Unification in the Description Logic $\mathcal{FL}_\bot$ ⋆

Barbara Morawska

Ahmedabad University, India, barbara.morawska@ahduni.edu.in

Abstract. The paper presents a proof that the unification in the description logic $\mathcal{FL}_\bot$ is decidable in ExpTime. $\mathcal{FL}_\bot$ adds to the constructors of the description logic $\mathcal{FL}_0$, bottom (inconsistency) which allows us to express a kind of negation. The result is obtained by a careful elimination of $\bot$ (flattening) from a given unification problem and solving the remaining problem with the $\mathcal{FL}_0$-unification procedure, which was presented in our previous papers.

1 Introduction

Unification in Descriptions Logics is a reasoning for a task of detecting equivalences between the concepts defined in an ontology. It can be also viewed as searching for an answer to a question of conditions under which such equivalence occurs. The conditions are then viewed as a set of new definitions provided for some yet undefined concepts.

Not much is known about the algorithms solving unification in Description Logics as yet. The problem was first proposed and solved for a small logic $\mathcal{FL}_0$ in [4]. It was shown that unification in $\mathcal{FL}_0$ is ExpTime complete. In [1], the authors extended the result for the same logic with regular operations on role strings, $\mathcal{FL}_{reg}$ and in [2] to $\mathcal{FL}_{\bot reg}$, which added inconsistency symbol to the constructors of $\mathcal{FL}_{reg}$. They reported in [2] that if one adds $\bot$ directly to $\mathcal{FL}_0$, their method of solving unification does not apply. The method was to reduce the unification to the problem of solving regular language equations.

Using a different method, we have repeated the result of [4] in [6]. The new method uses a different normal form of the concepts, and focuses on gradual building of the solution by adding small concepts to the substitution for the variables. In [7] the method was further developed and extended for the unification in $\mathcal{FL}_0$ modulo *flat* TBox.

Here we extend the result to the unification in $\mathcal{FL}_\bot$, obtaining the same complexity as for $\mathcal{FL}_0$.

2 Preliminaries

$\mathcal{FL}_\bot$ is a small description logic which extends the description logic $\mathcal{FL}_0$ with a concept expressing inconsistency, $\bot$ (bottom). Hence it is a description logic in which one can express the concepts constructed over a finite set of concept names $\mathbf{N}$ and binary predicate symbols (roles) $\mathbf{R}$, with conjunction, $\sqcap$, top constructor $\top$, value restriction $\forall r.C$, where C is an already formed concept, and $\bot$, bottom. Hence the concepts of $\mathcal{FL}_\bot$ may be seen as generated by the following grammar:

$$C \mapsto A \,|\, \top \,|\, \bot \,|\, C \sqcap C \,|\, \forall r.C$$

where $A \in \mathbf{N}$ and $r \in \mathbf{R}$. The $\mathcal{FL}_\bot$ concepts are interpreted as subsets of a non-empty domain, and role names as binary relations between elements of this domain. The universal value restriction $\forall r.C$ is then interpreted as a subset of elements of the domain such that, they are related by r with only elements in the interpretation of C.

One of the basic questions about the concepts in $\mathcal{FL}_\bot$ is what subsumption relations occur between them. A subsumption holds between C and D, $C \sqsubseteq D$, if an interpretation of C is a subset of an interpretation of D in any domain, in any interpretation of concept names and role names. We define equivalence as two subsumptions $C \equiv D$ iff $C \sqsubseteq D$ and $D \sqsubseteq C$.

Since we have only universal role restrictions of the form $\forall r.C$ in $\mathcal{FL}_\bot$, $\forall r$ may be distributed over conjunction. Hence $\forall r$ behaves as a homomorphism: $\forall r.(C_1 \sqcap C_n) \equiv \forall r.C_1 \sqcap \forall r.C_2$.

A concept of the form $\forall r_1.\forall r_2.\ldots.\forall r_m.A$ (in short $\forall r_1 r_2 \ldots r_m.A$) is called a **particle**. In a particle $\forall r_1 r_2 \ldots r_m.A$ a sequence of roles $r_1 \ldots r_m$ is called a **role string** and it may be empty. A particle: $\forall r_1 r_2 \ldots r_m.A$ where A is a concept name is called an A-particle. A particle: $\forall r_1 r_2 \ldots r_m.\bot$ is called a $\bot$-particle. A particle: $\forall r_1 r_2 \ldots r_m.\top$ is called a $\top$-particle. Each $\top$-particle reduces to $\top$.

Throughout the paper we will use implicitly the following equivalences as left-to-right reducing rules: $C \sqcap \top \equiv C$, $C \sqcap \bot \equiv \bot$, $\forall v.(C_1 \sqcap C_2) \equiv \forall v.C_1 \sqcap \forall v.C_2$, where C, C_1, C_2 are any $\mathcal{FL}_\bot$ concepts. The following basic properties of subsumption in $\mathcal{FL}_\bot$ are also used without proof: $\forall v.\top \equiv \top$ and $\bot \sqsubseteq C, C \sqsubseteq \top, \forall v_1.\bot \sqsubseteq \forall v_2.\bot$ if the role string v_1 is a prefix of v_2.

A concept C is in a **normal form** iff C is a conjunction of particles. Hence we view a concept C as a set of particles and use notation $C = \{P_1, P_2, \ldots, P_n\}$. In this perspective $\top$ is treated as the empty conjunction or the empty set of particles. Dually, $\bot$ may be understood as an infinite set of all particles, but we rather treat it as a **special concept name** interpreted always by an empty subset of an interpretation domain.
It is a common knowledge that the subsumption problem in $\mathcal{FL}_\bot$ is polynomial, e.g. [5, p. 81]. A simple procedure adapted to our normal form can be found in Appendix A.

If $C \sqsubseteq P$ holds, where P is a particle, then $P \in C$ or there is a $\bot$-particle $P' \in C$ such that $P' \sqsubseteq P$. For each subsumption $C \sqsubseteq^? D$ that holds in $\mathcal{FL}_\bot$ and for every particle $P \in D$ we choose $P' \in C$, such that $P' \sqsubseteq P$ and set that P', P are in **solving relation**, $P' \leftarrow P$.[1]

[1] There may be a choice of particles in C, but we can choose arbitrary.

