SAT Encoding of Unification in $\mathcal{ELH}_{R^+}$ w.r.t. Cycle-Restricted Ontologies*

Franz Baader, Stefan Borgwardt, and Barbara Morawska

Theoretical Computer Science, TU Dresden, Germany
{baader,stefborg,morawska}@tcs.inf.tu-dresden.de

Abstract. Unification in Description Logics has been proposed as an inference service that can, for example, be used to detect redundancies in ontologies. For the Description Logic $\mathcal{EL}$, which is used to define several large biomedical ontologies, unification is NP-complete. An NP unification algorithm for $\mathcal{EL}$ based on a translation into propositional satisfiability (SAT) has recently been presented. In this paper, we extend this SAT encoding in two directions: on the one hand, we add general concept inclusion axioms, and on the other hand, we add role hierarchies ($\mathcal{H}$) and transitive roles (R^+). For the translation to be complete, however, the ontology needs to satisfy a certain cycle restriction. The SAT translation depends on a new rewriting-based characterization of subsumption w.r.t. $\mathcal{ELH}_{R^+}$-ontologies.

1 Introduction

The Description Logic (DL) $\mathcal{EL}$, which offers the constructors conjunction ($\sqcap$), existential restriction ($\exists r.C$), and the top concept ($\top$), has recently drawn considerable attention since, on the one hand, important inference problems such as the subsumption problem are polynomial in $\mathcal{EL}$, even in the presence of general concept inclusion axioms (GCIs) [11,4]. On the other hand, though quite inexpressive, $\mathcal{EL}$ can be used to define biomedical ontologies, such as the large medical ontology SNOMED CT.[1]

Unification in DLs has been proposed in [8] as a novel inference service that can, for instance, be used to detect redundancies in ontologies. For example, assume that one developer of a medical ontology defines the concept of a *patient with severe injury of the frontal lobe* as

$$\exists \mathsf{finding}.(\mathsf{Frontal\_lobe\_injury} \sqcap \exists \mathsf{severity}.\mathsf{Severe}), \tag{1}$$

whereas another one represents it as

$$\exists \mathsf{finding}.(\mathsf{Severe\_injury} \sqcap \exists \mathsf{finding\_site}.\exists \mathsf{part\_of}.\mathsf{Frontal\_lobe}). \tag{2}$$

These two concept descriptions are not equivalent, but they are nevertheless meant to represent the same concept. They can obviously be made equivalent by

* Supported by DFG under grant BA 1122/14-1

[1] see http://www.ihtsdo.org/snomed-ct/

treating the concept names $\mathsf{Frontal\_lobe\_injury}$ and $\mathsf{Severe\_injury}$ as variables, and substituting the first one by $\mathsf{Injury} \sqcap \exists \mathsf{finding\_site}.\exists \mathsf{part\_of}.\mathsf{Frontal\_lobe}$ and the second one by $\mathsf{Injury} \sqcap \exists \mathsf{severity}.\mathsf{Severe}$. In this case, we say that the descriptions are unifiable, and call the substitution that makes them equivalent a *unifier*.

To motivate our interest in unification w.r.t. GCIs, role hierarchies, and transitive roles, assume that the developers use the descriptions (3) and (4) instead of (1) and (2):

$$\begin{aligned} &\exists \mathsf{finding}.\exists \mathsf{finding\_site}.\exists \mathsf{part\_of}.\mathsf{Brain} \sqcap \\ &\exists \mathsf{finding}.(\mathsf{Frontal\_lobe\_injury} \sqcap \exists \mathsf{severity}.\mathsf{Severe}) \end{aligned} \tag{3}$$

$$\begin{aligned} &\exists \mathsf{status}.\mathsf{Emergency} \sqcap \\ &\exists \mathsf{finding}.(\mathsf{Severe\_injury} \sqcap \exists \mathsf{finding\_site}.\exists \mathsf{part\_of}.\mathsf{Frontal\_lobe}) \end{aligned} \tag{4}$$

The descriptions (3) and (4) are not unifiable without additional background knowledge, but they are unifiable, with the same unifier as above, if the GCIs

$$\begin{aligned} \exists \mathsf{finding}.\exists \mathsf{severity}.\mathsf{Severe} &\sqsubseteq \exists \mathsf{status}.\mathsf{Emergency}, \\ \mathsf{Frontal\_lobe} &\sqsubseteq \exists \mathsf{proper\_part\_of}.\mathsf{Brain} \end{aligned}$$

are present in a background ontology and this ontology additionally states that $\mathsf{part\_of}$ is transitive and $\mathsf{proper\_part\_of}$ is a subrole of $\mathsf{part\_of}$.

Most of the previous results on unification in DLs did not consider such additional background knowledge. In [8] it was shown that, for the DL $\mathcal{FL}_0$, which differs from $\mathcal{EL}$ by offering value restrictions ($\forall r.C$) in place of existential restrictions, deciding unifiability is an ExpTime-complete problem. In [5], we were able to show that unification in $\mathcal{EL}$ is of considerably lower complexity: the decision problem is NP-complete. The original unification algorithm for $\mathcal{EL}$ introduced in [5] was a brutal "guess and then test" NP-algorithm, but we have since then also developed more practical algorithms. On the one hand, in [7] we describe a goal-oriented unification algorithm for $\mathcal{EL}$, in which nondeterministic decisions are only made if they are triggered by "unsolved parts" of the unification problem. On the other hand, in [6], we present an algorithm that is based on a reduction to satisfiability in propositional logic (SAT). In [7] it was also shown that the approaches for unification of $\mathcal{EL}$-concept descriptions (without any background ontology) can easily be extended to the case of an acyclic TBox as background ontology without really changing the algorithms or increasing their complexity. Basically, by viewing defined concepts as variables, an acyclic TBox can be turned into a unification problem that has as its unique unifier the substitution that replaces the defined concepts by unfolded versions of their definitions.

For GCIs, this simple trick is not possible, and thus handling them requires the development of new algorithms. In [1,2] we describe two such new algorithms: one that extends the brute-force "guess and then test" NP-algorithm from [5] and a more practical one that extends the goal-oriented algorithm from [7].

Both algorithms are based on a new characterization of subsumption w.r.t. GCIs in $\mathcal{EL}$, which we prove using a Gentzen-style proof calculus for subsumption. Unfortunately, these algorithms are complete only for cycle-restricted TBoxes, i.e., finite sets of GCIs that satisfy a certain restriction on cycles, which, however, does not prevent all cycles. For example, the cyclic GCI $\exists \mathsf{child}.\mathsf{Human} \sqsubseteq \mathsf{Human}$ satisfies this restriction, whereas the cyclic GCI $\mathsf{Human} \sqsubseteq \exists \mathsf{parent}.\mathsf{Human}$ does not.

