulae must be evaluated in $\mathcal{I}$, we could simply precede them with "$\mathcal{I}^{\#} \rightarrow$" where $\mathcal{I}^{\#}$ is the FOL-translation of $\mathcal{I}$, i.e., $\mathcal{I}^{\#} \coloneqq \bigwedge_{d \in A^{\mathcal{I}}} A(d) \wedge \bigwedge_{(d,e) \in r^{\mathcal{I}}} r(d,e)$. With that modification, we are then able to specify the role formulae as

$$\eta_{r,\mathcal{K},\mathcal{L}}(x,y) \coloneqq (\mathcal{I}^{\#} \rightarrow r(x,y)) \wedge \bigwedge_{\exists r.Q \in \mathcal{K}} \Big((\mathcal{I}^{\#} \rightarrow Q^{\#}(y)) \rightarrow \bigvee_{F \in \mathcal{L}} (\forall z \colon Q^{\#}(z) \rightarrow F^{\#}(z)) \Big).$$

Going with the second solution, however, leads to the effect that MSO transductions are not necessarily computable anymore because MSO is generally not decidable. If we were to restrict the conditions to decidable logics, then computability of transductions is ensured, but becomes less expressive.

Having the interpretation $\tau_{\mathsf{repair},D}(\mathcal{I}_{C,\mathcal{T}})$ computed, there are possibly multiple nodes that satisfy concept C and not concept D. Each of these nodes together with their reachable elements are countermodels to the unwanted consequence of the TBox. We define a transduction that cuts out reachable parts for every node that is in C and not in D. The result is a set of countermodels.

We are making use of an auxiliary predicate for reachability of two elements in an interpretation $\mathcal{I}$, denoted by $reach(x,y)$. An edge between two elements, expressed by $succ(x,y)$, is defined as $\bigvee_{r \in \Sigma_{\mathsf{R}}} r(x,y)$ for a given signature Σ.

$$reach(x,y) \coloneqq \forall X : x \in X \wedge (\forall x, y : x \in X \wedge succ(x,y) \rightarrow y \in X) \rightarrow y \in X$$

The formula $reach(x,y)$ is true for two elements in a given interpretation $\mathcal{I}$ if and only if they belong to the reflexive and transitive closure of the *succ* relation in the interpretation $\mathcal{I}$ [8].

The transduction τ_{reach} defined below is used to extract a set of countermodels from interpretations $\tau_{\mathsf{repair},D}(\mathcal{I})$. For this purpose, we use first-order parameter variables and no index set (non-copying). Recall that, for every variable assignment λ such that $(\mathcal{I},\lambda) \models \chi$, a transduction yields another output interpretation. Thus, we obtain a set of countermodels for each χ-satisfying assignment λ.

Definition 9. *Let Σ be a signature and let v be a first-order parameter in $\mathcal{W}$. The definition scheme $\mathsf{D}_{\mathsf{reach}}$ inducing the transduction τ_{reach} consists of the formulae:*

$$\begin{aligned} \chi(\mathcal{W}) &\coloneqq C^{\#}(v) \wedge \neg D^{\#}(v) \\ \delta(\mathcal{W},x) &\coloneqq reach(v,x) \\ \theta_A(\mathcal{W},x) &\coloneqq A(x) \text{ for each } A \in \Sigma_{\mathsf{C}} \\ \eta_r(\mathcal{W},x,y) &\coloneqq r(x,y) \text{ for each } r \in \Sigma_{\mathsf{R}}. \end{aligned}$$

Theorem 10. *Let $\mathcal{T}$ be an $\mathcal{EL}$ TBox, let $C \sqsubseteq D$ be an unwanted consequence of $\mathcal{T}$, and assume that $C \sqsubseteq D$ is no tautology. Then, $\tau_{\mathsf{reach}}(\tau_{\mathsf{repair},D}(\mathcal{I}_{C,\mathcal{T}}))$ is a set of countermodels to $C \sqsubseteq D$.*

