Complexity of Subsumption in Extensions of $\mathcal{EL}$

Diplomarbeit
zur Erlangung des akademischen Grades
Diplom-Informatiker

vorgelegt von
Christoph Haase
geboren am 7. August 1981 in Dresden

der Fakultät Informatik
der Technischen Universität Dresden

eingereicht am 15. August 2007

Betreuer: PD Dr. rer. nat. habil. Carsten Lutz
Verantwortlicher Hochschullehrer: Prof. Dr.-Ing. Franz Baader

Abstract

During the last years, it has been shown that the description logic $\mathcal{EL}$ is well-suited for tractable reasoning. In particular, reasoning is even tractable w.r.t. general concept inclusion axioms, and various extensions of $\mathcal{EL}$ and their effects on the complexity of subsumption w.r.t. general concept inclusion axioms have been studied.

In this thesis, we sharpen the border between tractability and intractability of subsumption in extensions of $\mathcal{EL}$ w.r.t. cyclic TBoxes. We provide two new extensions $\mathcal{EL}^{\sqcup,\sqcap,\neg}(\mathcal{D})$ and $\mathcal{EL}^{\sqcup,\sqcap,\geq}$ for which subsumption can be computed in polynomial time w.r.t. cyclic TBoxes. The first extends $\mathcal{EL}$ by role con- and disjunction in disjunctive normal form, primitive negation and p-admissible concrete domains, and the second by role con- and disjunction in disjunctive normal form and at-least restrictions. Moreover, we show that a combination of $\mathcal{EL}^{\sqcup,\sqcap,\neg}(\mathcal{D})$ and $\mathcal{EL}^{\sqcup,\sqcap,\geq}$ leads to intractability of subsumption w.r.t. cyclic TBoxes, as well as $\mathcal{EL}$ extended by negation, disjunction, transitive closure over role names, functionality and concrete domains with abstract feature chains. This justifies the fact that—except for inverse roles which remain an open problem—both $\mathcal{EL}^{\sqcup,\sqcap,\neg}(\mathcal{D})$ and $\mathcal{EL}^{\sqcup,\sqcap,\geq}$ are maximal in the sense that they cannot be further extended without losing tractability of subsumption w.r.t. cyclic TBoxes.

Statement of Academic Honesty

I declare that this thesis was composed by myself and that the work contained herein is my own, except where explicitly stated otherwise.

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbständig und unter ausschliesslicher Verwendung der angegebenen Quellen und Hilfsmittel verfasst, sowie Zitate aus anderen Werken durch Quellenangabe kenntlich gemacht habe.

Dresden, den 15. August 2007

Christoph Haase

Acknowledgments

First of all, I would like to thank my supervisor, Dr. Carsten Lutz, for the insightful and fruitful discussions at the Friday afternoons that guided me through this thesis. They played a key part in the success of this thesis. Moreover, I thank Prof. Franz Baader for introducing to me description logics and making it possible to write my thesis at the Chair for Automata Theory.

I thank my girlfriend Katharina for the love, understanding and patience she gave me—always, and in particular whilst working on this thesis. Furthermore, I thank my parents for the love and absolute support I received throughout my whole life. I am proud to be the brother of my two sisters, Claudia and Susanne, who make me joyful in their own ways.

During my studies, there have been many persons that influenced and supported me. Unfortunately, I can only list some of them. Notably, the lectures given by Dr. Hendrik Tews and Prof. Horst Reichel aroused my interest in theoretical computer science. The resulting collaboration within the VFiasco project was worth more than any course I attended. Manuel Zabelt has always been a good friend to me, thank you!

Last but not least, I would like to thank the Free State of Saxony and the Sokrates/ Erasmus Program of the European Union that made it possible to study without tuition fees and enabled me to study for almost one year in Bristol respectively.

Contents

1 Introduction **1**

1.1 Description Logics . . . 1

1.2 $\mathcal{EL}$ as a Tractable Description Logic . . . 3

1.3 Objective and Structure of this Thesis . . . 4

2 Preliminaries **5**

2.1 Introducing $\mathcal{EL}$ . . . 5

2.2 Extensions of $\mathcal{EL}$ . . . 7

2.3 $\mathcal{EL}$-Concept Descriptions as Graphs . . . 12

2.4 Terminologies . . . 15

2.4.1 Standard Terminology Boxes . . . 16

2.4.2 Normalized $\mathcal{EL}$-TBoxes . . . 18

2.4.3 General Terminology Boxes . . . 20

3 Tractable Extensions of $\mathcal{EL}$ **22**

3.1 $\mathcal{EL}^{\sqcup,\sqcap,\neg}(\mathcal{D})$ . . . 22

3.2 $\mathcal{EL}^{\sqcup,\sqcap,\geq}$ . . . 29

4 Intractable Extensions of $\mathcal{EL}$ **35**

4.1 A Combination of $\mathcal{EL}^{\sqcup,\sqcap,\neg}(\mathcal{D})$ and $\mathcal{EL}^{\sqcup,\sqcap,\geq}$ . . . 35

4.2 Arbitrary Role Con- and Disjunction . . . 37

4.3 Negation . . . 39

4.4 Disjunction . . . 39

4.5 Transitive Closure over Role Names . . . 44

4.6 Functionality . . . 55

4.7 Concrete Domains with Abstract Feature Chains . . . 63

4.8 Inverse Roles . . . 65

5 Conclusion **67**

Bibliography **70**

Chapter 1

Introduction

1.1 Description Logics

Description logics are a logic-based family of formalisms for knowledge representation, which is a subfield of Artificial Intelligence. They are used in knowledge-based systems that offer reasoning services which allow for finding implicit consequences from explicitly stored knowledge and are sound and complete. Approaches to represent knowledge began to arise in the 1970's. In contrast to early proposals, description logics enjoy a well-defined syntax and semantics. A detailed introduction to description logics and their history can be found in "The Description Logic Handbook" (Baader, Calvanese, McGuinness, Nardi & Patel-Schneider 2003). Particular description logics have individual names, e.g., $\mathcal{EL}$, $\mathcal{ALC}$ or $\mathcal{ALCN}$. In this thesis, we are going to investigate the complexity of the reasoning task *subsumption* in extensions of the description logic $\mathcal{EL}$.

Concept descriptions can been seen as the basis for expressing knowledge in description logics. They are built upon *concept names*, *role names* and *concept constructors*. The *semantics* of concept descriptions is given in terms of an *interpretation*. It consists of a non-empty set of individuals, the *interpretation domain*, and an *interpretation function*. The latter assigns concept names to sets of elements of the interpretation domain, and role names to a binary relation on the interpretation domain. The interpretation function is then inductively extended to arbitrary concept descriptions, and thus concept descriptions are interpreted as subsets of the interpretation domain.

Let us now consider an example of a concept description of the description logic $\mathcal{ALC}$ that describes the set of those humans who are female, have a child and whose children are all male:

$$\mathsf{Human} \sqcap \mathsf{Female} \sqcap \exists \mathsf{has\_child}.\top \sqcap \forall \mathsf{has\_child}.\mathsf{Male} \tag{1.1}$$

Here, Human, Female and Male are concept names, $\mathsf{has\_child}$ is a role name, and $\top$, $\sqcap$, $\forall$ and $\exists$ are concept constructors. In this context, $\top$ (*top*) can be read as a wild-card concept name, $\sqcap$ as "and" (*disjunction*), $\exists$ as "there exists" (*existential restriction*) and $\forall$ as "for all" (*value restriction*). The following interpretation $\mathcal{I}$ gives semantics to the concept description above, where $\Delta^{\mathcal{I}}$ is the interpretation domain and $\cdot^{\mathcal{I}}$ is the interpretation function:

- $\Delta^{\mathcal{I}} := \{\mathsf{ANNA}, \mathsf{MARIA}, \mathsf{BOB}, \mathsf{ALICE}\}$
- $\mathsf{Human}^{\mathcal{I}} := \Delta^{\mathcal{I}}; \mathsf{Female}^{\mathcal{I}} := \{\mathsf{ANNA}, \mathsf{MARIA}, \mathsf{ALICE}\}; \mathsf{Male}^{\mathcal{I}} := \{\mathsf{BOB}\}$
- $\mathsf{has\_child}^{\mathcal{I}} := \{(\mathsf{ANNA}, \mathsf{BOB}), (\mathsf{MARIA}, \mathsf{ALICE})\}$

Under $\mathcal{I}$, the concept description (1.1) is interpreted as $\{\mathsf{ANNA}\}$, from which we can conclude that Anna is a women that has a child and whose children are all male in the interpretation $\mathcal{I}$.

For the purpose of structuring knowledge and abbreviating complex concept descriptions, terminology formalisms—also known as TBoxes—have been introduced. In the context of the example from above, the following $\mathcal{ALC}$-TBox describes the relationships in families:

$$\begin{array}{rcl} \mathsf{Parent} & \equiv & \mathsf{Human} \sqcap \exists \mathsf{has\_child}.\top \\ \mathsf{Mother} & \equiv & \mathsf{Female} \sqcap \mathsf{Parent} \\ \mathsf{Father} & \equiv & \mathsf{Male} \sqcap \mathsf{Parent} \\ \mathsf{Mother\_of\_male} & \equiv & \mathsf{Mother} \sqcap \forall \mathsf{has\_child}.\mathsf{Male} \end{array}$$

An interpretation is a *model of a TBox* if it “respects” the *concept definitions* in the TBox. A TBox may contain cyclic definitions, i.e., concept definitions that directly or indirectly refer to themselves. Moreover, there are *general TBoxes* that allow to formulate so-called *general concept inclusion axioms*.

Particular description logics differ in which concept constructors they allow. For example, in $\mathcal{ALC}$ we cannot describe the set of mothers that have at least three children. However, for n being a positive integer and r a role name, if we introduce the additional concept constructor $\geq nr$, we could express this as follows:

$$\mathsf{Human} \sqcap \mathsf{Female} \sqcap \geq 3\ \mathsf{has\_child}$$

Amongst others, this concept constructor is present in the concept language $\mathcal{ALCN}$.

A main reasoning task in description logics is *subsumption*. Given two concept descriptions C, D, we say C *is subsumed by* D ($C \sqsubseteq D$) whenever for every interpretation $\mathcal{I}$, the set described by C is a subset of the set described by D ($C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$). If $C \sqsubseteq D$, we can think of D as being more general than C. Subsumption may also take TBoxes into consideration, i.e., C *is subsumed by* D *w.r.t. a TBox* $\mathcal{T}$ ($C \sqsubseteq_{\mathcal{T}} D$) whenever $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ for all models $\mathcal{I}$ of $\mathcal{T}$.

The complexity of reasoning is of special interest in description logics. This is mainly motivated by the fact that an extensive study of the complexity of reasoning in a particular concept language allows to make qualitative statements about its usability for certain application domains. For example, in embedded systems, reasoners would need to respond quickly, whereas in a web-information system it is acceptable to wait a second for a query response. There are several *sources of complexity* for reasoning in a concept language (Baader et al. 2003). The complexity may rise depending on the presence and combination of concept constructors as well as the presence and the sort of TBoxes. Furthermore, the interplay of some concept constructors can even lead to undecidability of a reasoning task. We say a reasoning task is *tractable* in a description logic when it can be decided in polynomial time w.r.t. the size of the input, i.e., a concept description, a TBox or both. Otherwise, reasoning is said to be *intractable*.

1.2 $\mathcal{EL}$ as a Tractable Description Logic

The quest for tractable description logics arose in the 1980's after the first intractability results were shown. This was mainly grounded in the opinion that a knowledge-based system should be able to answer "in time", and this was put on a level with tractability. In many cases, the small description logic $\mathcal{FL}_0$—which only allows for conjunction and value restriction—was the starting point of the search for tractable concept languages. Around this time, various extensions of $\mathcal{FL}_0$ have been considered for which subsumption is tractable without TBoxes, see e.g. (Donini, Lenzerini, Nardi & Nutt 1991). Moreover, polynomial time reasoners like Classic (Brachman, McGuiness, Patel-Schneider & Resnick 1990) were developed. However, it was Nebel who showed in 1990 that as soon as terminologies come into play, subsumption even in $\mathcal{FL}_0$ becomes intractable (Nebel 1990b). Since terminologies have been considered to be important in modeling knowledge, the search for tractable description logics was given up. Instead, one successfully concentrated on expressive description logics and highly optimized algorithms for practical implementations. It turned out that despite the high worst case complexity of expressive description logics, reasoning in real world problems is manageable. Nowadays, the Semantic Web is the most popular application of expressive description logics. Its underlying description logic is the Ontology Web Language (OWL) (Baader, Horrocks & Sattler 2005).

However, Baader showed in 2002 that in the description logic $\mathcal{EL}$ —which allows for conjunction and existential restriction—subsumption w.r.t. cyclic TBoxes can surprisingly be decided in polynomial time (Baader 2003). It furthermore turned out that even w.r.t. general TBoxes subsumption is polynomial (Brandt 2004). Building upon that, in (Baader, Brandt & Lutz 2005a) the description logic $\mathcal{EL}^{++}$ has been presented to be a very expressive extension of $\mathcal{EL}$ for which subsumption is tractable w.r.t. general TBoxes. Interestingly, $\mathcal{EL}$ had not been investigated in this depth before. Basically, this had historical reasons: Arcs in semantic networks and slots in frames, which can be seen as predecessors of description logics, were considered to be read as value restrictions rather than existential restrictions.

From a first impression, one might ask whether a relatively small concept language like $\mathcal{EL}$ is sufficient to find usage in practical applications. However, this is the case: For instance, the Systematized Nomenclature of Medicine (Snomed) corresponds to an acyclic $\mathcal{EL}$-TBox (Baader, Brandt & Lutz 2005a). Moreover, it has recently been shown that even relatively straight-forward implementations of $\mathcal{EL}$-subsumption algorithms can outperform current state of the art reasoners (Baader, Lutz & Suntisrivaraporn 2007).

1.3 Objective and Structure of this Thesis

As stated before, the complexity of reasoning in extensions of $\mathcal{EL}$ w.r.t. *general terminologies* has been extensively studied in (Baader, Brandt & Lutz 2005a). However, general terminologies are not always a compulsory premise for practical applications, as seen above for Smoned. It is an open question whether there are extensions of $\mathcal{EL}$ for which subsumption remains tractable w.r.t. *cyclic terminologies*—and this is the main objective of this thesis. We will sharpen the tractability border for subsumption in extensions of $\mathcal{EL}$ w.r.t. cyclic TBoxes and identify maximal extensions of $\mathcal{EL}$ such that subsumption is tractable. Moreover, we prove intractability of subsumption w.r.t. cyclic TBoxes for a variety of other extensions. We will furthermore present exact complexities of subsumption in extensions of $\mathcal{EL}$ without and with TBoxes.

This thesis is structured as follows: In Chapter 2, we are going to lay the foundations for this thesis. We formally introduce $\mathcal{EL}$, its syntax, semantics and the extensions that we are going to consider. Moreover, we introduce terminology boxes. Thereafter, we consider in Chapter 3 two extensions $\mathcal{EL}^{\sqcup,\sqcap,\neg}(\mathcal{D})$ and $\mathcal{EL}^{\sqcup,\sqcap,\geq}$ of $\mathcal{EL}$ for which the subsumption problem is tractable w.r.t. cyclic TBoxes. The first extends $\mathcal{EL}$ by primitive negation, con- and disjunction of role names in disjunctive normal form and p-admissible concrete domains. The latter is $\mathcal{EL}$ extended by role con- and disjunction in disjunctive normal form and at-least restrictions. A combination of both $\mathcal{EL}^{\sqcup,\sqcap,\neg}(\mathcal{D})$ and $\mathcal{EL}^{\sqcup,\sqcap,\geq}$ is considered in Chapter 4, as well as the extension of $\mathcal{EL}$ by negation, disjunction, transitive closure over role names, functionality and concrete domains with abstract feature chains, for which subsumption becomes intractable w.r.t. cyclic terminologies. We summarize the results of this thesis and briefly discuss future prospects in Chapter 5.

It is assumed that the reader has a basic understanding of mathematics. In particular, knowledge about the basic essentials of complexity theory, set theory and first-order predicate logic are required.

Chapter 2

Preliminaries

2.1 Introducing $\mathcal{EL}$

In description logics, *concept descriptions* are inductively defined with the help of *concept constructors* starting with pairwise disjoint countably infinite sets of *concept names* N_C and *role names* N_R. In the following, we introduce the concept descriptions of the concept language $\mathcal{EL}$ that is the basis of all concept languages considered in this thesis.

Definition 1 ($\mathcal{EL}$-concept descriptions) Let A be a concept name and r a role name. The set of $\mathcal{EL}$-concept descriptions is defined according to the following syntax rule:

$$
\begin{array}{lcll}
C, D & \longrightarrow & A & \text{(concept name)} \\
 & & \top & \text{(top concept)} \\
 & & \exists r.C & \text{(existential restriction)} \\
 & & C \sqcap D & \text{(conjunction)}
\end{array}
$$

◊

Subsequently, we will use A and B to denote concept names, r and s for role names, and C and D for concept descriptions. The size $|C|$ of an $\mathcal{EL}$-concept description C is defined to be the number of symbols used to write it down, and likewise for the extensions of $\mathcal{EL}$ considered in this thesis. For an $\mathcal{EL}$-concept description C, we inductively define the *role depth of* C by induction on the structure of C:

$$
\begin{array}{rcl}
rdepth(A) := rdepth(\top) & := & 0 \\
rdepth(\exists r.C) & := & rdepth(C) + 1 \\
rdepth(C_1 \sqcap C_2) & := & max(rdepth(C_1), rdepth(C_2))
\end{array}
$$

Now, let us consider two examples of concept descriptions and their informal meaning.

Example 1.

$$\mathsf{Female} \sqcap \exists \mathsf{has\_child}.\top$$

$$\mathsf{Male} \sqcap \exists \mathsf{likes}.(\mathsf{Female} \sqcap \exists \mathsf{child}.(\exists \mathsf{member\_in}.\mathsf{Choir}))$$

The first concept description describes persons who are female and have a child, and the second all men who like women who have children who are member in a choir. ■

The formal semantics of concept descriptions is given in terms of a Tarski-style interpretation.

Definition 2 ($\mathcal{EL}$ semantics) An *interpretation* $\mathcal{I}$ is a tuple $(\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where $\Delta^{\mathcal{I}}$ is the nonempty *interpretation domain*, and $\cdot^{\mathcal{I}}$ the *interpretation function*. The latter maps

- every concept name A to a subset of $\Delta^{\mathcal{I}}$
- every role name r to a subset of $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$.

The interpretation function is inductively extended to complex concept descriptions as follows:

$$\begin{array}{rcl} (\top)^{\mathcal{I}} & := & \Delta^{\mathcal{I}} \\ (C \sqcap D)^{\mathcal{I}} & := & C^{\mathcal{I}} \cap D^{\mathcal{I}} \\ (\exists r.C)^{\mathcal{I}} & := & \{x \in \Delta^{\mathcal{I}} \mid \exists y \in \Delta^{\mathcal{I}}.((x,y) \in r^{\mathcal{I}} \wedge y \in C^{\mathcal{I}})\} \end{array}$$

The union $\mathcal{I} = \mathcal{I}_1 \cup \mathcal{I}_2$ of two interpretations is defined to be the union of the components of $\mathcal{I}_1$ and $\mathcal{I}_2$. ◊

An $\mathcal{EL}$-concept description C is *satisfiable* iff there exists an interpretation $\mathcal{I}$ such that $C^{\mathcal{I}} \neq \emptyset$. A concept description C is *subsumed* by D iff $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ for all interpretations $\mathcal{I}$. We write $C \sqsubseteq D$ iff C is subsumed by D. Both satisfiability and subsumption are standard reasoning tasks in description logics. For $\mathcal{EL}$-concept descriptions, satisfiability is trivial in the sense that every $\mathcal{EL}$-concept description is satisfiable. Therefore, our main focus in this thesis will be on the complexity of subsumption. A special property of $\mathcal{EL}$ is that checking subsumption of $\mathcal{EL}$-concept descriptions C, D can be done in polynomial time in $|C| + |D|$, which will be proven later.

A further popular standard reasoning task is instance checking. Given a snapshot of a world, instance checking is to determine whether some object of the world belongs to a concept description. We will not consider instance checking in this thesis. For $\mathcal{EL}$ and its extensions, the complexity of instance checking has been widely investigated in (Krisnadhi 2007).

Apart the standard reasoning tasks, so-called *non-standard inferences* have been considered in the literature (Küsters 2001). That is, for instance, the least common subsumer of concept descriptions. In this thesis, we will not consider these reasoning tasks.

Name	Syntax	Semantics	Symbol
Role conjunction	$R \sqcap S$	$R^{\mathcal{I}} \cap S^{\mathcal{I}}$	$\sqcap$
Role disjunction	$R \sqcup S$	$R^{\mathcal{I}} \cup S^{\mathcal{I}}$	$\sqcup$
Inverse role	R^-	$\{(x, y)\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \mid (y, x) \in R^{\mathcal{I}}\}$	$\mathcal{I}$
Transitive closure	R^+	$\bigcup_{i \geq 1}(R^{\mathcal{I}})^i$	$+$

Table 2.1: Additional role constructors.

2.2 Extensions of $\mathcal{EL}$

It is not hard to see that the expressiveness of $\mathcal{EL}$ is rather limited for some applications. For example, we are not able to define a concept description that describes a mother that has more than two children. Therefore, we subsequently introduce several extensions of $\mathcal{EL}$ that increase expressiveness, but may also increase the complexity of reasoning.

Role Constructors

Firstly, we introduce *role constructors* as a possible extension, which allow to construct *complex roles* from role names. The role constructors considered in this thesis and their syntax and semantics are presented in Table 2.1. There and in the following, R and S denote complex roles and R and S are equivalent iff $R^{\mathcal{I}} = S^{\mathcal{I}}$ for all interpretations $\mathcal{I}$. Moreover, in the definition of transitive closure, for a relation $R \subseteq M \times M$ and $n > 0$,

$$\begin{aligned} R^0 &:= Id_M \\ R^1 &:= R \\ R^{n+1} &:= R^n \cup \{(x, y) \mid \exists z \in M.((x, z) \in R^n \wedge (z, y) \in R^n)\} \end{aligned}$$

$$R(x) := \{y \mid (x, y) \in R\}$$

For convenience, if $R(x) = \{y\}$ is a singleton, we sometimes write $R(x) = y$.

When complex roles are present in a given extension of $\mathcal{EL}$, they can be used in existential restrictions in an obvious way. For an interpretation $\mathcal{I}$, we define $x \in (\exists R.C)^{\mathcal{I}}$ iff $x \in \{y \mid \exists z \in \Delta^{\mathcal{I}}.((y, z) \in R^{\mathcal{I}} \wedge z \in C^{\mathcal{I}})\}$. Extending $\mathcal{EL}$ by a set of particular role constructor yields a particular $\mathcal{EL}$-language. Each such language is named by a string of the form

$$\mathcal{EL}[\mathcal{I}][^{\sqcap}][^{\sqcup}][^{+}]$$

For instance, $\mathcal{ELI}^{\sqcup}$ is the concept language allowing for top, concept names, conjunction, existential restriction, inverse roles and role disjunction.

The following three examples illustrate possible applications of the additional role constructors.

Example 2.

$$\exists \mathsf{has\_child}^{+}.\mathsf{Female}$$

$$\mathsf{Female} \sqcap \exists(\mathsf{has\_child} \sqcup \mathsf{has\_adopted}).\top$$

$$\mathsf{Male} \sqcap \exists \mathsf{has\_child}^{-}.\top$$

The first concept description describes persons that have a female offspring. Next, mothers are described, i.e., women that have a child or have adopted somebody. Lastly, sons are defined to be male persons that are the child of somebody. ■

Concept Constructors

Next, we consider additional concept constructors. Their syntax, semantics and symbols are presented in the second part in Table 2.2. There, for a set M, by $\#M$ we denote the cardinality of M. Existential restriction, conjunction and top occur in Table 2.2 for the sake of completeness. Negation is also called *complement*. It allows for fully negating concept descriptions, i.e., to express that some concept description must not hold. Primitive negation on the other hand only allows for negation only occurring in front of concept names. Functionality is a special case of at-most restrictions, which allows to define a maximum number of r-successors at some point. In this thesis, we will not consider at-most restrictions directly, but obviously hardness results for extensions of $\mathcal{EL}$ involving functionality carry over to extensions of $\mathcal{EL}$ with at-most restrictions. Value restriction and bottom are separated in Table 2.2, since we will only need them to define concept languages different that are not member in the $\mathcal{EL}$ family.

Let us have a look at some examples of concept descriptions that make use of the additional concept constructors.

Example 3.

$$\exists \mathsf{has\_child}.\mathsf{Female} \sqcup \exists \mathsf{has\_adopted}.\mathsf{Female}$$

$$\mathsf{Male} \sqcap \geq 3\ \mathsf{has\_child}$$

$$\mathsf{Female} \sqcap \exists \mathsf{has\_profession}.\mathsf{Academic} \sqcap \neg\exists \mathsf{has\_child}.\top$$

The first concept description describes persons that have a daughter or have adopted a daughter. Next, fathers with more than two children are described. Finally, the last concept description describes women who are academics and do not have a child. ■

Name	Syntax	Semantics	Symbol
Concept name	A	$A^{\mathcal{I}}$	
Top	$\top$	$\Delta^{\mathcal{I}}$	
Conjunction	$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$	
Existential restriction	$\exists r.C$	$\{x \in \Delta^{\mathcal{I}} \mid \exists y \in \Delta^{\mathcal{I}}.((x,y) \in r^{\mathcal{I}} \wedge y \in C^{\mathcal{I}})\}$	$\mathcal{E}$
Negation	$\neg C$	$\Delta^{\mathcal{I}} \backslash C^{\mathcal{I}}$	$\mathcal{C}$
Primitive negation	$\neg A$	$\Delta^{\mathcal{I}} \backslash A^{\mathcal{I}}$	$\neg$
Disjunction	$C \sqcup D$	$C^{\mathcal{I}} \cup D^{\mathcal{I}}$	$\mathcal{U}$
At-least restrictions	$\geq nr$	$\{x \in \Delta^{\mathcal{I}} \mid \#\{y \in \Delta^{\mathcal{I}} \mid (x,y) \in r^{\mathcal{I}}\} \geq n\}$	$\geq$
At-most restrictions	$\leq nr$	$\{x \in \Delta^{\mathcal{I}} \mid \#\{y \in \Delta^{\mathcal{I}} \mid (x,y) \in r^{\mathcal{I}}\} \leq n\}$	$\leq$
Functionality	$\leq 1r$	$\{x \in \Delta^{\mathcal{I}} \mid \#\{y \in \Delta^{\mathcal{I}} \mid (x,y) \in r^{\mathcal{I}}\} \leq 1\}$	$\mathcal{F}$
Bottom	$\bot$	$\emptyset$	
Value restriction	$\forall R.C$	$\{x \in \Delta^{\mathcal{I}} \mid \forall y \in \Delta^{\mathcal{I}}.((x,y) \in R^{\mathcal{I}} \rightarrow y \in C^{\mathcal{I}})\}$	$\mathcal{A}$

Table 2.2: Concept constructors considered in this thesis.