In the present paper, we still cannot get rid of cycle-restrictedness of the ontology, but extend the results of [2] in two other directions: (i) we add transitive roles (indicated by the subscript R^+ in the name of the DL) and role hierarchies (indicated by adding the letter $\mathcal{H}$ to the name of the DL) to the language, which are important for medical ontologies [17,15]; (ii) we provide an algorithm that is based on a translation into SAT, and thus allows us to employ highly optimized state-of-the-art SAT solvers [10] for implementing the unification algorithm. In order to obtain the SAT translation, using the characterization of subsumption from [2] is not sufficient, however. We had to develop a new rewriting-based characterization of subsumption.

In the next section, we introduce the DLs considered in this paper and the important inference problem subsumption. In Section 3 we define unification for these DLs and recall some of the existing results for unification in $\mathcal{EL}$. In particular, we introduce in this section the notion of cycle-restrictedness, which is required for the results on unification w.r.t. GCIs to hold. In Section 4 we then derive rewriting-based characterizations of subsumption. Section 5 contains the main result of this paper, which is a reduction of unification in $\mathcal{ELH}_{R^+}$ w.r.t. cycle-restricted ontologies to propositional satisfiability. The proof of correctness of this reduction strongly depends on the characterization of subsumption shown in the previous section.

2 The Description Logics $\mathcal{EL}$, $\mathcal{EL}^+$, and $\mathcal{ELH}_{R^+}$

The expressiveness of a DL is determined both by the formalism for describing concepts (the concept description language) and the terminological formalism, which can be used to state additional constraints on the interpretation of concepts and roles in a so-called ontology.

Syntax and Semantics

The concept description language considered in this paper is called $\mathcal{EL}$. Starting with a finite set N_C of *concept names* and a finite set N_R of *role names*, *$\mathcal{EL}$-concept descriptions* are built from concept names using the constructors *conjunction* ($C \sqcap D$), *existential restriction* ($\exists r.C$ for every $r \in N_R$), and *top* ($\top$).

Since in this paper we only consider $\mathcal{EL}$-concept descriptions, we will sometimes dispense with the prefix $\mathcal{EL}$.

Name	Syntax	Semantics
concept name	A	$A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$
role name	r	$r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$
top	$\top$	$\top^{\mathcal{I}} = \Delta^{\mathcal{I}}$
conjunction	$C \sqcap D$	$(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}$
existential restriction	$\exists r.C$	$(\exists r.C)^{\mathcal{I}} = \{x \mid \exists y : (x, y) \in r^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\}$
general concept inclusion	$C \sqsubseteq D$	$C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$
role inclusion	$r_1 \circ \cdots \circ r_n \sqsubseteq s$	$r_1^{\mathcal{I}} \circ \cdots \circ r_n^{\mathcal{I}} \subseteq s^{\mathcal{I}}$

Table 1. Syntax and semantics of $\mathcal{EL}$.

On the semantic side, concept descriptions are interpreted as sets. To be more precise, an *interpretation* $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ consists of a non-empty domain $\Delta^{\mathcal{I}}$ and an interpretation function $\cdot^{\mathcal{I}}$ that maps concept names to subsets of $\Delta^{\mathcal{I}}$ and role names to binary relations over $\Delta^{\mathcal{I}}$. This function is extended to concept descriptions as shown in the semantics column of Table 1.

Ontologies

A *general concept inclusion (GCI)* is of the form $C \sqsubseteq D$ for concept descriptions C, D, and a *role inclusion* is of the form $r_1 \circ \cdots \circ r_n \sqsubseteq s$ for role names $r_1, \ldots, r_n, s$. Both are called *axioms*. Role inclusions of the form $r \circ r \sqsubseteq r$ are called *transitivity axioms* and of the form $r \sqsubseteq s$ *role hierarchy axioms*. An interpretation $\mathcal{I}$ *satisfies* such an axiom if the corresponding condition in the semantics column of Table 1 holds, where $\circ$ in this column stands for composition of binary relations.

An $\mathcal{EL}^+$*-ontology* is a finite set of axioms. It is an $\mathcal{ELH}_{R^+}$*-ontology* if all its role inclusions are transitivity or role hierarchy axioms, and an $\mathcal{EL}$*-ontology* if it contains only GCIs. An interpretation is a *model* of an ontology if it satisfies all its axioms.

Subsumption, Equivalence, and Role Hierarchy

A concept description C is *subsumed* by a concept description D w.r.t. an ontology $\mathcal{O}$ (written $C \sqsubseteq_{\mathcal{O}} D$) if every model of $\mathcal{O}$ satisfies the GCI $C \sqsubseteq D$. We say that C is *equivalent* to D w.r.t. $\mathcal{O}$ ($C \equiv_{\mathcal{O}} D$) if $C \sqsubseteq_{\mathcal{O}} D$ and $D \sqsubseteq_{\mathcal{O}} C$. If $\mathcal{O}$ is empty, we also write $C \sqsubseteq D$ and $C \equiv D$ instead of $C \sqsubseteq_{\mathcal{O}} D$ and $C \equiv_{\mathcal{O}} D$, respectively. As shown in [11,4], subsumption w.r.t. $\mathcal{EL}^+$-ontologies (and thus also w.r.t. $\mathcal{ELH}_{R^+}$- and $\mathcal{EL}$-ontologies) is decidable in polynomial time.

Since conjunction is interpreted as intersection, the concept descriptions $(C \sqcap D) \sqcap E$ and $C \sqcap (D \sqcap E)$ are always equivalent. Thus, we dispense with parentheses and write nested conjunctions in flat form $C_1 \sqcap \cdots \sqcap C_n$. Nested existential restrictions $\exists r_1.\exists r_2.\ldots\exists r_n.C$ will sometimes also be written as $\exists r_1 r_2 \ldots r_n.C$, where $r_1 r_2 \ldots r_n$ is viewed as a word over the alphabet of role names, i.e., an element of N_R^*.

The *role hierarchy* induced by $\mathcal{O}$ is a binary relation $\trianglelefteq_\mathcal{O}$ on N_R, which is defined as the reflexive-transitive closure of the relation $\{(r,s) \mid r \sqsubseteq s \in \mathcal{O}\}$. Using elementary reachability algorithms, the role hierarchy can be computed in polynomial time in the size of $\mathcal{O}$. It is easy to see that $r \trianglelefteq_\mathcal{O} s$ implies that $r^\mathcal{I} \subseteq s^\mathcal{I}$ for all models $\mathcal{I}$ of $\mathcal{O}$.

3 Unification

In order to define unification, we first introduce the notion of a substitution operating on concept descriptions. For this purpose, we partition the set N_C of concepts names into a set N_v of concept variables (which may be replaced by substitutions) and a set N_c of concept constants (which must not be replaced by substitutions). A *substitution* σ maps every variable to an $\mathcal{EL}$-concept description. It can be extended from variables to $\mathcal{EL}$-concept descriptions as follows:

- $\sigma(A) := A$ for all $A \in N_c \cup \{\top\}$,
- $\sigma(C \sqcap D) := \sigma(C) \sqcap \sigma(D)$ and $\sigma(\exists r.C) := \exists r.\sigma(C)$.