Example 1. Consider a subsumption $\forall sr.\bot \sqcap \forall rrs.A \sqsubseteq \forall srs.A$. Here we can decide that the particle $P = \forall srs.A$ is solved by $P' = \forall sr.\bot$, $\forall sr.\bot \leftarrow \forall srs.A$.

3 Unification in $\mathcal{FL}_\bot$

In order to define unification, we have to assume a set of concept names **Var** (variables), disjoint from **N** (constants) and allow the variables to be substituted by $\mathcal{FL}_\bot$ concepts. Hence now we assume that the $\mathcal{FL}_\bot$ concepts are constructed over the set of constants and variables. If a concept does not contain variables, we call it *ground*.

Unification problem is defined as a set of goal subsumptions between $\mathcal{FL}_\bot$ concepts in normal form: $\Gamma = \{C_1 \sqsubseteq^? D_1, \ldots, C_n \sqsubseteq^? D_n\}$.

Each of the goal subsumptions contains concepts in normal form, and without loss of generality we can assume that each D_i is a particle. We call the particles in Γ, **goal particles**.

We define a **solution** or unifier for the unification problem as an assignment of ground concepts to variables such that the goal subsumptions hold. If γ is such an assignment, by $\gamma[P \to P']$ we mean replacement of the particle P by P' everywhere in the range of γ. If a variable is assigned by a solution an empty conjunction of particles, we understand that it is substituted by the top constructor, $\top$. The variables substituted with $\bot$ will be called $\bot$-variables, and the variables substituted with $\top$, $\top$-variables.[2]

Since in $\gamma(\Gamma)$ all subsumptions hold, we can identify a solving relation ($\leftarrow$) between ground particles in these subsumptions under γ. Since there may be some choice in defining a solving relation for $\gamma(\Gamma)$, from now on we assume that one such choice was made and the relation is defined.

A solving relation between the ground particles may not be visible in the goal subsumptions, because different particles in the substitution for one goal particle may be solved by the substitution of different goal particles, as in the following example.[3]

Example 2. Let the goal subsumption be: $\forall rs.Z \sqcap \forall r.Y \sqsubseteq^? \forall rsr.X$. If the solution is: $\gamma = [Z \mapsto rr.\bot, Y \mapsto \forall srs.\bot, X \mapsto \{\forall r.\bot, \forall s.\bot\}]$. Neither $\gamma(\forall rs.Z)$ nor $\gamma(\forall r.Y)$ is subsumed separately by $\gamma(\forall rsr.X)$.

The idea of a unification procedure which we are going to see, is to reduce a unification problem to a problem that can be solved by a $\mathcal{FL}_0$-unification procedure. The lemmas in the following subsections show the properties of $\mathcal{FL}_\bot$ unifiers that will be used to justify such a reduction.

[2] Notice that these concepts of $\bot$- or $\top$-variables make sense only relative to a solution.
[3] This is the important difference between $\mathcal{EL}$ and $\mathcal{FL}_\bot$.

3.1 Removing redundant particles

In Lemma 1 we observe that one can always remove *redundant* particles from the range of a unifier. This process is similar to the one we used for obtaining minimal $\mathcal{EL}$ unifiers w.r.t. the inverse of subsumption in [3].

Definition 1. *Let Γ be a unification problem, X a goal variable and γ a solution. A particle P in $\gamma(X)$ will be called* redundant *in $\gamma(X)$ if there is no particle P' in $\gamma(\Gamma)$, such that P solves P'.*

Lemma 1. *Let Γ be a unification problem and γ its solution. Let P be a redundant particle in $\gamma(X)$. Then a substitution γ', which is like γ with all $P \in \gamma(X)$ replaced by $\top$ is still a unifier.*

Proof. The claim follows from the properties of the $\mathcal{FL}_\bot$ subsumption. If P is redundant in $\gamma(X)$, there is no particle which is solved by the occurrence of P in $\gamma(X)$. Hence if P is replaced by $\top$, then no subsumption of the form $P \sqsubseteq P'$ which holds before the replacement is broken. Now if $P' \sqsubseteq P$ before the replacement, then it holds after the replacement too, because $P' \sqsubseteq \top$ by the properties of the $\mathcal{FL}_\bot$ subsumption. An example of this process may be found in Appendix B, Example 4. □

3.2 Cycles

We can eliminate redundant $\bot$-particles as well as any other redundant particles in this way. Removing all $\bot$ symbols from the range of γ may not be possible for two reasons.

- A cyclic solving relation between particles (Definition 2)
- A $\bot$-particle that is used in solving a ground goal particle.

In order to formalize the notion of cycle, we give the following definition.

Definition 2. *Let γ be a solution for a unification problem Γ.*

1. *A* path *of $\bot$-particles in $\gamma(\Gamma)$ is a sequence of $\bot$-particles: $\forall v_1.u_1.\bot, \ldots, \forall v_k.u_k.\bot$ such that:*
   (a) Each $\forall u_i.\bot$ is in $\gamma(X_i)$.
   (b) For odd j, $1 \leq j \leq k-1$, $\forall v_j.u_j.\bot \leftarrow \forall v_{j+1}.u_{j+1}.\bot$, $X_{j+1} = X_{j+2}$ and $\forall u_{j+1}.\bot = \forall u_{j+2}.\bot$
   We denote such a path by: $\forall v_1.u_1.\bot \stackrel{+}{\leftarrow} \forall v_k.u_k.\bot$.
2. *Two paths P_1, P_2* intersect *with each other if there is a $\bot$-particle $\forall u.\bot \in \gamma(X)$ such that $\forall v.u.\bot \in \forall v.\gamma(X)$ is in P_1, and $\forall v'.u.\bot \in \forall v'.\gamma(X)$ is in P_2.*
3. *A cycle in $\gamma(\Gamma)$ is a set of paths in $\gamma(\Gamma)$ such that each path intersects with at least one of the other paths in the set and there is at least one path in the set $\forall v_i.u_i.\bot, \ldots, \forall v_{i+l}.u_{i+l}.\bot$ such that $X_i = X_{i+l}$ and $\forall u_i.\bot = \forall u_{i+l}.\bot$.*

For a simple example illustrating this definition look at Example 5.

Even if a unification problem has no goal $\bot$-particles, it may not have a solution without $\bot$. This is illustrated by the following example.