As for role constructors, depending on the presence of additional concept constructors, the name of the resulting concept language is a string of the form

$$[\mathcal{A}][\mathcal{E}]\mathcal{L}[\mathcal{C}][\mathcal{U}][\mathcal{F}][^{-}][^{\geq}][^{\leq}]$$

Concrete Domains

One limitation of description logics is that *all* knowledge has to be described on an abstract level. In particular, it is hard to adequately express knowledge about concrete qualities, e.g., length, duration or temperature. To overcome this problem, Baader and Hanschke introduced concrete domains in (Baader & Hanschke 1991). They have widely been investigated by Lutz in (Lutz 2003).

Definition 3 (Concrete Domain) A *concrete domain* $\mathcal{D}$ is a pair $(\Delta_{\mathcal{D}}, \Phi_{\mathcal{D}})$, where $\Delta_{\mathcal{D}}$ is a set and $\Phi_{\mathcal{D}}$ is a set of predicates names. Each predicate name p is associated with an arity n and an n-ary predicate $p^{\mathcal{D}} \subseteq \Delta_{\mathcal{D}}^n$. ◊

Firstly, let us introduce an example of a concrete domain from (Lutz 2002). The concrete domain E is defined by setting

$$\begin{aligned}\Delta_{\mathsf{E}} &:= \mathbb{N}\\ \Phi_{\mathsf{E}} &:= \{\top_{\mathsf{E}}, \bot_{\mathsf{E}}\} \cup \{P_r \mid P \in \{=, \neq, <, >, \leq, \geq\}, r \in \Delta_{\mathsf{E}}\}\end{aligned}$$

Both $\top_{\mathsf{E}}$ and $\bot_{\mathsf{E}}$ have arity zero, and the other predicates are unary. All predicates have the obvious extension, e.g.,

$$(\geq_7)^{\mathsf{E}} = \{n \in \mathbb{N} \mid n \geq 7\}$$

When a concept language is equipped with a concrete domain $\mathcal{D}$, we write $\mathcal{D}$ in brackets after its name, e.g., $\mathcal{EL}(\mathsf{E})$. In the presence of concrete domains, we assume the set of role names to contain an infinite subset of *abstract features* N_{aF}. Moreover, there is a countably infinite set of *concrete features* N_{cF} such that $N_{cF} \cap (N_R \cup N_C) = \emptyset$. A *path of features* g is a concatenation $r_1 \dots r_n f$ of abstract features $r_1, \dots, r_n$ and a concrete feature f.

Definition 4 ($\mathcal{EL}(\mathcal{D})$ syntax and semantics) Let $\mathcal{D}$ be a concrete domain and N_{aF} be a countably infinite subset of N_R of abstract features. Let N_{cF} be a countably infinite set of concrete features such that $N_{cF} \cap (N_R \cup N_C) = \emptyset$. $\mathcal{EL}(\mathcal{D})$ is obtained from $\mathcal{EL}$ by allowing for the additional *concrete domain constructor*: For any n-ary predicate $p \in \Phi_{\mathcal{D}}$, $p(g_1, \dots, g_n)$ is an $\mathcal{EL}(\mathcal{D})$-concept description, where $g_1, \dots, g_n$ are *paths of features*. For an interpretation $\mathcal{I}$,

- $r \in N_{aF}$ is interpreted as a partial function from $\Delta^{\mathcal{I}}$ to $\Delta^{\mathcal{I}}$,

- $f \in N_{cF}$ is interpreted as a partial function from $\Delta^{\mathcal{I}}$ to $\Delta_{\mathcal{D}}$, and
- a path of features $g = r_1 \dots r_n f$ is interpreted as $r_1^{\mathcal{I}} \circ \dots \circ r_n^{\mathcal{I}} \circ f^{\mathcal{I}}$.

The semantics of the concrete domain constructor is as follows:

$$p(g_1, \dots, g_n)^{\mathcal{I}} \quad := \quad \{x \in \Delta^{\mathcal{I}} \mid \exists d_1, \dots, d_n \in \Delta_{\mathcal{D}}.(g_1^{\mathcal{I}}(x) = d_1 \wedge \dots \wedge g_n^{\mathcal{I}}(x) = d_n \wedge (d_1, \dots, d_n) \in p^{\mathcal{D}})\} \quad \diamondsuit$$

For our tractable extensions of $\mathcal{EL}$ in Chapter 3, we disallow abstract features, i.e., only concrete features must occur in concrete domain constructors. We call $p(f_1, \dots, f_n)$ an *atom*, where $f_i \in N_{cF}, 1 \leq i \leq n$. Given a conjunction ψ of atoms over the concrete features $f_1, \dots, f_n$, $\delta : N_{cF} \rightarrow \Delta_{\mathcal{D}}$ is called a *solution to* ψ iff $\psi[f_1/\delta(f_1), \dots, f_n/\delta(f_n)]$ evaluates to true in $\mathcal{D}$. We say ψ is *satisfiable* iff there exists a solution to ψ, and ψ_1 *implies* ψ_2 iff every solution to ψ_1 is a solution to ψ_2. For the remainder of this thesis, we restrict satisfiability and implication in the concrete domains that we deal with to be computable in polynomial time.

Definition 5 A concrete domain $\mathcal{D}$ is *p-admissible* iff satisfiability and implication in $\mathcal{D}$ can be decided in polynomial time. Furthermore, $\mathcal{D}$ is *convex* iff whenever a conjunction of atoms implies a disjunction of atoms, then it also implies one of its disjuncts. $\diamondsuit$

The following proposition was shown in (Lutz 2003).

Proposition 1 *The concrete domain* E *is p-admissible.*

However, E is *not* convex. For example,

$$\geq_7(f) \wedge \leq_8(f) \quad \text{implies} \quad =_7(f) \vee =_8(f),$$

but neither $=_7(f)$ is implied, nor $=_8(f)$. For $\mathcal{EL}^{++}$, concrete domains are required to be convex (Baader, Brandt & Lutz 2005a). This is *not* the case in this thesis. We close this section with two examples of $\mathcal{EL}(\mathsf{E})$-concept descriptions.

Example 4.

$$\mathsf{Human} \sqcap \geq_{13}(\mathsf{has\_age}) \sqcap \leq_{19}(\mathsf{has\_age})$$
$$\mathsf{Human} \sqcap \mathsf{Male} \sqcap \geq_{67}(\mathsf{has\_age})$$

The first concept description describes teenagers and the second male retirees. ∎

Name	Restriction	Symbol
Global functionality	$\forall\, \mathcal{I}, x \in \Delta^{\mathcal{I}}, r \in N_R.\#r^{\mathcal{I}}(x) \leq 1$	f
Totality	$\forall\, \mathcal{I}, x \in \Delta^{\mathcal{I}}, r \in N_R.\#r^{\mathcal{I}}(x) \geq 1$	t

Table 2.3: Additional concept constructors considered in this thesis.

Restrictions on Interpretations

The last extension of $\mathcal{EL}$ we consider are restrictions on interpretations. Namely, we will consider global functionality and totality requirements of the interpretation of role names. Both extensions are presented in Table 2.3. Global functionality requires that in all admissible interpretations, all nodes have at most at most one successor node of each role name, and totality requires them to have at least one successor. Note, that we will subsequently use the notation of "totality" and "functionality" for arbitrary relations as well.

2.3 $\mathcal{EL}$-Concept Descriptions as Graphs

An important observation is that $\mathcal{EL}$-concept descriptions can be viewed as directed labeled graphs. This allows to decide subsumption between $\mathcal{EL}$-concept descriptions in terms of the existence of a homomorphism between the graphs of the concept descriptions.

Definition 6 (Directed Labeled Graph) Let L_E be a set of *edge labels* and L_V a set of *node labels.* A directed labeled graph $\mathcal{G} = (V, E, \ell)$ over L_E and L_V consists of a set of *nodes* V, a set of *labeled edges* $E \subseteq V \times L_E \times V$, and a *node labeling function* $\ell : V \to L_V$. $\Diamond$

If not stated otherwise, in the following we will call a directed labeled graph just a graph. The *size of a graph* is defined to be the number of its vertices. For an edge $(v, r, w) \in E$, we call w an *(r-)successor* of v, and v an *(r-)predecessor* of w. We say $x_1 \xrightarrow{r_1} x_2 \xrightarrow{r_2} \dots \xrightarrow{r_n} x_{n+1}$ is a *path in* $\mathcal{G}$ of length n iff $(x_i, r_i, x_{i+1}) \in E$ for $1 \leq i \leq n$. For a path $p = x_1 \xrightarrow{r_1} \dots \xrightarrow{r_n} x_{n+1}$ and a node $v \in V$, $v \in p$ iff $v = x_i$ for some $1 \leq i \leq n+1$. A graph contains a cycle iff there is a path $x \xrightarrow{r_1} \dots \xrightarrow{r_n} x$ in $\mathcal{G}$.

$\mathcal{EL}$-description graphs are directed labeled graphs, whose edges are labeled with role names from N_R and vertices with finite subsets of N_C. $\mathcal{EL}$-concept descriptions are represented by *$\mathcal{EL}$-description trees.* An $\mathcal{EL}$-description tree is a connected $\mathcal{EL}$-description-graph t that does not contain any cycle, has a distinguished node called the *root* of t

that has no predecessor, and every other node has exactly one predecessor. As a naming convention, if not stated otherwise, for a concept description C, its corresponding $\mathcal{EL}$-description tree is called t_C and its root x_C. Nodes that do not have any successor are called *leaves*. For a tree, we use path and *branch* synonymously. Let t be a tree and $x \in V_t$ a node, $depth(x) := max\{n \mid x \xrightarrow{r_1} \ldots \xrightarrow{r_n} x_{n+1}\}$.

The $\mathcal{EL}$-description tree $t = (V_t, E_t, \ell_t)$ with root x corresponding to an $\mathcal{EL}$-concept description C is defined via induction on $d = rdepth(C)$:

$d = 0:$ So $C = P_1 \sqcap \ldots \sqcap P_k$ and $V_t := \{x\}$, $E_t := \emptyset$ and $\ell_t := x \mapsto \{P_1, \ldots, P_k\}$.

$d \to d+1:$ We have $C = P_1 \sqcap \ldots \sqcap P_k \sqcap \exists r_1.C_1 \sqcap \ldots \sqcap \exists r_m.C_m$. By the induction hypothesis, there exist trees $t_1, \ldots, t_m$ with roots $x_1, \ldots, x_m$ for the concept descriptions $C_1, \ldots, C_m$. W.l.o.g. we assume the set of vertices of $t_1, \ldots, t_m$ to be disjoint. We define $V_t := \bigcup_{1 \le i \le m} V_{t_i} \cup \{x\}$, $E_t := \bigcup_{1 \le i \le m} E_{t_i} \cup \{(x, r_i, x_i) \mid 1 \le i \le m\}$ and

$$\ell_t(v) := \begin{cases} \{P_1, \ldots, P_k\} & \text{if } v = x \\ \ell_{t_i}(v) & \text{if } v \in V_{t_i}, 1 \le i \le m \end{cases}$$

We can also view interpretations as graphs. Let $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ be an interpretation. The $\mathcal{EL}$-description graph $\mathcal{G}_{\mathcal{I}} = (V_{\mathcal{I}}, E_{\mathcal{I}}, \ell_{\mathcal{I}})$ corresponding to $\mathcal{I}$ is defined as follows:

- $x \in V_{\mathcal{I}}$ iff $x \in \Delta^{\mathcal{I}}$
- $(x, r, y) \in E_{\mathcal{I}}$ iff $(x, y) \in r^{\mathcal{I}}$
- $P \in \ell_{\mathcal{I}}(x)$ iff $x \in P^{\mathcal{I}}$

Note, that this definition also allows us to view $\mathcal{EL}$-description graphs as interpretations. In particular in the remainder of this thesis, we will heavily use $\mathcal{EL}$-description trees as interpretations. For convenience, given an $\mathcal{EL}$-concept description C, its corresponding $\mathcal{EL}$-description tree t_C with root x_C and some $\mathcal{EL}$-concept description D, we will lazily write "$x_C \in D^{t_C}$" instead of "$x_C \in D^{\mathcal{I}}$, where $\mathcal{I}$ is the interpretation corresponding to t_C".

Let us now introduce homomorphisms between $\mathcal{EL}$-description graphs and their relationship to interpretations.

Definition 7 (Homomorphism) Let $\mathcal{G}_1, \mathcal{G}_2$ be $\mathcal{EL}$-description graphs. A *homomorphism* from $x \in V_{\mathcal{G}_1}$ to $y \in V_{\mathcal{G}_2}$ is a functional binary relation $\mathcal{H} \subseteq V_{\mathcal{G}_1} \times V_{\mathcal{G}_2}$ such that:

- $(x, y) \in \mathcal{H}$
- For all $(v, w) \in \mathcal{H}$:
  - $\ell(v) \subseteq \ell(w)$

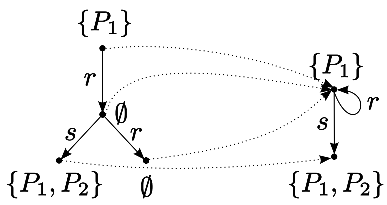

Figure 2.1: The $\mathcal{EL}$-description-graph t_C corresponding to the concept description $C = P_1 \sqcap \exists r.(\exists s.(P_1 \sqcap P_2) \sqcap \exists r.\top)$ (left) and $\mathcal{G}_\mathcal{I}$ corresponding to an interpretation $\mathcal{I}$ (right). The dashed arrows illustrate a homomorphism from t_C to $\mathcal{G}_\mathcal{I}$.

– For every $(v, r, v') \in E_{\mathcal{G}_1}$ there is $(w, r, w') \in E_{\mathcal{G}_2}$ such that $(v', w') \in \mathcal{H}$ ◊

Figure 2.1 illustrates the $\mathcal{EL}$-description tree for the $\mathcal{EL}$-concept description $C = P_1 \sqcap \exists r.(\exists s.(P_1 \sqcap P_2) \sqcap \exists r.\top)$, an interpretation $\mathcal{I}$ and a homomorphism from t_C to $\mathcal{I}$.

Lemma 1 *Let C be an $\mathcal{EL}$-concept description with the corresponding concept description tree t_C with root x_C, and let $\mathcal{I}$ be an interpretation with $x \in \Delta^\mathcal{I}$ and the corresponding $\mathcal{EL}$-description graph $\mathcal{G} = (V, E, \ell)$. Then, the following are equivalent:*

1. *$x \in C^\mathcal{I}$*
2. *There exists a homomorphism from x_C to x.*

Proof. The proof is by induction on $d = rdepth(C)$ in both directions.

$(1 \Rightarrow 2)$ For the induction base case, let $d = 0$ and $x \in (P_1 \sqcap \ldots \sqcap P_k)^\mathcal{I}$. Define $\mathcal{H} := \{(x_C, x)\}$, which obviously is a homomorphism. Now for the induction step, let $x \in (P_1 \sqcap \ldots \sqcap P_k \sqcap \exists r_1.C_1 \sqcap \ldots \sqcap \exists r_m.C_m)^\mathcal{I}$. There are $(x, x_i) \in r_i^\mathcal{I}$ such that $x_i \in C_i^\mathcal{I}$, $(x_C, r_i, x_{C_i}) \in E_{t_C}$ and by the induction hypothesis there exist homomorphisms $\mathcal{H}_i$ from x_{C_i} to x_i for $1 \leq i \leq m$. Hence, $\mathcal{H} := \bigcup_{1 \leq i \leq m} \mathcal{H}_i \cup \{(x_C, x)\}$ is a homomorphism from x_C to x.

$(2 \Rightarrow 1)$ For the induction base case, let $d = 0$ and $C = P_1 \sqcap \ldots \sqcap P_k$. Since $\ell_{t_C}(x_C) \subseteq \ell_\mathcal{I}(x)$, $x \in C^\mathcal{I}$. For the induction step, let $C = P_1 \sqcap \ldots \sqcap P_k \sqcap \exists r_1.C_1 \sqcap \ldots \sqcap \exists r_m.C_m$ and $\mathcal{H}$ be a homomorphism from x_C to x. There are $(x_C, r_i, x_{C_i}) \in E_{t_C}$ and by the homomorphism conditions, there are also $(x, x_i) \in r_i^\mathcal{I}$ for $1 \leq i \leq m$. Now $\mathcal{H}$ is a homomorphism from each x_{C_i} to x_i, and hence by the induction hypothesis $x_i \in C_i^\mathcal{I}$. Consequently, $x \in C^\mathcal{I}$.

The previous lemma allows us to establish the connection between the existence of a homomorphism between $\mathcal{EL}$-description trees and subsumption.

Algorithm 1 $\mathcal{EL}$-subsumption algorithm

Require: $\mathcal{EL}$-concept descriptions C, D
  $\mathcal{H} := \emptyset$
  for $i = 0$ to $depth(x_D)$ **do**
    for all $x \in V_{t_D}$ with $depth(x) = i$ and $y \in V_{t_C}$ **do**
      if $\ell_{t_D}(x) \subseteq \ell_{t_C}(y) \wedge \forall (x, r, x') \in E_{t_D}.\exists (y, r, y') \in E_{t_C}.(x', y') \in \mathcal{H}$ **then**
        $\mathcal{H} := \mathcal{H} \cup \{(x, y)\}$
      end if
    end for
  end for
  if $(x_D, x_C) \in \mathcal{H}$ **then**
    return $C \sqsubseteq D$
  else
    return $C \not\sqsubseteq D$
  end if

Lemma 2 *Let C, D be $\mathcal{EL}$-concept descriptions with their corresponding $\mathcal{EL}$-description trees t_C, t_D with roots x_C and x_D. Then, the following are equivalent:*

1. *$C \sqsubseteq D$*
2. *There exists a homomorphism from x_D to x_C.*

Proof. ($1 \Rightarrow 2$) We show the contrapositive. Assume there does not exist a homomorphism from x_D to x_C. Now the identity on the vertices of t_C is a homomorphism from x_C to x_C and hence $x_C \in C^{t_C}$. Since there does not exist a homomorphism from x_D to x_C, by the previous lemma $x_C \notin D^{t_C}$.

($2 \Rightarrow 1$) Let $\mathcal{H}$ be a homomorphism from x_D to x_C, and let $\mathcal{I}$ be an interpretation and $y \in C^{\mathcal{I}}$. By the previous lemma, there exists a homomorphism $\mathcal{H}'$ from x_C to y. Clearly, the composition $\mathcal{H}' \circ \mathcal{H}$ yields a homomorphism from x_D to y. Hence $y \in D^{\mathcal{I}}$. □

Taking the previous lemma together with Lemma 1, we can derive Algorithm 1 that decides subsumption between $\mathcal{EL}$-concept descriptions C, D in polynomial time in $|C| + |D|$. Basically, the algorithm labels *bottom-up* the $\mathcal{EL}$-description tree t_C of C with subsets of vertices of t_D and returns subsumption of C and D if at the end the root x_C of t_C is labeled by x_D.

Theorem 1 *Subsumption in $\mathcal{EL}$ can be decided in polynomial time.*

2.4 Terminologies

Most description logics do not only offer a concept language, but also additionally provide some terminology (TBox) formalism. In the following, we introduce standard TBoxes and general TBoxes. Although we make the definitions in the context of $\mathcal{EL}$, they can be adapted in an obvious way to arbitrary concept languages.

2.4.1 Standard Terminology Boxes

Standard terminology boxes introduce abbreviations for complex concepts and store terminological knowledge about the application domain. In the following, we call standard terminology boxes just TBoxes.

Definition 8 ($\mathcal{EL}$-TBox) Let C be an $\mathcal{EL}$-concept description and $A \in N_C$. Then, $A \equiv C$ is called a *concept definition*. A finite set $\mathcal{T}$ of concept definitions that does not contain multiple concept definitions is called an *$\mathcal{EL}$-TBox*. $\Diamond$

Formally, a set $\mathcal{T}$ contains multiple concept definitions if there are $\{A \equiv C, A \equiv D\} \subseteq \mathcal{T}$ such that $C \neq D$. If the underlying concept language is clear from the context, we will drop the language prefix and just talk about TBoxes. We call the concept names that occur on the left-hand side of a TBox *defined concepts*, and for a given TBox $\mathcal{T}$ their set is denoted by $N_{def}(\mathcal{T})$. All other concept names occurring in $\mathcal{T}$ are called *primitive concepts*, whose set is denoted by $N_{prim}(\mathcal{T})$. When we consider $\mathcal{EL}$ extended by primitive negation, we additionally do not allow negation to be used in front of defined concept names. Note, that the TBox definition allows for cyclic definitions, i.e., there might be concept definitions $\{A_1 \equiv D_1, \ldots, A_n \equiv D_n\} \subseteq \mathcal{T}$ such that

- D_i contains A_{i+1} for $1 \leq i < n$
- D_n contains A_1

The size $|\mathcal{T}|$ of a TBox $\mathcal{T}$ is defined as

$$|\mathcal{T}| := \sum_{A \equiv C \in \mathcal{T}} |A| + |C|$$

Let us now have a look at an example TBox.

Example 5. Consider the following acyclic TBox $\mathcal{T}_{\text{family}}$:

$$\begin{array}{rcl}
\mathsf{Parent} & \equiv & \exists \mathsf{has_child}.\top \\
\mathsf{Mother} & \equiv & \mathsf{Female} \sqcap \mathsf{Parent} \\
\mathsf{Father} & \equiv & \mathsf{Male} \sqcap \mathsf{Parent} \\
\mathsf{Grandmother} & \equiv & \mathsf{Mother} \sqcap \exists \mathsf{has_child}.\mathsf{Parent} \\
\mathsf{Grandfather} & \equiv & \mathsf{Father} \sqcap \exists \mathsf{has_child}.\mathsf{Parent}
\end{array}$$

It defines parts of the relations in families. ■

Algorithm 2 Unfolding algorithm

Require: Acyclic TBox $\mathcal{T}$ and $\mathcal{EL}$-concept description C
  while C contains some $A \in N_{def}(\mathcal{T})$ **do**
    Let $A \equiv D \in N_{def}(\mathcal{T})$
    Replace any occurrence of A in C by D
  end while
  return C

When we consider the semantics of TBoxes, we cannot allow arbitrary interpretations. The admissible interpretations have to fulfill the concept definitions of the TBox.

Definition 9 (Descriptive semantics) An interpretation $\mathcal{I}$ is a *model* of a TBox $\mathcal{T}$ iff $A^{\mathcal{I}} = C^{\mathcal{I}}$ for all concept definitions $A \equiv C \in \mathcal{T}$. ◊

For cyclic TBoxes, apart from descriptive semantics, least and greatest fixed point semantics have been introduced by Nebel (Nebel 1991). Informally speaking, they offer different ways of how to interpret cyclic concept definitions. In this thesis, we will only deal with descriptive semantics.

For the remainder of this thesis, subsumption w.r.t. TBoxes will be our main point of interest.

Definition 10 (Subsumption and satisfiability w.r.t. TBoxes) Let $\mathcal{T}$ be a TBox and C, D be $\mathcal{EL}$-concept descriptions. Then, C *is subsumed by* D *w.r.t.* $\mathcal{T}$ ($C \sqsubseteq_{\mathcal{T}} D$) iff $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ for all models $\mathcal{I}$ of $\mathcal{T}$. Moreover, C *is satisfiable w.r.t.* $\mathcal{T}$ iff there exists a model $\mathcal{I}$ of $\mathcal{T}$ such that there is $x \in C^{\mathcal{I}}$. ◊

Note, that it suffices to only consider subsumption of concept names w.r.t. TBoxes. Subsumption between arbitrary concept descriptions C, D w.r.t. a TBox $\mathcal{T}$ can be reduced to subsumption of concept names w.r.t. a TBox: $C \sqsubseteq_{\mathcal{T}} D$ iff $A \sqsubseteq_{\mathcal{T}'} B$, where $\mathcal{T}' = \mathcal{T} \dot\cup \{A \equiv C, B \equiv D\}$.

Concept definitions in acyclic TBoxes may be viewed as macro definitions and can be expanded in a natural way like macros. This allows us to rephrase subsumption w.r.t. an acyclic TBox as subsumption without a TBox. Formally, for a concept description C and an acyclic TBox $\mathcal{T}$, C is *unfolded w.r.t.* $\mathcal{T}$ iff no $A \in N_{def}(\mathcal{T})$ occurs in C. Using Algorithm 2, every concept C can be transformed to an unfolded concept $\hat{C}$ w.r.t. $\mathcal{T}$ such that $C \equiv_{\mathcal{T}} \hat{C}$. Termination of the algorithm is an immediate consequence of $\mathcal{T}$ being acyclic. For concept descriptions C, D and their unfoldings $\hat{C}, \hat{D}$ w.r.t. $\mathcal{T}$, it is easily seen that we have $\hat{C} \sqsubseteq \hat{D}$ iff $\hat{C} \sqsubseteq_{\mathcal{T}} \hat{D}$ iff $C \sqsubseteq_{\mathcal{T}} D$. Unfolding might lead to an exponential blowup of $|\hat{C}|$ in $|C| + |\mathcal{T}|$. Thus, unfolding is not suitable for providing complexity bounds, but can provide a helpful tool in proofs.

For an acyclic TBox $\mathcal{T}$ and $A \equiv C \in \mathcal{T}$, we define the *role depth of* A *w.r.t.* $\mathcal{T}$, $\mathit{rdepth}_{\mathcal{T}}(A) := \mathit{rdepth}(\hat{C})$. However, due to the potentially exponential blow-up of $|\hat{C}|$,

(n1)	replace $A \equiv \exists r.C' \in \mathcal{T}$	by	$\{A \equiv \exists r.B, B \equiv C\}$
(n2)	replace $A \equiv C \sqcap C' \in \mathcal{T}$	by	$\{A \equiv C \sqcap B, B \equiv C'\}$

where $C' \notin N_{def}(\mathcal{T})$ and (n2) is applied modulo commutativity

Table 2.4: Normalization rules

we cannot calculate the role depth w.r.t. $\mathcal{T}$ in polynomial time by naively unfolding C. Nevertheless, calculating first the role depths of the concept descriptions containing no defined concept name, and then bottom-up calculating the role depths for concept definitions that contain defined concept names for which the role depth has already been calculated gives us a way to calculate role depth w.r.t. acyclic TBoxes in polynomial time w.r.t. the size of the input TBox.

