Dismatching and Local Disunification in $\mathcal{EL}$ (Extended Abstract)

Franz Baader, Stefan Borgwardt, and Barbara Morawska[⋆]

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

Motivation

Unification in Description Logics was introduced in [6] as a novel inference service that can 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 head injury* using the $\mathcal{EL}$-concepts

$$\mathsf{Patient} \sqcap \exists \mathsf{finding}.(\mathsf{Head\_injury} \sqcap \exists \mathsf{severity}.\mathsf{Severe}), \qquad (1)$$

whereas another one represents it as

$$\mathsf{Patient} \sqcap \exists \mathsf{finding}.(\mathsf{Severe\_finding} \sqcap \mathsf{Injury} \sqcap \exists \mathsf{finding\_site}.\mathsf{Head}). \qquad (2)$$

Formally, these expressions are not equivalent, but they are nevertheless meant to represent the same concept. They can obviously be made equivalent by treating the concept names $\mathsf{Head\_injury}$ and $\mathsf{Severe\_finding}$ as variables, and substituting them by $\mathsf{Injury} \sqcap \exists \mathsf{finding\_site}.\mathsf{Head}$ and $\exists \mathsf{severity}.\mathsf{Severe}$, respectively. In this case, we say that the concepts are unifiable, and call the substitution that makes them equivalent a *unifier*. In [5], we were able to show that unification in $\mathcal{EL}$ is NP-complete. The main idea underlying the proof of this result is to show that any solvable $\mathcal{EL}$-unification problem has a local unifier, i.e., a unifier built from a polynomial number of *atoms* (concept names or existential restrictions), which are determined by the unification problem. This yields a brute-force NP-algorithm for unification, which guesses a local substitution and then checks (in polynomial time) whether it is a unifier.

Intuitively, a unifier of two $\mathcal{EL}$ concepts proposes definitions for the concept names that are used as variables: in our example, we know that, if we define $\mathsf{Head\_injury}$ as $\mathsf{Injury} \sqcap \exists \mathsf{finding\_site}.\mathsf{Head}$ and $\mathsf{Severe\_finding}$ as $\exists \mathsf{severity}.\mathsf{Severe}$, then the two concepts (1) and (2) are equivalent w.r.t. these definitions. Of course, this example was constructed such that the unifier (which is local) provides sensible definitions for the concept names used as variables. In general, the existence of a unifier only says that there is a structural similarity between the two concepts. The developer that uses unification needs to inspect the unifier(s) to see whether the definitions it suggests really make sense. For example, the substitution that replaces $\mathsf{Head\_injury}$ by $\mathsf{Patient} \sqcap \mathsf{Injury} \sqcap \exists \mathsf{finding\_site}.\mathsf{Head}$ and $\mathsf{Severe\_finding}$ by $\mathsf{Patient} \sqcap \exists \mathsf{severity}.\mathsf{Severe}$ is also a local unifier, which however does not make sense. Unfortunately, even small unification problems like the one

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

in our example can have too many local unifiers for manual inspection. We propose disunification to avoid local unifiers that do not make sense. In addition to positive constraints (requiring equivalence or subsumption between concepts), a disunification problem may also contain negative constraints (preventing equivalence or subsumption between concepts). In our example, the nonsensical unifier can be avoided by adding the dissubsumption constraint

$$\mathsf{Head\_injury} \not\sqsubseteq^? \mathsf{Patient} \tag{3}$$

to the equivalence constraint (1) $\equiv^?$ (2).

Disunification in DLs is closely related to unification and admissibility in modal logics [7,10–15], as well as (dis)unification modulo equational theories [5,6,8,9]. In the following, we shortly describe the ideas behind our work. More details can be found in [2,3].

Preliminaries

We designate certain concept names as *variables*, while all others are *constants*. An $\mathcal{EL}$-concept is *ground* if it contains no variables. We consider *(basic) disunification problems*, which are conjunctions of *subsumptions* $C \sqsubseteq^? D$ and *dissubsumptions* $C \not\sqsubseteq^? D$ between concepts C, D.[1] A *substitution* maps each variable to a ground concept, and can be extended to concepts as usual. A substitution σ *solves* a disunification problem Γ if the (dis)subsumptions of Γ become valid when applying σ on both sides. We restrict σ to a finite signature of concept and role names and do not allow variables to occur in a solution—it would not make sense for the new definitions to extend the vocabulary of the ontologies under consideration, nor to define variables in terms of themselves.