Example 3. $\Gamma = \{X \sqsubseteq^? A,\ X \sqsubseteq^? \forall r.Y,\ Y \sqsubseteq^? \forall r.X\}$

It is easy to see that X cannot be solved by $\top$. A solution for X may contain A or be $\bot$. The two next subsumptions create a cycle, which can be solved either by $\top$ or $\bot$-particles. Hence X has to be $\bot$ and then Y has to be $\bot$ or contain $\forall r.\bot$.

3.3 Reducing height of $\bot$-particles

We cannot remove all $\bot$-particles from the solution of a unification problem, but we can reduce their height.

Lemma 2. *Let Γ be a unification problem, γ is a solution. Then there is a unifier γ', that is the same as γ, except that for each $\bot$-particle $\forall v.u.\bot$ in $\gamma'(\forall v.X)$, there is a $\bot$-variable $\gamma'(Y) = \bot$ such that $\gamma'(\forall v'.Y) \overset{+}{\leftarrow} \forall v.u.\bot$.*

Proof. Let the $\bot$-particles in $\gamma(\Gamma)$ be:
$\forall v_1 u_1.\bot, \forall v_2 u_2.\bot, \ldots, \forall v_n u_n.\bot$. These $\bot$-particles are in γ substitution for the goal particles: $\forall v_1.X_1, \forall v_2.X_2, \ldots, \forall v_n.X_n$.

In order to minimize the role strings of these particles we cannot change v_i's, because these role strings come from Γ. We construct γ' by modifying suffixes u_i of the role strings in the $\bot$-particles in the range of γ.

Step 1. Remove u from every $\bot$-particle $\forall u.\bot$ in $\gamma(X)$ for every goal variable X that contains a $\bot$-particle. Now all variables with $\bot$-particles become $\bot$-variables if the reduction to $\bot$ is applied. In fact, we postpone performing such reduction till the end, in order to keep all solving relations from γ defined (even if they do not hold after the replacement).

Notice that for every particle P such that $\gamma(X) \sqsubseteq P$, the subsumption is preserved in γ', $\gamma'(X) = \bot \sqsubseteq P$, but the solving relation between $\bot$-particles may be broken in many places. For example take the subsumption: $\forall v_i.\gamma(X) \sqsubseteq \gamma(Y)$ where $\gamma(Y) = \{\forall v_j.\bot\}$, now after substituting X and Y with $\bot$, $\forall v_i.\bot \not\sqsubseteq \bot$.

All solving subsumptions of the form $\gamma'(\forall v_i.X_i) \sqsubseteq \gamma'(\forall v_j.X_j)$, where v_i is a prefix of v_j hold in γ'. The only solving relations that are broken are those where v_j is a proper prefix of v_i. One of these possibilities must occur, since γ is a unifier.[4]

Step 2. In this step we repair the solving relation of γ in all places that it is broken in γ'. We keep the following invariants:

1. If $\forall v_i.u_i'.\bot$ is in $\gamma'(\forall v_i.X_i)$, then $v_i u_i'$ is a prefix (not necessarily proper) of $v_i u_i$, where $\forall v_i.u_i.\bot \in \gamma(\forall v_i.X_i)$.
   Obviously, this is true before Step 2. Keeping this invariant ensures that the process of repairing the solving relation will terminate, because at worst we will recover the substitution γ in this way, and then all solving relations hold.
2. The second invariant is the part of the claim in the lemma: for each $\bot$-particle $\forall v.u.\bot$ in $\gamma'(\forall v.X)$, there is a $\bot$-variable $\gamma'(Y) = \bot$ such that $\gamma'(\forall v'.Y) \overset{+}{\leftarrow} \forall v.u.\bot$. Y is then called an *anchor* variable.

[4] Either v_i is a prefix (not necessarily proper) of v_j or v_j is a proper prefix of v_i.

The following process is to be performed exhaustively.
Assume:

- $\forall v_i u_i.\bot \in \gamma(X_i)$, $\forall v_j.u_j.\bot \in \gamma(X_j)$,
- there is the solving relation in γ: $\forall v_i u_i.\bot \leftarrow \forall v_j u_j.\bot$
- and in γ' the corresponding solving relation does not hold:
  $\forall v_i.u'_i.\bot \not\sqsubseteq \forall v_j.u'_j.\bot$,
  where $\forall v_i.u'_i.\bot \in \gamma'(\forall v_i.X_i)$ and $\forall v_j.u'_j.\bot \in \gamma'(\forall v_j.X_j)$.

There are two cases depending on $u'_i = \epsilon$ or $u'_i \neq \epsilon$

(i) Notice that if $u'_i = \epsilon$, X_i is an anchor variable, the assumptions and invariants tell us that $v_j u'_j$ is a proper prefix of v_i. $v_i = v_j u'_j r_i$. In this case, we replace $\forall v_j.u'_j.\bot$ with $\forall v_j.u'_j r_i.\bot \in \gamma'(\forall v_j.X_j)$. Notice that the solving relation is thus recovered and X_i is still an anchor variable (invariant 2). Notice also that since γ is a unifier, the invariant 1 is also satisfied. (The prefix for the $\bot$-particle is *enforced* by the goal subsumption and the $\bot$ in $\gamma'(X_i)$.)

(ii) If X_i is not an anchor variable in γ' (due to the corrections already done on the $\bot$-particles in $\gamma'(X_i)$), then by the invariant 2 there is an anchor variable Y, $\gamma'(Y) = \bot$, such that $\gamma'(\forall v.Y) \overset{+}{\leftarrow} \forall v_i.u'_i.\bot \in \gamma'(X_i)$.
We do similar correction on the $\bot$-particle $\forall v_j.u'_j.\bot$ in $\gamma'(\forall v_j.u'_j.X_j)$ as in the previous case. Since $\forall v_i.u'_i.\bot \not\sqsubseteq \forall v_j.u'_j.\bot$, $v_j u'_j$ has to be a proper prefix of $v_i u'_i$. Hence $v_i u'_i = v_j u'_j r_i$. We replace $\forall v_j.u'_j.\bot$ in $\gamma'(\forall v_j.X_j)$ by $\forall v_j.u'_j r_i.\bot$.
Notice that here too the invariant 1 is satisfied, since the modification preformed is the minimal requirement in order for the solving relation to hold. Hence the string roles have to agree with the string roles in γ.

After preforming all the transformations, we reduce the concepts in the substitution for the variables and remove the redundant particles. □

A simple example of this construction is in Appendix B, Example 6.