2.4.2 Normalized $\mathcal{EL}$-TBoxes

In the following we introduce two normal forms of $\mathcal{EL}$-TBoxes. They make the handling of TBoxes easier in algorithms and proofs.

Definition 11 (Normal form) An $\mathcal{EL}$-TBox $\mathcal{T}$ is in *normal form* iff for every $A \equiv C \in \mathcal{T}$, C is of the form $\top$ or B; $\exists r.B_1$; or $B_1 \sqcap B_2$, where $B \in N_C$, and $B_1, B_2 \in N_{def}(\mathcal{T})$.$\Diamond$

It is easily seen that by exhaustively applying the rules from Table 2.4, subsumption in $\mathcal{EL}$ w.r.t. TBoxes can be reduced to subsumption in $\mathcal{EL}$ w.r.t. a TBox in normal form in polynomial time. In Table 2.4, B is a fresh, previously unused concept name.

Proposition 2 *Subsumption w.r.t. an $\mathcal{EL}$-TBox can be reduced in polynomial time to subsumption w.r.t. an $\mathcal{EL}$-TBox in normal form.*

In order to represent $\mathcal{EL}$-TBoxes as graphs, Baader introduced in (Baader 2003) a further TBox normal form.

Definition 12 (Extended normal form) An $\mathcal{EL}$-TBox $\mathcal{T}$ is in *extended normal form* iff for each concept definition $A \equiv C \in \mathcal{T}$, C is of the form

$$P_1 \sqcap \ldots \sqcap P_k \sqcap \exists r_1.B_1 \sqcap \ldots \sqcap \exists r_\ell.B_\ell$$

for $k, \ell \geq 0$, $P_1, \ldots, P_k \in N_{prim}(\mathcal{T})$, $r_1, \ldots, r_\ell \in N_R$ and $B_1, \ldots, B_\ell \in N_{def}(\mathcal{T})$. $\Diamond$

Since we rely on this normal form for extensions of $\mathcal{EL}$ in Chapter 3, we explicitly show the normalization process. Subsequently, we follow Baader (Baader 2003) and firstly illustrate it with the help of an example. Let $\mathcal{T}$ be defined as follows:

$$\begin{aligned} A_1 &\equiv P_1 \sqcap \exists r.(P_2 \sqcap A_2) \\ A_2 &\equiv P_3 \sqcap A_3 \\ A_3 &\equiv A_2 \sqcap \exists r.P_1 \end{aligned}$$

It is not hard to see how we can enforce only defined concepts to occur in the scope of existential restrictions by introducing auxiliary concept definitions B_1 and B_2:

$$\begin{array}{rcl} A_1 & \equiv & P_1 \sqcap \exists r.B_1 \\ B_1 & \equiv & P_2 \sqcap A_2 \\ A_2 & \equiv & P_3 \sqcap A_3 \\ A_3 & \equiv & A_2 \sqcap \exists r.B_2 \\ B_2 & \equiv & P_1 \end{array}$$

However, $\mathcal{T}'$ is not yet in normal form. The definitions of B_1, A_2 and A_3 still contain defined concept names in their top-level.

Let us first have a look at the definitions of A_2 and A_3. We note that A_2 and A_3 mutually include each other on their top-level. Thus, both A_2 and A_3 are interpreted by the same set in all models of $\mathcal{T}$. Moreover, A_2 (and A_3) is subsumed by $P_3 \sqcap \exists r.B_2$. However, we cannot express such inclusions in $\mathcal{T}$, but employ Nebel's approach (Nebel 1990a) to turn this inclusion constraint into a concept definition by introducing a fresh primitive concept name $\bar{A}_2$ and defining $A_2 \equiv \bar{A}_2 \sqcap P_3 \sqcap \exists r.B_2$. In order to get rid of A_2 in the top-level of the definition of B_1, we just replace it by the new definition of A_2. So we end up with the following TBox, which is normalized:

$$\begin{array}{rcl} A_1 & \equiv & P_1 \sqcap \exists r.B_1 \\ B_1 & \equiv & P_2 \sqcap \bar{A}_2 \sqcap P_3 \sqcap \exists r.B_2 \\ A_2 & \equiv & \bar{A}_2 \sqcap P_3 \sqcap \exists r.B_2 \\ A_3 & \equiv & \bar{A}_2 \sqcap P_3 \sqcap \exists r.B_2 \\ B_2 & \equiv & P_1 \end{array}$$

A generalization of this approach for arbitrary TBoxes $\mathcal{T}$ is not too far away. First—as shown in the example—exhaustively introduce auxiliary concept definitions for complex concepts in existential restrictions, and replace the complex concepts by their newly introduced defined concept names. Now in order to remove defined concept names from the top-level of concept definitions of $\mathcal{T}$, we view the TBox as a directed graph $\mathcal{G} = (V, E)$, where $V := N_{def}(\mathcal{T})$ and $(A, B) \in E$ iff B occurs on the top-level of the definition of A. Let E^* be the reflexive transitive closure of E, which can be computed in polynomial time. We define the following equivalence relation on $N_{def}(\mathcal{T})$:

$$A \cong B \quad \text{iff} \quad (A, B) \in E^* \text{ and } (B, A) \in E^*$$

Now all equivalence classes $[C] := \{C' \mid C \cong C'\}$ are interpreted in every model by the same set. So we start with some equivalence class $[C]$, treat all concepts of C like A_2 and

A_3 in the example above, and replace the occurrence of any $C' \in [C]$ on the top-level of any concept definition by the newly introduced concept definition. We then continue with the next equivalent class until we have proceeded all equivalence classes. The fact that we only replace defined concept names on the top-level of concept definitions prevents an exponential blow-up of the size of $\mathcal{T}$, due to the idempotency of $\sqcap$. Summing up, this algorithm sketches the proof for the following proposition:

Proposition 3 *Subsumption w.r.t. an $\mathcal{EL}$-TBox can be reduced in polynomial time to subsumption w.r.t. a TBox in extended normal form.*

2.4.3 General Terminology Boxes

Besides our standard TBoxes, there is another popular terminology box formalism that we briefly introduce, *general terminology boxes (general TBoxes).*

Definition 13 (General $\mathcal{EL}$-TBox) Let C, D be $\mathcal{EL}$-concept descriptions. Then, $C \sqsubseteq D$ is called a *general concept inclusion (GCI).* A *general $\mathcal{EL}$-TBox $\mathcal{T}$* is a finite set of general concept inclusions. An interpretation $\mathcal{I}$ is a *model of $\mathcal{T}$* iff $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ for all $C \sqsubseteq D \in \mathcal{T}$. We say A *is subsumed by B w.r.t. $\mathcal{T}$* ($A \sqsubseteq_{\mathcal{T}} B$) for concept names A, B iff $A^{\mathcal{I}} \subseteq B^{\mathcal{I}}$ for all models $\mathcal{I}$ of $\mathcal{T}$. Moreover, A is *satisfiable w.r.t. $\mathcal{T}$* iff there exists a model $\mathcal{I}$ of $\mathcal{T}$ such that there is an $x \in A^{\mathcal{I}}$. ◊

As the name suggests, general TBoxes generalize TBoxes. Every TBox $\mathcal{T}$ can be transformed into an equivalent general TBox $\mathcal{T}'$: For every concept definition $A \equiv C \in \mathcal{T}$ add $A \sqsubseteq C$ and $C \sqsubseteq A$ to $\mathcal{T}'$. It is easily seen that $A \sqsubseteq_{\mathcal{T}} B$ if and only if $A \sqsubseteq_{\mathcal{T}'} B$.

General TBoxes are very expressive. Moreover, subsumption in $\mathcal{EL}$ w.r.t. general TBoxes is polynomial w.r.t. the size of the input general TBox (Brandt 2004). In the remainder of this subsection, we show that it is even PTime-complete. The reduction is done by reducing satisfiability of Horn formulas to non-subsumption in $\mathcal{EL}$ w.r.t. general TBoxes. Although the correspondence is quite obvious, to the best of the author's knowledge it has not been considered yet.

Definition 14 (Horn-satisfiability) A *Horn clause* is a clause that contains at most one positive literal and any finite number of negative literals. A *Horn formula* is a conjunction of Horn clauses. The *Horn-satisfiability problem* (Horn-SAT) is to determine for a given Horn formula whether there is a valuation of the atomic variables such that the Horn formula evaluates to true. ◊

Horn-SAT is a PTime-complete problem (Greenlaw, Hoover & Ruzzo 1992). It is widely believed that PTime-complete problems cannot be effectively parallized, thus proving PTime-completeness of subsumption in $\mathcal{EL}$ w.r.t. general TBoxes has also practical implications for implementations of subsumption algorithms.

Let H be an instance of Horn-SAT over the atomic variables $a_1, \ldots, a_g$ containing k clauses, each containing $m_i \geq 0$ negative literals $\neg n_1^i, \ldots, \neg n_{m_i}^i$ and at most one positive literal $p^i, 1 \leq i \leq k$. We translate H into a general TBox $\mathcal{T}$, and thus every clause into a GCI. In case the i-th clause contains one positive and at least one negative literal, we add the following GCI to $\mathcal{T}$, where $P^i, N_j^i \in \{A_1, \ldots, A_n\} \subseteq N_C, 1 \leq j \leq m_i$:

$$N_1^i \sqcap \ldots \sqcap N_{m_i}^i \sqsubseteq P^i$$

In case $m_i = 0$ we set the left-hand side of the GCI to $\top$, and if there is no positive literal present we set the right-hand side to some arbitrary, but fixed $P \in N_C$ not equal to any introduced concept name for a literal. Clearly, $\mathcal{T}$ is linear in the size of H.

Lemma 3 *Let H be an instance of Horn-SAT and $\mathcal{T}$ the general $\mathcal{EL}$-TBox corresponding to H. Then, the following are equivalent:*

1. *H is satisfiable*
2. *$\top \not\sqsubseteq_{\mathcal{T}} P$*

Proof. ($1 \Rightarrow 2$) Let $a_1, \ldots, a_g$ be the atomic variables used in H and $V : \{a_1, \ldots, a_g\} \rightarrow \{0, 1\}$ the valuation such that H is true under V. We define an interpretation $\mathcal{I}$ as follows: $\Delta^{\mathcal{I}} := \{x\}, P^{\mathcal{I}} := \emptyset, A_i^{\mathcal{I}} := \{x \mid V(a_i) = 1\}, 1 \leq i \leq g$. Clearly, $\top^{\mathcal{I}} = \{x\} \not\subseteq \emptyset = P^{\mathcal{I}}$, so it remains to show that $\mathcal{I}$ is a model of $\mathcal{T}$. Let $p^i \vee \neg n_1^i \vee \ldots \vee \neg n_{m_i}^i$ be the i-th clause in H. Since H is satisfiable we have $V(p^i)$ is true or $V(n_j^i)$ is false for some $1 \leq j \leq m_i$. Thus, we have $(N_1^i \sqcap \ldots \sqcap N_m^i)^{\mathcal{I}} \subseteq (P^i)^{\mathcal{I}}$. The argumentation holds similarly for clauses that do not contain a positive or negative literals.

($2 \Rightarrow 1$) Let $\mathcal{I}$ be a model of $\mathcal{T}$ such that $x \notin P^{\mathcal{I}}$. Let $a_1, \ldots, a_g$ be the atomic variables used in H. For $1 \leq i \leq g$ we define

$$V(a_i) := \begin{cases} 1 & \text{if } x \in A_i^{\mathcal{I}} \\ 0 & \text{if } x \notin A_i^{\mathcal{I}} \end{cases}$$

Then V is a solution to H: Let $p^i \vee \neg n_1^i \vee \ldots \vee \neg n_m^i$ be the i-th clause in H. Since $\mathcal{I}$ is a model we have $x \in (P^i)^{\mathcal{I}}$ or $x \notin (N_1^i \sqcap \ldots \sqcap N_{m_i}^i)^{\mathcal{I}}$. The latter is the case if for some $N_j^i, 1 \leq j \leq m_i, x \notin (N_j^i)^{\mathcal{I}}$. Thus V makes the clause true. It is not hard to see that V also makes clauses true that do not contain a positive literal or negative literals. □

Since PTime is a deterministic class, we have proved the following theorem.

Theorem 2 *Subsumption in $\mathcal{EL}$ w.r.t. general TBoxes is* PTime*-complete.*

Chapter 3

Tractable Extensions of $\mathcal{EL}$

As stated in the introduction, one of the main goals of this thesis is to find extensions of $\mathcal{EL}$ that provide more expressiveness and for which the subsumption problem w.r.t. cyclic TBoxes remains tractable. In this chapter, we consider two such extensions. In the first section, we extend $\mathcal{EL}$ by primitive negation, con- and disjunction of role names in disjunctive normal form (DNF) and p-admissible concrete domains. We show that subsumption is tractable w.r.t. cyclic TBoxes. The second section considers $\mathcal{EL}$ extended by at-least restrictions and con- and disjuncion of role names in DNF. Again, subsumption in this extension is tractable. A combination of the two extensions is considered in the next chapter, and it is shown there that subsumption in this combination is intractable.

3.1 $\mathcal{EL}^{\sqcup,\sqcap,\neg}(\mathcal{D})$

Let $\mathcal{EL}^{\sqcup,\sqcap,\neg}(\mathcal{D})$ be $\mathcal{EL}$ extended by primitive negation, role con- and disjunction and p-admissible concrete domains. We require the complex roles to be in disjunction normal form.

Definition 15 Let R be a complex role over the role constructors con- and disjunction. Then, R is in *disjunctive normal form (DNF)* iff R is of the form

$$(r_1^1 \sqcap \ldots \sqcap r_{k_1}^1) \sqcup \ldots \sqcup (r_1^n \sqcap \ldots \sqcap r_{k_n}^n)$$

$\Diamond$

It is *not* the case that any arbitrary complex role R can be transformed into an equivalent complex role in DNF that is polynomial in the size of R, e.g., for $R = (r_1^1 \sqcup \ldots \sqcup r_{k_1}^1) \sqcap \ldots \sqcap (r_1^n \sqcup \ldots \sqcup r_{k_n}^n)$. We say R *implies* S iff $R^\mathcal{I} \subseteq S^\mathcal{I}$ for all interpretations $\mathcal{I}$. It is easily seen that $\exists R.C \sqsubseteq \exists S.D$ iff R implies S and $C \sqsubseteq D$. Moreover, if R and S are in disjunctive normal form, checking for implication is polynomial in $|R| + |S|$, similar to checking implication of propositional formulas in disjunctive normal form.

Let us now briefly discuss the features of $\mathcal{EL}^{\sqcup,\sqcap,\neg}(\mathcal{D})$. Primitive negation allows for making concepts disjoint, e.g., we can define:

$$\begin{aligned}\mathsf{Parent} &\equiv \mathsf{Human} \sqcap \exists \mathsf{has\_child}.\top \\ \mathsf{Mother} &\equiv \mathsf{Parent} \sqcap \mathsf{Female} \sqcap \neg\mathsf{Male} \\ \mathsf{Father} &\equiv \mathsf{Parent} \sqcap \mathsf{Male} \sqcap \neg\mathsf{Female}\end{aligned}$$

Role disjunction is another advantage. It allows for expressing some limited sort of "real" disjunction:

$$\exists r.C \sqcup \exists s.C \quad \text{iff} \quad \exists(r \sqcup s).C$$

Last but not least, in particular the property that concrete domains do *not* have to be convex makes $\mathcal{EL}^{\sqcup,\sqcap,\neg}(\mathcal{D})$ really expressive. For example, we can express

$$\mathsf{Teenager} \equiv \mathsf{Human} \sqcap \geq_{13}(\mathsf{has\_age}) \sqcap \leq_{19}(\mathsf{has\_age})$$

Non-convex concrete domains have a high potential to find application in areas where it is important to reason about concrete qualities, e.g., in medical applications.

In the following, we present an algorithm that decides subsumption in $\mathcal{EL}^{\sqcup,\sqcap,\neg}(\mathcal{D})$ w.r.t. cyclic TBoxes in polynomial time w.r.t. the size of the input TBox. We define an extended normal form of $\mathcal{EL}^{\sqcup,\sqcap,\neg}(\mathcal{D})$-TBoxes that is required by the subsumption algorithm.

Definition 16 An $\mathcal{EL}^{\sqcup,\sqcap,\neg}(\mathcal{D})$-TBox $\mathcal{T}$ is in *extended normal form* iff for each concept definition $A \equiv C \in \mathcal{T}$, C is of the form

$$\bigsqcap_{1\leq i\leq j} P_i \sqcap \bigsqcap_{1\leq i\leq k} \neg N_i \sqcap \bigsqcap_{1\leq i\leq \ell} \exists R_i.B_i \sqcap \bigsqcap_{1\leq i\leq m} p_i(f_1^i, \ldots, f_{n_i}^i)$$

for $j, k, \ell, m \geq 0$, $P_1, \ldots, P_j, N_1, \ldots, N_k \in N_{prim}(\mathcal{T})$; $R_1, \ldots, R_\ell$ being complex roles in DNF; $B_1, \ldots, B_\ell \in N_{def}(\mathcal{T})$; $p_1, \ldots, p_m \in \Phi^{\mathcal{D}}$, $n_i \geq 0$ and $f_1^i, \ldots, f_{n_i}^i \in N_{cF}, 1 \leq i \leq m.\Diamond$

Firstly, let us first introduce some abbreviations. Let $\mathcal{T}$ be a TBox in extended normal form and $A \equiv C \in \mathcal{T}$. Then,

- $P_{\mathcal{T}}(A) := \{P_i \mid 1 \leq i \leq j\}$
- $\bar{P}_{\mathcal{T}}(A) := \{N_i \mid 1 \leq i \leq k\}$
- $E_{\mathcal{T}}(A) := \{\exists R_i.B_i \mid 1 \leq i \leq \ell\}$
- $F_{\mathcal{T}}(A) := \{p_i(f_1^i, \ldots, f_{n_i}^i) \mid 1 \leq i \leq m\}$
- $\bigwedge_A^{\mathcal{D}}$ is the conjunction of $p(f_1, \ldots, f_m) \in F_{\mathcal{T}}(A)$ in $\mathcal{D}$

It is not hard to see that we can modify the algorithm from Section 2.4.2 that brings $\mathcal{EL}$-TBoxes into extended normal form to an algorithm that brings $\mathcal{EL}^{\sqcup,\sqcap,\neg}(\mathcal{D})$-TBoxes into extended normal form. Obviously, negated primitive concept names and the concrete domain concept constructors can be treated in the same way as primitive concept names. Complex roles can also be treated like simple role names by the algorithm.

Proposition 4 *Subsumption w.r.t. $\mathcal{EL}^{\sqcup,\sqcap,\neg}(\mathcal{D})$-TBoxes can be reduced in polynomial time to subsumption w.r.t. $\mathcal{EL}^{\sqcup,\sqcap,\neg}(\mathcal{D})$-TBoxes in extended normal form.*

The presence of primitive negation and concrete domains may lead to unsatisfiable concept definitions. A defined concept name $A \in N_{def}(\mathcal{T})$ in extended normal form can be unsatisfiable w.r.t. $\mathcal{T}$ for three reasons:

1. There is $P \in P_{\mathcal{T}}(A)$ such that $P \in \bar{P}_{\mathcal{T}}(A)$.
2. $\bigwedge_A^{\mathcal{D}}$ is unsatisfiable
3. There is $\exists R.B \in E_{\mathcal{T}}(A)$ and B is unsatisfiable w.r.t. $\mathcal{T}$

We can view checking satisfiability of the concept definitions as part of the normalization process: Firstly, we apply the usual normalization algorithm. Then, we check for every concept definition if one of the first two reasons for unsatisfiability applies. If so for $A \equiv C$, we replace this concept definition by $A \equiv \bot$. Afterwards, we can exhaustively check every $A \equiv C \in \mathcal{T}$ if C contains $\exists R.B$ and $B \equiv \bot \in \mathcal{T}$. Again, if so, $A \equiv C$ is replaced by $A \equiv \bot$. At the end, the right-hand side of every concept definition of an unsatisfiable concept name w.r.t. $\mathcal{T}$ is replaced by $\bot$. It is easily seen that this extended normalization can be done in polynomial time and does not blow up the TBox.

Lemma 4 *Let $\mathcal{T}$ be a TBox in extended normal form and $A, B \in N_{def}(\mathcal{T})$. Then, if A is unsatisfiable w.r.t. $\mathcal{T}$, checking $A \sqsubseteq_{\mathcal{T}} B$ or $B \sqsubseteq_{\mathcal{T}} A$ can be done in polynomial time.*

Proof. As seen above, checking satisfiability of defined concepts can be done in polynomial time. Clearly, we have $A \sqsubseteq_{\mathcal{T}} B$ for all $B \in N_{def}(\mathcal{T})$. If also B is unsatisfiable w.r.t. $\mathcal{T}$, we have $B \sqsubseteq_{\mathcal{T}} A$. Otherwise, $B \not\sqsubseteq_{\mathcal{T}} A$. □

Since checking subsumption for unsatisfiable concept definitions is rather trivial, we assume for our subsumption algorithm that every concept definition in the input TBox is satisfiable.

The $\mathcal{EL}^{\sqcup,\sqcap,\neg}(\mathcal{D})$-subsumption algorithm, Algorithm 3, takes a TBox $\mathcal{T}$ in extended normal form as input, that does not contain unsatisfiable concept definitions. It computes a subsumption relation $S \subseteq N_{def}(\mathcal{T}) \times N_{def}(\mathcal{T})$. Starting from the identity on the concept names, it exhaustively checks for every tuple of defined concept names of $\mathcal{T}$ if all of the completion conditions from Table 3.1 apply, and if so, these tuples are added to S. When the algorithm finishes, we have $(A, B) \in S$ iff $A \sqsubseteq_{\mathcal{T}} B$.

Let us first have a look at the complexity of the algorithm.

(s1)	$P_\mathcal{T}(B) \subseteq P_\mathcal{T}(A)$
(s2)	$\bar{P}_\mathcal{T}(B) \subseteq \bar{P}_\mathcal{T}(A)$
(s3)	For each $(\exists R_B.B') \in E_\mathcal{T}(B)$ there is $(\exists R_A.A') \in E_\mathcal{T}(A)$ such that R_A implies R_B and $(A', B') \in S$
(s4)	$\bigwedge_A^\mathcal{D}$ implies $\bigwedge_B^\mathcal{D}$

Table 3.1: $\mathcal{EL}^{\sqcup,\sqcap,\neg}(\mathcal{D})$ completion conditions

Algorithm 3 $\mathcal{EL}^{\sqcup,\sqcap,\neg}(\mathcal{D})$-subsumption algorithm

Input: $\mathcal{EL}^{\sqcup,\sqcap,\neg}(\mathcal{D})$-TBox $\mathcal{T}$ in extended normal form not containing any unsatisfiable defined concept name
  $S := \{(A, A) \mid A \in N_{def}(\mathcal{T})\}$
  while there are $(A, B) \notin S$ and completion condition (s1)-(s4) apply for A and B **do**
    $S := S \cup \{(A, B)\}$
  end while
  return S

Lemma 5 *Algorithm 3 runs in polynomial time.*

Proof. The algorithm produces a sequence of relations $S_0, \ldots, S_n$. In every successor relation, one tuple is added, and thus $n \leq |N_{def}(\mathcal{T})|^2$. In every step, the completion conditions from Table 3.1 have to be checked for at most $|N_{def}(\mathcal{T})|^2$ tuples. Checking the completion conditions is polynomial, since implication between complex roles in DNF and implication in $\mathcal{D}$ can be checked in polynomial time. □

Concerning the correctness, it is obvious that the algorithm terminates, and it is not hard to see that it is sound. Proving completeness is the harder part. When the algorithm has terminated for some TBox $\mathcal{T}$ and we have some $(A_0, B_0) \notin S$, some completion condition from Table 3.1 does not hold for (A_0, B_0). Depending on which it is, we construct an interpretation by guided unwinding of $\mathcal{T}$. Then, at the root of the interpretation A_0 does hold whereas B_0 does not. However, since role disjunction and the concrete domains are not convex, this interpretation does not directly lead to a model of $\mathcal{T}$.

Consider the following example TBox $\mathcal{T}$:

$$\begin{array}{rcl} A_1 & \equiv & \exists(r \sqcup s).B \\ A_2 & \equiv & \exists r.B \\ A_3 & \equiv & \exists s.B \\ B & \equiv & \top \end{array}$$

Obviously, $A_1 \not\sqsubseteq A_2$ and $A_1 \not\sqsubseteq A_3$, but $A_1 \sqsubseteq A_2 \sqcup A_3$. So, if we want to construct a counter-model $\mathcal{I}$ of $\mathcal{T}$ for $A_1 \not\sqsubseteq A_2$ such that $x \in A_1^{\mathcal{I}} \setminus A_2^{\mathcal{I}}$, we have $x \in A_3^{\mathcal{I}}$. However, for S being the subsumption relation produced by the algorithm, we would not and must not have $(A_1, A_3) \in S$. For that reason, S cannot completely guide the construction of a counter-model. However, it can lead to an interpretation that can easily be extended to become a counter-model. In the completeness proof, the function $\mathcal{O}_{\mathcal{T}}$ is used therefor.

Definition 17 Let *Int* be the set of all interpretations. A TBox $\mathcal{T}$ induces a function $\mathcal{O}_{\mathcal{T}}$ on *Int* with $\mathcal{O}_{\mathcal{T}}(\mathcal{I}) = \mathcal{J}$ iff

- $\Delta^{\mathcal{J}} = \Delta^{\mathcal{I}}$
- $P^{\mathcal{J}} = P^{\mathcal{I}}$ for all $P \in N_{prim}(\mathcal{T})$
- $r^{\mathcal{J}} = r^{\mathcal{I}}$ for all role names r occurring in $\mathcal{T}$
- $f^{\mathcal{J}} = f^{\mathcal{I}}$ for all concrete feature names occurring in $\mathcal{T}$
- $A^{\mathcal{J}} = C^{\mathcal{I}}$ for all $A \equiv C \in \mathcal{T}$ ◊

Informally speaking, the exhaustive application of $\mathcal{O}_{\mathcal{T}}$ allows for making a model of $\mathcal{T}$ of a given interpretation, due to the monotonicity of $\mathcal{O}_{\mathcal{T}}$.

