Hybrid Unification in the Description Logic $\mathcal{EL}$

Franz Baader, Oliver Fernández Gil and Barbara Morawska*
{baader,morawska}@tcs.inf.tu-dresden.de, olitof11@gmail.com

Theoretical Computer Science, TU Dresden, Germany

1 Introduction

The description logic $\mathcal{EL}$ belongs to the family of logic-based knowledge representation formalisms. It allows a user to define concepts with the help of *concept names* (N_c), *role names* (N_r) and constructors: conjunction ($\sqcap$), existential restriction ($\exists r.C$ for $r \in N_r$ and a concept C) and top constructor ($\top$).

Unification in Description Logics has been introduced in [6]. A unification problem in such logic is defined as a set of subsumptions between concepts which contain occurrences of a distinct set of concept names (called *variables*) and asks for definitions of these concept names, which would make the subsumptions valid.

Unification in $\mathcal{EL}$ corresponds to unification modulo semilattices with monotone operators [5]. In [4], we were able to show that unification in $\mathcal{EL}$ is NP-complete. The problem is how to extend the unification in $\mathcal{EL}$ to such unification with a background ontology in the form of a set of definitions of some concept names occuring in the unification problem, or more generally in the form of additional statements about concept inclusions. If the background ontology is just a set of non-cyclic definitions, unification in $\mathcal{EL}$ is NP-complete [5]. If the background ontology satisfies some cycle restriction, it is still NP-complete [2]. At the moment it is not known what is the status of the unification problem in $\mathcal{EL}$ with a background ontology in the general case.

In this paper, instead of restricting the background ontology, we allow cyclic definitions to be used as unifiers. Moreover, we interpret these definitions in a *greatest fixpoint* semantics, while the background ontology is still interpreted in the usual *descriptive* semantics. We show that if the concept of unification in $\mathcal{EL}$ is modified in this way, such unification is NP-complete. Detailed proofs and examples can be found in [3].

2 The Description Logic $\mathcal{EL}$

Concept descriptions written in the language of $\mathcal{EL}$ are interpreted over an *interpretation* $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ which 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 inductively extended to concept descriptions as follows:
$\top^{\mathcal{I}} := \Delta^{\mathcal{I}}$, $(C \sqcap D)^{\mathcal{I}} := C^{\mathcal{I}} \cap D^{\mathcal{I}}$, $(\exists r.C)^{\mathcal{I}} := \{x \mid \exists y : (x,y) \in r^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\}$

A *concept definition* is an expression of the form $X \equiv C$ where X is a concept name and C is a concept description, and a *general concept inclusion* (GCI) is an expression of the form $C \sqsubseteq D$, where C, D are concept descriptions. An interpretation $\mathcal{I}$ is a *model* of this concept definition (this GCI) if it satisfies $X^{\mathcal{I}} = C^{\mathcal{I}}$ ($C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$). This semantics for GCIs and concept definitions is usually called *descriptive semantics*.

A *TBox* is a finite set $\mathcal{T}$ of concept definitions that does not contain multiple definitions of the same concept name. Note that we do *not* prohibit cyclic dependencies among the concept

*Supported by DFG under grant BA 1122/14-2

definitions in a TBox. An *acyclic TBox* is a TBox without cyclic dependencies. An *ontology* is a finite set of GCIs. The interpretation $\mathcal{I}$ is a *model* of a TBox (ontology) iff it is a model of all concept definitions (GCIs) contained in it.

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 $\mathcal{O}$ is also a model of 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 $D \sqsubseteq_{\mathcal{O}} C$. As shown in [7], subsumption w.r.t. $\mathcal{EL}$-ontologies is decidable in polynomial time.

3 Hybrid Ontologies

We assume that the set of concept names N_C is partitioned into the set of *primitive concepts* N_{prim} and the set of *defined concepts* N_{def}.

Definition 1 (Hybrid $\mathcal{EL}$-ontologies)**.** A *hybrid $\mathcal{EL}$-ontology* is a pair $(\mathcal{O}, \mathcal{T})$, where $\mathcal{O}$ is an $\mathcal{EL}$-ontology containing only concept names from N_{prim}, and $\mathcal{T}$ is a (possibly cyclic) $\mathcal{EL}$-TBox such that $X \equiv C \in \mathcal{T}$ if and only if $X \in N_{def}$.