From Lemma 2, we know that if there is a cycle in a solution for a unification problem, there are anchor variables in the unification problem. We can guess them to be $\bot$-variables and solve the subsumptions in which they occur. The solved subsumptions are removed from the unsolved part of the problem, hence the cycles are broken. What is left, are at most some non-cyclic paths, with which we will deal with in the next subsection.

3.4 Reducing $\mathcal{FL}_\bot$ solution to one in $\mathcal{FL}_0$

Now we formulate a kind of reduction lemma, which relates unification in $\mathcal{FL}_\bot$ to that in $\mathcal{FL}_0$. It is similar to Lemma 9 in [2], where unification in $\mathcal{FL}_{\bot reg}$ was related to the unification in $\mathcal{FL}_{reg}$. Our lemma is not so general as theirs and the approach and proof are different.

Lemma 3. *Let Γ be an $\mathcal{FL}_\bot$ unification problem such that $\bot$ does not appear as a symbol in Γ. If γ is a $\mathcal{FL}_\bot$ solution of Γ and there are no cycles among $\bot$-particles in $\gamma(\Gamma)$, then there is also an $\mathcal{FL}_0$ solution for Γ.*

Proof. Let γ be an $\mathcal{FL}_\bot$ solution of Γ. By Lemma 1 we can assume that γ does not have any redundant particles. Since we assume that there are no cycles between $\bot$-particles, we know that $\bot$-particles may only be on paths starting with $\bot$-particles solving some ground A-particles from Γ.

The proof shows how we can eliminate the $\bot$ symbols from the solution γ.

First we construct a substitution γ_B from γ, such that γ_B is γ with $\bot$ replaced by a new constant B. Hence $\gamma_B = \gamma[\bot \to B]$.

Obviously, γ_B is an $\mathcal{FL}_0$ substitution, but it is not in general a unifier of Γ. We show how to extend γ_B so that it unifies Γ.

Let P be a particle in the range of γ and P^B be the particle P after the replacement in γ_B.

Hence: $P^B = (\forall v.A)^B = \begin{cases} \forall v.A & \text{if } A \text{ is not } \bot \\ \forall v.B & \text{if } A \text{ is } \bot \end{cases}$

Each particle P that occurs in the range of γ is now changed to P^B in the range of γ_B. Notice that in the definition above, v may be empty.

For any goal subsumption $C \sqsubseteq^? D$: if $\gamma(C) \sqsubseteq \gamma(D)$, but $\gamma_B(C) \not\sqsubseteq \gamma_B(D)$, there is a particle P such that $\gamma_B(C) \not\sqsubseteq P^B$, $P^B \in \gamma_B(D)$.

In this case there is a $\bot$-particle $\forall v.\bot \in \gamma(C)$, such that $\forall v.\bot \sqsubseteq P^B$, but now $\forall v.\bot$ is replaced by $\forall v.B$ in $\gamma_B(C)$.

$\forall v.\bot \in \gamma(\forall v'.X)$, where $\forall v'.X \in C$. This is so, because $\bot$ cannot occur as such in C by assumption. v' may be empty, but it must be a prefix of v, $v = v'v''$. P^B has to have the form $\forall v v_i.A$, where v_i may be empty and A is either B or any other constant.

In order to regain a unifier, we extend $\gamma_B(X)$ as follows:
$\gamma_B(X) \leftarrow \gamma_B(X) \cup \{\forall v'' v_i.A\}$.

After this extension $\gamma_B(\forall v'.X)$ has two particles (among other possible particles): $\{\forall v'v''.B, \forall v'v''v_i.A\} = \{\forall v.B, \forall v v_i.A\}$ which replaced $\forall v.B$.[5] Hence after the extension $\gamma_B(C) \sqsubseteq P^B$. In such way we repair each dis-subsumption among the goal subsumptions and the extended γ_B becomes a unifier.

Obviously, this extension of γ_B that corrects dis-subsumptions locally, has to terminate in polynomial time, because since there is no cycle in $\gamma(\Gamma)$, extending γ_B, we will finally get to the $\bot$-particles in the range of γ that are not solved by any other particle (dead-ends for solving relation). If such $\bot$-particle is changed to B-particle in the range of γ_B it will not trigger any more extensions and the additional particles added to the range because of such a B-particle will not need to be solved by any other particle.

The only situation that a new particle might be augmented ad infinitum is when the $\bot$-particles are in a cyclic solving relation. Example 7 illustrates the

[5] If we were to replace B back with $\bot$, the second particle would be reduced immediately in the presence of $\forall v.\bot$.

symbols used in the above proof. Example 8 illustrates how the construction breaks when there is a cycle in $\gamma(\Gamma)$.

□

4 Unification procedure

The unification procedure is as follows:

1. We run $\perp$-elimination (flattening) on Γ. This is explained in the next subsection (subsection 4.1). This step can fail or result with a partial solution $\gamma_\perp$ that maps some variables to $\perp$ and a small unification problem Γ'.
   If the flattening step does not fail, then Γ is $\mathcal{FL}_\perp$-unifiable iff Γ' is.[6]
2. In subsection 4.2 we consider the unification problem Γ'. Γ' does not contain $\perp$ symbol, but it may contain $\perp$-variables. We can see that Γ' is of such form that cycles are impossible in any solution of Γ'. Hence by Lemma 3, in order to decide if there is an $\mathcal{FL}_\perp$ unifier of Γ', it is enough to decide the existence of an $\mathcal{FL}_0$-unifier. Hence we run an $\mathcal{FL}_0$-unification procedure, which enforces the so called decreasing rule for decomposition variables. We have to ensure that the $\mathcal{FL}_0$-unifier will not increase the number of variables substituted with $\perp$. For this task we may use the algorithm described in [7] with a small modification.
   This procedure fails or terminates with success in at most exponential time in the size of Γ'.

4.1 Elimination of $\perp$ (flattening)

In this section we allow $\perp$ symbol in a unification problem.

The most important part of the unification procedure is a flattening step, where we eliminate bottom from the goal subsumptions and flatten them.[7] Elimination of $\perp$ requires guessing which variables are $\perp$-variables. This information is kept as a partial solution. Such guessing causes some goal subsumptions to be removed as solved. We can also **fail**, if a subsumption is unsolvable due to such guess. First we set the following notation.