Lemma 6 *Let $\mathcal{T}$ be an $\mathcal{EL}^{\sqcup,\sqcap,\neg}(\mathcal{D})$-TBox, $\mathcal{I}$ an interpretation such that $A^{\mathcal{I}} \subseteq C^{\mathcal{I}}$ for all $A \equiv C \in \mathcal{T}$ and $\mathcal{J} = \mathcal{O}_{\mathcal{T}}(\mathcal{I})$. Then, $A^{\mathcal{J}} \subseteq C^{\mathcal{J}}$ for all $A \equiv C \in \mathcal{T}$.*

Proof. Instead of proving the lemma directly, we show by structural induction $D^{\mathcal{I}} \subseteq D^{\mathcal{J}}$ for all $\mathcal{EL}^{\sqcup,\sqcap,\neg}(\mathcal{D})$-concept descriptions D, since then $A^{\mathcal{J}} = C^{\mathcal{I}} \subseteq C^{\mathcal{J}}$. For the induction base case, let $D = A$ for $A \equiv C \in \mathcal{T}$. We have $A^{\mathcal{I}} \subseteq C^{\mathcal{I}}$ and by definition of $\mathcal{O}_{\mathcal{T}}$, $A^{\mathcal{J}} = C^{\mathcal{I}}$. Thus, $A^{\mathcal{I}} \subseteq A^{\mathcal{J}}$. The cases of primitive concepts, negated primitive concepts and concrete domain constructors are trivial. For the induction step, let $D = \exists R.D_1$. We have $R^{\mathcal{I}} = R^{\mathcal{J}}$, the induction hypothesis yields $D_1^{\mathcal{I}} \subseteq D_1^{\mathcal{J}}$ and thus $D^{\mathcal{I}} \subseteq D^{\mathcal{J}}$. For $D = D_1 \sqcap D_2$, we have by the induction hypothesis $D_i^{\mathcal{I}} \subseteq D_i^{\mathcal{J}}, i \in \{1, 2\}$ and hence $D^{\mathcal{I}} \subseteq D^{\mathcal{J}}$. □

We are now prepared for proving soundness and completeness of Algorithm 3.

Lemma 7 (Soundness and completeness) *Let $S_0, \ldots, S_n = S$ be the sequence of relations produced by the algorithm. Then, the following are equivalent:*

1. $A_0 \sqsubseteq_{\mathcal{T}} B_0$
2. $(A_0, B_0) \in S$

Proof. $(2 \Rightarrow 1)$ Instead of proving the statement directly, we show by induction on i the following statement:

$$(A, B) \in S_i \text{ implies } A \sqsubseteq_{\mathcal{T}} B$$

For the induction base case, we clearly have $A \sqsubseteq_{\mathcal{T}} A'$. Now for the induction step, take $(A, B) \in S_{i+1} \backslash S_i$ and a model $\mathcal{I}$ of $\mathcal{T}$. Recall that conditions (s1)-(s4) are fulfilled for A and B, and let $x \in A^{\mathcal{I}}$. In order to show $x \in B^{\mathcal{I}}$, we show $x \in D^{\mathcal{I}}$ for every conjunct D of the concept definition of B:

$D \in P_{\mathcal{T}}(B)$: We have by (s1) $P_{\mathcal{T}}(B) \subseteq P_{\mathcal{T}}(A)$, and hence $x \in D^{\mathcal{I}}$.

$D \in \bar{P}_{\mathcal{T}}(B)$: By (s2), $\bar{P}_{\mathcal{T}}(B) \subseteq \bar{P}_{\mathcal{T}}(A)$. Hence $x \in D^{\mathcal{I}}$.

$D \in E_{\mathcal{T}}(B)$: Let $D = \exists R_B.B'$. By (s3), there is $(\exists R_A.A') \in E_{\mathcal{T}}(A)$ such that R_A implies R_B and $(A', B') \in S$. Induction hypothesis yields $A' \sqsubseteq_{\mathcal{T}} B'$, hence $x \in D^{\mathcal{I}}$.

$D \in F_{\mathcal{T}}(B)$: By (s4), we have $\bigwedge_A^{\mathcal{D}}$ implies $\bigwedge_B^{\mathcal{D}}$. Thus, for all $p(f_1, \ldots, f_k) \in F_{\mathcal{T}}(B)$, $x \in p(f_1, \ldots, f_k)^{\mathcal{I}}$. Hence $x \in D^{\mathcal{I}}$.

$(1 \Rightarrow 2)$ We show the contrapositive and assume $(A_0, B_0) \notin S$. We construct a model $\mathcal{I}$ of $\mathcal{T}$ such that there is an $x \in A_0^{\mathcal{I}}$ and $x \notin B_0^{\mathcal{I}}$. We unwind $\mathcal{T}$ starting at A_0 and inductively define a sequence $\mathcal{M}_i = (\mathcal{J}_i, p_i, n_i), i \geq 0$, where $\mathcal{J}_i$ is an interpretation and $p_i, n_i \subseteq \Delta^{\mathcal{J}_i} \times N_{def}(\mathcal{T})$. During the construction process, we ensure that p_i is totally functional and n_i partial functional. The intention of p_i is to indicate which concept definition of $\mathcal{T}$ is or will be unwinded at every point of $\mathcal{J}_i$. Additionally, if n_i is defined at some point it states which defined concept must not hold at this point. Moreover, for the construction of the $\mathcal{M}_i$, we ensure that the following invariant holds for all $y \in \Delta^{\mathcal{J}_i}$:

$$p_i(y) = A \text{ and } n_i(y) = B \quad \text{implies} \quad (A, B) \notin S \tag{3.1}$$

We have $(A, B) \notin S$ if some completion condition from Table 3.1 does not hold, and that is what we are going to exploit during the construction of the $\mathcal{M}_i$.

$\mathcal{M}_0$: We set $\Delta^{\mathcal{J}_0} := \{x\}$. All other sets, relations and concrete features are interpreted by the empty set, empty relation, and are undefined respectively. We define $p_0 := \{(x, A_0)\}$ and $n_0 := \{(x, B_0)\}$. Clearly, the invariant (3.1) holds in $\mathcal{M}_0$.

$\mathcal{M}_{i+1}$: For the beginning, we set $\mathcal{M}_{i+1} := \mathcal{M}_i$. We iterate over all leaves $y \in \Delta^{\mathcal{J}_i}$ and extend $\mathcal{J}_{i+1}$. Let $p_i(y) = A$. We add y to $A^{\mathcal{J}_{i+1}}$ and to every $P^{\mathcal{J}_{i+1}}$ for $P \in P_{\mathcal{T}}(A)$.

Firstly, we consider the case in which n_i is undefined in y, which we call the *default case*. In this case, we add a fresh node y' to $\Delta^{\mathcal{J}_{i+1}}$ for every $\exists R_A.A' \in E_{\mathcal{T}}(A)$. Moreover, we choose a role conjunct $r_1 \sqcap \ldots \sqcap r_k$ of R_A, add (y, y') to $r_j^{\mathcal{J}_{i+1}}, 1 \leq j \leq k$, set $p_{i+1}(y') := A'$ and leave n_{i+1} undefined in y'. Now for the concrete features, let δ be a solution to $\bigwedge_A^{\mathcal{D}}$, which contains concrete features names $f_1, \ldots, f_k$. We set $f_j(y)^{\mathcal{J}_{i+1}} := \delta(f_j), 1 \leq j \leq k$ and leave $f_j^{\mathcal{J}_{i+1}}$ unchanged for all the other nodes of $\Delta^{\mathcal{J}_{i+1}}$.

For the other case, if $n_i(y) = B$, we have $(A, B) \notin S$ by (3.1). We choose exactly one completion condition that does not hold for A and B and continue depending on it:

(s1): Then we can extend M_{i+1} as in the default case.

(s2): So $\bar{P}(B) \not\subseteq \bar{P}(A)$, i.e., there is some $P \in \bar{P}(B) \backslash \bar{P}(A)$. Then we extend M_{i+1} as in the default case, but additionally add y to $P^{\mathcal{J}_{i+1}}$.

(s3): There is some $(\exists R_B.B') \in E_{\mathcal{T}}(B)$ and for all $(\exists R_A.A') \in E_{\mathcal{T}}(A)$ we have (a) R_A does not imply R_B, or (b) $(A', B') \notin S$. Basically, we proceed as in the default case, except for the existential restrictions $(\exists R_A.A') \in E_{\mathcal{T}}(A)$, for which we choose in case of (a) some role disjunct of R_A that does not imply R_B. Otherwise, in case of (b), we set $n_{i+1}(y') := B'$, for the successor node y' of y introduced by $(\exists R_A.A')$. We note that (3.1) holds.

(s4): There is a solution δ to $\bigwedge_A^{\mathcal{D}}$ that is not a solution to $\bigwedge_B^{\mathcal{D}}$. Again, we proceed as in the default case, except that we interpret $f_j(y)^{\mathcal{J}_{i+1}} := \delta(d_j), 1 \leq j \leq k$.

Let $\mathcal{M}_\omega := (\mathcal{J}_\omega, p_\omega, n_\omega)$, where $\mathcal{J}_\omega := \bigcup_{i \geq 0} \mathcal{J}_i$, $p_\omega := \bigcup_{i \geq 0} p_i$ and $n_i := \bigcup_{i \geq 0} n_i$. Obviously, $\mathcal{J}_\omega$ is not yet a model of $\mathcal{T}$. In order to obtain a model from $\mathcal{J}_\omega$, we extend $\mathcal{J}_\omega$ infinitely w.r.t. $\mathcal{T}$, i.e., define $\mathcal{I}_0 := \mathcal{I}_\omega$ and $\mathcal{I}_{n+1} := \mathcal{O}_{\mathcal{T}}(\mathcal{I}_n)$. Then

$$\mathcal{I} := \bigcup_{i \geq 0} \mathcal{I}_i$$

We have that the following three facts hold:

Fact 1: $A^{\mathcal{I}} \subseteq C^{\mathcal{I}}$ for all $(A \equiv C) \in \mathcal{T}$. By the construction of $\mathcal{J}_\omega$ this holds for $\mathcal{I}_0$, and the application of $\mathcal{O}_{\mathcal{T}}$ preserves this fact by Lemma 6.

Fact 2: $C^{\mathcal{I}} \subseteq A^{\mathcal{I}}$ for all $(A \equiv C) \in \mathcal{T}$. Assume there is $y \in C^{\mathcal{I}}$ and $y \notin A^{\mathcal{I}}$. By the construction of $\mathcal{I}$, there is an i such that $y \in C^{\mathcal{I}_i}$. But then, $y \in A^{\mathcal{O}_{\mathcal{T}}(\mathcal{I}_i)} = A^{\mathcal{I}_{i+1}}$ and hence $y \in A^{\mathcal{I}}$.

Fact 3: $n_\omega(y) = B$ implies $y \notin B^{\mathcal{I}}$ for all $y \in \Delta^{\mathcal{I}}$. In order to show this, we prove by induction on i the following statement:

$$n_\omega(y) = B \quad \text{implies} \quad y \notin B^{\mathcal{I}_i} \tag{3.2}$$

For the induction base case, we have constructed $\mathcal{J}_\omega$ such that (3.2) holds. Now for the induction step, let $n_\omega(y) = B$ and $p_\omega(y) = A$, and assume $y \in B^{\mathcal{I}_{i+1}}$. By (3.1), we have that $(A, B) \notin S$. The application of $\mathcal{O}_\mathcal{T}$ does neither change the interpretation of the primitive concept names, roles names nor concrete features. Hence, (s3) does not hold for (A, B), since otherwise $y \notin B^{\mathcal{I}_{i+1}}$. So, there is an existential restriction $(\exists R_B.B') \in E_\mathcal{T}(B)$ and there is no $(\exists R_A.A') \in E_\mathcal{T}(A)$ with $(A', B') \in S$. For every R-successor y' of y in $\mathcal{J}_\omega$, if R implies R_B, we have by the construction of $\mathcal{J}_\omega$ that $n_\omega(y') = B'$. Now $y' \notin B'^{\mathcal{I}_i}$ by the induction hypothesis, and hence $y \notin B^{\mathcal{I}_{i+1}}$.

Taking together the three facts, we have that $\mathcal{I}$ is a model of $\mathcal{T}$, $x \in A_0^{\mathcal{I}}$ and $x \notin B_0^{\mathcal{I}}$. □

Theorem 3 *Let $\mathcal{D}$ be a p-admissible concrete domain. Then, subsumption in $\mathcal{EL}^{\sqcup,\sqcap,\neg}(\mathcal{D})$ w.r.t. cyclic TBoxes is in* PTime.

Interestingly, the sole extension of $\mathcal{EL}$ by role disjunction or non-convex p-admissible concrete domains directly leads to ExpTime-completeness of the subsumption problem w.r.t. general TBoxes (Baader, Brandt & Lutz 2005a). This big gap between cyclic and general TBoxes is quite surprising.

3.2 $\mathcal{EL}^{\sqcup,\sqcap,\geq}$

We now consider $\mathcal{EL}^{\sqcup,\sqcap,\geq}$, which extends $\mathcal{EL}$ by at-least restrictions, and the role con- and disjunction constructor. Again, we restrict the complex roles to be in DNF and do not allow them to occur inside number restrictions. We provide an algorithm for subsumption in $\mathcal{EL}^{\sqcup,\sqcap,\geq}$ w.r.t. cyclic TBoxes which runs in polynomial time w.r.t. the size of the input TBox. The algorithm and its correctness proof are very similar to the $\mathcal{EL}^{\sqcup,\sqcap,\neg}(\mathcal{D})$ case. For that reason, this section is a bit compact.

The algorithm also requires the input $\mathcal{EL}^{\sqcup,\sqcap,\geq}$-TBoxes to be in extended normal form.

Definition 18 An $\mathcal{EL}^{\sqcup,\sqcap,\geq}$-TBox $\mathcal{T}$ is in *extended normal form* iff for each concept definition $A \equiv C \in \mathcal{T}$, C is of the form

$$\bigsqcap_{1 \leq i \leq j} P_i \sqcap \bigsqcap_{1 \leq i \leq k} \exists R_i.B_i \sqcap \bigsqcap_{1 \leq i \leq \ell} \geq n_i r_i$$

for $j, k, \ell \geq 0$, $P_1, \ldots, P_j \in N_{prim}(\mathcal{T})$; $R_1, \ldots, R_\ell$ being complex roles in DNF; $B_1, \ldots, B_\ell \in N_{def}(\mathcal{T})$; $n_i \geq 2, 1 \leq i \leq \ell$; $r_1, \ldots, r_\ell \in N_R$ and $r_i \neq r_j, 1 \leq i < j \leq \ell$. ◊

(s1)	$P_{\mathcal{T}}(B) \subseteq P_{\mathcal{T}}(A)$
(s2)	For each $(\exists R_B.B') \in E_{\mathcal{T}}(B)$ there is $(\exists R_A.A') \in E_{\mathcal{T}}(A)$ such that R_A implies R_B and $(A', B') \in S$
(s3)	For each $(\geq mr) \in N(B)$ there is $(\geq nr) \in N(A)$ such that $n \geq m$

Table 3.2: $\mathcal{EL}^{\sqcup,\sqcap,\geq}$ completion conditions

Algorithm 4 $\mathcal{EL}^{\sqcup,\sqcap,\geq}$ -subsumption algorithm

Input: $\mathcal{EL}^{\sqcup,\sqcap,\geq}$-TBox $\mathcal{T}$ in extended normal form
  $S := \{(A, A) \mid A \in N_{def}(\mathcal{T})\}$
  while there are $(A, B) \notin S$ and completion conditions (s1)-(s3) hold for A and B **do**
    $S := S \cup \{(A, B)\}$
  end while

Let $\mathcal{T}$ be an $\mathcal{EL}^{\sqcup,\sqcap,\geq}$-TBox into extended normal form and $A \equiv C \in \mathcal{T}$. We use the following abbreviations to have easy access to the conjuncts of C:

- $P_{\mathcal{T}}(A) := \{P_i \mid 1 \leq i \leq j\}$
- $E_{\mathcal{T}}(A) := \{\exists R_i.B_i \mid 1 \leq i \leq k\}$
- $N_{\mathcal{T}}(A) := \{\geq n_i r_i \mid 1 \leq i \leq \ell\}$

In order to bring an arbitrary $\mathcal{EL}^{\sqcup,\sqcap,\geq}$-TBox $\mathcal{T}$ into extended normal form, we apply the normalization algorithm for $\mathcal{EL}$-TBoxes from Section 2.4.2 first, which can easily be modified to also work with at-least restrictions. Basically, at-least restrictions can be handled in an obvious way like primitive concept names. In the normalization process we additionally remove $\geq mr$ in all concept definitions that contain $\geq nr$ and $\geq mr$ such that $n > m$. We thus end up with an $\mathcal{EL}^{\sqcup,\sqcap,\geq}$-TBox in extended normal form. For $\mathcal{EL}^{\sqcup,\sqcap,\geq}$, we do not have to take care of concept satisfiability, since every $\mathcal{EL}^{\sqcup,\sqcap,\geq}$-concept description is satisfiable.

Proposition 5 *Subsumption w.r.t. an $\mathcal{EL}^{\sqcup,\sqcap,\geq}$-TBox can be reduced in polynomial time to subsumption in $\mathcal{EL}^{\sqcup,\sqcap,\geq}$ w.r.t. an $\mathcal{EL}^{\sqcup,\sqcap,\geq}$-TBox in extended normal form.*

The $\mathcal{EL}^{\sqcup,\sqcap,\geq}$-subsumption algorithm, Algorithm 4, takes a TBox $\mathcal{T}$ in extended normal form as input and computes a subsumption relation $S \subseteq N_{def}(\mathcal{T}) \times N_{def}(\mathcal{T})$ in polynomial time w.r.t. the size of the input TBox. It is similar to the previous subsumption algorithm. When it terminates, which obviously is always the case, we have $(A, B) \in S$ iff $A \sqsubseteq_{\mathcal{T}} B$.

Lemma 8 *Algorithm 4 runs in polynomial time.*

Proof. The algorithm produces a sequence of relations $S_0, \ldots, S_n$. In every successor relation, one tuple is added, and thus $n \leq |N_{def}(\mathcal{T})|^2$. In every step, the completion conditions from Table 3.2 have to be checked for at most $|N_{def}(\mathcal{T})|^2$ tuples. Checking the completion conditions is polynomial, since implication between complex roles in DNF can be checked in polynomial time. □

In the proof of completeness of the $\mathcal{EL}^{\sqcup,\sqcap,\geq}$-subsumption algorithm, we also rely on the monotonicity of $\mathcal{O}_\mathcal{T}$.

Lemma 9 *Let $\mathcal{T}$ be an $\mathcal{EL}^{\sqcup,\sqcap,\geq}$-TBox, $\mathcal{I}$ an interpretation such that $A^\mathcal{I} \subseteq C^\mathcal{I}$ for all $A \equiv C \in \mathcal{T}$ and $\mathcal{J} = \mathcal{O}_\mathcal{T}(\mathcal{I})$. Then, $A^\mathcal{J} \subseteq C^\mathcal{J}$ for all $A \equiv C \in \mathcal{T}$.*

Proof. Instead of proving the lemma directly, we show by structural induction $D^\mathcal{I} \subseteq D^\mathcal{J}$ for all $\mathcal{EL}^{\sqcup,\sqcap,\geq}$-concept descriptions D. For the induction start, let $D = A$ for $A \equiv C \in \mathcal{T}$. We have $A^\mathcal{I} \subseteq C^\mathcal{I}$ and by definition of $\mathcal{O}_\mathcal{T}$, $A^\mathcal{J} = C^\mathcal{I}$. Thus, $A^\mathcal{I} \subseteq A^\mathcal{J}$. The cases of primitive concepts and number restrictions are trivial. For the induction step, let $D = \exists R.D_1$. We have $R^\mathcal{I} = R^\mathcal{J}$, the induction hypothesis yields $D_1^\mathcal{I} \subseteq D_1^\mathcal{J}$ and thus $D^\mathcal{I} \subseteq D^\mathcal{J}$. For $D = D_1 \sqcap D_2$, we have by the induction hypothesis $D_i^\mathcal{I} \subseteq D_i^\mathcal{J}, i \in \{1, 2\}$ and hence $D^\mathcal{I} \subseteq D^\mathcal{J}$. □

Lemma 10 (Soundness and completeness) *Let $S_0, \ldots, S_n = S$ be the sequence of relations produced by the algorithm. Then, the following are equivalent:*

1. $A_0 \sqsubseteq_\mathcal{T} B_0$
2. $(A_0, B_0) \in S$

Proof. ($2 \Rightarrow 1$) Instead of proving the statement directly we show by induction on i the following:

$$(A, B) \in S_i \text{ implies } A \sqsubseteq_\mathcal{T} B$$

For the induction base case, clearly $A \sqsubseteq_\mathcal{T} A$. Now for the induction step, take $(A, B) \in S_{i+1} \setminus S_i$ and a model $\mathcal{I}$ of $\mathcal{T}$. Recall that completion conditions (s1)-(s3) are fulfilled for A and B. Let $x \in A^\mathcal{I}$, we show $x \in D^\mathcal{I}$ for every conjunct D of the concept definition of B, and thus $x \in B^\mathcal{I}$:

- $D \in P_\mathcal{T}(B)$: By (s1) we have $P_\mathcal{T}(B) \subseteq P_\mathcal{T}(A)$ and hence $x \in D^\mathcal{I}$.
- $D \in E_\mathcal{T}(B)$: Let $D = \exists R_B.B'$. By (s2), there is $(\exists R_A.A') \in E_\mathcal{T}(A)$ such that R_A implies R_B and $(A', B') \in S$. Induction hypothesis yields $A' \sqsubseteq_\mathcal{T} B'$, hence $x \in D^\mathcal{I}$.
- $D \in N_\mathcal{T}(B)$: Let $D = (\geq mr)$. Then, by (s3) there is $(\geq nr) \in N_\mathcal{T}(A), n \geq m$ and $x \in (\geq nr)^\mathcal{I}$. Hence $x \in D^\mathcal{I}$.

(1 $\Rightarrow$ 2) We show the contrapositive. Assume $(A_0, B_0) \notin S$. As in the $\mathcal{EL}^{\sqcup,\sqcap,\neg}(\mathcal{D})$ case, we construct a model $\mathcal{I}$ of $\mathcal{T}$ such that there is $x \in A_0^{\mathcal{I}}$ and $x \notin B_0^{\mathcal{I}}$. We unwind $\mathcal{T}$ starting at A_0. Therefore, we inductively define a sequence $\mathcal{M}_i = (\mathcal{J}_i, p_i, n_i), i \geq 0$ where $\mathcal{J}_i$ is an interpretation, $p_i \subseteq \Delta^{\mathcal{J}_i} \times \mathcal{P}(N_{def}(\mathcal{T}))$, and $n_i \subseteq \Delta^{\mathcal{J}_i} \times N_{def}(\mathcal{T})$. We ensure p_i to be totally functional, and n_i to be partial functional. The intention of p_i is that it keeps the set of concept definitions that have been or will be unwinded at some point of $\mathcal{J}_i$. In the process of the construction of the M_i, we need the possibility to expand more than one defined concept in a single point. For that reason, p_i maps to a subset of $N_{def}(\mathcal{T})$. Again, if n_i is defined for some node of $\mathcal{J}_i$, it indicates which defined concept must not hold there. We construct the $\mathcal{M}_i$ such that the following invariant holds for all $y \in \Delta^{\mathcal{J}_i}$:

$$p_i(y) = \{A\} \text{ and } n_i(y) = B \quad \text{implies} \quad (A, B) \notin S \tag{3.3}$$

Note, that $p_i(y)$ is a singleton in the precondition of (3.3). If $(A, B) \notin S$, we have that some completion condition from Table 3.2 does not hold, and this will guide the construction of the M_i.

In the following, if we do not explicitly define p_i or n_i for some node y, then $p_i(y) := \emptyset$ and n_i is undefined in y. Let $m := max\{n \mid A \in N_{def}(\mathcal{T}), (\geq nr) \in N_{\mathcal{T}}(A)\}$.

$\mathcal{M}_0$: For $\mathcal{M}_0$, we set $\Delta^{\mathcal{J}_0} := \{x, z_1, \ldots, z_m\}$. All other concepts and roles are interpreted by the empty set and the empty relation respectively. We define $n_0 := \{(x, B_0)\}$ and

$$p_0(y) \quad := \begin{cases} \{A_0\} & \text{if } y = x \\ \emptyset & \text{otherwise} \end{cases}$$

We note that (3.3) holds in M_0.

$\mathcal{M}_{i+1}$: At the beginning we set $\mathcal{M}_{i+1} := \mathcal{M}_i$. We iterate over all leaves $y \in \Delta^{\mathcal{J}_i}$, all $A \in p_i(y)$, and extend $\mathcal{J}_{i+1}$.

Firstly, we consider the *default case* where n_i is not defined in y. We fix the interpretation of the concept names in y, add y to $A^{\mathcal{J}_{i+1}}$ and to $P^{\mathcal{J}_{i+1}}$ for all $P \in P_{\mathcal{T}}(A)$. For the existential restrictions, we introduce *only one* fresh successor y'. So for each $(\exists R_A.A') \in E_{\mathcal{T}}(A)$, we choose a role conjunction $r_1 \sqcap \ldots \sqcap r_k$ from R_A, add (y, y') to $r_j^{\mathcal{J}_{i+1}}, 1 \leq j \leq k$ and A' to $p_{i+1}(y')$. Now for the at-least restrictions $(\geq nr) \in N_{\mathcal{T}}(A)$: If (y, y') has been added to $r^{\mathcal{J}_{i+1}}$, we add $\{(y, z_1), \ldots, (y, z_{n-1})\}$ to $r^{\mathcal{J}_{i+1}}$, and $\{(y, z_1), \ldots, (y, z_n)\}$ to $r^{\mathcal{I}}$ otherwise. We thus have ensured that y has *exactly* n r-successors.

For the other case let $n_i(y) = B$. We have $(A, B) \notin S$ by invariant (3.3). Depending on which completion condition from Table 3.2 does not hold, we extend M_{i+1}:

(s1): Then we can extend M_{i+1} as in the default case.

(s2): So there is an existential restriction $(\exists R_B.B') \in E_\mathcal{T}(B)$ and for every existential restriction $(\exists R_A.A') \in E_\mathcal{T}(A)$ (a) R_A does not imply R_B or (b) $(A', B') \notin S$. For those existential restrictions that fall under case (a) we introduce a fresh node y', choose a role conjunction $r_1 \sqcap \ldots \sqcap r_k$ from R_A that does not imply R_B, add (y, y') to $r_j^{\mathcal{J}_{i+1}}, 1 \leq j \leq k$ and set $p_{i+1}(y') := \{A'\}$. For existential restrictions that fall under (b), we choose a role conjunction $r_1 \sqcap \ldots \sqcap r_k$ from R_A, introduce a fresh y', add (y, y') to $r_j^{\mathcal{J}_{i+1}}$, set $p_{i+1}(y') := \{A'\}$ and $n_{i+1}(y') := B'$. Again, the invariant (3.3) holds in M_{i+1}. In order to ensure that the at-least restrictions hold, for each $(\geq nr) \in N_\mathcal{T}(A)$, we add $\{(y, z_1), \ldots, (y, z_m)\}$ to $r^{\mathcal{I}_{i+1}}$.