A *primitive interpretation* $\mathcal{J}$ is defined like an interpretation, with the only difference that it does not provide an interpretation for the defined concepts.

Given a primitive interpretation $\mathcal{J}$, we say that the (full) interpretation $\mathcal{I}$ is *based on* $\mathcal{J}$ if it has the same domain as $\mathcal{J}$ and its interpretation function coincides with $\mathcal{J}$ on N_{prim} and N_r.

Given two interpretations $\mathcal{I}_1$ and $\mathcal{I}_2$ based on the same primitive interpretation $\mathcal{J}$, we define $\mathcal{I}_1 \preceq_{\mathcal{J}} \mathcal{I}_2$ iff $X^{\mathcal{I}_1} \subseteq X^{\mathcal{I}_2}$ for all $X \in N_{def}$.

It is easy to see that the relation $\preceq_{\mathcal{J}}$ is a partial order on the set of interpretations based on $\mathcal{J}$. In [1] the following was shown: given an $\mathcal{EL}$-TBox $\mathcal{T}$ and a primitive interpretation $\mathcal{J}$, there exists a unique model $\mathcal{I}$ of $\mathcal{T}$ such that

- $\mathcal{I}$ is based on $\mathcal{J}$;
- $\mathcal{I}' \preceq_{\mathcal{J}} \mathcal{I}$ for all models $\mathcal{I}'$ of $\mathcal{T}$ that are based on $\mathcal{J}$.

We call such a model $\mathcal{I}$ a *gfp-model* of $\mathcal{T}$.

Definition 2 (Semantics of hybrid $\mathcal{EL}$-ontologies)**.** The interpretation $\mathcal{I}$ is a *hybrid model* of the hybrid $\mathcal{EL}$-ontology $(\mathcal{O}, \mathcal{T})$ iff $\mathcal{I}$ is a gfp-model of $\mathcal{T}$ and the primitive interpretation $\mathcal{J}$ it is based on is a model of $\mathcal{O}$.

It is well-known that gfp-semantics coincides with descriptive semantics for acyclic TBoxes.

Let $(\mathcal{O}, \mathcal{T})$ be a hybrid $\mathcal{EL}$-ontology and C, D $\mathcal{EL}$-concept descriptions. Then C *is subsumed by D w.r.t.* $(\mathcal{O}, \mathcal{T})$ (written $C \sqsubseteq_{gfp,\mathcal{O},\mathcal{T}} D$) iff every hybrid model of $(\mathcal{O}, \mathcal{T})$ is also a model of the GCI $C \sqsubseteq D$. As shown in [8, 10], subsumption w.r.t. hybrid $\mathcal{EL}$-ontologies is decidable in polynomial time.

Our algorithms for hybrid unification in $\mathcal{EL}$ are based on the Gentzen style calculus HC$(\mathcal{O}, \mathcal{T}, \Delta)$ from [10]. HC$(\mathcal{O}, \mathcal{T}, \Delta)$ is parametrized by a hybrid ontology $(\mathcal{O}, \mathcal{T})$ and a set of subsumptions Δ. It decides if $C \sqsubseteq_{gfp,\mathcal{O},\mathcal{T}} D$ holds where C, D are concept descriptions occurring in Δ.

4 Hybrid unification in $\mathcal{EL}$

Definition 3. Let $\mathcal{O}$ be an $\mathcal{EL}$-ontology containing only concept names from N_{prim}. An $\mathcal{EL}$-*unification problem* w.r.t. $\mathcal{O}$ is a finite set of GCIs $\Gamma = \{C_1 \sqsubseteq D_1, \ldots, C_n \sqsubseteq D_n\}$ (which may also contain concept names from N_{def}). The TBox $\mathcal{T}$ is a *hybrid unifier* of Γ w.r.t. $\mathcal{O}$ if $(\mathcal{O}, \mathcal{T})$ is a hybrid $\mathcal{EL}$-ontology that entails all the GCIs in Γ, i.e. , $C_1 \sqsubseteq_{gfp,\mathcal{O},\mathcal{T}} D_1, \ldots, C_n \sqsubseteq_{gfp,\mathcal{O},\mathcal{T}} D_n$. We call such a TBox $\mathcal{T}$ a *classical unifier* of Γ w.r.t. $\mathcal{O}$ if $\mathcal{T}$ is acyclic.