Example 11. Coming back to Example 5, the application of first $\tau_{\mathsf{repair},D}$ and then τ_{reach} to the canonical model $\mathcal{I}_{C,\mathcal{T}}$ yields a set consisting of three countermodels to $A \sqcap \exists r.B \sqsubseteq A_1 \sqcap \exists r.(B_1 \sqcap B_2)$ with roots $d_{C,\{A_1\}}, d_{C,\{\exists r.(B_1 \sqcap B_2)\}}$, and $d_{C,\{A_1,\exists r.(B_1 \sqcap B_2)\}}$, respectively. For instance, in addition to the root element $d_{C,\{A_1\}}$, the first countermodel consists of the r-successors $d_{B,\emptyset}$, $d_{B,\{B_1\}}$, $d_{B,\{B_2\}}$, and $d_{B,\{B_1,B_2\}}$, each of which is coloured with green in Figure 2.

While the canonical model $\mathcal{I}_{C,\mathcal{T}}$ of a concept C w.r.t. a TBox $\mathcal{T}$ is computable in polynomial time [7], the computation of the image of $\mathcal{I}_{C,\mathcal{T}}$ under the transduction $\tau_{\mathsf{repair},D}$ needs exponential time [4]. The computationally hardest task to check in transduction τ_{reach} is the reachability predicate — and this problem is NL-complete [15], which leads to the following proposition.

Proposition 12. *The set of countermodels $\tau_{\mathsf{reach}}(\tau_{\mathsf{repair},D}(\mathcal{I}_{C,\mathcal{T}}))$ is computable in exponential time.*

4 Constructing Repairs by Axiomatizing the Logical Intersection

The goal of this section is to explain how a countermodel can be used for computing a repair. Specifically, we only consider the case of repairing an $\mathcal{EL}$ TBox $\mathcal{T}$ for an unwanted $\mathcal{EL}$ concept inclusion $C \sqsubseteq D$. It is irrelevant for our purposes for what reason the concept inclusion is unwanted — for instance, it might be an erroneous consequence, or it might be sensitive information that needs to be hidden. We start with the formal definition of a repair, which is Definition 1 in [3] customized to our setting.

Definition 13. *Consider a TBox $\mathcal{T}$ that entails a concept inclusion $C \sqsubseteq D$. A* repair *of $\mathcal{T}$ for $C \sqsubseteq D$ is a TBox that is entailed by $\mathcal{T}$ and that does not entail $C \sqsubseteq D$.*

In contrast to classical repairs, we not only consider subsets of $\mathcal{T}$ as repair candidates, but arbitrary TBoxes entailed by $\mathcal{T}$. This leaves much more room for fine-grained repairs.

Now assume that $\mathcal{J}$ is a countermodel to the unwanted consequence $C \sqsubseteq D$, e.g., one that is constructed from the canonical model $\mathcal{I}_{C,\mathcal{T}}$ according to Theorem 10 or, alternatively, one that has been built by an expert in the domain that underlies $\mathcal{T}$. In order to construct a repair, the idea is to axiomatize the logical intersection of the TBox $\mathcal{T}$ and the countermodel $\mathcal{J}$. This *logical intersection* $\mathcal{T} \vartriangle \mathcal{J}$ consists of all concept inclusions that are both entailed by $\mathcal{T}$ and satisfied by $\mathcal{J}$. It immediately follows that $\mathcal{T}$ entails $\mathcal{T} \vartriangle \mathcal{J}$ and further that $\mathcal{T} \vartriangle \mathcal{J}$ does not entail $C \sqsubseteq D$. The only reason that prevents us from directly using $\mathcal{T} \vartriangle \mathcal{J}$ as a repair is, in general, its infinite size. To overcome this obstacle, we need to axiomatize it with only finitely many concept inclusions. We solve the axiomatization task by means of the results in Section 6 in [17], which we will only briefly describe in the following due to a lack of space.