(s3): We extend M_{i+1} as in the default case.

Let $\mathcal{M}_\omega := (\mathcal{J}_\omega, p_\omega, n_\omega)$, where $\mathcal{J}_\omega := \bigcup_{i\geq 0} \mathcal{J}_i$, $p_\omega = \bigcup_{i\geq 0} p_i$ and $n_\omega = \bigcup_{i\geq 0} n_i$. Obviously, $\mathcal{J}_\omega$ is not a model of $\mathcal{T}$. For that reason, we apply $\mathcal{T}$ infinitely often to $\mathcal{J}_\omega$. Let $\mathcal{I}_0 := \mathcal{J}_\omega$ and $\mathcal{I}_{n+1} := \mathcal{O}_\mathcal{T}(\mathcal{I}_n)$. We define

$$\mathcal{I} := \bigcup_{i\geq 0} \mathcal{I}_i.$$

The following three facts hold:

Fact 1: $A^\mathcal{I} \subseteq C^\mathcal{I}$ for all $(A \equiv C) \in \mathcal{T}$. By the construction of $\mathcal{J}_\omega$ this holds for $\mathcal{I}_0$, and the application of $\mathcal{O}_\mathcal{T}$ preserves this fact due to the monotonicity of $\mathcal{O}_\mathcal{T}$.

Fact 2: $C^\mathcal{I} \subseteq A^\mathcal{I}$ for all $(A \equiv C) \in \mathcal{T}$. Assume there is $y \in C^\mathcal{I}$. By the construction of $\mathcal{I}$, there is an i such that $y \in C^{\mathcal{I}_i}$. But then $y \in A^{\mathcal{O}_\mathcal{T}(\mathcal{I}_i)} = A^{\mathcal{I}_{i+1}}$, and hence $y \in A^\mathcal{I}$.

Fact 3: $n_\omega(y) = B$ implies $y \notin B^\mathcal{I}$ for all $y \in \Delta^\mathcal{I}$. In order to show the fact, we prove by induction on i the following statement:

$$n_\omega(y) = B \quad \text{implies} \quad y \notin B^{\mathcal{I}_i} \tag{3.4}$$

For the induction base case, we have constructed $\mathcal{J}_\omega$ such that (3.4) holds. Now for the induction step, let $n_\omega(y) = B$ and $p_\omega(y) = \{A\}$, and assume $y \in B^{\mathcal{I}_{i+1}}$. By (3.3), we have that $(A, B) \notin S$. The application of $\mathcal{O}_\mathcal{T}$ does neither change the interpretation of the primitive concept nor roles names. Thus, if (s1) does not hold for (A, B), $y \notin B^{\mathcal{I}_{i+1}}$. If (s3) does not hold for (A, B), there is $(\geq mr) \in N_\mathcal{T}(B)$ and no $(\geq nr) \in N_\mathcal{T}(A)$ such that $n \geq m$. By the construction, we have ensured that y has less than m r-successors. Hence, $y \notin B^{\mathcal{I}_{i+1}}$. Lastly, in case (s2) does not hold for (A, B), there is an existential restriction $(\exists R_B.B') \in E_\mathcal{T}(B)$ and there is no $(\exists R_A.A') \in E_\mathcal{T}(A)$ such that R_A implies R_B and $(A', B') \in S$. For every

R-successor y' of y in $\mathcal{J}_\omega$, if R implies R_B, we have by the construction of $\mathcal{J}_\omega$ that $n_\omega(y') = B'$. Now $y' \notin B'^{\mathcal{I}_i}$ by the induction hypothesis, and hence $y \notin B^{\mathcal{I}_{i+1}}$.

Taking together the three facts, we have that $\mathcal{I}$ is a model of $\mathcal{T}$, $x \in A_0^{\mathcal{I}}$ and $x \notin B_0^{\mathcal{I}}$. □

Theorem 4 *Subsumption in $\mathcal{EL}^{\sqcup,\sqcap,\geq}$ can be decided in polynomial time.*

Again, it is interesting that the sole extension of $\mathcal{EL}$ by at-least restrictions directly leads to ExpTime-completeness of subsumption w.r.t. general TBoxes (Baader, Brandt & Lutz 2005a).

Chapter 4

Intractable Extensions of $\mathcal{EL}$

In this chapter, we investigate extensions of $\mathcal{EL}$ for which the subsumption problem becomes intractable w.r.t. cyclic TBoxes. Firstly, we show that in a combination of the extensions presented in the previous chapter, subsumption is intractable. Moreover, we justify the requirement of the complex roles to be in DNF made in Chapter 3. We then continue having a look at subsumption in $\mathcal{EL}$ extended by negation, disjunction, transitive closure over role names, functionality and concrete domains with abstract feature chains. We close the chapter with a short discussion on $\mathcal{EL}$ extended by inverse roles, which remains an open question of this thesis.

For most extensions of $\mathcal{EL}$, concept satisfiability is trivial in the sense that every concept description is satisfiable without or w.r.t. acyclic and cyclic TBoxes. For that reason, we only consider concept satisfiability in $\mathcal{EL}$ if it is *not* trivial.

4.1 A Combination of $\mathcal{EL}^{\sqcup,\sqcap,\neg}(\mathcal{D})$ and $\mathcal{EL}^{\sqcup,\sqcap,\geq}$

In the previous chapter, we have seen two different extensions of $\mathcal{EL}$, for which the subsumption problem can be decided in polynomial time w.r.t. the size of input TBox. Naturally, the question comes up, whether we can combine both extensions in order to obtain an even more expressive, tractable description logic. We show in the following, that both the combination of at-least restrictions and primitive negation, and the combination of at-least restrictions and concrete domains lead to intractability of subsumption. Thus, subsumption in the combination of the two tractable description logics presented in the previous chapter is intractable.

Let $\mathcal{EL}^{\geq,\neg}$ be $\mathcal{EL}$ extended by at-least restrictions and primitive negation. As we will see in the following, subsumption in $\mathcal{EL}^{\geq,\neg}$ is CO-NP-complete. The upper bound comes from the description logic $\mathcal{ALUN}$, which allows for the concept constructors bottom, value restriction, conjunction, disjunction, primitive negation, number restrictions, and unqualified existential restriction ($\exists r.\top$). It has been shown in (Francesco M. Donini & Nutt 1997) that subsumption in $\mathcal{ALUN}$ is CO-NP-complete. Given two $\mathcal{EL}^{\geq,\neg}$-concept descriptions C, D, we have $C \sqsubseteq D$ iff $\neg D \sqsubseteq \neg C$. Bringing $\neg C$ and $\neg D$ into negation normal form gives us two $\mathcal{ALUN}$-concept descriptions, and thus subsumption in $\mathcal{EL}^{\geq,\neg}$ is also in CO-NP.

$$\begin{aligned}
\mathsf{Colors} &:= \exists r.(R \sqcap \neg G \sqcap \neg B) \sqcap \\
&\quad\ \exists r.(\neg R \sqcap G \sqcap \neg B) \sqcap \\
&\quad\ \exists r.(\neg R \sqcap \neg G \sqcap B) \\
\mathsf{Coloring} &:= \bigsqcap_{v \in V} \left(\exists r.(P_v \sqcap \bigsqcap_{\{v,w\} \in E} \neg P_w) \right) \\
C &:= \mathsf{Colors} \sqcap \mathsf{Coloring}
\end{aligned}$$

Table 4.1: The $\mathcal{EL}^{\geq,\neg}$-concept description C for the reduction of an instance $\mathcal{G} = (V, E)$ of 3-Colorability to non-subsumption in $\mathcal{EL}^{\geq,\neg}$.

For the lower bound, we reduce Graph 3-Colorability to non-subsumption in $\mathcal{EL}^{\geq,\neg}$.

Definition 19 (Graph 3-Colorability) For a given graph $\mathcal{G} = (V, E)$, the Graph 3-Colorability Problem (3-Colorability) is to determine whether there exists a *coloring function* $f : V \to \{1, 2, 3\}$ such that $f(u) \neq f(v)$ whenever $\{u, v\} \in E$. $\Diamond$

3-Colorability is known to be NP-complete (Garey & Johnson 1990). For a given instance $\mathcal{G} = (V, E)$ of 3-Colorability, Table 4.1 defines an $\mathcal{EL}^{\geq,\neg}$-concept description C that we will use for the reduction. Obviously, C is linear in the size of $\mathcal{G}$. Firstly, let us explain the intention of C. For an interpretation $\mathcal{I}$ and $x \in C^{\mathcal{I}}$, Colors ensures that x has three r-successors with three different concept names, R, G and B, that represent three colors. Then, $\mathsf{Coloring}$ requires x to have a successor for each vertex of $\mathcal{G}$ that does not overlap with any adjacent vertices. Clearly, C itself does not constrain the number of r-successors of x, but if there is a coloring function for $\mathcal{G}$, then there is an interpretation where x has exactly three r-successors.

Lemma 11 *Let $\mathcal{G} = (V, E)$ be an instance of 3-Colorability. Then, the following are equivalent:*

1. *$\mathcal{G}$ has a solution*
2. *$C \not\sqsubseteq (\geq 4r)$*

Proof. ($1 \Rightarrow 2$) Let f be the coloring function. We define an interpretation $\mathcal{I}$ with $\Delta^{\mathcal{I}} := \{x, y_1, y_2, y_3\}$ and set $r^{\mathcal{I}} := \{(x, y_i) \mid i \in \{1, 2, 3\}\}$, $R^{\mathcal{I}} := \{y_1\}$, $G^{\mathcal{I}} := \{y_2\}$ and $B^{\mathcal{I}} := \{y_3\}$. Now for every $v \in V$, we set $P_v^{\mathcal{I}} := \{y_{f(v)}\}$. Obviously, we have $x \in C^{\mathcal{I}}$ and $x \notin (\geq 4r)^{\mathcal{I}}$.

($2 \Rightarrow 1$) Let $\mathcal{I}$ be an interpretation such that $x \in C^{\mathcal{I}}$, $\{(x, y_i) \mid i \in \{1, 2, 3\}\} \subseteq r^{\mathcal{I}}$ and $x \notin (\geq 4r)^{\mathcal{I}}$. We define a coloring function f. For every vertex $v \in V$, there are possibly

$$
\begin{aligned}
\mathsf{Colors} &:= \exists r.({=_d}(f_R) \sqcap {\neq_d}(f_G) \sqcap {\neq_d}(f_B)) \sqcap \\
&\quad\ \exists r.({\neq_d}(f_R) \sqcap {=_d}(f_G) \sqcap {\neq_d}(f_B)) \sqcap \\
&\quad\ \exists r.({\neq_d}(f_R) \sqcap {\neq_d}(f_G) \sqcap {=_d}(f_B)) \sqcap \\
\mathsf{Coloring} &:= \bigsqcap_{v \in V} \left(\exists r.({=_d}(f_v) \sqcap \bigsqcap_{\{v,w\} \in E} {\neq_d}\ (f_w)) \right) \\
C &:= \mathsf{Colors} \sqcap \mathsf{Coloring}
\end{aligned}
$$

Table 4.2: The $\mathcal{EL}^{\geq}(\mathcal{D})$-concept description C for the reduction of an instance $\mathcal{G} = (V, E)$ of 3-Colorability to non-subsumption in $\mathcal{EL}^{\geq}(\mathcal{D})$, for f_R, f_G, f_B, $f_v \in N_{cF}, v \in V$.

more than one $y_i, i \in \{1,2,3\}$ such that $y_i \in P_v^{\mathcal{I}}$. Nevertheless, through Coloring it is ensured that in every y_i for every adjacent w of v, $y_i \notin P_w^{\mathcal{I}}$. Thus, for every $v \in V$, we set $f(v) := i$, where $i \in \{1,2,3\}$ is chosen such that $y_i \in P_v^{\mathcal{I}}$. □

Theorem 5 *Subsumption in $\mathcal{EL}^{\geq,\neg}$ is* co-NP*-complete.*

One proves similarly intractability of subsumption in $\mathcal{EL}$ extended by at-least restriction and a wide variety of concrete domains, $\mathcal{EL}^{\geq}(\mathcal{D})$. Any concrete domain with unary predicates $=_d$ and $\neq_d$ for some $d \in \Delta_{\mathcal{D}}$ with the obvious extension can be used together with at-least restrictions to also encode an instance $\mathcal{G} = (V, E)$ of 3-Colorability. The concept description C used for the reduction is presented in Table 4.2, which is very similar to Table 4.1. One can easily verify that $\mathcal{G}$ has a solution iff $C \not\sqsubseteq (\geq 4r)$.

Theorem 6 *Subsumption in $\mathcal{EL}^{\geq}(\mathcal{D})$ is* co-NP*-hard.*

For subsumption w.r.t. TBoxes in a combination of all extensions from $\mathcal{EL}^{\sqcup,\sqcap,\neg}(\mathcal{D})$ and $\mathcal{EL}^{\sqcup,\sqcap,\geq}$, to the best of the author's knowledge, no upper bounds are currently known.

4.2 Arbitrary Role Con- and Disjunction

A restriction made in the previous chapter was to require the complex roles to be in disjunctive normal form for both tractable extensions of $\mathcal{EL}$. We will see in the following that allowing an arbitrary nesting of con- and disjunction role constructors leads to co-NP-hardness of subsumption. In the following, $\mathcal{EL}^{\sqcap,\sqcup}$ denotes $\mathcal{EL}$ extended by *arbitrary* role con- and disjunction.

We show co-NP-hardness of subsumption in $\mathcal{EL}^{\sqcap,\sqcup}$ by reducing 3-Colorability to non-subsumption in $\mathcal{EL}^{\sqcap,\sqcup}$. Let $\mathcal{G} = (V, E)$ be an instance of 3-Colorability and $|V| = n$. We

$$\begin{aligned}
\mathsf{Coloring} &:= \bigsqcap_{1\leq i\leq n} (c_1^i \sqcup c_2^i \sqcup c_3^i)\\
\mathsf{Multicolored} &:= \bigsqcup_{1\leq i\leq n} ((c_1^i \sqcap c_2^i) \sqcup (c_1^i \sqcap c_3^i) \sqcup (c_2^i \sqcap c_3^i))\\
\mathsf{Neighbors} &:= \bigsqcup_{\substack{\{v_j,v_k\}\in E\\ i\in\{1,2,3\}}} (c_i^j \sqcap c_i^k)
\end{aligned}$$

Table 4.3: The definitions of the complex roles used in the reduction from 3-Colorability to non-subsumption in $\mathcal{EL}^{\sqcap,\sqcup}$.

introduce role names $c_1^1, \ldots, c_1^n, c_2^1, \ldots, c_2^n$ and $c_3^1, \ldots, c_3^n$ that represent the three possible colors of each vertex. The complex roles used to encode 3-Colorability for $\mathcal{G}$ are shown in Table 4.3. The intention of the definition of Coloring is that it requires each node of the graph to be colored by at least one color. The remaining two complex role definitions describe wrong coloring: Multicolored signals that a vertex is multicolored and Neighbors that two adjacent vertices have the same color. Clearly, the sizes of Coloring, Multicolored and Neighbors are linear w.r.t. to the size of $\mathcal{G}$.

Lemma 12 *Let $\mathcal{G} = (V, E)$ be an instance of 3-Colorability. Then, the following are equivalent.*

1. *$\mathcal{G}$ has a solution*
2. $\exists\mathsf{Coloring}.\top \not\sqsubseteq \exists(\mathsf{Multicolored} \sqcup \mathsf{Neighbors}).\top$

Proof. (1⇒2) Let f be the coloring function. We define an interpretation $\mathcal{I}$ with $\Delta^{\mathcal{I}} := \{x, y\}$. For each vertex $v_i, 1 \leq i \leq n$, we set $(c^i_{f(V_i)})^{\mathcal{I}} := \{(x, y)\}$ and for $j \neq f(V_i)$, c_j^i is interpreted by the empty relation. Clearly, $x \in (\exists\mathsf{Coloring}.\top)^{\mathcal{I}} \setminus (\exists(\mathsf{Multicolored} \sqcup \mathsf{Neighbors}).\top)^{\mathcal{I}}$.

(2⇒1) Let $\mathcal{I}$ be an interpretation and $(x, y) \in \mathsf{Coloring}^{\mathcal{I}} \setminus (\mathsf{Multicolored} \sqcup \mathsf{Neighbors})^{\mathcal{I}}$. We define a coloring function f as follows:

$$f(v_i) := j \qquad \text{if } (x, y) \in (c_j^i)^{\mathcal{I}}, j \in \{1, 2, 3\}$$

The facts that $(x, y) \in \mathsf{Coloring}^{\mathcal{I}}$ and $(x, y) \notin \mathsf{Multicolored}^{\mathcal{I}}$ ensure that f is well-defined. Moreover $(x, y) \notin \mathsf{Neighbors}^{\mathcal{I}}$ guarantees that f does not assign the same colors to any adjacent vertices. □

Theorem 7 *Subsumption in $\mathcal{EL}^{\sqcap,\sqcup}$ is* CO-NP-*hard.*

Lastly, it is not too hard to see that this lower bound for subsumption in $\mathcal{EL}^{\sqcap,\sqcup}$ is tight. Similar to $\mathcal{EL}$, one can use a homomorphism approach that in the induction step checks for implication of the complex roles, which is in NP for arbitrary combinations of role con- and disjunction. For subsumption in $\mathcal{EL}^{\sqcap,\sqcup}$ w.r.t. TBoxes, the best known upper bound is ExpTime, coming from $\mathcal{EL}^{\sqcap,\sqcup}$ being a notational fragment of $\mathcal{ALC}_{\text{reg}}$, which is $\mathcal{ALC}$ extended by regular expressions over role names. Schild showed (Schild 1991) that $\mathcal{ALC}_{\text{reg}}$ corresponds to Propositional Dynamic Logic (PDL) (Fischer & Ladner 1979) and for that reason subsumption in $\mathcal{ALC}_{\text{reg}}$ is ExpTime-complete.

4.3 Negation

Let $\mathcal{ELC}$ denote $\mathcal{EL}$ extended by negation. Since $\mathcal{ELC}$ is a notational variant of $\mathcal{ALC}$, which allows for the concept constructors top, negation and conjunction, the complexity of subsumption in $\mathcal{ALC}$ (Schmidt-Schauß & Smolka 1991, Schild 1994) directly carries over to $\mathcal{ELC}$, and concept satisfiability is no longer trivial.

Theorem 8 *Concept satisfiability, subsumption and subsumption w.r.t. acyclic TBoxes in $\mathcal{ELC}$ is* PSpace-*complete and* ExpTime-*complete w.r.t. cyclic and general TBoxes.*

Note, that in the proofs of hardness of satisfiability in $\mathcal{ALC}$ only one role name is used (Schild 1991, Schild 1994). Thus, for reductions we may assume that only *one* role name occurs in $\mathcal{ELC}$-TBoxes.

We now introduce a normal form of $\mathcal{ELC}$ TBoxes. Basically, it is the same normal form as for $\mathcal{EL}$-TBoxes from Section 2.4.2, but additionally restricts negation symbols to only occur in front of defined concept names.

Definition 20 An $\mathcal{ELC}$-TBox $\mathcal{T}$ is in *normal form* iff for each $A \equiv C \in \mathcal{T}$, C is either of the form $\top$ or P; $\neg B$; $\exists r.B$; or $B_1 \sqcap B_2$, for $P \in N_C$ and $B, B_1, B_2 \in N_{def}(\mathcal{T})$. ◊

Using a similar normalization algorithm as in Section 2.4.2, it is not hard to see that every $\mathcal{ELC}$-TBox $\mathcal{T}$ can be transformed in polynomial time into an equivalent $\mathcal{ELC}$-TBox in normal form that is linear in the size of $\mathcal{T}$.

Proposition 6 *Subsumption in $\mathcal{ELC}$ w.r.t. TBoxes can be reduced in polynomial time to subsumption in $\mathcal{ELC}$ w.r.t. TBoxes in normal form.*

4.4 Disjunction

We now consider the extension of $\mathcal{EL}$ by disjunction, denoted by $\mathcal{ELU}$. Brandt showed in (Brandt 2006) that subsumption in $\mathcal{ELU}$ is CO-NP-complete. In the following, we present exact complexities of subsumption in $\mathcal{ELU}$ w.r.t. to acyclic and cyclic TBoxes.

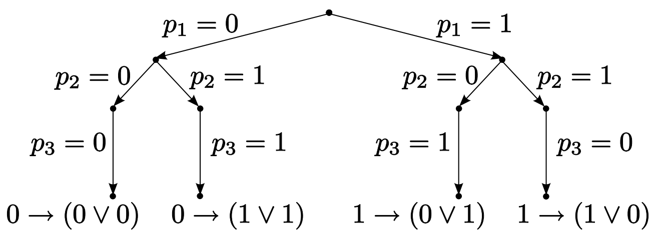

Figure 4.1: An example of a quantifier tree for $Q = \forall p_1.\forall p_2.\exists p_3.p_1 \rightarrow (p_2 \vee p_3)$.

For the case of acyclic TBoxes, subsumption in $\mathcal{ELU}$ is PSPACE-complete. In order to obtain the lower bound, we reduce validity of Quantified Boolean Formulas to non-subsumption in $\mathcal{ELU}$.

Definition 21 (Quantified Boolean Formula) A *Quantified Boolean Formula (QBF)* is of the form

$$Q_1 p_1 \ldots . Q_n p_n . \phi(p_1, \ldots, p_n)$$

where $Q_i \in \{\exists, \forall\}$ and $\phi(p_1, \ldots, p_n)$ is a propositional formula using only propositional variables $p_1, \ldots, p_n$. ◊

Validity of a QBF $Q = Q_1 p_1 \ldots . Q_n p_n . \phi(p_1, \ldots, p_n)$ is defined via induction on the length of the quantifier prefix. For a propositional formula ϕ we define $\phi[p/i], i \in \{0, 1\}$ to be the propositional formula that is obtained by replacing p by i in ϕ. The QBF Q is valid iff

1. $Q_1 = \forall$: $Q_2 p_1 \ldots . Q_n p_n . \phi[p_1/0]$ *and* $Q_2 p_1 \ldots . Q_n p_n . \phi[p_1/1]$ is valid
2. $Q_1 = \exists$: $Q_2 p_1 \ldots . Q_n p_n . \phi[p_1/0]$ *or* $Q_2 p_1 \ldots . Q_n p_n . \phi[p_1/1]$ is valid

We thus obtain a Boolean combination of truth values that is valid iff it evaluates to 1. Deciding validity of QBFs is known to be PSPACE-complete (Garey & Johnson 1990).

We can view QBFs as *quantifier trees*. Let $Q = Q_1 p_1 \ldots . Q_n p_n . \phi(p_1, \ldots, p_n)$ be a QBF, its quantifier tree is characterized as follows:

- Each level of the tree corresponds to one quantifier
- Each path starting from the root to a leaf has length n
- In $\forall$-levels each node has two successors, one for $p_i = 0$ and one for $p_i = 1$
- In $\exists$-levels each node has one successor: It suffices to explore one of the possibilities $p_i = 0$ or $p_i = 1$

$$
\begin{array}{rcl}
L_i & \equiv & \prod_{1\le j<i} (P_j \sqcup \bar{P}_j) \sqcap \begin{cases} \exists r.(P_i \sqcap L_{i+1}) \sqcap \exists r.(\bar{P}_i \sqcap L_{i+1}) & \text{if } Q_i = \forall \\ \exists r.L_{i+1} & \text{if } Q_i = \exists \end{cases} \\
L_{n+1} & \equiv & \prod_{1\le j\le n} (P_j \sqcup \bar{P}_j) \sqcap \tau(\phi(p_1,\dots,p_n)) \\
M & \equiv & \bigsqcup_{1\le k\le j\le n} \underbrace{\exists r.\dots\exists r}_{j \text{ times}}.\big((P_k \sqcap \exists r.\bar{P}_k) \sqcup (\bar{P}_k \sqcap \exists r.P_k) \sqcup (P_k \sqcap \bar{P}_k)\big)
\end{array}
$$

Table 4.4: The TBox $\mathcal{T}_Q$ for the reduction of an instance $Q = Q_1p_1.\dots Q_np_n.\phi(p_1,\dots,p_n)$ of QBF to non-subsumption in $\mathcal{ELU}$ for $1 \le i \le n$.

- On each branch, the Boolean formula ϕ evaluates to 1

It is not hard to see that a QBF Q is valid iff there exists a quantifier tree of Q. Figure 4.1 exemplarily shows the quantifier tree of $Q = \forall p_1.\forall p_2.\exists p_3.p_1 \rightarrow (p_2 \vee p_3)$.

Let $Q = Q_1p_1.\dots Q_np_n.\phi(p_1,\dots,p_n)$ be a QBF. It is assumed w.l.o.g. that ϕ is in negation normal form, i.e., negation occurs only in front of the $p_i, 1 \le i \le n$. We define a TBox $\mathcal{T}_Q$, shown in Table 4.4, that corresponds to Q and introduce concept names $P_1,\dots,P_n,\bar{P}_1,\dots,\bar{P}_n$ that represent the Boolean variables $p_1,\dots,p_n$ and their negation respectively. It is easily seen that $\mathcal{T}_Q$ is linear in the size of Q and acyclic.

The TBox $\mathcal{T}_Q$ contains the defined concepts $L_1,\dots,L_n$ that represent the quantifier levels, and L_{n+1} which represents ϕ. In the definition of L_{n+1}, $\tau(\phi(p_1,\dots,p_n))$ is the concept description corresponding to $\phi(p_1,\dots,p_n)$. It is obtained by simply replacing $\wedge$ by $\sqcap$, $\vee$ by $\sqcup$, p_i by P_i and $\neg p_i$ by $\bar{P}_i$. Furthermore, the additional defined concept M covers wrong behavior, i.e., M holds if there is up to depth n an incorrect propagation of the truth values or if some P_i and $\bar{P}_i$ hold simultaneously.

Lemma 13 *Let $Q = Q_1p_1.\dots Q_np_n.\phi(p_1,\dots,p_n)$ be a QBF and $\mathcal{T}_Q$ the TBox corresponding to Q. Then, the following are equivalent:*

1. *Q is valid*
2. *$L_1 \not\sqsubseteq_{\mathcal{T}_Q} M$*

Proof. $(1 \Rightarrow 2)$ Q is valid and thus there exists a quantifier tree. Clearly, this tree yields a model $\mathcal{I}$ of $\mathcal{T}_Q$: Its interpretation domain are the vertices of the tree, and if two vertices v_1, v_2 are connected, then $(v_1, v_2) \in r^{\mathcal{I}}$. Moreover, if $p_i = 0$ at the incoming edge of some node v, then $v \in \bar{P}_i$ and $w \in \bar{P}_i$ for all nodes w below v. The case in which $p_i = 1$ at the incoming edge of node v is defined similarly. For $L_i, 1 \le i \le n+1$, we define $v \in L_i^{\mathcal{I}}$ iff

$depth(v) = n + 1 - i$. Moreover, we set $M^{\mathcal{I}} := \emptyset$. Thus, $\mathcal{I}$ is a model for $\mathcal{T}_Q$, L_1 holds at the root of $\mathcal{I}$. Hence, $L_1 \not\sqsubseteq_{\mathcal{T}_Q} M$.