Notice that N_{prim} and N_{def} respectively correspond to the sets of concept constants and concept variables in previous papers on unification in DLs. A substitution σ can be expressed as concept definitions $X \equiv E$ in a corresponding acyclic TBox. In contrast, hybrid unifiers cannot be translated into substitutions since the unfolding process would not terminate for a cyclic TBox.

Our hybrid unification algorithm works on a *flat* unification problem and assumes a *flattened* ontology. In order to define this form we need the following notions.

An *atom* is a concept name or an existential restriction. 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 or $\exists r.\top$.

The GCI $C \sqsubseteq D$ is called *flat* if C is a conjunction of $n \geq 0$ flat atoms and D is a flat atom. The unification problem Γ w.r.t. the ontology $\mathcal{O}$ is called *flat* if both Γ and $\mathcal{O}$ consist of flat GCIs.

Given a unification problem Γ w.r.t. an ontology $\mathcal{O}$, we can compute in polynomial time (see [3]) a flat ontology $\mathcal{O}'$ and a flat unification problem Γ' such that Γ has a (hybrid or classical) unifier w.r.t. $\mathcal{O}$ iff Γ' has a (hybrid or classical) unifier w.r.t. $\mathcal{O}'$. For this reason, we will assume in the following that all unification problems are flat.

The main reason why hybrid unification in $\mathcal{EL}$ is in NP is that any unification problem that has a unifier also has a local unifier. For classical unification w.r.t. background ontologies this is only true if the background ontology is cycle-restricted [2].

Given a flat unification problem Γ w.r.t. an ontology $\mathcal{O}$, we denote by At the set of atoms occurring as sub-descriptions in GCIs in Γ or $\mathcal{O}$. The set of *non-variable atoms* is defined as by $\mathrm{At}_{\mathsf{nv}} := \mathrm{At} \setminus N_{def}$.

In order to define local unifiers, we consider assignments ζ of subsets ζ_X of $\mathrm{At}_{\mathsf{nv}}$ to defined concepts $X \in N_{def}$. Such an assignment induces a TBox

$$T_\zeta := \{X \equiv \bigsqcap_{D \in \zeta_X} D \mid X \in N_{def}\}.$$

We call such a TBox *local*. The (hybrid or classical) unifier $\mathcal{T}$ of Γ w.r.t. $\mathcal{O}$ is called *local unifier* if $\mathcal{T}$ is local, i.e., there is an assignment ζ such that $\mathcal{T} = T_\zeta$.

5 Hybrid $\mathcal{EL}$-unification is NP-complete

The fact that hybrid $\mathcal{EL}$-unification w.r.t. arbitrary $\mathcal{EL}$-ontologies is in NP is an easy consequence of the following proposition.

Proposition 4. *Consider a flat $\mathcal{EL}$-unification problem Γ w.r.t. an $\mathcal{EL}$-ontology $\mathcal{O}$. If Γ has a hybrid unifier w.r.t. $\mathcal{O}$ then it has a* local *hybrid unifier w.r.t. $\mathcal{O}$.*

In fact, the NP-algorithm simply guesses a local TBox and then checks (using the polynomial-time algorithm for hybrid subsumption) whether it is a hybrid unifier.

To prove the proposition, we assume that $\mathcal{T}$ is a hybrid unifier of Γ w.r.t. $\mathcal{O}$. We use this unifier to define an assignment $\zeta^{\mathcal{T}}$ as follows:

$$\zeta^{\mathcal{T}}_X := \{D \in \mathrm{At}_{\mathsf{nv}} \mid X \sqsubseteq_{gfp,\mathcal{O},\mathcal{T}} D\}.$$

Let $\mathcal{T}'$ be the TBox induced by this assignment. To show that $\mathcal{T}'$ is indeed a hybrid unifier of Γ w.r.t. $\mathcal{O}$, we consider the set of GCIs

$$\Delta := \{C_1 \sqcap \ldots \sqcap C_m \sqsubseteq D \mid C_1, \ldots, C_m, D \in \mathrm{At}\},$$

and show that, for any GCI $C_1 \sqcap \ldots \sqcap C_m \sqsubseteq D \in \Delta$, a proof of $C_1 \sqcap \ldots \sqcap C_m \sqsubseteq D$ by $\mathrm{HC}(\mathcal{O}, \mathcal{T}, \Delta)$ implies a proof of $C_1 \sqcap \ldots \sqcap C_m \sqsubseteq D$ also in $\mathrm{HC}(\mathcal{O}, \mathcal{T}', \Delta)$.