A concept description C is *ground* if it does not contain variables, and a substitution is *ground* if all concept descriptions in its range are ground. Obviously, a ground concept description is not modified by applying a substitution, and if we apply a ground substitution to any concept description, then we obtain a ground description. An ontology is *ground* if it does not contain variables.

Definition 1. *Let $\mathcal{O}$ be a ground ontology. A* unification problem *w.r.t. $\mathcal{O}$ is a finite set $\Gamma = \{C_1 \sqsubseteq^? D_1, \ldots, C_n \sqsubseteq^? D_n\}$ of subsumptions between $\mathcal{EL}$-concept descriptions. A substitution σ is a* unifier *of Γ w.r.t. $\mathcal{O}$ if σ* solves *all the subsumptions in Γ w.r.t. $\mathcal{O}$, i.e., if $\sigma(C_1) \sqsubseteq_\mathcal{O} \sigma(D_1), \ldots, \sigma(C_n) \sqsubseteq_\mathcal{O} \sigma(D_n)$. We say that Γ is* unifiable *w.r.t. $\mathcal{O}$ if it has a unifier w.r.t. $\mathcal{O}$.*

We call Γ w.r.t. $\mathcal{O}$ an $\mathcal{EL}$-, $\mathcal{EL}^+$-, or $\mathcal{ELH}_{R^+}$-unification problem depending on whether and what kind of role inclusions are contained in $\mathcal{O}$.

Three remarks regarding the definition of unification problems are in order. First, note that some of the previous papers on unification in DLs used equivalences $C \equiv^? D$ instead of subsumptions $C \sqsubseteq^? D$. This difference is, however, irrelevant since $C \equiv^? D$ can be seen as a shorthand for the two subsumptions $C \sqsubseteq^? D$ and $D \sqsubseteq^? C$, and $C \sqsubseteq^? D$ has the same unifiers as $C \sqcap D \equiv^? C$.

Second, note that—as in [2]—we have restricted the background ontology $\mathcal{O}$ to be ground. This is not without loss of generality. In fact, if $\mathcal{O}$ contained variables, then we would need to apply the substitution also to its axioms, and instead of requiring $\sigma(C_i) \sqsubseteq_\mathcal{O} \sigma(D_i)$ we would thus need to require $\sigma(C_i) \sqsubseteq_{\sigma(\mathcal{O})} \sigma(D_i)$, which would change the nature of the problem considerably. The treatment of unification w.r.t. acyclic TBoxes in [7] actually considers a more general setting, where some of the primitive concepts occurring in the TBox may be variables. The restriction to ground general TBoxes is, however, appropriate for the application scenario sketched in the introduction. In this scenario, there is a fixed background ontology, which is extended with definitions of new concepts by several knowledge engineers. Unification w.r.t. the background ontology is used to check whether some of these new definitions actually are redundant, i.e., define the same intuitive concept. Here, some of the primitive concepts newly introduced by one knowledge engineer may be further defined by another one, but we assume that the knowledge engineers use the vocabulary from the background ontology unchanged, i.e., they define *new* concepts rather than adding definitions for concepts that already occur in the background ontology. An instance of this scenario can, e.g., be found in [12], where different extensions of SNOMED CT are checked for overlaps, albeit not by using unification, but by simply testing for equivalence.

Third, though arbitrary substitutions σ are used in the definition of a unifier, it is actually sufficient to consider ground substitutions such that all concept descriptions $\sigma(X)$ in the range of σ contain only concept and role names occurring in Γ or $\mathcal{O}$. It is an easy consequence of well-known results from unification theory [9] that Γ has a unifier w.r.t. $\mathcal{O}$ iff it has such a ground unifier.

Relationship to Equational Unification

Unification was originally not introduced for Description Logics, but for equational theories [9]. In [7] it was shown that unification in $\mathcal{EL}$ (w.r.t. the empty ontology) is the same as unification in the equational theory *SLmO* of semilattices with monotone operators [16]. As argued in [2], unification in $\mathcal{EL}$ w.r.t. a ground $\mathcal{EL}$-ontology corresponds to unification in *SLmO* extended with a finite set of ground identities. In contrast to GCIs, role inclusions add non-ground identities to *SLmO* (see [16] and [3] for details).

This unification-theoretic point of view sheds some light on our decision to restrict unification w.r.t. general TBoxes to the case of general TBoxes that are ground. In fact, if we lifted this restriction, then we would end up with a generalization of rigid E-unification [14,13], in which the theory *SLmO* extended with the identities expressing role inclusions is used as a background theory. To the best of our knowledge, such variants of rigid E-unification have not been considered in the literature, and are probably quite hard to solve.

Flat Ontologies and Unification Problems

To simplify the technical development, it is convenient to normalize the TBox and the unification problem appropriately. To introduce this normal form, we need the notion of an atom.

An *atom* is a concept name or an existential restriction. Obviously, every $\mathcal{EL}$-concept description C is a finite conjunction of atoms, where $\top$ is considered to be the empty conjunction. We call the atoms in this conjunction the *top-level atoms* of C. An atom is called *flat* if it is a concept name or an existential restriction of the form $\exists r.A$ for a concept name A.

The GCI $C \sqsubseteq D$ or subsumption $C \sqsubseteq^? D$ is called *flat* if C is a conjunction of $n \geq 0$ flat atoms and D is a flat atom. The ontology $\mathcal{O}$ (unification problem Γ) is called *flat* if all the GCIs in $\mathcal{O}$ (subsumptions in Γ) are flat. Given a ground ontology $\mathcal{O}$ and a unification problem Γ, we can compute in polynomial time (see [3]) a flat ontology $\mathcal{O}'$ and a flat unification problem Γ' such that

- Γ has a unifier w.r.t. $\mathcal{O}$ iff Γ' has a unifier w.r.t. $\mathcal{O}'$;
- the type of the unification problem ($\mathcal{EL}$, $\mathcal{EL}^+$, or $\mathcal{ELH}_{R^+}$) is preserved.

For this reason, we will assume in the following that all ontologies and unification problems are flat.

Cycle-Restricted Ontologies

The decidability and complexity results for unification w.r.t. $\mathcal{EL}$-ontologies in [2], and also the corresponding ones in the present paper, only hold if the ontologies satisfy a restriction that prohibits certain cyclic subsumptions.