($2 \Rightarrow 1$) Let $\mathcal{I}$ be a model of $\mathcal{T}_Q$ and $x \in L_1^{\mathcal{I}} \setminus M^{\mathcal{I}}$. Due to the tree model property (Baader et al. 2003), we may assume that x is the root of a tree. Let t be the sub-tree obtained from $\mathcal{I}$ that contains all nodes reachable via r-edges along paths starting from x of length n. In order to make a quantifier tree of t, for every node y with $depth(y) = i$ and a successor node y', the connecting edge is labeled with $p_{n+1-i} = 0$ if $y' \in \bar{P}_i^{\mathcal{I}}$, and with $p_{n+1-i} = 1$ if $y' \in P_i^{\mathcal{I}}$. If there exist $\exists$-levels that have more than one successor, in each such level all successor branches except one are dropped. Likewise, if there are $\forall$-levels that have more than two successors, only two successors branches that define different truth values are kept and the rest is dropped. Since $x \notin M^{\mathcal{I}}$, we have ensured that all P_i's are correctly propagated along the tree, and that no P_i and $\bar{P}_i$ hold simultaneously. Thus, we have constructed a quantifier tree for Q and Q is valid. □

The lemma yields a PSpace lower bound for non-subsumption in $\mathcal{ELU}$ w.r.t. acyclic TBoxes. Since PSpace is a deterministic class, we have that subsumption in $\mathcal{ELU}$ is PSpace-hard w.r.t. acyclic TBoxes. The PSpace upper bound is an immediate consequence of $\mathcal{ELU}$ being a notational fragment of $\mathcal{ALC}$.

Theorem 9 *Subsumption in $\mathcal{ELU}$ w.r.t. acyclic TBoxes is* PSpace*-complete.*

For the case of subsumption in $\mathcal{ELU}$ w.r.t. cyclic TBoxes, we reduce *satisfiability* in $\mathcal{ELC}$ to non-subsumption in $\mathcal{ELU}$. Thus, let $\mathcal{T} = \{A_1 \equiv C_1, \ldots, A_n \equiv C_n\}$ be a cyclic $\mathcal{ELC}$-TBox in normal form with only one role name occurring in it. Let $\bar{A}_1, \ldots, \bar{A}_n$ be concept names not occurring in $\mathcal{T}$. Their intention is to simulate negation, i.e., $\bar{A}$ should hold at some point iff A does not hold at this point. In order to obtain an $\mathcal{ELU}$-TBox $\mathcal{T}'$ corresponding to $\mathcal{T}$, for $B \in N_{def}(\mathcal{T})$ we

1. replace any $A \equiv \exists r.A_j \in \mathcal{T}$ by
$$A \equiv \exists r.(A_j \sqcap \bigsqcap_{1 \le i \le n} (A_i \sqcup \bar{A}_i))$$
2. replace any $A \equiv \neg A_j \in \mathcal{T}$ by $A \equiv \bar{A}_j$, $1 \le j \le n$.

Finally, we introduce the following additional concept definition:
$$M \quad \equiv \quad \exists r.M \sqcup \bigsqcup_{1 \le i \le n} (A_i \sqcap \bar{A}_i)$$

The replacements in the first step ensure that A_j or $\bar{A}_j$ hold at every relevant point of connected models, except for the root node, which will be treated analogously. The replacements in the second step remove any negations occurring in $\mathcal{T}$, making $\mathcal{T}'$ an $\mathcal{ELU}$-TBox. The additional concept definition M signals a simultaneous occurrence of some A_j and $\bar{A}_j$ at some point of connected models $1 \le j \le n$. It is easily seen that $\mathcal{T}'$ is linear in the size of $\mathcal{T}$.

Lemma 14 *Let $\mathcal{T} = \{A_1 \equiv C_1, \ldots, A_n \equiv C_n\}$ be an $\mathcal{ELC}$-TBox in normal form with only one role name r occurring in it, $\mathcal{T}'$ the $\mathcal{ELU}$-TBox corresponding to $\mathcal{T}$ and $A \in N_{def}(\mathcal{T})$. Furthermore, let*

$$F := \bigsqcap_{1 \leq i \leq n} (A_i \sqcup \bar{A}_i).$$

Then, the following are equivalent:

1. *A is satisfiable w.r.t. $\mathcal{T}$*
2. *$A \sqcap F \not\sqsubseteq_{\mathcal{T}'} M$*

Proof. ($1 \Rightarrow 2$) Let $\mathcal{I}$ be a model of $\mathcal{T}$ and $x \in A^{\mathcal{I}}$. Let $\mathcal{I}'$ be defined as $\mathcal{I}$ and additionally interpret $\bar{A}_i^{\mathcal{I}'} := \Delta^{\mathcal{I}'} \setminus A_i^{\mathcal{I}'}, 1 \leq i \leq n$ and $M^{\mathcal{I}'} := \emptyset$. The construction ensures that at every point of $\mathcal{I}'$ either A_i or $\bar{A}_i$ hold for every $1 \leq i \leq n$. Consequently, $\mathcal{I}'$ is also a model of $\mathcal{T}'$, and $x \in (A \sqcap F)^{\mathcal{I}'} \setminus M^{\mathcal{I}'}$. Hence, $A \sqcap F \not\sqsubseteq_{\mathcal{T}'} M$.

($2 \Rightarrow 1$) Let $\mathcal{I}$ be a model of $\mathcal{T}'$ and $x \in (A \sqcap F)^{\mathcal{I}} \setminus M^{\mathcal{I}}$. In order to obtain a model $\mathcal{I}'$ of $\mathcal{T}$, we need to strip $\mathcal{I}$. For a relation $R \subseteq M \times M$, in order to obtain its *reflexive* transitive closure, we redefine $R^1 := R^0 \cup R$ and for $i \geq 0$, we set

$$\begin{aligned} \Delta^{\mathcal{I}'_i} &:= (r^{\mathcal{I}})^i(x) \cap F^{\mathcal{I}} \\ r^{\mathcal{I}'_i} &:= r^{\mathcal{I}} \cap (\Delta^{\mathcal{I}'_i} \times \Delta^{\mathcal{I}'_i}) \\ A^{\mathcal{I}'_i} &:= A^{\mathcal{I}} \cap \Delta^{\mathcal{I}'_i}, A \in N_C. \end{aligned}$$

Then,

$$\Delta^{\mathcal{I}'} := \bigcup_{i \geq 0} \Delta^{\mathcal{I}'_i}, \quad r^{\mathcal{I}'} := \bigcup_{i \geq 0} r^{\mathcal{I}'_i}, \quad A_j^{\mathcal{I}'} := \bigcup_{i \geq 0} r^{\mathcal{I}'_i}, 1 \leq j \leq n.$$

In order to prove that $\mathcal{I}'$ is a model of $\mathcal{I}$, we prove by induction on m the following statement, where $A_j \equiv C_j \in \mathcal{T}, 1 \leq j \leq n$:

$$A_j^{\mathcal{I}'_m} \subseteq C_j^{\mathcal{I}'_{m+1}}$$

It suffices to only consider the case $C_j = \exists r.B$ and $C_j = \neg B$ in the induction step. First of all, it is easily seen that $\Delta^{\mathcal{I}'_0} \subseteq \Delta^{\mathcal{I}'_1} \subseteq \ldots$. Let $C_j = \exists r.B$ and $y \in A_i^{\mathcal{I}'_{m+1}} \setminus A_i^{\mathcal{I}'_m}$. Since $y \in (\exists r.(B \sqcap F))^{\mathcal{I}}$, there is a $z \in (B \sqcap F)^{\mathcal{I}}$ such that $(y, z) \in r^{\mathcal{I}}$. Consequently, $z \in \Delta^{\mathcal{I}'_{m+2}}$ and $(y, z) \in r^{\mathcal{I}'_{m+2}}$. Hence, $y \in (\exists r.B)^{\mathcal{I}'_{m+2}}$. For the other case, let $C_j = \neg B$ and $y \in A^{\mathcal{I}'_{m+1}}$. We have $y \in \bar{B}^{\mathcal{I}}$ and since $y \in F^{\mathcal{I}} \setminus M^{\mathcal{I}}$, $y \notin B^{\mathcal{I}'_{m+2}}$. It is not hard to see that $C^{\mathcal{I}'} \subseteq A^{\mathcal{I}'}$ for all $A \equiv C \in \mathcal{T}$. Thus, we have that $\mathcal{I}'$ is a model of $\mathcal{T}$ and $x \in A^{\mathcal{I}'}$. □

The reduction gives an ExpTime lower bound for non-subsumption in $\mathcal{ELU}$ w.r.t. cyclic TBoxes. Again, since ExpTime is a deterministic class, this also yields an ExpTime lower bound for subsumption in $\mathcal{ELU}$.The ExpTime upper bound is an immediate consequence of $\mathcal{ELU}$ being a notational fragment of $\mathcal{ALC}$.

Theorem 10 *Subsumption in $\mathcal{ELU}$ w.r.t. cyclic TBoxes is* ExpTime*-complete.*

4.5 Transitive Closure over Role Names

We are now going to investigate the complexity of reasoning in $\mathcal{EL}^+$, that is $\mathcal{EL}$ extended by transitive closure over role names. It turns out that $\mathcal{EL}^+$ is as difficult as $\mathcal{ELU}$ from a complexity theory point of view.

There is a close relationship between subsumption in $\mathcal{EL}^+$ and query containment in the XPath fragment $\mathrm{XP}^{\{[],*,//\}}$, which is investigated by Miklau and Suciui in (Miklau & Suciu 2002). XPath (Clark & DeRose 1999) is a query language to select sets of nodes in XML-documents. An XPath query q_1 is contained in a query q_2 iff for all XML-documents the set of nodes selected by q_1 is a subset of the nodes selected by q_2. Due to the close relationship between $\mathrm{XP}^{\{[],*,//\}}$ and $\mathcal{EL}^+$, we will subsequently employ techniques from (Miklau & Suciu 2002).

Let us first introduce some notions that we will frequently use in the following. We call an existential restriction $\exists r^+.C$ a *transitive existential restriction.* For $\mathcal{EL}^+$-concept descriptions, we define *rdepth* similarly as for $\mathcal{EL}$ such that transitive existential restrictions are treated in the same way as existential restrictions. A path $x \xrightarrow{r} \dots \xrightarrow{r} y$ in a graph is called an *r-chain from x to y.* $\mathcal{EL}^+$-descriptions trees are defined in an obvious way like $\mathcal{EL}$-description trees which additionally may have *transitive edges*, i.e., edges labeled with $r^+, r \in N_R$. Let C be an $\mathcal{EL}^+$-concept description. In order to translate the corresponding $\mathcal{EL}^+$-description tree t_C into an $\mathcal{EL}$-description tree, we expand the transitive edges of t. To do so, we need an order on the edges of t_C. Let $\preceq_{N_R}$ be a well-order on N_R and $V_{t_C} = \{v_1, \dots, v_n\}$. We define $(v_{i_1}, r_1^+, v_{j_1}) \in E_{t_C} \preceq (v_{i_2}, r_2^+, v_{j_2}) \in E_{t_C}$ iff $r_1 \preceq_{N_R} r_2$, $i_1 \leq i_2$ and $j_1 \leq j_2$. Let $e_1 = (v_1, r^+, v_1'), \dots, e_n = (v_n, r^+, v_n')$ be all r^+-edges of t_C such that $e_1 \preceq \dots \preceq e_n$. Then, for $1 \leq i \leq n$, e_i is called the *i-th transitive edge of t_C*. Let t_C contain m different role names, that occur as transitive edges in t_C, each $k_i \geq 0$ times, $1 \leq i \leq m$. We call $U = (u_1^1, \dots, u_{k_1}^1, \dots, u_1^m, \dots, u_{k_m}^m)$ an *expansion vector of t_C*, whose components are all positive integers. The *canonical $\mathcal{EL}$-description tree induced by t_C and U*, written as $t_C[U]$, is obtained by replacing the i-th transitive edge of $r_j, (v, r_j^+, w) \in E_{t_C}$, by $(v, r_j, v_2), \dots, (v_{u_i^j}, r_j, w)$ for fresh nodes $v_2, \dots, v_{u_i^j - 1}$. For an $\mathcal{EL}^+$-concept description C with the corresponding $\mathcal{EL}^+$-description tree t_C and an expansion vector U of t_C, we denote by $C[U]$ the $\mathcal{EL}$-concept description that corresponds to $t_C[U]$. We define $\mathbb{U}_{t_C}$ to be the set of all expansion vectors of t_C, and for $n > 0$, $\mathbb{U}_{t_C}^n$ to be the set of all expansion vectors of t_C whose components are all smaller or equal to n. Similarly, for an $\mathcal{EL}^+$-concept description C, $\mathbb{U}_C := \mathbb{U}_{t_C}$ and $\mathbb{U}_C^n := \mathbb{U}_{t_C}^n$. We recall that we make no distinction between an $\mathcal{EL}$-description tree t and the interpretation $\mathcal{I}$ induced by t.

Figure 4.2 illustrates the expansion of the $\mathcal{EL}^+$-description tree t_C corresponding to $C = \exists s^+.P_1 \sqcap \exists r.(\exists s.(P_1 \sqcap P_2) \sqcap \exists r^+.\top)$ with the expansion vector $U \in \mathbb{U}_C^2 = (r_1^1, s_1^1) =$

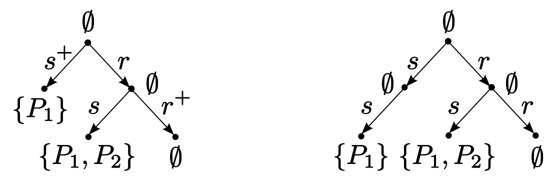

Figure 4.2: Expansion of t_C with expansion vector $U = (r_1^1, s_1^1) = (1, 2)$, for $r \preceq_{N_R} s$ and $C = \exists s^+.P_1 \sqcap \exists r.(\exists s.(P_1 \sqcap P_2) \sqcap \exists r^+.\top)$.

$(1, 2)$, where $r \preceq_{N_R} s$. The result is an $\mathcal{EL}$-description tree, that corresponds to $C' = \exists s.(\exists s.P_1) \sqcap \exists r.(\exists s.(P_1 \sqcap P_2) \sqcap \exists r.\top)$.

Before we begin having a look at subsumption in $\mathcal{EL}^+$, let us figure out two special properties of $\mathcal{EL}^+$. Firstly, transitive existential restrictions allow some sort of disjunction that we will later use to encode truth values. Let $\mathcal{I}$ be an interpretation, $x \in \Delta^\mathcal{I}$ and $P \in N_C$, then

$$x \in (\exists r^+.P)^\mathcal{I} \text{ iff } x \in (\exists r.P \sqcup \exists r.\exists r^+.P)^\mathcal{I}.$$

Secondly, in (Miklau & Suciu 2002) a neat technique has been introduced in order to express $C \sqsubseteq C_1 \sqcup \ldots \sqcup C_n$ for $\mathcal{EL}^+$-concept descriptions $C, C_1, \ldots, C_n$.

Lemma 15 *Let $C, C_1, \ldots, C_k$ be $\mathcal{EL}^+$-concept descriptions, $k > 0$. For $m > 0$, we define*

$$\begin{aligned}
\underline{C} &:= \bigsqcap_{1 \le i \le k} C_i \\
\underline{C}^1(D) &:= \exists r.\underline{C} \sqcap D \\
\underline{C}^{m+1}(D) &:= \exists r.\underline{C} \sqcap \exists r.\underline{C}^m(D) \\
\\
C_\ell &:= \exists r.\underline{C}^{k-1}(\exists r.(\exists r.C \sqcap \exists r.\underline{C}^{k-1}(\top))) \\
C_r &:= \exists r^+.(\exists r.C_1 \sqcap \exists r.(\exists r.C_2 \sqcap \ldots \exists r.C_k \ldots))
\end{aligned}$$

Then, the following are equivalent:

1. $C \sqsubseteq C_1 \sqcup \ldots \sqcup C_k$
2. $C_\ell \sqsubseteq C_r$

A detailed proof can be found in (Miklau & Suciu 2002). Figure 4.3 visualizes C_ℓ and C_r. Intuitively, it is clear that $\underline{C} \sqsubseteq C_j, 1 \le j \le k$ and that if $C \sqsubseteq C_j$ for some $1 \le j \le k$, r^+

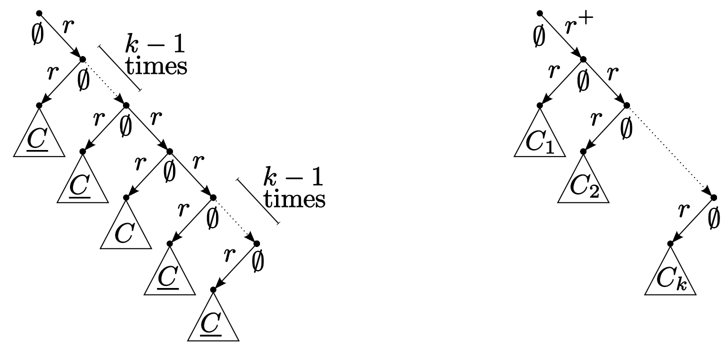

Figure 4.3: The $\mathcal{EL}^+$-concept description trees of C_ℓ and C_r constructed in Lemma 15.

in C_r can be replaced by an r-chain of length j such that there exists a homomorphism from the root of C_r to the root of C_ℓ. For the other direction, if $C \not\sqsubseteq C_j$ for all $1 \leq j \leq k$, we cannot expand r^+ in C_r such that there exists a homomorphism from the root of C_r to the root of C_ℓ, since every possible expansion will "shift" some C_j such that C would need to be subsumed by $C_j, 1 \leq j \leq k$.

The lemma provides a strong tool, and we will implicitly use it, i.e., directly write $C \sqsubseteq C_1 \sqcup \ldots \sqcup C_k$. Moreover, we write $\exists r^*.C$ in order to abbreviate $C \sqcup \exists r^+.C$.

Subsumption in $\mathcal{EL}^+$

In order to obtain a lower bound for subsumption in $\mathcal{EL}^+$, we reduce 3-SAT to non-subsumption in $\mathcal{EL}^+$. The approach for the reduction is due to Miklau and Suciu (Miklau & Suciu 2002).

Definition 22 (3-SAT) A *3-SAT-formula* is a propositional formula in conjunctive normal form, such that each clause contains exactly three literals. The *3-SAT problem* (3-SAT) is to determine for a given 3-SAT-formula, whether there exists a valuation of the propositional variables such that the 3-SAT-formula evaluates to true. ◊

It is well-known that 3-SAT is NP-complete (Garey & Johnson 1990). Let ϕ be an instance of 3-SAT over the propositional variables $p_1, \ldots, p_n$. Then, ϕ has no solution iff every possible valuation makes some clause false. We encode the possible truth values of the propositional variables with the help of transitive existential restrictions. For an interpretation $\mathcal{I}$ and $x \in \Delta^\mathcal{I}$, our intention is that $x \in (\exists r.P_i)^\mathcal{I}$ encodes that p_i is true in x, and $x \in (\exists r.\exists r^+.P_i)$ that p_i is false in x. Thus, stating $\exists r^+.P_i$ at some point gives us a way to choose a truth value.

Lemma 16 *Let $\phi = \psi_1 \wedge \ldots \wedge \psi_k, \psi_j = \ell_1^j \vee \ell_2^j \vee \ell_3^j, 1 \leq j \leq k$ be an instance of 3-SAT over the propositional variables $p_1, \ldots, p_n$. Furthermore, let*

$$\tau(\ell) := \begin{cases} \exists r.P_i & \text{if } \ell = \neg p_i \\ \exists r.\exists r^+.P_i & \text{if } \ell = p_i \end{cases}$$

and $C_i := \tau(\ell_1^i) \sqcap \tau(\ell_2^i) \sqcap \tau(\ell_3^i), 1 \leq i \leq k$. Then, the following are equivalent:

1. *ϕ has a solution*
2. *$\exists r^+.P_1 \sqcap \ldots \sqcap \exists r^+.P_n \not\sqsubseteq C_1 \sqcup \ldots \sqcup C_k$*

Proof. ($1 \Rightarrow 2$) Let $V : \{p_1, \ldots, p_n\} \to \{0,1\}$ be the valuation function such that ϕ is true under V. We define an interpretation $\mathcal{I}$ with $\Delta^{\mathcal{I}} := \{x, y_2, \ldots, y_{2k+1}\}$ and set $P_i^{\mathcal{I}} := y_{2i}$ if $V(p_i) = 1$ and $P_i^{\mathcal{I}} := y_{2i+1}$ if $V(p_i) = 0$ for $1 \leq i \leq k$. We furthermore add (x, y_{2i}) to $r^{\mathcal{I}}$ if $V(p_i) = 1$, and (x, y_{2i}) and (y_{2i}, y_{2i+1}) to $r^{\mathcal{I}}$ if $V(p_i) = 0$. Clearly, $x \in (\exists r^+.P_1 \sqcap \ldots \sqcap \exists r^+.P_k)^{\mathcal{I}}$. Now assume $x \in C_i^{\mathcal{I}}$ for some $1 \leq i \leq k$. By the construction of C_i and $\mathcal{I}$ this would imply that the i-th clause is false under V, which contradicts V being a valuation function that makes ϕ true.

($2 \Rightarrow 1$) Let $\mathcal{I}$ be an interpretation such that $x \in (\exists r^+.P_1 \sqcap \ldots \sqcap \exists r^+.P_k)^{\mathcal{I}} \setminus (C_1 \sqcup \ldots \sqcup C_k)^{\mathcal{I}}$. We define a valuation function V for $1 \leq i \leq n$ as follows:

$$V(p_i) := \begin{cases} 0 & \text{if } x \in (\exists r.\exists r^+.P_i)^{\mathcal{I}} \setminus (\exists r.P_i)^{\mathcal{I}} \\ 1 & \text{otherwise} \end{cases}$$

Assume some clause ψ_i is false under $V, 1 \leq i \leq k$. This is the case if V makes every literal of ψ_i false. However, by the definition of V and the construction of C_i, this implies $x \in C_i^{\mathcal{I}}$. □

For the upper bound, we prove containment of subsumption in $\mathcal{EL}^+$ in co-NP by generalizing the proof of containment of XPath query containment in co-NP from (Miklau & Suciu 2002). The generalization is needed since XPath queries allow for neither multiple role names nor multiple labels on nodes.

We define homomorphisms between $\mathcal{EL}^+$-description graphs and $\mathcal{EL}$-description graphs that serve as the basis for testing subsumption.

Definition 23 Let $\mathcal{G}_1$ be an $\mathcal{EL}^+$-description graph, $\mathcal{G}_2$ an $\mathcal{EL}$-description graph and $x_i \in \mathcal{G}_i, i \in \{1,2\}$. A functional binary relation $\mathcal{H} \subseteq V_{\mathcal{G}_1} \times V_{\mathcal{G}_2}$ is a *homomorphism* from x_1 to x_2 iff

- $(x_1, x_2) \in \mathcal{H}$
- If $(v, w) \in \mathcal{H}$ then:

(h1) $\ell(v) \subseteq \ell(w)$

(h2) For all $(v, r, v') \in E_{\mathcal{G}_1}$ there is $(w, r, w') \in E_{\mathcal{G}_2}$ and $(v', w') \in \mathcal{H}$

(h3) For all $(v, r^+, v') \in E_{\mathcal{G}_1}$ there is an r-chain $w \xrightarrow{r} \ldots \xrightarrow{r} w'$ in $\mathcal{G}_2$ such that $(v', w') \in \mathcal{H}$. ◊

If $\mathcal{G}_1$ and $\mathcal{G}_2$ are trees, checking the existence of a homomorphism is polynomial in $|\mathcal{G}_1| + |\mathcal{G}_2|$ (Miklau & Suciu 2002).

Again, we establish a connection between the description tree of an $\mathcal{EL}^+$-concept description C and an interpretation of in terms of a homomorphism.

Lemma 17 *Let C be an $\mathcal{EL}^+$-concept description with the corresponding $\mathcal{EL}^+$-concept description tree t_C, and $\mathcal{I}$ an interpretation with the corresponding $\mathcal{EL}$-description graph $\mathcal{G}_\mathcal{I}$. Then, the following are equivalent:*

1. *$x \in C^\mathcal{I}$*
2. *There exists a homomorphism from the root x_C of t_C to x.*

Proof. The proof is by induction on $rdepth(C)$. We only show the induction step for $C = P_1 \sqcap \ldots \sqcap P_k \sqcap \exists r_1.C_1 \sqcap \ldots \sqcap \exists r_m.C_m \sqcap \exists s_1^+.D_1 \sqcap \ldots \sqcap \exists s_n^+.D_n$.

(1 ⇒ 2) Since $x \in C^\mathcal{I}$, $\ell_{t_C}(x_C) \subseteq \ell_\mathcal{I}(x)$. We have that there are y_{C_i} such that $(x, y_{C_i}) \in r_i^\mathcal{I}$ and $y_{C_i} \in C_i^\mathcal{I}, 1 \leq i \leq m$, and y_{D_i} such that $(x, y_{D_i}) \in (s_i^\mathcal{I})^+$ and $y_{D_i} \in D_i^\mathcal{I}, 1 \leq i \leq n$. By the induction hypothesis, there exist homomorphisms $\mathcal{H}_{C_i}$ from x_{C_i} to y_{C_i}, where $(x_C, r_i, x_{C_i}) \in E_{t_C}, 1 \leq i \leq m$. Likewise, there exist homomorphisms $\mathcal{H}_{D_i}$ from x_{D_i} to y_{D_i}, where $(x_C, r_i^+, x_{D_i}) \in E_{t_C}$, $1 \leq i \leq n$. Now $\mathcal{H} := \{(x_C, x)\} \cup \bigcup_{1 \leq i \leq m} \mathcal{H}_{C_i} \cup \bigcup_{1 \leq i \leq n} \mathcal{H}_{D_i}$ is obviously a homomorphism from x_C to x.