NP-hardness does *not* follow directly from NP-hardness of classical $\mathcal{EL}$-unification. In fact an $\mathcal{EL}$-unification problem that does not have a classical unifier may well have a hybrid unifier. Instead, we reduce $\mathcal{EL}$-matching modulo equivalence to hybrid $\mathcal{EL}$-unification.

An *$\mathcal{EL}$-matching problem modulo equivalence* is an $\mathcal{EL}$-unification problem of the form $\{C \sqsubseteq D, D \sqsubseteq C\}$ such that D does not contain elements of N_{def}. A *matcher* of such a problem is a classical unifier of it. As shown in [9], testing whether a matching problem modulo equivalence has a matcher or not is an NP-complete problem. Thus, NP-hardness of hybrid $\mathcal{EL}$-unification w.r.t. $\mathcal{EL}$-ontologies is an immediate consequence of the following lemma, whose (non-trivial) proof can be found in [3].

Lemma 5. *If an $\mathcal{EL}$-matching problem modulo equivalence has a hybrid unifier w.r.t. the empty ontology, then it also has a matcher.*

To sum up, we have thus determined the exact worst-case complexity of hybrid $\mathcal{EL}$-unification.

Theorem 6. *The problem of testing whether an $\mathcal{EL}$-unification problem w.r.t. an arbitrary $\mathcal{EL}$-ontology has a hybrid unifier or not is* NP*-complete.*

6 Conclusions

In this paper, we have proved that hybrid $\mathcal{EL}$-unification w.r.t. arbitrary $\mathcal{EL}$-ontologies is NP-complete. In [3] we have developed also a goal-oriented NP-algorithm for hybrid $\mathcal{EL}$-unification that is better than the brute-force "guess and then test" algorithm used to show the "in NP" result. The decidability and complexity of classical $\mathcal{EL}$-unification w.r.t. arbitrary $\mathcal{EL}$-ontologies is an important topic for future research. We hope that hybrid unification may also be helpful in this context.

References

[1] Baader, F.: Terminological cycles in a description logic with existential restrictions. In: Gottlob, G., Walsh, T. (eds.) Proc. of the 18th Int. Joint Conf. on Artificial Intelligence (IJCAI 2003). pp. 325–330. Morgan Kaufmann, Los Altos, Acapulco, Mexico (2003)

[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 (KR 2012). pp. 568–572. AAAI Press/The MIT Press (2012)

[3] Baader, F., Fernández Gil, O., Morawska, B.: Hybrid unification in the description logic $\mathcal{EL}$. LTCS-Report 13-07, Chair for Automata Theory, Institute for Theoretical Computer Science, Technische Universität Dresden, Dresden, Germany (2013), see http://lat.inf.tu-dresden.de/research/reports.html.

[4] 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 (RTA 2009). Lecture Notes in Computer Science, vol. 5595, pp. 350–364. Springer-Verlag (2009)

[5] Baader, F., Morawska, B.: Unification in the description logic $\mathcal{EL}$. Logical Methods in Computer Science 6(3) (2010)

[6] Baader, F., Narendran, P.: Unification of concept terms in description logics. J. of Symbolic Computation 31(3), 277–305 (2001)

[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.) Proc. of the 16th Eur. Conf. on Artificial Intelligence (ECAI 2004). pp. 298–302 (2004)

[8] Brandt, S., Model, J.: Subsumption in $\mathcal{EL}$ w.r.t. hybrid tboxes. In: Proc. of the 28th German Annual Conf. on Artificial Intelligence (KI'05). pp. 34–48. Lecture Notes in Artificial Intelligence, Springer-Verlag (2005)

[9] Küsters, R.: Non-standard Inferences in Description Logics, Lecture Notes in Artificial Intelligence, vol. 2100. Springer-Verlag (2001)

[10] Novakovic, N.: A proof-theoretic approach to deciding subsumption and computing least common subsumer in w.r.t. hybrid TBoxes. In: Hölldobler, S., Lutz, C., Wansing, H. (eds.) Proc. of the 11th Eur. Conf. on Logics in Artificial Intelligence (JELIA'2004). Lecture Notes in Computer Science, vol. 5293, pp. 311–323. Springer-Verlag (2008)