Definition 2. *The $\mathcal{EL}^+$-ontology $\mathcal{O}$ is called* cycle-restricted *iff there is no nonempty word $w \in N_R^+$ and $\mathcal{EL}$-concept description C such that $C \sqsubseteq_{\mathcal{O}} \exists w.C$.*

Note that cycle-restrictedness is not a syntactic condition on the form of the axioms in $\mathcal{O}$, but a semantic one on what follows from $\mathcal{O}$. Nevertheless, for $\mathcal{ELH}_{R^+}$-ontologies, this condition can be decided in polynomial time [3]. Basically, one first shows that the $\mathcal{ELH}_{R^+}$-ontology $\mathcal{O}$ is cycle-restricted iff $A \not\sqsubseteq_{\mathcal{O}} \exists w.A$ holds for all nonempty words $w \in N_R^+$ and all $A \in N_C \cup \{\top\}$. Then, one shows that $A \sqsubseteq_{\mathcal{O}} \exists w.A$ for some $w \in N_R^+$ and $A \in N_C \cup \{\top\}$ implies that there are $n \geq 1$ role names $r_1, \ldots, r_n$ and $A_1, \ldots, A_n \in N_C \cup \{\top\}$ such that

$$(*) \quad A \sqsubseteq_{\mathcal{O}} \exists r_1.A_1, A_1 \sqsubseteq_{\mathcal{O}} \exists r_2.A_2, \ldots, A_{n-1} \sqsubseteq_{\mathcal{O}} \exists r_n.A_n \text{ and } A_n = A.$$

Using the polynomial-time subsumption algorithm for $\mathcal{ELH}_{R^+}$, we can build a graph whose nodes are the elements of $N_C \cup \{\top\}$ and where there is an edge from A to B with label r iff $A \sqsubseteq_{\mathcal{O}} \exists r.B$. Then we can use standard reachability algorithms to check whether this graph contains a cycle of the form $(*)$. The restriction to $\mathcal{ELH}_{R^+}$ stems from the fact that the proof of correctness of this algorithm is based on Lemma 7 below, which we cannot show for $\mathcal{EL}^+$.

The main reason why we need cycle-restrictedness of $\mathcal{O}$ is that it ensures that a substitution always induces a strict partial order on the variables.[2] To be more precise, assume that γ is a substitution. For $X, Y \in N_v$ we define

$$X >_\gamma Y \text{ iff } \gamma(X) \sqsubseteq_{\mathcal{O}} \exists w.\gamma(Y) \text{ for some } w \in N_R^+. \tag{5}$$

Transitivity of $>_\gamma$ is an easy consequence of transitivity of subsumption, and cycle-restrictedness of $\mathcal{O}$ yields irreflexivity of $>_\gamma$.

Lemma 3. *If $\mathcal{O}$ is a cycle-restricted $\mathcal{EL}^+$-ontology, then $>_\gamma$ is a strict partial order on N_v.*

[2] Why we need this order will become clear in Section 5.

4 Subsumption w.r.t. $\mathcal{EL}^+$- and $\mathcal{ELH}_{R^+}$-Ontologies

Subsumption w.r.t. $\mathcal{EL}^+$-ontologies can be decided in polynomial time [4]. For the purpose of deciding unification, however, we do not simply want a decision procedure for subsumption, but are more interested in a characterization of subsumption that helps us to find unifiers. The characterization of subsumption derived here is based on a rewrite relation that uses axioms as rewrite rules from right to left.

Proving Subsumption by Rewriting

Throughout this subsection, we assume that $\mathcal{O}$ is a flat $\mathcal{EL}^+$-ontology. Intuitively, an axiom of the form $A_1 \sqcap \ldots \sqcap A_n \sqsubseteq B \in \mathcal{O}$ is used to replace B by $A_1 \sqcap \ldots \sqcap A_n$ and an axiom of the form $r_1 \circ \ldots \circ r_n \sqsubseteq s \in \mathcal{O}$ to replace $\exists s.C$ by $\exists r_1 \ldots r_n.C$. In order to deal with associativity, commutativity, and idempotency of conjunction, it is convenient to represent concept descriptions as sets of atoms rather than as conjunctions of atoms.

Given an $\mathcal{EL}$-concept description C, the *description set* $\mathsf{s}(C)$ associated with C is defined by induction:

- $\mathsf{s}(A) := \{A\}$ for $A \in N_C$ and $\mathsf{s}(\top) := \emptyset$;
- $\mathsf{s}(C \sqcap D) := \mathsf{s}(C) \cup \mathsf{s}(D)$ and $\mathsf{s}(\exists r.C) := \{\exists r.\mathsf{s}(C)\}$.

For example, if $C = A \sqcap \exists r.(A \sqcap \exists r.\top)$, then $\mathsf{s}(C) = \{A, \exists r.\{A, \exists r.\emptyset\}\}$. We call *set positions* the positions in $\mathsf{s}(C)$ at which there is a set. In our example, we have three set positions, corresponding to the sets $\{A, \exists r.\{A, \exists r.\emptyset\}\}$, $\{A, \exists r.\emptyset\}$, and $\emptyset$. The set position that corresponds to the whole set $\mathsf{s}(C)$ is called the *root position*.

Our *rewrite rules* are of the form $N \leftarrow M$, where N, M are description sets. Such a rule *applies* at a set position p in $\mathsf{s}(C)$ if the corresponding set $\mathsf{s}(C)|_p$ contains M, and its *application* replaces $\mathsf{s}(C)|_p$ by $(\mathsf{s}(C)|_p \setminus M) \cup N$ (see [3] for a more formal definition of set positions and of the application of rewrite rules).

Given a flat $\mathcal{EL}^+$-ontology $\mathcal{O}$, the corresponding rewrite system $R(\mathcal{O})$ consists of the following rules:

- *Concept inclusion* ($\mathbf{R}_c$): For every $C \sqsubseteq D \in \mathcal{O}$, $R(\mathcal{O})$ contains the rule
$$\mathsf{s}(C) \leftarrow \mathsf{s}(D).$$
- *Role inclusion* ($\mathbf{R}_r$): For every $r_1 \circ \cdots \circ r_n \sqsubseteq s \in \mathcal{O}$ and every $\mathcal{EL}$-concept description C, $R(\mathcal{O})$ contains the rule
$$\mathsf{s}(\exists r_1 \ldots r_n.C) \leftarrow \mathsf{s}(\exists s.C).$$
- *Monotonicity* ($\mathbf{R}_m$): For every atom D, $R(\mathcal{O})$ contains the rule
$$\mathsf{s}(D) \leftarrow \emptyset.$$

Definition 4. *Let N, M be description sets. We write $N \leftarrow_{\mathcal{O}} M$ if N can be obtained from M by the application of a rule in $R(\mathcal{O})$. The relation $\overset{*}{\leftarrow}_{\mathcal{O}}$ is defined to be the reflexive, transitive closure of $\leftarrow_{\mathcal{O}}$, i.e., $N \overset{*}{\leftarrow}_{\mathcal{O}} M$ iff there is a chain*

$$N = M_\ell \leftarrow_{\mathcal{O}} M_{\ell-1} \leftarrow_{\mathcal{O}} \ldots \leftarrow_{\mathcal{O}} M_0 = M$$

of $\ell \geq 0$ rule applications. We call such a chain a derivation *of N from M w.r.t. $\mathcal{O}$. A rewriting step in such a derivation is called a* root step *if it applies a rule of the form $(\boldsymbol{R}_c)$ at the root position. We write $N \overset{(n)}{\longleftarrow}_{\mathcal{O}} M$ to express that there is a derivation of N from M w.r.t. $\mathcal{O}$ that uses* at most n root steps.