(2 ⇒ 1) Let $\mathcal{H}$ be a homomorphism from x_C to x. By (h3), there are $(x, y_{D_i}) \in (r_i^\mathcal{I})^+$ such that for all $(x_C, r_i^+, x_{D_i}) \in E_{t_C}$, $(x_{D_i}, y_{D_i}) \in \mathcal{H}, 1 \leq i \leq m$. Clearly, $\mathcal{H}$ is also a homomorphism from x_{D_i} to y_{D_i}. Thus by the induction hypothesis, $y_{D_i} \in D_i^\mathcal{I}$. The argumentation holds similarly for the existential restrictions in C. For the concept names $P_1, \ldots, P_k$, we have $x \in P_i^\mathcal{I}, 1 \leq i \leq k$, since $\ell_{t_C}(x_C) \subseteq \ell_\mathcal{I}(x)$. Hence, $x \in C^\mathcal{I}$. □

Lemma 18 *Let C, D be $\mathcal{EL}^+$-concept descriptions. Then, the following are equivalent:*

1. *$C \sqsubseteq D$*
2. *$C[U] \sqsubseteq D$ for all expansion vectors $U \in \mathbb{U}_C$*

Proof. We show the contrapositive in both directions.

(1 ⇒ 2) Let $\mathcal{I}$ be an interpretation such that $x \in C[U]^\mathcal{I} \setminus D^\mathcal{I}$. By the previous lemma, there exists a homomorphism $\mathcal{H}$ from $x_{C[U]}$ to x. Now define $\mathcal{H}' := \mathcal{H} \cap V_{t_C} \times V_{t_C}$. Obviously, $\mathcal{H}'$ is a homomorphism from the root x_C of t_C to x. Hence, $x \in C^\mathcal{I}$.

(2 ⇒ 1) Let $\mathcal{I}$ be an interpretation such that $x \in C^{\mathcal{I}} \setminus D^{\mathcal{I}}$. By Lemma 17, there exists a homomorphism $\mathcal{H}$ from the root x_C of t_C to x. For the i-th transitive r_j^+-edge of t_C, $(v, r_j^+, v') \in E_{t_C}$, let $(v, y) \in \mathcal{H}$ and $(v', y') \in \mathcal{H}$. We define the component u_i^j of U to be the length of the shortest r-chain from y to y'. Now $\mathcal{H}$ can be extended in an obvious way to become a homomorphism $\mathcal{H}'$ from the root $x_{C[U]}$ of $t_{C[U]}$ to x. Again by Lemma 17, $x \in C[U]^{\mathcal{I}}$ and consequently $C[U] \not\sqsubseteq D$. □

The following lemma is a consequence of the previous lemma together with Lemma 17.

Lemma 19 *Let C, D be $\mathcal{EL}^+$-concept descriptions. Then, the following are equivalent:*

1. $C \sqsubseteq D$
2. $x_{C[U]} \in D^{t_C[U]}$ *for all* $U \in \mathbb{U}_C$

Proof. (1 ⇒ 2) By the previous lemma, for all $U \in \mathbb{U}_C$, $C[U] \sqsubseteq D$. Since $x_{C[U]} \in C^{t_{C[U]}}$, $x_{C[U]} \in D^{t_{C[U]}}$.

(2 ⇒ 1) We show the contrapositive and assume $C \not\sqsubseteq D$. By the previous lemma, there exists an expansion vector $U \in \mathbb{U}_C$ such that $C[U] \not\sqsubseteq D$ and by lemma 17, there does not exist a homomorphism from x_D to $x_{C[U]}$. Hence, $x_{C[U]} \notin D^{t_{C[U]}}$. □

Thus, in order to show $C \not\sqsubseteq D$, it suffices to search all expansion vectors of C for a canonical interpretation witnessing non-subsumption. However, there are infinitely many expansion vectors. For that reason, we now show that it suffices to only expand each transitive existential restriction in C to at most $rdepth(D) + 1$.

Lemma 20 *Let C, D be $\mathcal{EL}^+$-concept descriptions such that $C \not\sqsubseteq D$, and let $n := rdepth(D) + 1$. Then, there exists an expansion vector $W \in \mathbb{U}_C^n$ such that $C[W] \not\sqsubseteq D$.*

Proof. By the previous lemma, there exists an expansion vector

$$U = (u_1^1, \ldots, u_{k_1}^1, \ldots, u_1^m, \ldots, u_{k_m}^m)$$

of t_C such that $x_{C[U]} \notin D^{t_C[U]}$. Now define

$$\begin{aligned} W \quad := \quad & (\min(u_1^1, n), \ldots, \min(u_{k_1}^1, n), \ldots, \\ & \min(u_1^m, n), \ldots, \min(u_{k_m}^m, n)) \end{aligned}$$

We show that $x_{C[W]} \in D^{t_C[W]}$ implies $x_{C[U]} \in D^{t_C[U]}$, an obvious contradiction.

By Lemma 17, there exists a homomorphism $\mathcal{H}$ from x_D to $x_{C[W]}$. We show how $\mathcal{H}$ can be modified to become a homomorphism $\mathcal{H}'$ from x_D to $x_{C[U]}$. Let $E' \subseteq E_{t_D}$ be set the of transitive edges (v, r^+, v') whose expansion differs in $t_{C[W]}$ and $t_{C[U]}$. In order to have proper naming of the nodes along expansions of transitive edges, w.l.o.g. we assume that $e = (v, r^+, v') \in E'$ is expanded in $t_C[W]$ by an r-chain of length n as

$$v = v_1 \xrightarrow{r} \ldots \xrightarrow{r} v_{n+1} = v',$$

and in $t_C[U]$ by an r-chain of length $u > n$ as

$$v = v_1 \xrightarrow{r} \ldots \xrightarrow{r} v_{n+1} \xrightarrow{r} \ldots \xrightarrow{r} v_{u+1} = v'.$$

Let $V_e := \{v_2, \ldots, v_n\}$. We define the set W_e of all paths of maximal length of t_D that are affected by the expansion of e, where $N_R^+ := N_R \cup \{r^+ \mid r \in N_R\}$:

$$\begin{aligned} W_e \;:= \{w_1 \xrightarrow{r_1} \ldots \xrightarrow{r_k} w_{k+1} \mid \quad & \forall 1 \le i \le k+1. \mathcal{H}(w_i) \in V_e \wedge \\ & \forall w' \in V_{t_D}, r \in N_R^+.((w', r, w_1) \notin E_{t_D} \vee \\ & \qquad (w', r, w_1) \in E_{t_D} \rightarrow \mathcal{H}(w') \notin V_e) \wedge \\ & \forall w' \in V_{t_D}, r \in N_R^+.((w_{k+1}, r, w') \notin E_{t_D} \vee \\ & \qquad (w_{k+1}, r, w') \in E_{t_D} \rightarrow \mathcal{H}(w') \notin V_e)\} \end{aligned}$$

Let us now define $\mathcal{H}'$. For all edges $e \in E'$, $p = w_1 \xrightarrow{r_1} \ldots \xrightarrow{r_k} w_{k+1} \in W_e$, let $w_i \in p, 1 \le k+1$. We define $\mathcal{H}'(w_i)$ by distinguishing three cases. Let u be the length that e is expanded to in $t_C[U]$.

(a) If w_1 has an incoming r^+-edge, we define $\mathcal{H}'(w_i) := v_{q+u-n}$, where $\mathcal{H}(w_i) = v_q, 2 \le q \le n$.

(b) If (a) does not apply and there is a smallest $j \le k$ such that $r_j = r^+$ and $i > j$, $\mathcal{H}'(w_i) := v_{q+u-n}$, where $\mathcal{H}(w_i) = v_q, 2 \le q \le n$.

(c) If (a) and (b) do not apply, $\mathcal{H}'(w_i) := \mathcal{H}(w_i)$

For all other nodes $w \in V_{t_D}$, that are not affected by the different expansions of transitive edges in $t_C[W]$ and $t_C[U]$, $\mathcal{H}'(w) := \mathcal{H}(w)$.

We claim that $\mathcal{H}'$ is a homomorphism from x_D to $x_{C[U]}$. Let $r_1, \ldots, r_m \in N_R^+$ be the role names occurring in t_D and for $1 \le i \le m$

$$\begin{aligned} r_i^{t_D} \;&:=\; \{(w, w') \mid (w, r, w') \in E_{t_D}\} \\ R^* \;&:=\; \bigcup_{\substack{i \ge 0 \\ 1 \le j \le m}} (r_j^{t_D})^i. \end{aligned}$$

For a node $w \in V_{t_D}$, $R^*(w)$ consists of all nodes of the subtree starting at w. For all nodes $w \in V_{t_D}$, we prove by induction on *depth*(w) the following statement:

$$\mathcal{H}'_w := \mathcal{H}' \cap (R^*(w) \times R^*(w)) \text{ is a homomorphism from } w \text{ to } \mathcal{H}'(w) \tag{4.1}$$

For the induction base case, let $depth(w) = 0$. If $\mathcal{H}'_w(w) = \mathcal{H}(w)$, then $\mathcal{H}'_w$ is obviously a homomorphism from w to $\mathcal{H}'_w(w)$. Otherwise, let e be the transitive edge such that there is a $p \in W_e$ and $\mathcal{H}(w) = v_i \in p$, and let $u > n$ be the length that e is expanded in $t_C[U]$. Hence, $\mathcal{H}'_w(w) = v_{i+u-n}$. We have $\ell_{t_D}(w) = \emptyset$, since $\ell_{t_C[W]}(v_i) = \emptyset$ and thus $\mathcal{H}'_w$ is a homomorphism from w to v_{i+u-n}.

For the induction step, let us exemplarily consider the case $\mathcal{H}'_w(w) = \mathcal{H}(w) = v$. Clearly, (h1) holds. For proving (h2), let $(w, r, w') \in E_{t_D}$. Thus, w' has no incoming r^+-edge and consequently, $\mathcal{H}'_w(w') = \mathcal{H}(w')$ and $(\mathcal{H}'_w(w), r, \mathcal{H}'_w(w')) \in E_{t_{C[U]}}$. By 4.1 and the induction hypothesis, we have that $\mathcal{H}'_{w'}$ is a homomorphism from w' to the subtree starting at $\mathcal{H}'_{w'}(w')$. Moreover, $\mathcal{H}'_{w'} \subseteq \mathcal{H}'_w$. Now for the r^+-successors, let $(w, r^+, w') \in E_{t_D}$. In case $\mathcal{H}'_w(w') \neq \mathcal{H}(w')$, let e be the transitive edge such that there is a path $p \in W_e$ and $\mathcal{H}(w') = v_i \in p$, and let $u > n$ be the length that e is expanded in $t_{C[U]}$. By the construction, $\mathcal{H}'_w(w') = v_{i+u-n}$ and there is an r-chain between v_i and v_{i+u-n} in $t_{C[U]}$. Hence, (h3) holds for $\mathcal{H}'_w$. Moreover, $\mathcal{H}'_{w'}$ is a homomorphism from w' to the subtree starting at $\mathcal{H}'_w(w')$ and $\mathcal{H}'_{w'} \subseteq \mathcal{H}'_w$. Summing up, if $\mathcal{H}'_w(w) = \mathcal{H}(w)$, then $\mathcal{H}'_w$ is a homomorphism from w to $\mathcal{H}'(w)$, since the homomorphism conditions are fulfilled for w and all nodes of subtrees of successors of w. The last case, when $\mathcal{H}'_w(w) \neq \mathcal{H}(w)$, can be handled similarly to the previous case. One can exploit the fact that the relevant r- and r^+-successors of w are also shifted in the image of $\mathcal{H}'_w$ along r-chains in $t_{C[U]}$.

Since $\mathcal{H}(x_D) = x_{C[W]} = x_{C[U]} = \mathcal{H}'_{x_D}(x_D)$, we thus have that $\mathcal{H}' = \mathcal{H}'_{x_D}$ is a homomorphism from x_D to $x_{C[U]}$ and hence $C[U] \sqsubseteq D$. □

We are now prepared to prove the main lemma of this subsection from which we can immediately derive an algorithm for deciding non-subsumption in $\mathcal{EL}^+$ that runs in non-deterministic polynomial time.

Lemma 21 *Let C, D be $\mathcal{EL}^+$-concept descriptions, $n := rdepth(D) + 1$ and $m := rdepth(C) \cdot n$. Then the following are equivalent:*

1. *$C \not\sqsubseteq D$*
2. *There is $U \in \mathbb{U}^n_C$ and for all $W \in \mathbb{U}^m_D$, $C[U] \not\sqsubseteq D[W]$*

Proof. ($1 \Rightarrow 2$) By the previous lemma, we have that there exists an expansion vector $U \in \mathbb{U}^n_C$ such that $C[U] \not\sqsubseteq D$. Thus, in particular $C[U] \not\sqsubseteq D[W]$ for $W \in \mathbb{U}^m_D$.

($2 \Rightarrow 1$) We show the contrapositive and assume $C \sqsubseteq D$. For every expansion vector $U \in \mathbb{U}^n_{t_C}$, we obviously have

$$rdepth(C[U]) \leq rdepth(C) \cdot n.$$

By Lemma 19, we have $x_{C[U]} \in D^{t_C[U]}$ for all expansion vectors $U \in \mathbb{U}^n_C$. Thus, there exists a homomorphism $\mathcal{H}$ from x_D to $x_{C[U]}$. For every $(v, r^+, v') \in E_{t_D}$ and

Algorithm 5 $\mathcal{EL}^+$-subsumption algorithm

Input: $\mathcal{EL}^+$-concept descriptions C, D
  $n := rdepth(D) + 1$
  non-deterministically guess $U \in \mathbb{U}_C^n$ such that $C[U] \not\sqsubseteq D$
  return $C \not\sqsubseteq D$

$(v, w), (v', w') \in \mathcal{H}$, the length of the r-chain between w and w' is less or equal than m, since t_C is a tree not deeper than m. Thus, we can find an expansion vector $W \in \mathbb{U}_W^m$ as shown in the proof of Lemma 18 such that $x_{C[U]} \in D[W]^{t_{C[U]}}$. Hence, $C[U] \sqsubseteq D[W]$. □

Algorithm 5, which decides non-subsumption between $\mathcal{EL}^+$-concept descriptions, can directly be derived from the previous lemma. Recall, that checking for subsumption between $\mathcal{EL}$- and $\mathcal{EL}^+$-concept descriptions can be done in polynomial time. Thus, it is obvious that the algorithm runs in non-deterministic polynomial time. Its soundness and completeness follow from the previous lemma.

Theorem 11 *Subsumption in $\mathcal{EL}^+$ is* co-NP*-complete.*

Subsumption in $\mathcal{EL}^+$ w.r.t. TBoxes

Let us now examine the complexity of subsumption in $\mathcal{EL}^+$ w.r.t. acyclic and cyclic TBoxes. For the lower bound, we reduce concept *satisfiability* in $\mathcal{ELC}$ w.r.t. TBoxes to non-subsumption in $\mathcal{EL}^+$ w.r.t. TBoxes. Let $\mathcal{T} = \{A_1 \equiv C_1, \ldots, A_n \equiv C_n\}$ be an $\mathcal{ELC}$-TBox in normal form. Again, we assume that only *one* role name r occurs in $\mathcal{T}$. In order to obtain an $\mathcal{EL}^+$-TBox $\mathcal{T}'$ corresponding to $\mathcal{T}$, we

- introduce fresh concept names $P_1, \ldots, P_n$,
- replace every existential restriction $A_i \equiv \exists r.A_j$ by
$$A_i \equiv \exists r.(A_j \sqcap \bigsqcap_{1 \le i \le n} \exists s^+.P_i)$$
- replace every $A_i \equiv \neg A_j$ by $A_i \equiv \exists s.\exists s^+.P_j$.

Basically, we do the same thing as in the $\mathcal{ELU}$ case. The extension ensures that at the relevant points of connected models, $\exists s^+.P_i$ holds for every $1 \le i \le n$, except for the root node which will be treated analogously. Our intention is that $\exists s.\exists s^+.P_i$ holds at some point of an interpretation iff A_i does not hold at the respective point. Therefore, we introduce a concept M that signals points of models that contradict this intention.

Lemma 22 *Let $\mathcal{T} = \{A_1 \equiv C_1, \ldots, A_n \equiv C_n\}$ be an $\mathcal{ELC}$-TBox in normal form, A a defined concept in $\mathcal{T}$ and $\mathcal{T}'$ the $\mathcal{EL}^+$-TBox corresponding to $\mathcal{T}$. Furthermore, let*

$$\begin{aligned} F &:= \bigsqcap_{1 \le i \le n} \exists s^+.P_i \\ M &:= \bigsqcup_{1 \le i \le n} \exists r^*.(\exists s.P_i \sqcap \exists s.\exists s^+.P_i) \sqcup \bigsqcup_{1 \le i \le n} \exists r^*.(A_i \sqcap \exists s.\exists s^+.P_i) \end{aligned}$$

Then, the following are equivalent:

1. *A is satisfiable w.r.t. $\mathcal{T}$*
2. *$A \sqcap F \not\sqsubseteq_{\mathcal{T}'} M$*

Proof. ($1 \Rightarrow 2$) Let $\mathcal{I}$ be a model of $\mathcal{T}$ such that $x \in A^{\mathcal{I}}$. We extend $\mathcal{I}$ to obtain a model $\mathcal{I}'$ of $\mathcal{T}'$. For every $y \in A_i^{\mathcal{I}}, 1 \le i \le n$, we add some fresh y_1 to $\Delta^{\mathcal{I}'}$, (y, y_1) to $s^{\mathcal{I}'}$ and y_1 to $P_i^{\mathcal{I}'}$. Likewise for every $y \notin A_i^{\mathcal{I}}$, we add some fresh y_1, y_2 to $\Delta^{\mathcal{I}'}$, (y, y_1) and (y_1, y_2) to $s^{\mathcal{I}'}$, and y_2 to $P_i^{\mathcal{I}'}$. We thus have $y \notin A_i^{\mathcal{I}'}$ iff $y \in (\exists s.\exists s.P_i)^{\mathcal{I}'}$ and $\mathcal{I}'$ is a model of $\mathcal{T}'$. Clearly, $x \in (A \sqcap F)^{\mathcal{I}'}$. On the other hand, $x \notin M^{\mathcal{I}'}$. On all points reachable via an r-chain starting from x, the construction of $\mathcal{I}'$ ensures that on any s-branch, P_i occurs either on the first node of the branch or at some deeper node.

($2 \Rightarrow 1$) Let $\mathcal{I}$ be a model of $\mathcal{T}$ and $x \in (A \sqcap F)^{\mathcal{I}} \setminus M^{\mathcal{I}}$. In order to obtain a model $\mathcal{I}'$ of $\mathcal{T}$, we need to strip $\mathcal{I}$. For a relation $R \subseteq M \times M$, we redefine $R^1 := R^0 \cup R$ and set

$$\begin{aligned} \Delta^{\mathcal{I}'_i} &:= (r^{\mathcal{I}})^i(x) \cap F^{\mathcal{I}} \\ r^{\mathcal{I}'_i} &:= r^{\mathcal{I}} \cap (\Delta^{\mathcal{I}'_i} \times \Delta^{\mathcal{I}'_i}) \\ A^{\mathcal{I}'_i} &:= A^{\mathcal{I}} \cap \Delta^{\mathcal{I}'_i}, A \in N_C. \end{aligned}$$

Then,

$$\Delta^{\mathcal{I}'} := \bigcup_{i \ge 0} \Delta^{\mathcal{I}'_i}, \quad r^{\mathcal{I}'} := \bigcup_{i \ge 0} r^{\mathcal{I}'_i}, \quad A_j^{\mathcal{I}'} := \bigcup_{i \ge 0} A_j^{\mathcal{I}'_i}, 1 \le j \le n.$$

In order to prove that $\mathcal{I}'$ is a model of $\mathcal{T}$, we prove by induction on m the following statement, where $A_j \equiv C_j \in \mathcal{T}, 1 \le j \le n$:

$$A_j^{\mathcal{I}'_m} \subseteq C_j^{\mathcal{I}'_{m+1}}$$

It suffices to only consider the case $C_j = \exists r.B$ and $C_j = \neg A_i$ in the induction step. First of all, it is easily seen that $\Delta^{\mathcal{I}'_0} \subseteq \Delta^{\mathcal{I}'_1} \subseteq \ldots$. Let $C_j = \exists r.B$ and $y \in A_j^{\mathcal{I}'_{m+1}} \setminus A_j^{\mathcal{I}'_m}$. Since $y \in (\exists r.(B \sqcap F))^{\mathcal{I}}$, there is a z such that $(y, z) \in r^{\mathcal{I}}$ and $z \in (B \sqcap F)^{\mathcal{I}}$. Consequently, $z \in \Delta^{\mathcal{I}'_{m+2}}$ and $(y, z) \in r^{\mathcal{I}'_{m+2}}$. Hence, $y \in (\exists r.B)^{\mathcal{I}'_{m+2}}$. For the other case, let $C_j = \neg A_i$ and $y \in A_j^{\mathcal{I}'_{m+1}}$. We have $y \in (\exists s.\exists s^+.P_i)^{\mathcal{I}}$ and since $y \in F^{\mathcal{I}} \setminus M^{\mathcal{I}}$, $x \notin A_i^{\mathcal{I}'_{m+2}}$. It is not hard to see that $C^{\mathcal{I}'} \subseteq A^{\mathcal{I}'}$ for all $A \equiv C \in \mathcal{T}$. Thus, we have that $\mathcal{I}'$ is a model of $\mathcal{T}$ and $x \in A^{\mathcal{I}'}$. □

The lemma gives us a PSpace lower bound for non-subsumption in $\mathcal{EL}^+$ w.r.t. to acyclic TBoxes, and an ExpTime lower bound for non-subsumption in $\mathcal{EL}^+$ w.r.t. cyclic TBoxes. The ExpTime upper bound for subsumption in $\mathcal{EL}^+$ w.r.t. cyclic TBoxes comes from $\mathcal{EL}^+$ being a notational fragment of $\mathcal{ALC}_{\text{reg}}$.

Theorem 12 *Subsumption in $\mathcal{EL}^+$ is* ExpTime*-complete w.r.t. cyclic TBoxes.*

However, subsumption in $\mathcal{ALC}_{\text{reg}}$ is also ExpTime-complete w.r.t. acyclic TBoxes. In order to obtain a PSpace-upper bound, we reduce subsumption in $\mathcal{EL}^+$ w.r.t. acyclic TBoxes to subsumption in $\mathcal{ELU}$ w.r.t. acyclic TBoxes, whereof we already know that it is PSpace-complete. We exploit the fact that we can "simulate" bounded expansion vectors in $\mathcal{ELU}$. For instance, let $C = \exists r^+.P$ be an $\mathcal{EL}^+$-concept description, $n > 0$ and $\mathcal{I}$ an interpretation. Then, there exists an expansion vector $(i) \in U_C^n$ such that $x \in C[(i)]^\mathcal{I}$ iff

$$x \in (\exists r.P \sqcup \ldots \sqcup \underbrace{\exists r.\ldots\exists r}_{n\text{ times}}.P)^\mathcal{I}$$

Now it is kind of obvious what to do for the reduction. Basically, for a given TBox, we replace all transitive existential restrictions by an appropriate number of disjunctions. Since these disjunctions "simulate" expansion vectors, we then unfold the desired concept definitions and use Lemma 21 to establish the connection between $\mathcal{EL}^+$ and $\mathcal{ELU}$.

Firstly, we introduce a normal form for $\mathcal{EL}^+$-TBoxes.

Definition 24 An $\mathcal{EL}^+$-TBox $\mathcal{T}$ is in *normal form* iff for each $A \equiv C \in \mathcal{T}$, C is either of the form $\top$ or P; $\exists r.B$; $\exists r^+.B$; or $B_1 \sqcap B_2$, for $P \in N_C$ and $B, B_1, B_2 \in N_{def}(\mathcal{T})$. $\diamondsuit$

Using a similar normalization algorithm as in Section 2.4.2, it is not hard to see that every $\mathcal{EL}^+$-TBox $\mathcal{T}$ can be transformed in polynomial time into an equivalent $\mathcal{EL}^+$-TBox in normal form, that is linear in the size of $\mathcal{T}$.

Let $\mathcal{T} = \{A_1 \equiv C_1, \ldots, A_k \equiv C_k\}$ be an acyclic $\mathcal{EL}^+$-TBox in normal form. For $A, B \in N_{def}(\mathcal{T})$, let $n := \mathit{rdepth}_\mathcal{T}(B) + 1$ and $m := \mathit{rdepth}_\mathcal{T}(A) \cdot n$. Let $\mathcal{T}_1$ be the TBox obtained from $\mathcal{T}$ by

- replacing every $\exists r^+.A'$ by

$$\bigsqcup_{1 \leq i \leq n} \underbrace{\exists r.\ldots\exists r}_{i\text{ times}}.A'$$

- renaming every defined concept $A \in N_{def}(\mathcal{T})$ to A_1

Similarly, let $\mathcal{T}_2$ be the TBox obtained from $\mathcal{T}$ by

- replacing every $\exists r^+.A'$ by

$$\bigsqcup_{1 \leq i \leq m} \underbrace{\exists r.\ldots\exists r}_{i\text{ times}}.A'$$

- renaming every defined concept $A \in N_{def}(\mathcal{T})$ to A_2

Lemma 23 *Let $\mathcal{T}$ be an $\mathcal{EL}^+$-TBox in normal form, $A, B \in N_{def}(\mathcal{T})$ and $\mathcal{T}' := \mathcal{T}_1 \cup \mathcal{T}_2$ be the $\mathcal{ELU}$-TBox obtained from $\mathcal{T}$ as shown above. Then, the following are equivalent:*

1. $A \sqsubseteq_{\mathcal{T}} B$
2. $A_1 \sqsubseteq_{\mathcal{T}'} B_2$

Proof. We show the contrapositive in both directions.
(1 ⇒ 2) We have $A_1 \not\sqsubseteq_{\mathcal{T}'} B_2$. Thus, for the unfoldings $\hat{A}_1$ and $\hat{B}_2$ of A_1 and B_2 w.r.t. $\mathcal{T}'$, we have $\hat{A}_1 \not\sqsubseteq \hat{B}_2$. Let $\mathcal{I}$ be an interpretation such that $x \in \hat{A}_1^{\mathcal{I}} \setminus \hat{B}_2^{\mathcal{I}}$. Since in $\hat{A}_1$ we have replaced transitive existential restrictions by n disjunctions, it is not hard to see that there is an expansion vector $U \in \mathbb{U}^n_{\hat{A}}$ such that $x \in \hat{A}[U]^{\mathcal{I}}$ for the unfolding $\hat{A}$ of A w.r.t. $\mathcal{T}$. Likewise, for all expansion vectors $W \in \mathbb{U}^m_{\hat{B}}$, $x \notin \hat{B}[W]^{\mathcal{I}}$ for the unfolding $\hat{B}$ of B w.r.t. $\mathcal{T}$. By Lemma 21, $\hat{A} \not\sqsubseteq \hat{B}$ and hence $A \not\sqsubseteq_{\mathcal{T}} B$.
(2 ⇒ 1) We have $A \not\sqsubseteq_{\mathcal{T}} B$. Thus, for the unfoldings $\hat{A}$ and $\hat{B}$ of A and B w.r.t. $\mathcal{T}$, $\hat{A} \not\sqsubseteq \hat{B}$. By Lemma 21, there exists an expansion vector $U \in \mathbb{U}^n_{\hat{A}}$ such that for all expansion vectors $W \in \mathbb{U}^m_{\hat{B}}$, $\hat{A}[U] \not\sqsubseteq \hat{B}[W]$. Thus, for the unfoldings $\hat{A}_1$ and $\hat{B}_2$ of A_1 and B_2 w.r.t. $\mathcal{T}'$, $\hat{A}_1 \not\sqsubseteq \hat{B}_2$. Hence, $A_1 \not\sqsubseteq_{\mathcal{T}'} B_2$. □

Theorem 13 *Subsumption in $\mathcal{EL}^+$ w.r.t. acyclic TBoxes is* PSpace*-complete.*

4.6 Functionality

Functionality restricts role names to be interpreted as partial functions. We start having a look at the case of global functionality and then continue with local functionality.