If P is a particle, $P \neq \perp$ and r a role name ($r \in \mathbf{R}$), we define P^{-r} in the following way:

$$P^{-r} = \begin{cases} P^r & \text{if } P \text{ is a variable and then } P^r \text{ is a decomposition variable} \\ P' & \text{, if } P = \forall r.P' \\ \top & \text{, if } P \text{ is a constant or } P = \forall s.P'', \text{ where } s \neq r \end{cases}$$

P^r is the so called a **decomposition variable**. The meaning of this variable is expressed in the following property: for any solution γ and any ground particle Q, $\forall r.Q \in \gamma(P)$ if and only if $Q \in \gamma(P^r)$.

For each role name r and a variable P, there can be only one decomposition variable denoted by P^r. It will be constrained by the so called increasing

[6] With an additional requirement of preserving the so called decreasing rule.
[7] The procedure presented here is very similar to the flattening in [7].

goal subsumption $P \sqsubseteq^? \forall r.P^r$ which corresponds to the "if" part of the above property, and by the decreasing rule, which corresponds to the "only if" part.

Property 1 (Decreasing rule). If $\forall r.C \in \gamma(P)$, then $C \in \gamma(P^r)$, where P is a variable and C is a particle.

For the decreasing rule, we cannot have a suitable subsumption which could be added to a unification problem, but our algorithm for solving $\mathcal{FL}_\bot$ unification should secure that this implication is true for any variable P and concept C.

A goal subsumption $C \sqsubseteq^? D$, where D is a particle, is called *non-flat* if $D = \forall r.D'$ or there is a particle of the form $\forall r.C' \in C$.

If s is a non-flat goal subsumption, $s = C_1 \sqcap \cdots \sqcap C_n \sqsubseteq^? D$, where $C_1, \ldots, C_n, D$ are particles not equal to $\bot$, we define $s^{-r} = C_1^{-r} \sqcap \cdots \sqcap C_n^{-r} \sqsubseteq^? D^{-r}$.

If P is a particle, $P \neq \bot$ and A is a constant, then we define P^A in the following way:

$$P^A = \begin{cases} P & \text{if } P \text{ is a constant } A \text{ or a variable} \\ \top & \text{in all other cases} \end{cases}$$

If s is a goal subsumption, $s = C_1 \sqcap \cdots \sqcap C_n \sqsubseteq^? D$, where $C_1, \ldots, C_n, D$ are particles not equal to $\bot$, we define $s^A = C_1^A \sqcap \cdots \sqcap C_n^A \sqsubseteq^? D^A$.[8]

Implicit rule At each step of the following procedure, we implicitly apply the following rule that removes trivially solved equations or fails:

- if there is a goal subsumption $C \sqsubseteq^? P$ such that $\bot \in C$ or $P \in C$ or $P \equiv \top$, then remove this subsumption from the current unification problem.
- if there is a goal subsumption $C \sqsubseteq^? P$ such that $\forall v.\top \in C$, then delete the particle from C,
- **fail** at once if $C \sqsubseteq^? \bot$ is in the goal and $\bot \notin C$.

Step 1. In the first step we guess which variables in the goal contain bottom and we replace them with $\bot$. We keep the partial solution for the eliminated variables as a set of assignments ($[X \mapsto \bot]$).

Step 2. We look at the non-flat subsumptions.

For a non-flat goal subsumption: $s = C_1 \sqcap \cdots \sqcap C_n \sqsubseteq^? D$ we do the following.

1. If $D = \forall r.D'$, then replace s with s^{-r}.
   If $D' = \bot$, s^{-r} has $\bot$ on its right hand side, hence the implicit rule applies.
   If a new decomposition variable X^r is created in the process of constructing s^{-r}, we add the increasing subsumption of the form $X \sqsubseteq^? \forall r.X^r$. If the decomposition variable X^r is already created, we use it in s^{-r} as needed.
   The increasing subsumption is part of the unification problem, but no increasing subsumption is subject to the flattening procedure.
   We guess if X^r is $\bot$ or not. If it is $\bot$ then X^r is replaced by $\bot$ in all subsumptions except the increasing one. (The implicit rule applies.)
2. If D is a constant, then we replace s with s^D.

[8] The particles that are not constant A or variables are deleted from s.

3. If D is a variable, s contains a particle of the form $\forall r.C' \in C$.
   Since D is guessed not to contain $\bot$, we *split* D, i.e. we delete s and add the following goal subsumptions:
   (a) for each $r \in \mathbf{R}$, we add s^{-r}.
   (b) we guess for each constant A, if it should be in the substitution for D and if this is the case, we add the following goal subsumptions: $D \sqsubseteq^? A$ and $C_1^A \sqcap \cdots \sqcap C_n^A \sqsubseteq^? A$
       If any decomposition variable is created in the splitting, it is dealt with as in the case 1.

Lemma 4. *The process of $\bot$-elimination terminates in nondeterministic polynomial time.*

Proof. Step 1 terminates because there are only finitely many variables in a given unification problem. Step 2 terminates, since there are only polynomially many occurrences of particles of the form $\forall r.C$ in a given unification problem and each transformation in the step removes at least one of them. □

The following theorem states completeness and soundness of $\bot$-elimination w.r.t. unification in $\mathcal{FL}_\bot$.

Theorem 1. *Let Γ be an $\mathcal{FL}_\bot$ unification problem and Γ' the problem transformed by the $\bot$-elimination steps. A substitution γ is a solution of Γ iff there is a substitution γ' that solves Γ' and obeys the decreasing rule.*

Proof. The "only if" direction (completeness): if γ is a unifier of Γ, then γ' is a unifier of Γ', where γ' is the extension of γ:

$\gamma'(X^r) = \{P \mid \forall r.P \in \gamma'(X)\}$.

Notice that γ' obeys the decreasing rule.

It is enough to show the implication for one step of the procedure of $\bot$-elimination.