First of all, we introduce a common abstraction of the two notions of TBoxes and interpretations, namely both induce so-called closure operators. Such a *closure operator (clop)* ϕ maps each concept description E to a concept description E^ϕ such that $E^\phi \sqsubseteq_\emptyset E$ (*extensive*), $E \sqsubseteq_\emptyset F$ implies $E^\phi \sqsubseteq_\emptyset F^\phi$ (*monotone*), and $(E^\phi)^\phi \equiv_\emptyset E^\phi$ (*idempotent*). We say that a concept inclusion $E \sqsubseteq F$ is *valid* for ϕ if the closure E^ϕ is subsumed by F, i.e., $E^\phi \sqsubseteq_\emptyset F$.

As described in Section 4.3 in [17], the TBox $\mathcal{T}$ induces the clop $\phi_\mathcal{T}$ that maps each concept description E to its *most specific consequence* w.r.t. $\mathcal{T}$, which means that the closure $E^{\phi_\mathcal{T}}$ satisfies $E \sqsubseteq_\mathcal{T} E^{\phi_\mathcal{T}}$ and $E^{\phi_\mathcal{T}} \sqsubseteq_\emptyset F$ for each concept F where $E \sqsubseteq_\mathcal{T} F$. Put simply, the closure $E^{\phi_\mathcal{T}}$ can be computed by saturating the concept E with the concept inclusions in $\mathcal{T}$ or by unravelling the canonical model $\mathcal{I}_{E,\mathcal{T}}$ into a concept. The important property of this clop is that a concept inclusion is entailed by $\mathcal{T}$ if and only if it is valid for $\phi_\mathcal{T}$.

Example 14. Consider the concept $C = A \sqcap \exists r.B$ and the TBox $\mathcal{T} = \{A \sqsubseteq A_1, B \sqsubseteq B_1 \sqcap B_2\}$ from Example 5. The most specific consequence of C w.r.t. $\mathcal{T}$ is the concept $C^{\phi_\mathcal{T}} = A \sqcap A_1 \sqcap \exists r.(B \sqcap B_1 \sqcap B_2)$. It is subsumed by the concept $D = A_1 \sqcap \exists r.(B_1 \sqcap B_2)$ in Example 5, i.e., the concept inclusion $C \sqsubseteq D$ is valid for the induced closure operator $\phi_\mathcal{T}$.

Furthermore, as shown in [9] (or in Section 4.1 in [17]), the interpretation $\mathcal{J}$ induces the clop $\phi_\mathcal{J}$. Given a concept description E, it is first mapped to the extension $E^\mathcal{J}$, and then the closure $E^{\phi_\mathcal{J}}$ is obtained as the model-based most specific concept of $E^\mathcal{J}$ w.r.t. $\mathcal{J}$. Formally, the *model-based most specific concept* of a subset $X \subseteq \Delta^\mathcal{J}$ is a concept F such that $X \subseteq F^\mathcal{J}$ and $F \sqsubseteq_\emptyset G$ for each concept G where $X \subseteq G^\mathcal{J}$. It can be obtained by first constructing the $|X|$-fold product of $\mathcal{J}$ and then unravelling the product into a concept. Now a concept inclusion is satisfied by $\mathcal{J}$ if and only if it is valid for $\phi_\mathcal{J}$.

Example 15. Consider the interpretation $\mathcal{J}$ the domain of which consists of the four elements in Figure 2 with a double outline. We are going to compute the closure of the concept C from Example 5. Firstly, we determine the extension $C^\mathcal{J}$, which consists only of $d_{C,\{\exists r.(B_1 \sqcap B_2)\}}$. Secondly, we construct the model-based most specific concept of $C^\mathcal{J}$, which yields the closure $C^{\phi_\mathcal{J}} = A \sqcap A_1 \sqcap \exists r.(B \sqcap B_2) \sqcap \exists r.(B \sqcap B_1) \sqcap \exists r.B$. This concept is not subsumed by the concept D from Example 5, i.e., the CI $C \sqsubseteq D$ is not valid for the induced closure operator $\phi_\mathcal{J}$.