In the following, we consider a *flat* disunification problem Γ, i.e. one that contains only (dis)subsumptions where both sides are conjunctions of *flat atoms* of the form A or $\exists r.A$, for a role name r and a concept name A. We denote by At the set of all such atoms that occur in Γ, by Var the set of variables occurring in Γ, and by $\mathsf{At}_{\mathsf{nv}} := \mathsf{At} \setminus \mathsf{Var}$ the set of *non-variable atoms* of Γ. Let $S\colon \mathsf{Var} \to 2^{\mathsf{At}_{\mathsf{nv}}}$ be an *assignment*, i.e. a function that assigns to each variable $X \in \mathsf{Var}$ a set $S_X \subseteq \mathsf{At}_{\mathsf{nv}}$. The relation $>_S$ on Var is defined as the transitive closure of $\{(X,Y) \in \mathsf{Var}^2 \mid Y \text{ occurs in an atom of } S_X\}$. If $>_S$ is irreflexive, then S is called *acyclic*. In this case, we can define the substitution σ_S inductively along $>_S$ as follows: if X is minimal, then $\sigma_S(X) := \bigsqcap_{D \in S_X} D$; otherwise, assume that $\sigma_S(Y)$ is defined for all $Y \in \mathsf{Var}$ with $X > Y$, and define $\sigma_S(X) := \bigsqcap_{D \in S_X} \sigma_S(D)$. All substitutions of this form are called *local*.

Results

Unification in $\mathcal{EL}$ is *local*: each problem Γ can be transformed into an equivalent flat problem that has a local solution iff Γ is solvable, and hence (general) solvability of unification problems in $\mathcal{EL}$ is in NP [5]. However, disunification in $\mathcal{EL}$

[1] A *unification problem* contains only subsumptions.

is *not local* in this sense: consider

$$\Gamma := \{X \sqsubseteq^? B,\ A \sqcap B \sqcap C \sqsubseteq^? X,\ \exists r.X \sqsubseteq^? Y,\ \top \not\sqsubseteq^? Y,\ Y \not\sqsubseteq^? \exists r.B\}$$

with variables X, Y and constants A, B, C. If we set $\sigma(X) := A \sqcap B \sqcap C$ and $\sigma(Y) := \exists r.(A \sqcap C)$, then σ is a solution of Γ that is not local. This is because $\exists r.(A \sqcap C)$ is not a substitution of any non-variable atom in Γ. Assume now that Γ has a local solution γ. Since γ must solve the first dissubsumption, $\gamma(Y)$ cannot be $\top$, and due to the third subsumption, none of the constants A, B, C can be a conjunct of $\gamma(Y)$. The remaining atoms $\exists r.\gamma(X)$ and $\exists r.B$ are ruled out by the last dissubsumption since both $\gamma(X)$ and B are subsumed by B. This shows that Γ cannot have a local solution, although it is solvable.

The decidability and complexity of general solvability of disunification problems is still open. However, we can show that each *dismatching* problem Γ, which is a disunification problem where one side of each dissubsumption must be ground, can be polynomially reduced to a flat problem that has a local solution iff Γ is solvable. This shows that deciding solvability of dismatching problems in $\mathcal{EL}$ is in NP. The main idea is to introduce auxiliary variables and flat atoms that allow us to solve the dissubsumptions using a local substitution. For example, we replace the dissubsumption $\top \not\sqsubseteq^? Y$ from above with $Y \sqsubseteq^? \exists r.Z$. The rule we applied here is the following:

Solving Left-Ground Dissubsumptions:

Condition: This rule applies to $\mathfrak{s} = C_1 \sqcap \cdots \sqcap C_n \not\sqsubseteq^? X$ if X is a variable and $C_1, \ldots, C_n$ are ground atoms.
Action: Choose one of the following options:
- Choose a constant $A \in \Sigma$, replace $\mathfrak{s}$ by $X \sqsubseteq^? A$. If $C_1 \sqcap \cdots \sqcap C_n \sqsubseteq A$, then *fail*.
- Choose a role $r \in \Sigma$, introduce a new variable Z, replace $\mathfrak{s}$ by $X \sqsubseteq^? \exists r.Z$, $C_1 \not\sqsubseteq^? \exists r.Z, \ldots, C_n \not\sqsubseteq^? \exists r.Z$.