1. For Step 1 and the implicit rule, we just state that the removed subsumptions are trivially satisfied by γ as well as by γ' augmented with saved partial solution.
2. For Step 2, we assume that a non-flat goal subsumption s was selected for the transformation. γ' solves s.
   We have several cases to consider.
   (a) $D = \forall r.D'$. Since γ' solves s, $\gamma'(s)$ has to have either $\bot$ on its right hand side or a particle $\forall r.C'$ ($C' \sqsubseteq D'$). The first possibility is excluded, since then s would be removed by the implicit rule. Hence there must be $\forall r.C'$ in the particles on its left hand side of $\gamma'(s)$. This particle is either a part of s, or there is $\forall r.X$ in s and $\forall r.C' \in \gamma'(\forall r.X)$ or there is a variable Y in s and $\forall r.C' \in \gamma'(Y)$. In the first case, C' is on the left hand side of s^{-r}, in the second case, $C' \in \gamma'(X)$, in the third case, (by decreasing rule) $C' \in \gamma'(X^r)$. In all these cases γ' solves s^{-r}.

(b) D is a constant. Since s is solved by γ' and there is no $\bot$ at the top level on the left hand side of $\gamma'(s)$, D must be on the left hand side of $\gamma'(s)$. Hence γ' solves s^D as required.

(c) D is a variable.
We have two cases here.
   – For a given role name r, there is are particles $\forall r.P \in \gamma'(D)$. Then since s is solved by γ' and $\bot$ is not at the top level of s, and because of the definition of γ' for the decomposition variables obeying the decreasing rule, γ' solves also s^{-r}. If there is no particle of the form $\forall r.P$ in $\gamma'(D)$, then $\gamma'(X^r) = \top$, and s^{-r} is also solved by γ'.
   – If there is a constant A in $\gamma'(D)$, then γ' solves $D \sqsubseteq^? A$ and $C_1^A \sqcap \cdots \sqcap C_n^A \sqsubseteq^? A$.

For the "if" direction (soundness) we have to show that, if γ' solves Γ' and obeys the decreasing rule, then γ solves Γ. Notice that γ is the restriction of γ' and the saved partial solution assignments, to the variables in Γ. We have to follow the steps of flattening in the opposite direction. Assume that $s \in \Gamma$ and $s = C_1 \sqcap \cdots \sqcap C_n \sqsubseteq^? D$.

1. If s was removed in Step 1 or by the implicit rule, it is solved by any substitution extended with the saved partial solution, hence it is solved by γ' plus saved partial solution assignments. The cases below assume that s was not removed by the implicit rule.
2. If $D = \forall r.D'$, we know that s^{-r} replaced s and γ' solves s^{-r}. Since γ' satisfies the increasing subsumptions, then γ' solves s too. $D = \forall r.\bot$ is a special case, when s^{-r} had to be removed by the implicit rule.
3. If D is a constant, s was replaced by s^D and γ' solves s^D. Since s differs from s^D by having more particles on the left hand side, γ' solves s.
4. If D is a variable, we consider all particles P in $\gamma'(D)$.
   – If $P = \forall r.P'$, there is a subsumption s^{-r} in the goal, γ' solves s^{-r}. Thus since γ' solves the increasing subsumptions and obeys the decreasing rule, it solves also subsumption $C_1 \sqcap \cdots \sqcap C_n \sqsubseteq^? P$.
   – If P is a constant, then there is a subsumption s^A in the goal and γ' solves this subsumption. Hence γ' solves also subsumption $C_1 \sqcap \cdots \sqcap C_n \sqsubseteq^? P$.

   Hence since for each particle in $\gamma'(D)$, the corresponding subsumptions are solved by γ', then γ' solves s too. In the case there are no particles in $\gamma'(D)$, $\gamma'(D) = \top$ and γ' solves s.

□

4.2 Applying $\mathcal{FL}_0$-unification

In the next theorem we justify the possbility of using $\mathcal{FL}_0$-unification on Γ' obtained from $\bot$-elimination.

Theorem 2. *Let γ be a $\mathcal{FL}_\bot$-unifier of Γ, then there is Γ' a unification problem obtained from Γ by $\bot$-elimination, such that γ', an extension of γ, is a unifier of Γ' and $\gamma'(\Gamma')$ has no cycles.*

Proof. Let Γ' be obtained from Γ by $\bot$-elimination, γ' is a solution for Γ', where γ' is an extension of γ to new variables. By Lemma 1 and Lemma 2, we assume that γ' has no redundant particles, and every $\bot$-particle in the cycle is connected with a path to an anchor $\bot$-variable. Since Γ' was obtained by $\bot$-elimination, $\gamma'(\Gamma)$ has no $\bot$-variables except some decomposition variables occurring in the increasing subsumptions. Hence if there is a cycle in $\gamma'(\Gamma')$, these are the anchor variables for the cycles.

For a $\bot$-variable X^r to be an anchor for a cycle, Γ' must contain a subsumption $C \sqcap \forall u_1.X^r \sqsubseteq^? \forall u_2.Y$, where u_1, u_2 may be empty. It is impossible that u_1 is not empty, because Γ' contains only flat subsumptions.

This cannot be an increasing subsumption, because X^r is $\bot$-variable and thus the subsumption is impossible in Γ', because X^r is $\bot$. Hence there cannot be any cycle in $\gamma'(\Gamma')$. Example 9 shows how a cycle is solved by $\bot$-elimination. □

Theorem 3. *(main result) For an $\mathcal{FL}_\bot$ unification problem Γ is decidable in at most ExpTime.*

Proof. By Lemma 4, $\bot$-elimination is a procedure non-deterministic polynomial in the size of Γ. If successful, it returns $\gamma_\bot$ a partial solution for the variables eliminated from the goal. The exponential time needed for the $\mathcal{FL}_0$-unification dominates the non-deterministic polynomial time of the first step.

Soundness: if both stages of the procedure terminate successfully, then the partial solution $\gamma_\bot$ combined with $\gamma_{\mathcal{FL}_0}[B \mapsto \bot]$ is a solution for Γ. This is because of soundness of $\bot$-elimination, Theorem 2 and soundness of $\mathcal{FL}_0$-unification procedure. The $\mathcal{FL}_0$-unifier should not increase the number of $\bot$-variables (variables with B in the substitution).[9]

Completeness: if there is a solution γ of Γ, by completeness of $\bot$-elimination, Theorem 1, there is a way to perform $\bot$-elimination in such a way that it will not fail, and we will get a unification problem Γ'. Γ' has a unifier with no cycles and obeys the decreasing rule, hence by Lemma 3, there is a $\mathcal{FL}_0$-unifier of Γ' that obeys the decreasing rule. Hence by completeness of $\mathcal{FL}_0$-unification procedure we will get the positive answer. □

5 Conclusions

The constructions presented in this paper show that the unification in $\mathcal{FL}_\bot$ can be solved in ExpTime. The question remains if the problem is also ExpTime hard. In the solving procedure we are passing the flattened problem to a unification procedure for $\mathcal{FL}_0$ with a flat TBox. This is possible, because we can see our flattened problem as a $\mathcal{FL}_0$ unification problem with the empty TBox. The next question arises, if it is possible to extend the algorithm to the case of $\mathcal{FL}_\bot$ with a flat TBox. One can try to extend the method of solving unification presented in this paper to other extensions of $\mathcal{FL}_0$ and other description logics.