The set of all closure operators is a lattice [13], i.e., each two closure operators have an infimum and a supremum. Specifically, the *infimum* $\phi_\mathcal{T} \vartriangle \phi_\mathcal{J}$ maps each concept E to the least common subsumer of the closures $E^{\phi_\mathcal{T}}$ and $E^{\phi_\mathcal{J}}$. Recall that the *least common subsumer* of two concepts E and F is a concept G such that $E \sqsubseteq_\emptyset G$, $F \sqsubseteq_\emptyset G$, and $G \sqsubseteq_\emptyset H$ for each concept H where $E \sqsubseteq_\emptyset H$ and $F \sqsubseteq_\emptyset H$. According to [5], least common subsumers in $\mathcal{EL}$ can be computed by means of graph products.

As shown in [17], a concept inclusion is valid for $\phi_\mathcal{T} \vartriangle \phi_\mathcal{J}$ if and only if it is valid both for $\phi_\mathcal{T}$ and for $\phi_\mathcal{J}$, i.e., this infimum describes the logical intersection $\mathcal{T} \vartriangle \mathcal{J}$. Summing up, the benefit of switching to these abstract representations is that the logical intersection can be characterized by means of a closure operator.

Example 16. Reconsider the closures $C^{\phi_{\mathcal{T}}}$ and $C^{\phi_{\mathcal{J}}}$ from Example 14 and Example 15, respectively. The least common subsumer of these two concepts is the closure of C w.r.t. the infimum $\phi_{\mathcal{T}} \vartriangle \phi_{\mathcal{J}}$ and evaluates to $C^{\phi_{\mathcal{T}} \vartriangle \phi_{\mathcal{J}}} = A \sqcap A_1 \sqcap \exists r.(B \sqcap B_2) \sqcap \exists r.(B \sqcap B_1) \sqcap \exists r.B$. It follows that the concept inclusion $C \sqsubseteq D$ from Example 5 is not valid for the infimum $\phi_{\mathcal{T}} \vartriangle \phi_{\mathcal{J}}$, i.e., the logical intersection $\mathcal{T} \vartriangle \mathcal{J}$ does not entail $C \sqsubseteq D$.

Finally, we utilize the technique in Section 6.6 in [17] in order to construct the *canonical base of concept inclusions* for the infimum $\phi_{\mathcal{T}} \vartriangle \phi_{\mathcal{J}}$, which is a TBox $\mathcal{B}$ such that a concept inclusion is valid for $\phi_{\mathcal{T}} \vartriangle \phi_{\mathcal{J}}$ if and only if it is entailed by $\mathcal{B}$. That way, we obtain a finite axiomatization of the logical intersection, which almost qualifies as a repair of the given TBox $\mathcal{T}$ for the unwanted consequence $C \sqsubseteq D$. Applying the axiomatization method to the unrestricted clop $\phi_{\mathcal{T}} \vartriangle \phi_{\mathcal{J}}$ produces a TBox formulated in an extension of $\mathcal{EL}$ with greatest fixed points. The concepts in such an extension need not be tree-shaped anymore but can contain cycles. Since the given TBox $\mathcal{T}$ is assumed to be expressed in plain $\mathcal{EL}$, we also want the repair to be a usual $\mathcal{EL}$ TBox. This can be achieved by simply restricting $\phi_{\mathcal{T}} \vartriangle \phi_{\mathcal{J}}$ to a maximal role depth n, yielding the restricted clop $(\phi_{\mathcal{T}} \vartriangle \phi_{\mathcal{J}})\restriction_n$ — the closure of a concept C w.r.t. $(\phi_{\mathcal{T}} \vartriangle \phi_{\mathcal{J}})\restriction_n$ is the unraveling of the (possibly cyclic) closure $C^{\phi_{\mathcal{T}} \vartriangle \phi_{\mathcal{J}}}$ into a tree-shaped $\mathcal{EL}$ concept with depth not exceeding n. Specifically, this means that we axiomatize the logical intersection $\mathcal{T} \vartriangle \mathcal{J}$ only up to role depth n. Suitable choices for n are, e.g., the maximal role depth of a concept occurring in $\mathcal{T}$ or its doubled value — the concrete choice of n depends on the use case.