For example, if $\mathcal{O}$ contains the axioms $\top \sqsubseteq \exists r.B$ and $s \sqsubseteq r$, then the following is a derivation w.r.t. $\mathcal{O}$:

$$\{A, \exists s.\{A\}\} \leftarrow_{\mathcal{O}} \{A, \exists r.\{A\}\} \leftarrow_{\mathcal{O}} \{A, \exists r.\{A, \exists r.\{B\}\}\} \leftarrow_{\mathcal{O}} \{A, \exists r.\{A, \exists r.\emptyset\}\}$$

This is a derivation without a root step, which first applies a rule of the form $(\mathbf{R}_m)$, then one of the form $(\mathbf{R}_c)$ (not at the root position), and finally one of the form $(\mathbf{R}_r)$. This shows $\mathsf{s}(A \sqcap \exists s.A) \overset{(0)}{\longleftarrow}_{\mathcal{O}} \mathsf{s}(A \sqcap \exists r.(A \sqcap \exists r.\top))$.

The following theorem states that subsumption w.r.t. $\mathcal{O}$ corresponds to the existence of a derivation w.r.t. $\mathcal{O}$ whose root steps are bounded by the number of GCIs in $\mathcal{O}$ (see [3] for a proof of this result).

Theorem 5. *Let $\mathcal{O}$ be a flat $\mathcal{EL}^+$-ontology containing n GCIs and C, D be two $\mathcal{EL}$-concept descriptions. Then $C \sqsubseteq_{\mathcal{O}} D$ iff $\mathsf{s}(C) \overset{(n)}{\longleftarrow}_{\mathcal{O}} \mathsf{s}(D)$.*

A Structural Characterization of Subsumption in $\mathcal{ELH}_{R^+}$

Our translation of unification problems into propositional satisfiability problems depends on a structural characterization of subsumption, which we can unfortunately only show for $\mathcal{ELH}_{R^+}$ ontologies. Throughout this subsection, we assume that $\mathcal{O}$ is a flat $\mathcal{ELH}_{R^+}$-ontology. We say that r is *transitive* if the transitivity axiom $r \circ r \sqsubseteq r$ belongs to $\mathcal{O}$.

Definition 6. *Let C, D be atoms. We say that C is* structurally subsumed *by D w.r.t. $\mathcal{O}$ ($C \sqsubseteq^{\mathsf{s}}_{\mathcal{O}} D$) iff*

- *$C = D$ is a concept name,*
- *$C = \exists r.C'$, $D = \exists s.D'$, $C' \sqsubseteq_{\mathcal{O}} D'$, and $r \trianglelefteq_{\mathcal{O}} s$, or*
- *$C = \exists r.C'$, $D = \exists s.D'$, and $C' \sqsubseteq_{\mathcal{O}} \exists t.D'$ for a transitive role t with $r \trianglelefteq_{\mathcal{O}} t \trianglelefteq_{\mathcal{O}} s$.*

On the one hand, structural subsumption is a stronger property than $C \sqsubseteq_{\mathcal{O}} D$ since it requires the atoms C and D to have "compatible" top-level structures. On the other hand, it is weaker than subsumption w.r.t. the empty ontology, i.e., whenever $C \sqsubseteq D$ holds for two atoms C and D, then $C \sqsubseteq^{\mathsf{s}}_{\mathcal{O}} D$, but not necessarily vice versa. If $\mathcal{O} = \emptyset$, then the three relations $\sqsubseteq$, $\sqsubseteq^{\mathsf{s}}_{\mathcal{O}}$, $\sqsubseteq_{\mathcal{O}}$ coincide on atoms. Like

$\sqsubseteq$ and $\sqsubseteq_\mathcal{O}$, $\sqsubseteq^\mathsf{s}_\mathcal{O}$ is reflexive, transitive, and closed under applying existential restrictions (see [3] for proofs of the results mentioned in this paragraph).

Using the connection between subsumption and rewriting stated in Theorem 5, we can now prove a characterization of subsumption in the presence of an $\mathcal{ELH}_{R^+}$-ontology $\mathcal{O}$ that expresses subsumption in terms of structural subsumptions and derivations w.r.t. $\leftarrow_\mathcal{O}$. Recall that all $\mathcal{EL}$-concept descriptions are conjunctions of atoms, that $C \sqsubseteq_\mathcal{O} D_1 \sqcap \cdots \sqcap D_m$ iff $C \sqsubseteq_\mathcal{O} D_j$ for all $j \in \{1, \ldots, m\}$, and $C \sqsubseteq_\mathcal{O} D$ iff there is an ℓ such that $\mathsf{s}(C) \xleftarrow{(\ell)}_\mathcal{O} \mathsf{s}(D)$.

Lemma 7. *Let $\mathcal{O}$ be a flat $\mathcal{ELH}_{R^+}$-ontology, $C_1, \ldots, C_n$, D be atoms, and $\ell \geq 0$. Then $\mathsf{s}(C_1 \sqcap \cdots \sqcap C_n) \xleftarrow{(\ell)}_\mathcal{O} \mathsf{s}(D)$ iff there is*

1. *an index $i \in \{1, \ldots, n\}$ such that $C_i \sqsubseteq^\mathsf{s}_\mathcal{O} D$; or*
2. *a GCI $A_1 \sqcap \cdots \sqcap A_k \sqsubseteq B$ in $\mathcal{T}$ such that*
   a) for every $p \in \{1, \ldots, k\}$ we have $\mathsf{s}(C_1 \sqcap \cdots \sqcap C_n) \xleftarrow{(\ell-1)}_\mathcal{O} \mathsf{s}(A_p)$,
   b) $\mathsf{s}(C_1 \sqcap \cdots \sqcap C_n) \xleftarrow{(\ell)}_\mathcal{O} \mathsf{s}(B)$, and
   c) $B \sqsubseteq^\mathsf{s}_\mathcal{O} D$.

A detailed proof of this lemma is given in [3]. Here, we only want to point out that this proof makes extensive use of the transitivity of $\sqsubseteq^\mathsf{s}_\mathcal{O}$, and that this is the main reason why we cannot deal with general $\mathcal{EL}^+$-ontologies. In fact, while it is not hard to extend the definition of structural subsumption to more general kinds of ontologies, it is currently not clear to us how to do this such that the resulting relation is transitive; and without transitivity of structural subsumption, we cannot show a characterization analogous to the one in Lemma 7.