[9] In the $\mathcal{FL}_0$-unification procedure in [7], instead of computing a shortcut for B, we just check if $\{X \mid X \sqsubseteq^? B \in \Gamma'\}$ is a valid shortcut.

References

1. Baader, F., Küsters, R.: Unification in a description logic with transitive closure of roles. In: Goble, C.A., McGuinness, D.L., Möller, R., Patel-Schneider, P.F. (eds.) Working Notes of the 2001 International Description Logics Workshop (DL-2001), Stanford, CA, USA, August 1-3, 2001. CEUR Workshop Proceedings, vol. 49. CEUR-WS.org (2001), http://ceur-ws.org/Vol-49/BaaderKuesters-36start.ps
2. Baader, F., Küsters, R.: Unification in a description logic with inconsistency and transitive closure of roles. In: Horrocks, I., Tessaris, S. (eds.) Proceedings of the 2002 International Workshop on Description Logics (DL2002), Toulouse, France, April 19-21, 2002. CEUR Workshop Proceedings, vol. 53. CEUR-WS.org (2002), http://ceur-ws.org/Vol-53/BaaderKuesters-unification.ps
3. Baader, F., Morawska, B.: Unification in the description logic $\mathcal{EL}$. Logical Methods in Computer Science **6**(3) (2010), special Issue of the 20th International Conference on Rewriting Techniques and Applications; also available at http://arxiv.org/abs/1006.2289
4. Baader, F., Narendran, P.: Unification of concept terms in description logics. Journal of Symbolic Computation **31**(3), 277–305 (2001). https://doi.org/10.1006/jsco.2000.0426
5. Baader, F., Nutt, W.: Basic description logics. In: Baader, F., Calvanese, D., McGuinness, D.L., Nardi, D., Patel-Schneider, P.F. (eds.) The Description Logic Handbook: Theory, Implementation, and Applications, pp. 43–95. Cambridge University Press (2003)
6. Borgwardt, S., Morawska, B.: Finding finite Herbrand models. In: Bjørner, N., Voronkov, A. (eds.) Proc. of the 18th Int. Conf. on Logic for Programming, Artificial Intelligence, and Reasoning (LPAR-18). Lecture Notes in Computer Science, vol. 7180, pp. 138–152. Springer (2012), https://doi.org/10.1007/978-3-642-28717-6_13
7. Morawska, B.: Unification in $\mathcal{FL}_0$ modulo a Flat TBox. In: Borgwardt, S., Meyer, T. (eds.) Proceedings of the 33rd International Workshop on Description Logics (DL 2020) co-located with the 17th International Conference on Principles of Knowledge Representation and Reasoning (KR 2020), Online Event [Rhodes, Greece], September 12th to 14th, 2020. CEUR Workshop Proceedings, vol. 2663. CEUR-WS.org (2020), http://ceur-ws.org/Vol-2663/paper-15.pdf

A Subsumption in $\mathcal{FL}_\bot$

We will be using the following notation. If P is a particle, $P \neq \bot$ and r a role name ($r \in \mathbf{R}$), we define P^{-r} in the following way:

$$P^{-r} = \begin{cases} P', & \text{if } P = \forall r.P' \\ \top, & \text{if } P \text{ is a constant or } P = \forall s.P'', \text{ where } s \neq r \end{cases}$$

Now if C is a set of the particles $\{P_1, \dots, P_n\}$ such that $\bot \notin C$, then $C^{-r} = \{P_1^{-r}, \dots, P_n^{-r}\}$. For the $\mathcal{FL}_\bot$ concepts C, D in normal form, $C \sqsubseteq D$ holds if for every particle P in D, $C \sqsubseteq P$. Hence in the following steps we decide only $C \sqsubseteq P$ for $P \in D$.

Step 1. If $\bot \in C$ then return `true`.

Step 2. If P is $\bot$, then return `false`.

Step 3. If P is a constant, then return $P \in C$.

Step 4. If $P = \forall r.P'$, then return $C^{-r} \sqsubseteq P'$

Obviously the procedure either terminates at once (Step 1 or Step 2), or terminates after a polynomial inclusion test (Step 3), or calls itself on a strictly smaller problem (Step 4). Hence it has to terminate in the polynomial time.

B Additional examples

The following example illustrates the construction from the proof of Lemma 1.

Example 4. Let $\Gamma = \{\forall rr.\bot \sqcap X \sqsubseteq^? \forall r.X\}$ and $\gamma = [X \mapsto \{A, \forall r.A\}]$. γ is a unifier. The solving relation is as indicated by the arrows over the subsumption.

$$\forall rr.\bot \sqcap A \sqcap \forall r.A \sqsubseteq \forall r.A \sqcap \forall rr.A$$

Now, there is no particle P, which is solved by A. Hence we can remove it from $\gamma(X)$. We obtain another unifier, $\gamma' = [X \mapsto \{\forall r.A\}]$.

$$\forall rr.\bot \sqcap \forall r.A \sqsubseteq \forall rr.A$$

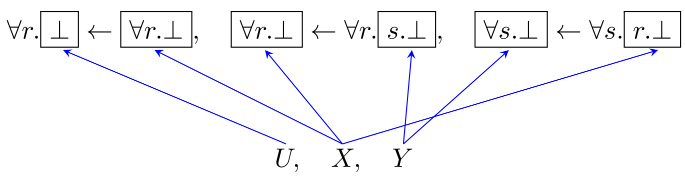
Now, we discover that no particle is solved by $\forall r.A$ in $\gamma'(X)$. By removing it, we obtain another unifier, $\gamma'' = [X \mapsto \top]$

$$\forall rr.\bot \sqcap \top \equiv \forall rr.\bot \sqsubseteq \top \tag{1}$$

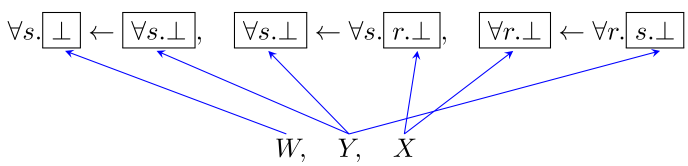
The following simple example illustrates the definition of a cycle (Definition 2).