According to the rule, we can choose a constant or create a new existential restriction with a fresh variable, and use it in the new subsumption and dissubsumptions. In our example the left hand side of the dissubsumption $\top \not\sqsubseteq^? Y$ is empty, hence only a subsumption is produced.

However, the brute-force NP-algorithm for checking local solvability of the resulting flat disunification problem is hardly practical. For this reason, we have extended the rule-based algorithm from [5] and the SAT reduction from [4] by additional rules and propositional clauses, respectively, to deal with dissubsumptions. The reason we extend both algorithms is that, in the case of unification, they have proved to complement each other well in first evaluations [1]: the goal-oriented algorithm needs less memory and finds minimal solutions faster, while the SAT reduction generates larger data structures, but outperforms the goal-oriented algorithm on unsolvable problems. The SAT reduction has been implemented in our prototype system UEL.[2] First experiments show that dismatching is indeed helpful for reducing the number and the size of unifiers. The runtime performance of the solver for dismatching problems is comparable to the one for pure unification problems.

[2] version 1.3.0, available at http://uel.sourceforge.net/

References

1. Baader, F., Borgwardt, S., Mendez, J.A., Morawska, B.: UEL: Unification solver for $\mathcal{EL}$. In: Kazakov, Y., Lembo, D., Wolter, F. (eds.) Proc. of the 25th Int. Workshop on Description Logics (DL'12). CEUR Workshop Proceedings, vol. 846, pp. 26–36 (2012)
2. Baader, F., Borgwardt, S., Morawska, B.: Dismatching and local disunfication in $\mathcal{EL}$. LTCS-Report 15-03, Chair for Automata Theory, TU Dresden, Germany (2015), see http://lat.inf.tu-dresden.de/research/reports.html.
3. Baader, F., Borgwardt, S., Morawska, B.: Dismatching and local disunification in $\mathcal{EL}$. In: Fernández, M. (ed.) Proc. of the 26th Int. Conf. on Rewriting Techniques and Applications (RTA'15). LIPIcs, vol. 36. Dagstuhl Publishing (2015), to appear.
4. 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 (LPAR'10). Lecture Notes in Computer Science, vol. 6397, pp. 97–111. Springer-Verlag (2010)
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. Babenyshev, S., Rybakov, V.V., Schmidt, R., Tishkovsky, D.: A tableau method for checking rule admissibility in $S4$. In: Proc. of the 6th Workshop on Methods for Modalities (M4M-6). Copenhagen (2009)
8. Buntine, W.L., Bürckert, H.J.: On solving equations and disequations. J. of the ACM 41(4), 591–629 (1994)
9. Comon, H.: Disunification: A survey. In: Lassez, J.L., Plotkin, G. (eds.) Computational Logic: Essays in Honor of Alan Robinson, pp. 322–359. MIT Press, Cambridge, MA (1991)
10. Ghilardi, S.: Unification through projectivity. Journal of Logic and Computation 7(6), 733–752 (1997)
11. Ghilardi, S.: Unification in intuitionistic logic. Journal of Logic and Computation 64(2), 859–880 (1999)
12. Iemhoff, R., Metcalfe, G.: Proof theory for admissible rules. Annals of Pure and Applied Logic 159(1-2), 171–186 (2009)
13. Rybakov, V.V.: Admissibility of logical inference rules, Studies in Logic and the Foundations of Mathematics, vol. 136. North-Holland Publishing Co., Amsterdam (1997)
14. Rybakov, V.V.: Multi-modal and temporal logics with universal formula - reduction of admissibility to validity and unification. Journal of Logic and Computation 18(4), 509–519 (2008)
15. Wolter, F., Zakharyaschev, M.: Undecidability of the unification and admissibility problems for modal and description logics. ACM Transactions on Computational Logic 9(4), 25:1–25:20 (2008)