5 Reduction of Unification w.r.t. Cycle-Restricted $\mathcal{ELH}_{R^+}$-Ontologies to SAT

The main idea underlying the NP-membership results in [5] and [2] is to show that any $\mathcal{EL}$-unification problem that is unifiable w.r.t. the empty ontology and w.r.t. a cycle-restricted $\mathcal{EL}$-ontology, respectively, has a so-called local unifier. Here, we generalize the notion of a local unifier to the case of unification w.r.t. cycle-restricted $\mathcal{ELH}_{R^+}$-ontologies, but then go a significant step further. Instead of using an algorithm that "blindly" generates all local substitutions and then checks whether they are unifiers, we reduce the search for a local unifier to a propositional satisfiability problem.

Local Unifiers

Let Γ be a flat unification problem and $\mathcal{O}$ be a flat, cycle-restricted $\mathcal{ELH}_{R^+}$-ontology. We denote by At the set of atoms occurring as subdescriptions in subsumptions in Γ or axioms in $\mathcal{O}$ and define

$$\mathsf{At}_\mathsf{tr} := \mathsf{At} \cup \{\exists t.D' \mid \exists s.D' \in \mathsf{At},\ t \trianglelefteq_\mathcal{O} s,\ t \text{ transitive}\}.$$

Furthermore, we define the set of *non-variable atoms* by $\mathsf{At_{nv}} := \mathsf{At_{tr}} \setminus N_v$. Though the elements of $\mathsf{At_{nv}}$ cannot be variables, they may contain variables if they are of the form $\exists r.X$ for some role r and a variable X. We call a function S that associates every variable $X \in N_v$ with a set $S_X \subseteq \mathsf{At_{nv}}$ an *assignment*. Such an assignment induces the following relation $>_S$ on N_v: $>_S$ is the transitive closure of

$$\{(X, Y) \in N_v \times N_v \mid Y \text{ occurs in an element of } S_X\}.$$

We call the assignment S *acyclic* if $>_S$ is irreflexive (and thus a strict partial order). Any acyclic assignment S induces a unique substitution σ_S, which can be defined by induction along $>_S$:

- If X is a minimal element of N_v w.r.t. $>_S$, then we set $\sigma_S(X) := \bigsqcap_{D \in S_X} D$.
- Assume that $\sigma(Y)$ is already defined for all Y such that $X >_S Y$. Then we define $\sigma_S(X) := \bigsqcap_{D \in S_X} \sigma_S(D)$.

We call a substitution σ *local* if it is of this form, i.e., if there is an acyclic assignment S such that $\sigma = \sigma_S$. Since N_v and $\mathsf{At_{nv}}$ are finite, there are only finitely many local substitutions. Thus, if we know that any solvable unification problem has a local unifier, then we can enumerate (or guess, in a nondeterministic machine) all local substitutions and then check whether any of them is a unifier. Thus, in general many substitutions will be generated that only in the subsequent check turn out not to be unifiers. In contrast, our SAT reduction will ensure that only unifiers are generated.

The Reduction

Here, we reduce unification w.r.t. cycle-restricted $\mathcal{ELH}_{R^+}$-ontologies to the satisfiability problem for propositional logic, which is NP-complete. This shows that this unification problem is in NP. But more importantly, it immediately allows us to apply highly optimized SAT solvers for solving such unification problems.

As before, we assume that Γ is a flat unification problem and $\mathcal{O}$ is a flat, cycle-restricted $\mathcal{ELH}_{R^+}$-ontology. Let $\mathcal{T}$ be the subset of $\mathcal{O}$ that consists of the GCIs in $\mathcal{O}$. We define the set

$$\text{Left} := \mathsf{At} \cup \{C_1 \sqcap \cdots \sqcap C_n \mid C_1 \sqcap \cdots \sqcap C_n \sqsubseteq^? D \in \Gamma \text{ for some } D \in \mathsf{At}\}$$

that contains all atoms of Γ and $\mathcal{O}$ and all left-hand sides of subsumptions from Γ. For $L \in \text{Left}$ and $C \in \mathsf{At}$, we write "$C \in L$" if C is a top-level atom of L.

The propositional variables we use for the reduction are of the form $[L \sqsubseteq D]^i$ for $L \in \text{Left}$, $D \in \mathsf{At_{tr}}$, and $i \in \{0, \ldots, |\mathcal{T}|\}$. The intuition underlying these variables is that every satisfying propositional valuation induces an acyclic assignment S such that the following holds for the corresponding substitution σ_S: $[L \sqsubseteq D]^i$ is evaluated to true by the assignment iff $\mathsf{s}(\sigma_S(L))$ can be derived from $\mathsf{s}(\sigma_S(D))$ using at most i root steps, i.e., $\mathsf{s}(\sigma_S(L)) \xleftarrow{(i)}_{\mathcal{O}} \mathsf{s}(\sigma_S(D))$.

Additionally, we use the propositional variables $[X > Y]$ for $X, Y \in N_v$ to express the strict partial order $>_S$ induced by the acyclic assignment S.

The auxiliary function Dec is defined as follows for $C \in \mathsf{At}$, $D \in \mathsf{At}_{\mathsf{tr}}$:

$$\mathrm{Dec}(C \sqsubseteq D) = \begin{cases} \mathbf{1} & \text{if } C = D \\ [C \sqsubseteq D]^{|\mathcal{T}|} & \text{if } C \text{ and } D \text{ are ground} \\ \mathrm{Trans}(C \sqsubseteq D) & \text{if } C = \exists r.C',\ D = \exists s.D', \text{ and } r \trianglelefteq_{\mathcal{O}} s\ , \\ [C \sqsubseteq D]^{|\mathcal{T}|} & \text{if } C \text{ is a variable} \\ \mathbf{0} & \text{otherwise} \end{cases}$$

$$\mathrm{Trans}(C \sqsubseteq D) = [C' \sqsubseteq D']^{|\mathcal{T}|} \vee \bigvee_{\substack{t \text{ transitive} \\ r \trianglelefteq_{\mathcal{O}} t \trianglelefteq_{\mathcal{O}} s}} [C' \sqsubseteq \exists t.D']^{|\mathcal{T}|}.$$

Note that $C' \in \mathsf{At}$ and $D', \exists t.D' \in \mathsf{At}_{\mathsf{tr}}$ by the definition of $\mathsf{At}_{\mathsf{tr}}$ and since Γ and $\mathcal{O}$ are flat. Here, $\mathbf{0}$ and $\mathbf{1}$ are Boolean constants representing the truth values 0 (*false*) and 1 (*true*), respectively.

The unification problem will be reduced to satisfiability of the following set of propositional formulae. For simplicity, we do not use only clauses here. However, our formulae can be transformed into clausal form by introducing polynomially many auxiliary propositional variables and clauses.