Example 5. Let the unification problem be:
$\Gamma = \{\forall r.U \sqsubseteq^? X,\ X \sqsubseteq^? \forall r.Y,\ Y \sqsubseteq^? \forall s.X,\ \forall s.W \sqsubseteq^? Y\}$
And let the solution γ be:
$\gamma = [U \mapsto \bot,\ W \mapsto \bot,\ X \mapsto \forall r.\bot,\ Y \mapsto \forall s.\bot]$.

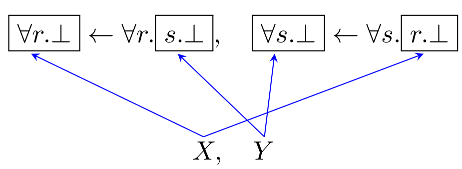
1. An example of a path is:



   The arrows indicate which $\bot$-particles are assigned to which variables.
2. The following path obviously intersects with the one above.



3. A cycle of $\bot$-particles includes all particles in $\gamma(\Gamma)$, because all of them are connected by the solving relation. The actual cyclic relation is between the particles involving $\gamma(X)$ and $\gamma(Y)$:



The variables U and W are the so called *anchor* variables.

The following example illustrates the construction in the proof of Lemma 2.

Example 6. Let $\Gamma = \{s_1 = \forall r.U \sqsubseteq^? Z,\ s_2 = \forall rs.Z \sqcap \forall r.Y \sqsubseteq^? \forall rsr.X,\ s_3 = X \sqsubseteq^? \forall r.A,\ s_4 = X \sqsubseteq^? \forall s.B\}$

Let $\gamma = [U \mapsto \bot, X \mapsto \{\forall r.\bot, \forall s.\bot\}, Y \mapsto \forall srs.\bot, Z \mapsto \forall rr.\bot]$

Step 1. $\gamma' = [U \mapsto \bot, X \mapsto \{\bot, \bot\}, Y \mapsto \bot, Z \mapsto \bot]$

Step 2. $s_1 = \forall r.\bot \not\sqsubseteq \bot$. Hence the particle $\bot$ in $\gamma'(Z)$ is changed to $\forall r.\bot$.

In s_2 solving relation is satisfied: $\forall rs.r.\bot \sqsubseteq \forall rsr.\bot$.

Another solving relation in s_2 is also satisfied: $\forall r.\bot \sqsubseteq \forall rsr.\bot$.

s_3 and s_4 are also satisfied. In fact their role is to make all the $\bot$ particles non-redundant.

Finally we get $\gamma' = [U \mapsto \bot, X \mapsto \bot, Y \mapsto \bot, Z \mapsto \forall r.\bot]$.

Notice that the particles in the goal subsumptions should be non-redundant, if these modifications are to be executed. If the particles are redundant, they can be replaced by $\top$.

The following example illustrates concepts in the proof of Lemma 3.

Example 7. Let $\gamma(X) = \{\forall rr.\bot\}$ and let $\forall s.X$ be a goal particle, such that $\forall srr.\bot \in \gamma(\forall s.X)$ solves $\forall srrs.A$. Hence $\forall srr.\bot \sqsubseteq \forall srrs.A$, but $\forall srr.B \not\sqsubseteq \forall srrs.A$.

$\forall v.C$ in the proof is $\forall s.rr.\bot$ in this example, hence $v = srr$.
v' in this example is the role string of the goal particle $\forall s.X$, hence $v' = s$ and $v'' = rr$, $v = v'v''$.

P^B mentioned in the proof is $\forall s.rrs.A$. Notice that $srrs$ is longer than srr, hence the particle was solved because of the $\bot$-particle, $\forall s.rr.\bot$.

In the proof we have that $P^B = \forall v'.v''v_i.A$. $v_i = s$ in this example. The particle $\forall v''v_i.A$ is being added to $\gamma_B(X)$.

$\gamma_B(X) = \{\forall rr.B, \forall rrs.A\}$

Another example illustrates how the above construction is not possible when the $\bot$-particles form a cycle.

Example 8. Let Γ contain the following goal subsumptions: $X \sqsubseteq^? \forall r.Y, \quad Y \sqsubseteq^? \forall s.X$ Let the unifier γ be $[X \mapsto \bot, Y \mapsto \bot]$.

Hence the following particles form a cycle: $\bot \leftarrow \forall r.\bot, \bot \leftarrow \forall s.\bot$.

If we replace $\bot$ with B in the range of γ, we obtain $\gamma_B = [X \mapsto B, Y \mapsto B]$. Then the solving relation breaks.

Trying to correct the first subsumption, we have to add $\forall r.B$ to $\gamma_B(X)$, and then $\forall s.B$ to $\gamma_B(Y)$. $\gamma_B = [X \mapsto \{B, \forall r.B\}, Y \mapsto \{B, \forall s.B\}]$.

Now we have to add $\forall rs.B$ to $\gamma_B(X)$ and $\forall sr.B$ to $\gamma_B(Y)$, and so on *ad infinitum.*

The next example illustrates how a cycle is solved by $\bot$-elimination process. (Theorem 1)

Example 9. Let $X \sqsubseteq^? \forall r.Y$, $Y \sqsubseteq^? \forall s.X$ belong to a unification problem Γ. If γ is the solution, the subsumptions must be solved by some cyclic particles in $\gamma(\Gamma)$.

If there is a cycle in $\gamma(\Gamma)$, we have to guess at least one variable to be $\bot$. Let $[X \mapsto \bot]$ and Y is not a $\bot$-variable. Then the first subsumption is solved and removed. The second subsumption has the form: $Y \sqsubseteq^? \forall s.\bot$. It will be flattened to: $Y^s \sqsubseteq^? \bot$ and $Y \sqsubseteq^? \forall s.Y^s$ (the increasing subsumption). We are forced to guess $[Y^s \mapsto \bot]$ (or fail), which yields the solution $[X \mapsto \bot, Y \mapsto \forall s.\bot]$.