Another reason why a role-depth bound needs to be employed is to guarantee finiteness of the canonical base. As shown in Section 6.3 in [17], not every logical intersection is finitely axiomatizable. For instance, the logical intersection of the TBoxes $\{A \sqsubseteq B_1\}$ and $\{A \sqsubseteq B_2\}$ cannot be described by a finite $\mathcal{EL}$ TBox, as it entails the concept inclusion $\exists r^n.(A \sqcap B_1) \sqcap \exists r^n.(A \sqcap B_2) \sqsubseteq \exists r^n.(A \sqcap B_1 \sqcap B_2)$ for each number n. There might also be a TBox and an interpretation for which their logical intersection is not finitely axiomatizable by an $\mathcal{EL}$ TBox, but this is only a claim for now.

In summary, we obtain the following result.

Theorem 17. *For each countermodel $\mathcal{J}$ to $C \sqsubseteq D$ and for each role-depth bound $n \in \mathbb{N}$, the canonical base $\mathsf{Can}((\phi_{\mathcal{T}} \vartriangle \phi_{\mathcal{J}})\restriction_n)$ is a repair of $\mathcal{T}$ for $C \sqsubseteq D$.*

The actual computation of the above canonical base utilizes methods from *Formal Concept Analysis (FCA)*. We will not go into detail here and rather refer interested readers to Section 6 in [17].

Example 18. Reconsider the TBox $\mathcal{T}$ as well as the concept inclusion $C \sqsubseteq D$ from Example 5 and one of the countermodels from Example 11, namely the one with root $d_{C,\{\exists r.(B_1 \sqcap B_2)\}}$, which consists of the domain elements depicted in Figure 2 by the four nodes with a double outline. Denote this countermodel by $\mathcal{J}$. Of the canonical base of $\phi_{\mathcal{T}} \vartriangle \phi_{\mathcal{J}}$ restricted to role depth 1, we computed the part consisting of all CIs where each conjunction has at most three conjuncts:

- $A \sqsubseteq A_1$,
- $B \sqcap E \sqsubseteq B_1 \sqcap B_2$ where E is a concept name different from B, B_1, B_2, or E is an existential restriction $\exists s. \top$ for a role name s,
- $\exists r. B \sqsubseteq \exists r. (B \sqcap B_1) \sqcap \exists r. (B \sqcap B_2)$,
- $\exists s. B \sqsubseteq \exists s. (B \sqcap B_1 \sqcap B_2)$ where s is a role name different from r,
- $\exists r. (B \sqcap B_1) \sqcap \exists r. (B \sqcap B_2) \sqcap E \sqsubseteq \exists r. (B \sqcap B_1 \sqcap B_2)$ where E is a concept name different from A, A_1, B, or E is an existential restriction $\exists r. F$ for a concept name F different from A, B, B_1, B_2, or E is the existential restriction $\exists r. (B_1 \sqcap B_2)$, or E is an existential restriction $\exists s. \top$ for a role name $s \neq r$

We observe that one of the r-successors of the root $d_{C,\{\exists r.(B_1 \sqcap B_2)\}}$ would suffice to constitute a countermodel to $C \sqsubseteq D$. Specifically, for the countermodel consisting only of the root and the single r-successor $d_{B,\{B_1\}}$, the CI $B \sqsubseteq B_2$ is contained in $\mathcal{T} \vartriangle \mathcal{J}$ and thus entailed by the repair.

The above example shows that a repair in form of a canonical base (CB-repair) might get considerably larger than the input TBox. Alternatively, a countermodel $\mathcal{J}$ could be used to only weaken the axioms in the TBox $\mathcal{T}$ in order to get a repair: for each CI $E \sqsubseteq F$ in $\mathcal{T}$, replace its right side F with the least common subsumer of F and $E^{\phi_{\mathcal{J}}}$. The resulting TBox is entailed by $\mathcal{T}$ and has $\mathcal{J}$ as a model, which implies that it does not entail the unwanted consequence $C \sqsubseteq D$. It follows that each such repair qualifies as a *gentle repair* in the sense of [3]. As benefits, such repairs are cheaper to compute than CB-repairs and they never contain more CIs than the input TBox. However, the corresponding CB-repair usually retains more other consequences.