Definition 8. *Let Γ be a flat unification problem and $\mathcal{O}$ a flat, cycle-restricted $\mathcal{ELH}_{R^+}$-ontology. The set $C(\Gamma, \mathcal{O})$ contains the following propositional formulae:*

(I) Translation of the subsumptions of Γ. *For every $L \sqsubseteq^? D$ in Γ, we introduce a clause asserting that this subsumption must hold:*

$$\rightarrow [L \sqsubseteq D]^{|\mathcal{T}|}.$$

(II) Translation of the relevant properties of subsumption.

1) For all ground atoms $C \in \mathsf{At}$, $D \in \mathsf{At}_{\mathsf{tr}}$ and $i \in \{0, \dots, |\mathcal{T}|\}$ such that $C \not\sqsubseteq_{\mathcal{O}} D$, we introduce a clause preventing this subsumption:

$$[C \sqsubseteq D]^i \rightarrow \quad .$$

2) For every variable Y, $B \in \mathsf{At}_{\mathsf{nv}}$, $i, j \in \{0, \dots, |\mathcal{T}|\}$, and $L \in \mathrm{Left}$, we introduce the clause

$$[L \sqsubseteq Y]^i \wedge [Y \sqsubseteq B]^j \rightarrow [L \sqsubseteq B]^{\min\{|\mathcal{T}|, i+j\}}.$$

3) For every $L \in \mathrm{Left} \setminus N_v$ and $D \in \mathsf{At}_{\mathsf{tr}}$, we introduce the following formulae, depending on L and D:

a) If D is a ground atom and L is not a ground atom, we introduce

$$[L \sqsubseteq D]^i \rightarrow \bigvee_{C \in L} \mathrm{Dec}(C \sqsubseteq D) \vee \bigvee_{\substack{A_1 \sqcap \dots \sqcap A_k \sqsubseteq B \in \mathcal{O} \\ B \sqsubseteq_{\mathcal{O}} D}} ([L \sqsubseteq A_1]^{i-1} \wedge \dots \wedge [L \sqsubseteq A_k]^{i-1})$$

for all $i \in \{1, \dots, |\mathcal{T}|\}$ and

$$[L \sqsubseteq D]^0 \rightarrow \bigvee_{C \in L} \mathrm{Dec}(C \sqsubseteq D).$$

b) If D is a non-variable, non-ground atom, we introduce

$$[L \sqsubseteq D]^i \to \bigvee_{C \in L} \mathrm{Dec}(C \sqsubseteq D) \vee \bigvee_{A \text{ atom of } \mathcal{O}} ([L \sqsubseteq A]^i \wedge \mathrm{Dec}(A \sqsubseteq D))$$

for all $i \in \{1, \ldots, |\mathcal{T}|\}$ and

$$[L \sqsubseteq D]^0 \to \bigvee_{C \in L} \mathrm{Dec}(C \sqsubseteq D).$$

(III) Translation of the relevant properties of $>$.

1) Transitivity and irreflexivity of $>$ is expressed by the clauses

$$[X > X] \to \quad \text{and } [X > Y] \wedge [Y > Z] \to [X > Z]$$

for all $X, Y, Z \in N_v$.

2) The connection between $>$ and $\sqsubseteq$ is expressed using the clause

$$[X \sqsubseteq \exists r.Y]^i \to [X > Y]$$

for every $X, Y \in N_v$, $\exists r.Y \in \mathsf{At}_{\mathsf{tr}}$, and $i \in \{0, \ldots, |\mathcal{T}|\}$.

It is easy to see that the set $C(\Gamma, \mathcal{O})$ can be constructed in time polynomial in the size of Γ and $\mathcal{O}$. In particular, subsumptions $B \sqsubseteq_{\mathcal{O}} D$ between ground atoms B, D can be checked in polynomial time in the size of $\mathcal{O}$ [4].

There are several differences between $C(\Gamma, \mathcal{O})$ and the clauses constructed in [6] to solve unification in $\mathcal{EL}$ w.r.t. the empty ontology. The propositional variables employed in [6] are of the form $[C \not\sqsubseteq D]$ for atoms C, D of Γ, i.e., they stand for non-subsumption rather than subsumption. The use of single atoms C instead of whole left-hand sides L also leads to a different encoding of the subsumptions from Γ in part (I). The clauses in (III) are identical up to negation of the variables $[X \sqsubseteq \exists r.Y]^i$. But most importantly, in [6] the properties of subsumption expressed in (II) need only deal with subsumption w.r.t. the empty ontology, whereas here we have to take a cycle-restricted $\mathcal{ELH}_{R^+}$-ontology into account. We do this by expressing the characterization of subsumption given in Lemma 7. This is also the reason why the propositional variables $[L \sqsubseteq D]^i$ have an additional index i: in fact, in Lemma 7 we refer to the number of root steps in the derivation that shows the subsumption, and this needs to be modeled in our SAT reduction.

Theorem 9. *The unification problem Γ is solvable w.r.t. $\mathcal{O}$ iff $C(\Gamma, \mathcal{O})$ is satisfiable.*

Since $C(\Gamma, \mathcal{O})$ can be constructed in polynomial time and SAT is in NP, this shows that unification w.r.t. cycle-restricted $\mathcal{ELH}_{R^+}$-ontologies is in NP. NP-hardness follows from the known NP-hardness of $\mathcal{EL}$-unification w.r.t. the empty ontology [5].

Corollary 10. *Unification w.r.t. cycle-restricted $\mathcal{ELH}_{R^+}$-ontologies is an NP-complete problem.*

To prove Theorem 9, we must show soundness and completeness of the reduction.

Soundness of the reduction. Let τ be a valuation of the propositional variables that satisfies $C(\Gamma, \mathcal{O})$. We must show that then Γ has a unifier w.r.t. $\mathcal{O}$. To this purpose, we use τ to define an assignment S by

$$S_X := \{D \in \mathsf{At}_{\mathsf{nv}} \mid \exists i \in \{0, \ldots, |\mathcal{T}|\} : \tau([X \sqsubseteq D]^i) = 1\}.$$

Using the clauses in (III), it is not hard to show [3] that $X >_S Y$ implies $\tau([X > Y]) = 1$. Due to the irreflexivity clause in (III), this yields that the assignment S is acyclic. Thus, it induces a substitution σ_S. A proof of the following lemma can be found in [3].

Lemma 11. *If $\tau([L \sqsubseteq D]^i) = 1$ for $L \in$ Left, $D \in \mathsf{At}_{\mathsf{tr}}$, and $i \in \{0, \ldots, |\mathcal{T}|\}$, then $\sigma_S(L) \sqsubseteq_{\mathcal{O}} \sigma_S(D)$.*

Because of the clauses in (I), this lemma immediately implies that σ_S is a unifier of Γ w.r.t. $\mathcal{O}$.