5 Concluding Remarks

This article discusses an approach to repairing $\mathcal{EL}$ TBoxes such that they do not entail unwanted consequences in the form of CIs. In particular, this approach is realized by first transforming the canonical model of the left-hand side of the CI w.r.t. the given TBox into a countermodel to the CI. We adapt a technique for computing ABox repairs and we use transductions as formalism for the model transformation. Instead of only constructing one countermodel, our approach is also equipped with a technique that yields a set of countermodels to the CI. A TBox repair is finally obtained as the axiomatization of the logical intersection of the original TBox and a constructed countermodel.

As seen in Example 18, we could further refine the second transduction τ_{reach} such that not all reachable elements are included, but only enough to obtain a countermodel. To complete our complexity results, a further study on the complexity of computing a TBox repair from a given countermodel will be necessary as a next step. In [3], the notion of *optimal repairs* is defined by requiring that only a minimal amount of other consequences is lost, and it was further shown there that optimal repairs need not exist in general — specifically, if the ontology consists of both an ABox and a TBox. Characterizing the existence of optimal TBox repairs and then extending our approach such that it yields such repairs, if they exist, are also interesting future work.

References

1. Baader, F.: Terminological Cycles in a Description Logic with Existential Restrictions. In: Gottlob, G., Walsh, T. (eds.) IJCAI-03, Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence, Acapulco, Mexico, August 9-15, 2003. pp. 325–330. Morgan Kaufmann (2003)
2. Baader, F., Brandt, S., Lutz, C.: Pushing the $\mathcal{EL}$ Envelope. In: Kaelbling, L.P., Saffiotti, A. (eds.) IJCAI-05, Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence, Edinburgh, Scotland, UK, July 30 - August 5, 2005. pp. 364–369. Professional Book Center (2005)
3. Baader, F., Kriegel, F., Nuradiansyah, A., Peñaloza, R.: Making Repairs in Description Logics More Gentle. In: Thielscher, M., Toni, F., Wolter, F. (eds.) Principles of Knowledge Representation and Reasoning: Proceedings of the Sixteenth International Conference, KR 2018, Tempe, Arizona, 30 October - 2 November 2018. pp. 319–328. AAAI Press (2018)
4. Baader, F., Kriegel, F., Nuradiansyah, A., Peñaloza, R.: Computing Compliant Anonymisations of Quantified ABoxes w.r.t. $\mathcal{EL}$ Policies. In: Pan, J.Z., Tamma, V.A.M., d'Amato, C., Janowicz, K., Fu, B., Polleres, A., Seneviratne, O., Kagal, L. (eds.) The Semantic Web - ISWC 2020 - 19th International Semantic Web Conference, Athens, Greece, November 2-6, 2020, Proceedings, Part I. Lecture Notes in Computer Science, vol. 12506, pp. 3–20. Springer (2020)
5. Baader, F., Küsters, R., Molitor, R.: Computing Least Common Subsumers in Description Logics with Existential Restrictions. In: Dean, T. (ed.) Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence, IJCAI 99, Stockholm, Sweden, July 31 - August 6, 1999. 2 Volumes, 1450 pages. pp. 96–103. Morgan Kaufmann (1999)
6. Baader, F., Lutz, C., Brandt, S.: Pushing the $\mathcal{EL}$ Envelope Further. In: Clark, K., Patel-Schneider, P.F. (eds.) Proceedings of the Fourth OWLED Workshop on OWL: Experiences and Directions, Washington, DC, USA, 1-2 April 2008. CEUR Workshop Proceedings, vol. 496. CEUR-WS.org (2008)
7. Brandt, S.: Polynomial Time Reasoning in a Description Logic with Existential Restrictions, GCI Axioms, and - What Else? In: de Mántaras, R.L., Saitta, L. (eds.) Proceedings of the 16th Eureopean Conference on Artificial Intelligence, ECAI'2004, including Prestigious Applicants of Intelligent Systems, PAIS 2004, Valencia, Spain, August 22-27, 2004. pp. 298–302. IOS Press (2004)
8. Courcelle, B., Engelfriet, J.: Graph structure and monadic second-order logic: a language-theoretic approach, vol. 138. Cambridge University Press (2012)
9. Distel, F.: Learning Description Logic Knowledge Bases from Data using Methods from Formal Concept Analysis. Doctoral thesis, Technische Universität Dresden, Dresden, Germany (2011)
10. Du, J., Qi, G., Fu, X.: A practical fine-grained approach to resolving incoherent OWL 2 DL terminologies. In: Li, J., Wang, X.S., Garofalakis, M.N., Soboroff, I., Suel, T., Wang, M. (eds.) Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management, CIKM 2014, Shanghai, China, November 3-7, 2014. pp. 919–928. ACM (2014)
11. Ebbinghaus, H.D., Flum, J., Thomas, W.: Mathematical logic. Springer Science & Business Media (2013)
12. Hieke, W., Turhan, A.Y.: Towards Model Transformation in Description Logics–Investigating the Case of Transductions. CEUR Workshop Proceedings of the 6th Workshop on Formal and Cognitive Reasoning **2680**, 69–82 (2020)

13. Higuchi, A.: Lattices of Closure Operators. Discrete Mathematics **179**(1-3), 267–272 (1998)
14. Horridge, M., Parsia, B., Sattler, U.: Laconic and precise justifications in OWL. In: Sheth, A.P., Staab, S., Dean, M., Paolucci, M., Maynard, D., Finin, T.W., Thirunarayan, K. (eds.) The Semantic Web - ISWC 2008, 7th International Semantic Web Conference, ISWC 2008, Karlsruhe, Germany, October 26-30, 2008. Proceedings. Lecture Notes in Computer Science, vol. 5318, pp. 323–338. Springer (2008)
15. Jones, N.D.: Space-bounded reducibility among combinatorial problems. Journal of Computer and System Sciences **11**(1), 68–85 (1975)
16. Kalyanpur, A., Parsia, B., Sirin, E., Grau, B.C.: Repairing unsatisfiable concepts in OWL ontologies. In: Sure, Y., Domingue, J. (eds.) The Semantic Web: Research and Applications, 3rd European Semantic Web Conference, ESWC 2006, Budva, Montenegro, June 11-14, 2006, Proceedings. Lecture Notes in Computer Science, vol. 4011, pp. 170–184. Springer (2006)
17. Kriegel, F.: Constructing and Extending Description Logic Ontologies using Methods of Formal Concept Analysis. Doctoral thesis, Technische Universität Dresden, Dresden, Germany (2019)
18. Lam, J.S.C., Sleeman, D.H., Pan, J.Z., Vasconcelos, W.W.: A fine-grained approach to resolving unsatisfiable ontologies. J. Data Semant. **10**, 62–95 (2008)
19. Lutz, C., Wolter, F.: Deciding inseparability and conservative extensions in the description logic $\mathcal{EL}$. Journal of Symbolic Computation **45**(2), 194–228 (2010)
20. Reiter, R.: A theory of diagnosis from first principles. Artificial Intelligence **32**(1), 57–95 (1987)
21. Schlobach, S.: Debugging and semantic clarification by pinpointing. In: Gómez-Pérez, A., Euzenat, J. (eds.) The Semantic Web: Research and Applications, Second European Semantic Web Conference, ESWC 2005, Heraklion, Crete, Greece, May 29 - June 1, 2005, Proceedings. Lecture Notes in Computer Science, vol. 3532, pp. 226–240. Springer (2005)
22. Schlobach, S.: Diagnosing terminologies. In: Veloso, M.M., Kambhampati, S. (eds.) Proceedings, The Twentieth National Conference on Artificial Intelligence and the Seventeenth Innovative Applications of Artificial Intelligence Conference, July 9-13, 2005, Pittsburgh, Pennsylvania, USA. pp. 670–675. AAAI Press / The MIT Press (2005)
23. 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), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018. pp. 1981–1988. AAAI Press (2018)
24. Vardi, M.Y.: The complexity of relational query languages. In: Proceedings of the fourteenth annual ACM symposium on Theory of computing. pp. 137–146 (1982)