Completeness of the reduction. Given a unifier γ of Γ w.r.t. $\mathcal{O}$, we can define a valuation τ that satisfies $C(\Gamma, \mathcal{O})$ as follows.

Let $L \in$ Left and $D \in \mathsf{At}_{\mathsf{tr}}$ and $i \in \{0, \ldots, |\mathcal{T}|\}$. We set $\tau([L \sqsubseteq D]^i) := 1$ iff $\mathsf{s}(\gamma(L)) \xleftarrow{(i)}_{\mathcal{O}} \mathsf{s}(\gamma(D))$. According to Theorem 5, we thus have $\tau([L \sqsubseteq D]^i) = 0$ for all $i \in \{0, \ldots, |\mathcal{T}|\}$ iff $\gamma(L) \not\sqsubseteq_{\mathcal{O}} \gamma(D)$. Otherwise, there is an $i \in \{0, \ldots, |\mathcal{T}|\}$ such that $\tau([L \sqsubseteq D]^j) = 1$ for all $j \geq i$, and $\tau([L \sqsubseteq D]^j) = 0$ for all $j < i$.

To define the valuation of the remaining propositional variables $[X > Y]$ with $X, Y \in N_v$, we set $\tau([X > Y]) = 1$ iff $X >_\gamma Y$, where $>_\gamma$ is defined as in (5), i.e., $X >_\gamma Y$ iff $\gamma(X) \sqsubseteq_{\mathcal{O}} \exists w.\gamma(Y)$ for some $w \in N_R^+$.

The following lemma, whose proof can be found in [3], shows completeness of our reduction using Lemma 7.

Lemma 12. *The valuation τ satisfies $C(\Gamma, \mathcal{O})$.*

Note that cycle-restrictedness of $\mathcal{O}$ is needed in order to satisfy the irreflexivity clause $[X > X] \rightarrow$ (see Lemma 3). We cannot dispense with this clause since it is needed in the proof of soundness to obtain acyclicity of the assignment S constructed there. In fact, only because S is acyclic can we define the substitution σ_S, which is then shown to be a unifier.

6 Conclusions

We have shown that unification w.r.t. cycle-restricted $\mathcal{ELH}_{R^+}$-ontologies can be reduced to propositional satisfiability. This improves on the results in [1,2] in two respects. First, it allows us to deal also with ontologies that contain transitivity and role hierarchy axioms, which are important for medical ontologies. Second, the SAT reduction can easily be implemented and enables us to make use of highly optimized SAT solvers, whereas the goal-oriented algorithm in [1], while having the potential of becoming quite efficient, requires a high amount of additional optimization work. The main topic for future research is to investigate whether we can get rid of cycle-restrictedness.

References

1. Baader, F., Borgwardt, S., Morawska, B.: Unification in the description logic $\mathcal{EL}$ w.r.t. cycle-restricted TBoxes. LTCS-Report 11-05, Theoretical Computer Science, TU Dresden (2011), see http://lat.inf.tu-dresden.de/research/reports.html.
2. Baader, F., Borgwardt, S., Morawska, B.: Extending unification in $\mathcal{EL}$ towards general TBoxes. In Proc. of the 13th Int. Conf. on Principles of Knowledge Representation and Reasoning. AAAI Press (2012), short paper.
3. Baader, F., Borgwardt, S., Morawska, B.: SAT encoding of unification in $\mathcal{ELH}_{R^+}$ w.r.t. cycle-restricted ontologies. LTCS-Report 12-02, Theoretical Computer Science, TU Dresden (2012), see http://lat.inf.tu-dresden.de/research/reports.html.
4. Baader, F., Brandt, S., Lutz, C.: Pushing the $\mathcal{EL}$ envelope. In: Kaelbling, L.P., Saffiotti, A. (eds.) Proc. of the 19th Int. Joint Conf. on Artificial Intelligence. pp. 364–369. Morgan Kaufmann, Los Altos, Edinburgh (UK) (2005)
5. Baader, F., Morawska, B.: Unification in the description logic $\mathcal{EL}$. In: Treinen, R. (ed.) Proc. of the 20th Int. Conf. on Rewriting Techniques and Applications. LNCS, vol. 5595, pp. 350–364. Springer-Verlag (2009)
6. Baader, F., Morawska, B.: SAT encoding of unification in $\mathcal{EL}$. In: Fermüller, C.G., Voronkov, A. (eds.) Proc. of the 17th Int. Conf. on Logic for Programming, Artificial Intelligence, and Reasoning. LNCS, vol. 6397, pp. 97–111. Springer-Verlag (2010)
7. Baader, F., Morawska, B.: Unification in the description logic $\mathcal{EL}$. Logical Methods in Computer Science 6(3) (2010)
8. Baader, F., Narendran, P.: Unification of concept terms in description logics. J. of Symbolic Computation 31(3), 277–305 (2001)
9. Baader, F., Snyder, W.: Unification theory. In: Robinson, J.A., Voronkov, A. (eds.) Handbook of Automated Reasoning, pp. 445–532. The MIT Press (2001)
10. Biere, A., Heule, M., van Maaren, H., Walsh, T. (eds.): Handbook of Satisfiability. IOS Press (2009)
11. 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.) Proc. of the 16th Eur. Conf. on Artificial Intelligence. pp. 298–302 (2004)
12. Campbell, J.R., Lopez Osornio, A., de Quiros, F., Luna, D., Reynoso, G.: Semantic interoperability and SNOMED CT: A case study in clinical problem lists. In: Kuhn, K., Warren, J., Leong, T.Y. (eds.) Proc. of the 12th World Congress on Health (Medical) Informatics. pp. 2401–2402. IOS Press (2007)
13. Degtyarev, A., Voronkov, A.: The undecidability of simultaneous rigid E-unification. Theor. Comput. Sci. 166(1&2), 291–300 (1996)
14. Gallier, J.H., Narendran, P., Plaisted, D.A., Snyder, W.: Rigid E-unification: NP-completeness and applications to equational matings. Inf. Comput. 87(1/2), 129–195 (1990)
15. Seidenberg, J., Rector, A.L.: Representing transitive propagation in owl. In: Embley, D.W., Olivé, A., Ram, S. (eds.) Proc. of the 25th International Conference on Conceptual Modeling. LNCS, vol. 4215, pp. 255–266. Springer-Verlag (2006)
16. Sofronie-Stokkermans, V.: Locality and subsumption testing in $\mathcal{EL}$ and some of its extensions. In: Proc. Advances in Modal Logic (2008)
17. Suntisrivaraporn, B., Baader, F., Schulz, S., Spackman, K.: Replacing SEP-triplets in SNOMED CT using tractable description logic operators. In: Bellazzi, R., Abu-Hanna, A., Hunter, J. (eds.) Proceedings of the 11th Conference on Artificial Intelligence in Medicine. LNCS, vol. 4594, pp. 287–291. Springer-Verlag (2007)