Global Functionality

Let $\mathcal{EL}^f$ be $\mathcal{EL}$ extended by global functionality, i.e., $\mathcal{EL}^f$ has the same syntax as $\mathcal{EL}$, but additionally restricts role names to be interpreted as partial functions. It has been shown in (Baader, Brandt & Lutz 2005b) that $\mathcal{EL}^f$ is closely related to the description logic $\mathcal{FL}_0^{tf}$, which enforces role names to be interpreted as total functions in interpretations of $\mathcal{FL}_0$. The latter allows for value restriction, conjunction and top only. Using Lemma 23 from (Baader, Brandt & Lutz 2005b), we obtain the following proposition.

Proposition 7 *Subsumption in $\mathcal{FL}_0^{tf}$ is in* PTime*,* co-NP*-complete w.r.t. acyclic TBoxes, and* PSpace*-complete w.r.t. cyclic TBoxes.*

The connection between $\mathcal{EL}^f$ and $\mathcal{FL}_0^{tf}$ is quite obvious. The requirement that role names have to be interpreted as partial *functions* in $\mathcal{EL}^f$ makes $\exists$-quantifiers "act" like $\forall$-quantifiers. Basically, an $\mathcal{EL}^f$-concept description can be translated into an $\mathcal{FL}_0^{tf}$-concept description by just replacing the $\exists$-quantifiers by $\forall$-quantifiers. However, the top-concept requires a special treatment. In order to reduce subsumption in $\mathcal{FL}_0^{tf}$ to $\mathcal{EL}^f$ and vice versa, we define how to convert TBoxes appropriately.

It is not hard to see that any $\mathcal{FL}_0^{tf}$-TBox can be transformed into a TBox in normal form that is linear in their size, similar to the normal form of $\mathcal{EL}$-TBoxes presented in Section 2.4.2.

Definition 25 An $\mathcal{FL}_0^{tf}$-TBox $\mathcal{T}$ is in *normal form* iff for each concept definition $A \equiv C \in \mathcal{T}$, C is of the form P, $\exists r.B$ or $B_1 \sqcap B_2$ for $P \in N_C$ and $B, B_1, B_2 \in N_{def}(\mathcal{T})$. $\Diamond$

Note, that no top-concept occurs in a TBox in normal form. That is justified by the facts that in $\mathcal{FL}_0^{tf}$, $\forall r.\top \equiv \top$ and $C \sqcap \top \equiv C$.

Let $\mathcal{T}$ be an $\mathcal{FL}_0^{tf}$-TBox in normal form. The $\mathcal{EL}^f$-TBox $\mathcal{T}'$ corresponding to $\mathcal{T}$ is obtained from $\mathcal{T}$ by replacing every concept definition $A \equiv \forall r.B \in \mathcal{T}$ by $A \equiv \exists r.B$.

Lemma 24 *Let $\mathcal{T}$ be an $\mathcal{FL}_0^{tf}$-TBox in normal form and $\mathcal{T}'$ the $\mathcal{EL}^f$-TBox corresponding to $\mathcal{T}$. Then, the following are equivalent:*

1. $A \sqsubseteq_{\mathcal{T}} B$
2. $A \sqsubseteq_{\mathcal{T}'} B$

Proof. ($1 \Rightarrow 2$) We show $x \in (\forall r.B)^{\mathcal{I}}$ implies $x \in (\exists r.B)^{\mathcal{I}}$ for all models of $\mathcal{T}$, and thus every model $\mathcal{I}$ of $\mathcal{T}$ is also a model of $\mathcal{T}'$. So, let $x \in (\forall r.B)^{\mathcal{I}}$. Because r is interpreted in $\mathcal{I}$ as a total function, there is $y \in B^{\mathcal{I}}$ and $(x, y) \in r^{\mathcal{I}}$. Consequently, $x \in (\exists r.B)^{\mathcal{I}}$.

($2 \Rightarrow 1$) Let $\mathcal{I}$ be a model of $\mathcal{T}'$. Let $\mathcal{I}'$ be obtained from $\mathcal{I}$ by adding one distinct y to the interpretation domain and, for all role names r occurring in $\mathcal{T}$, adding (y, y) to $r^{\mathcal{I}'}$, and (x, y) to $r^{\mathcal{I}'}$ iff $x \in \Delta^{\mathcal{I}}$ has no r-successor in $\mathcal{I}$. Now $\mathcal{I}'$ is also a model of $\mathcal{T}'$, since no top-concept occurs in $\mathcal{T}'$. Moreover, all role names are interpreted in $\mathcal{I}'$ as total functions and $x \in (\exists r.B)^{\mathcal{I}'}$ iff $x \in (\forall r.B)^{\mathcal{I}'}$. Hence, $\mathcal{I}'$ is also a model a model of $\mathcal{T}$. $\square$

For the reduction in the other direction, let $\mathcal{T}$ be an $\mathcal{EL}^f$-TBox in normal form and $\mathcal{T}'$ be the $\mathcal{FL}_0^{tf}$-TBox corresponding to $\mathcal{T}$. It is obtained from $\mathcal{T}$ by replacing every $A \equiv \exists r.B \in \mathcal{T}$ by $A \equiv \forall r.(P \sqcap B)$, where P is a fresh concept name.

Lemma 25 *Let $\mathcal{T}$ be an $\mathcal{EL}^f$-TBox in normal form and $\mathcal{T}'$ be the $\mathcal{FL}_0^{tf}$-TBox corresponding to $\mathcal{T}$. Then, the following are equivalent:*

1. $A \sqsubseteq_{\mathcal{T}} B$
2. $A \sqsubseteq_{\mathcal{T}'} B$

Proof. (1 ⇒ 2) Let $\mathcal{I}$ be a model of $\mathcal{T}$ and let $\mathcal{I}'$ be obtained from $\mathcal{I}$ by setting $\Delta^{\mathcal{I}'} := \Delta^{\mathcal{I}} \dot\cup \{y\}$, interpreting $P^{\mathcal{I}'} := \Delta^{\mathcal{I}}$,adding (x, y) to $r^{\mathcal{I}'}$ iff x has no r-successor in $\mathcal{I}$ and (y, y) to $r^{\mathcal{I}'}$ for all role names r occurring in $\mathcal{T}$. For all $A \equiv \top \in \mathcal{T}$, we moreover set $A^{\mathcal{I}'} := A^{\mathcal{I}} \cup \{y\}$. We have $y \in (\exists r.B)^{\mathcal{I}}$ iff $y \in (\forall r.(P \sqcap B))^{\mathcal{I}'}$ and $\mathcal{I}'$ is a model of $\mathcal{T}'$.

(2 ⇒ 1) We show the contrapositive. Let $\mathcal{I}$ be a model of $\mathcal{T}$ such that there is $x \in A^{\mathcal{I}} \setminus B^{\mathcal{I}}$. The interpretation $\mathcal{I}'$ is obtained from $\mathcal{I}$ by setting $\Delta^{\mathcal{I}'} := \Delta^{\mathcal{I}} \dot\cup \{y\}$, interpreting $P^{\mathcal{I}'} := \Delta^{\mathcal{I}}$, adding (x, y) to $r^{\mathcal{I}'}$ iff x has no r-successor in $\mathcal{I}$ and (y, y) to $r^{\mathcal{I}'}$ for all role names r occurring in $\mathcal{T}$. For all $A \equiv \top \in \mathcal{T}$, we moreover set $A^{\mathcal{I}'} := A^{\mathcal{I}} \cup \{y\}$. We have $y \in (\exists r.B)^{\mathcal{I}}$ iff $y \in (\forall r.(P \sqcap B))^{\mathcal{I}'}$ and $\mathcal{I}'$ is a model of $\mathcal{T}'$ witnessing $A \not\sqsubseteq_{\mathcal{T}} B$.□

Theorem 14 *Subsumption in $\mathcal{EL}^f$ is* CO-NP*-complete w.r.t. acyclic TBoxes and* PSPACE*-complete w.r.t. cyclic TBoxes.*

Usually, one is not interested in requiring *all* role names to be interpreted as partial functions in $\mathcal{EL}^f$. In this case, subsumption is CO-NP-hard w.r.t. acyclic TBoxes and PSPACE-hard w.r.t. cyclic TBoxes respectively. The reduction from subsumption in $\mathcal{FL}_0^{tf}$ w.r.t. TBoxes to subsumption in $\mathcal{EL}^f$ w.r.t. TBoxes where not all role names have to be interpreted by partial functions can be done in the same manner as above. However, it is unknown whether the lower bounds are tight. We address to known upper bounds and subsumption in $\mathcal{EL}^f$ in the following subsection.

Local Functionality

Let $\mathcal{ELF}$ be $\mathcal{EL}$ extended by the functionality concept constructor. This constructor allows for restricting role names to be locally interpreted as partial functions. In order to have an induction argument, we define for any role name $r \in N_R$, $rdepth(\leq 1r) := 0$. For the other cases, *rdepth* is defined as in the $\mathcal{EL}$ case.

Let us now consider subsumption in $\mathcal{ELF}$ without TBoxes. We define a function τ that translates $\mathcal{ELF}$-concept descriptions into $\mathcal{EL}$-concept descriptions and preserves subsumption relations. Informally speaking, what τ does is that it merges existential restrictions that are functionally constrained and replaces functionality restrictions by additional concept names. Let C, D be $\mathcal{ELF}$-concept descriptions. We define τ via induction on $rdepth(C)$. For $rdepth(C) = 0$, i.e., $C = P_1 \sqcap \ldots P_k \sqcap \leq 1s_1 \sqcap \leq 1s_m$, $\tau(C) := P_1 \sqcap \ldots \sqcap P_k \sqcap P_{s_1} \sqcap \ldots \sqcap P_{s_m}$.

For $rdepth(C) = n + 1$, we have that C is of the form

$$P_1 \sqcap \ldots \sqcap P_k \sqcap \exists r_1.C_1 \sqcap \ldots \sqcap \exists r_m.C_m \sqcap \leq 1s_1 \sqcap \ldots \sqcap \leq 1s_n$$

Let $I := \{s_1, \ldots, s_n\}$ and $N_R^C := \{r_1, \ldots, r_m\}$. We define

$$\tau(C) := P_1 \sqcap \ldots \sqcap P_k \sqcap \bigsqcap_{r_i \in N_R^C \setminus I} \exists r_i.\tau(C_i) \sqcap \bigsqcap_{r_i \in I \cap N_R^C} \exists r_i.\tau(\bigsqcap_{r_j = r_i} C_j) \sqcap \bigsqcap_{s_i \in I} P_{s_i}$$

The P_s are chosen such that they do not occur in C. Obviously, for all $\mathcal{ELF}$-concept descriptions C, $\tau(C)$ yields an $\mathcal{EL}$-concept description that is *linear* in the size of C.

Lemma 26 *Let C be an $\mathcal{ELF}$-concept description. Then, the following are equivalent:*

1. $C \sqsubseteq D$
2. $\tau(C) \sqsubseteq \tau(D)$

Proof. We show the contrapositive in both directions.

(2 $\Rightarrow$ 1) We prove this direction by induction on $max(rdepth(C), rdepth(D))$. For the induction step, let $max(rdepth(C), rdepth(D)) = o + 1$. We have

$$\begin{aligned} C &= P_1^C \sqcap \ldots \sqcap P_{k_C}^C \sqcap \exists r_1^C.C_1 \sqcap \ldots \sqcap \exists r_{m_C}^C.C_{m_C} \sqcap \leq 1 s_1^C \sqcap \ldots \sqcap \leq 1 s_{n_C}^C \\ D &= P_1^D \sqcap \ldots \sqcap P_{k_D}^D \sqcap \exists r_1^D.D_1 \sqcap \ldots \sqcap \exists r_{m_D}^D.D_{m_D} \sqcap \leq 1 s_1^D \sqcap \ldots \sqcap \leq 1 s_{n_D}^D \end{aligned}$$

and thus, for $I_C := \{s_1^C, \ldots, s_{n_C}^C\}$, $N_R^C := \{r_1^C, \ldots, r_{m_C}^C\}$, $I_D := \{s_1^D, \ldots, s_{n_C}^C\}$ and $N_R^D := \{r_1^D, \ldots, r_{m_D}^D\}$,

$$\tau(C) = P_1^C \sqcap \ldots \sqcap P_{k_C}^C \sqcap \bigsqcap_{r_i^C \in N_R^D \setminus I_C} \exists r_i^C.\tau(C_i) \sqcap \bigsqcap_{r_i^C \in N_R^C \cap I_C} \exists r_i^C.\tau(\bigsqcap_{r_j^C = r_i^C} C_j) \sqcap \bigsqcap_{s_i^C \in I_C} P_{s_i^C}$$

$$\tau(D) = P_1^D \sqcap \ldots \sqcap P_{k_D}^D \sqcap \bigsqcap_{r_i^D \in N_R^D \setminus I_D} \exists r_i^D.\tau(D_i) \sqcap \bigsqcap_{r_i^D \in N_R^D \cap I_D} \exists r_i^D.\tau(\bigsqcap_{r_j^D = r_i^D} D_j) \sqcap \bigsqcap_{s_i^D \in I_D} P_{s_i^D}$$

If there is a primitive concept name $P_i^D, 1 \leq i \leq k_D$ on the top-level of D not occurring on the top-level of C, then obviously $\tau(C) \not\sqsubseteq \tau(D)$. Likewise, if there is a functionality constraint $\leq 1 s_i^D, 1 \leq i \leq n_D$ on the top-level of D not occurring on the top-level of C, then $\tau(C) \not\sqsubseteq \tau(D)$. Lastly, if there is an existential restriction $\exists r_i^D.D_i, 1 \leq i \leq m_D$ on the top-level of D, such that $C \not\sqsubseteq \exists r_i^D.D_i$ we distinguish two cases:

(a) r_i^D is bound by a functionality restriction in C

(b) r_i^D is not bound by a functionality restriction in C

In case of (a), we have for $1 \leq j \leq m_C$:

$$\exists r_i^D. \bigsqcap_{r_j^C = r_i^D} C_j \quad \not\sqsubseteq \quad \exists r_i^D.D_i$$

By the induction hypothesis,

$$\tau(\bigsqcap_{r_j^C = r_i^D} C_j) \quad \not\sqsubseteq \quad \tau(D_i)$$

and hence $\tau(C) \not\sqsubseteq \tau(D)$. In case of (b), we have for every j such that $r_i^D = r_j^C$, $\exists r_j^C.C_j \not\sqsubseteq \exists r_i^D.D_i$. Again, by the induction hypothesis, $\tau(C_j) \not\sqsubseteq \tau(D_i)$ and hence $\tau(C) \not\sqsubseteq \tau(D)$.

($1 \Rightarrow 2$) For the induction step, let $max(rdepth(C), rdepth(D)) = o + 1$. We have

$$\begin{aligned} C &= P_1^C \sqcap \ldots \sqcap P_{k_C}^C \sqcap \exists r_1^C.C_1 \sqcap \ldots \sqcap \exists r_{m_C}^C.C_{m_C} \sqcap \leq 1 s_1^C \sqcap \ldots \sqcap \leq 1 s_{n_C}^C \\ D &= P_1^D \sqcap \ldots \sqcap P_{k_D}^D \sqcap \exists r_1^D.D_1 \sqcap \ldots \sqcap \exists r_{m_D}^D.D_{m_D} \sqcap \leq 1 s_1^D \sqcap \ldots \sqcap \leq 1 s_{n_D}^D \end{aligned}$$

and thus, for $I_C := \{s_1^C, \ldots, s_{n_C}^C\}$, $N_R^C := \{r_1^C, \ldots, r_{m_C}^C\}$, $I_D := \{s_1^D, \ldots, s_{n_C}^C\}$ and $N_R^D := \{r_1^D, \ldots, r_{m_D}^D\}$,

$$\tau(C) = P_1^C \sqcap \ldots \sqcap P_{k_C}^C \sqcap \bigsqcap_{r_i^C \in N_R^D \setminus I_C} \exists r_i^C.\tau(C_i) \sqcap \bigsqcap_{r_i^C \in N_R^C \cap I_C} \exists r_i^C.\tau(\bigsqcap_{r_j^C = r_i^C} C_j) \sqcap \bigsqcap_{s_i^C \in I_C} P_{s_i^C}$$

$$\tau(D) = P_1^D \sqcap \ldots \sqcap P_{k_D}^D \sqcap \bigsqcap_{r_i^D \in N_R^D \setminus I_D} \exists r_i^D.\tau(D_i) \sqcap \bigsqcap_{r_i^D \in N_R^D \cap I_D} \exists r_i^D.\tau(\bigsqcap_{r_j^D = r_i^D} D_j) \sqcap \bigsqcap_{s_i^D \in I_D} P_{s_i^D}$$

If there is a primitive concept name $P_i^D, 1 \leq i \leq k_D$ on the top-level of $\tau(D)$ not occurring on the top-level of $\tau(C)$, then obviously $C \not\sqsubseteq D$. Likewise, if there is a P_s on the top-level of $\tau(D)$ not occurring on the top-level of $\tau(C)$, then $\leq 1s$ occurs on the top-level of D, but not on the top-level of C. Hence, $C \not\sqsubseteq D$. Lastly, concerning the existential restriction, we exemplarily consider the most special case. The other cases follow similarly. Let $\exists r_i^D.\tau(\sqcap_{r_j^D = r_i^D} D_j), r_i^D \in N_R^D \cap I_D$ be an existential restriction on the top-level of $\tau(D)$ such that r_i^D is bound by a functional constraint in D and

$$\tau(C) \not\sqsubseteq \exists r_i^D.\tau(\bigsqcap_{r_j^D = r_i^D} D_j)$$

Exemplarily, assume $r_i^D \in N_R^C \cap I_C$. We then have

$$\exists r_i^D.\tau(\bigsqcap_{r_j^C = r_i^D} C_j) \not\sqsubseteq \exists r_i^D.\tau(\bigsqcap_{r_j^D = r_i^D} D_j)$$

By the induction hypothesis,

$$\bigsqcap_{r_j^C = r_i^D} C_j \not\sqsubseteq \bigsqcap_{r_j^D = r_i^D} D_j$$

Hence,

$$\bigsqcap_{r_j^C = r_i^D} \exists r_j^C.C_j \sqcap \leq 1 r_i^D \not\sqsubseteq \bigsqcap_{r_j^D = r_i^D} \exists r_j^D.D_j \sqcap \leq 1 r_i^D$$

Consequently, $C \not\sqsubseteq D$. □

Let us have a look at an example to actually see what τ does.

Example 6. Let C, D be defined as follows:

$$\begin{aligned} C &:= P_1 \sqcap \exists r.(P_2 \sqcap \exists s.\top) \sqcap \exists r.P_1 \sqcap \leq 1r \sqcap \leq 1s \\ D &:= \exists r.(P_1 \sqcap \exists s.\top) \sqcap \leq 1s \end{aligned}$$

Then,

$$\begin{aligned} \tau(C) &= P_1 \sqcap \exists r.(P_1 \sqcap P_2 \sqcap \exists s.\top) \sqcap P_r \sqcap P_s \\ \tau(D) &= \exists r.(P_1 \sqcap \exists s.\top) \sqcap P_s \end{aligned}$$

Obviously, both $C \sqsubseteq D$ and $\tau(C) \sqsubseteq \tau(D)$. ■

Theorem 15 *Subsumption in $\mathcal{ELF}$ is in* PTIME.

Concerning subsumption in $\mathcal{EL}^f$, given $\mathcal{EL}^f$-concept descriptions C, D, it is not hard to see how we translate C and D to $\mathcal{ELF}$-concept descriptions C' and D' such that $C \sqsubseteq D$ iff $C' \sqsubseteq D'$. Basically, in every *rdepth*-level of C and D, every role name r occurring in C or D has to be functionally constrained in C' with $\leq 1r$. This translation leads only to a polynomial blow-up in the size of C' and D'.

Theorem 16 *Subsumption in $\mathcal{EL}^f$ is in* PTIME.

However, we cannot employ the approach of merging existential restrictions and introducing additional primitive concept names for subsumption in $\mathcal{ELF}$ w.r.t. TBoxes, which turns out to be intractable. Basically, this comes from the possible exponential blow-up that would occur if we would merge existential restrictions that are functionally constrained in a concept definition.

In the following, we show CO-NP-hardness of subsumption in $\mathcal{ELF}$ w.r.t. acyclic TBoxes and reduce 3-SAT to non-subsumption in $\mathcal{ELF}$ w.r.t. acyclic TBoxes. Let ϕ be an instance of 3-SAT, containing k clauses over n propositional variables $p_1, \ldots, p_n$. The TBox $\mathcal{T}_\phi$ for the reduction is presented in Table 4.5. Firstly, we note that $\mathcal{T}_\phi$ is indeed acyclic and linear in the size of ϕ. The concept definitions $C_1^i, 1 \leq i \leq$ in $\mathcal{T}_\phi$ encode paths that represent those valuations for which the i-th clause of ϕ becomes false. A path $x_1 \xrightarrow{r_{i_1}} \ldots \xrightarrow{r_{i_n}} x_{n+1}, r_j \in \{0,1\}, 1 \leq j \leq n$ represents the valuation $V(p_j) := i_j, 1 \leq j \leq n$. If every valuation makes some clause of ϕ false, we have through the functionality constraints in each C_j^i that in every model $\mathcal{I}$ of $\mathcal{T}_\phi$, we can span a complete binary tree starting at $x \in C^\mathcal{I}$, and thus also $x \in L_1^\mathcal{I}$.

Lemma 27 *Let $\phi = \psi_1 \wedge \ldots \wedge \psi_k, \psi_j = \ell_1^j \vee \ell_2^j \vee \ell_3^j, 1 \leq j \leq k$ be an instance of 3-SAT and $\mathcal{T}_\phi$ the $\mathcal{ELF}$-TBox corresponding to ϕ. Then, the following are equivalent:*

1. *ϕ has a solution*
2. *$C \not\sqsubseteq_{\mathcal{T}_\phi} L_1$*

$$
\begin{array}{rcl}
C_i^j & \equiv & \begin{cases} \exists r_0.C_{i+1}^j \sqcap \leq 1 r_0 & \text{if } p_i \in \{\ell_1^j, \ell_2^j, \ell_3^j\} \\ \exists r_1.C_{i+1}^j \sqcap \leq 1 r_1 & \text{if } \neg p_i \in \{\ell_1^j, \ell_2^j, \ell_3^j\} \\ \exists r_0.C_{i+1}^j \sqcap \exists r_1.C_{i+1}^j \sqcap \leq 1 r_0 \sqcap \leq 1 r_1 & \text{otherwise} \end{cases} \\
C_{n+1}^j & \equiv & \top \\
C & \equiv & \bigsqcap_{1 \leq j \leq k} C_1^j \\
L_i & \equiv & \exists r_0.L_{i+1} \sqcap \exists r_1.L_{i+1} \\
L_{n+1} & \equiv & \top
\end{array}
$$

Table 4.5: The $\mathcal{ELF}$-TBox $\mathcal{T}_\phi$ for the reduction of an instance $\phi = \ell_1^1 \vee \ell_2^1 \vee \ell_3^1 \wedge \ldots \wedge \ell_1^k \vee \ell_2^k \vee \ell_3^k$ of 3-SAT over the propositional variables $p_1, \ldots, p_n$, where $1 \leq i \leq n$ and $1 \leq j \leq k$.

Proof. For the proof, let $\hat{C}$ and $\hat{L_1}$ be the unfolding of C and L_1 w.r.t. $\mathcal{T}_\phi$. Furthermore, by $\mathcal{J}$ we denote the interpretation corresponding to the $\mathcal{EL}$-description tree of $\tau(\hat{C})$ with root x. For a given valuation V of ϕ, we define

$$C_V := \exists r_{V(p_1)}.(\ldots(\exists r_{V(p_n)}.\top)\ldots)$$

(1 $\Rightarrow$ 2) There exists a valuation V such that V does not make any clause of ϕ false. Hence, $\mathcal{J}$ is an incomplete binary tree with root x, because there is no path $x \xrightarrow{r_{V(p_1)}} \ldots \xrightarrow{r_{V(p_n)}} y$. So because of $x \in \hat{C}^{\mathcal{J}}$ and $x \notin C_V^{\mathcal{J}}$, we have $C \not\sqsubseteq_{\mathcal{T}_\phi} C_V$. However, we obviously have $\hat{L_1} \sqsubseteq C_V$. Hence $C \not\sqsubseteq_{\mathcal{T}_\phi} L_1$.

(2 $\Rightarrow$ 1) We have $C \not\sqsubseteq_{\mathcal{T}_\phi} L_1$ and $\hat{C} \not\sqsubseteq \hat{L_1}$. Since $\hat{L_1}$ is an $\mathcal{EL}$-concept description, we have $\tau(\hat{L_1}) = \hat{L_1}$ and by Lemma 26, $\tau(\hat{C}) \not\sqsubseteq \hat{L_1}$. Consequently, $x \notin \hat{L_1}^{\mathcal{J}}$. The only reason for the latter fact is, that there are some $a_1, \ldots, a_n \in \{0, 1\}$ such that there is no path $x \xrightarrow{r_{a_1}} \ldots \xrightarrow{r_{a_n}} y$ in $\mathcal{J}$. Thus, $V(p_j) := a_j, 1 \leq j \leq n$ is a valuation that does not make any clause of ϕ false, and hence V is a solution to ϕ. □

Theorem 17 *Subsumption in $\mathcal{ELF}$ w.r.t. acyclic TBoxes is* co-NP*-hard.*

It is a justified question, why we have not made a reduction from subsumption in $\mathcal{EL}^f$ to subsumption in $\mathcal{ELF}$. Basically, the 3-SAT reduction hopefully gives a better intuition on why subsumption in $\mathcal{ELF}$ is co-NP-hard[1]. Nevertheless, for the case of

[1]Indeed, one could similarly prove co-NP-hardness of the language inclusion problem of acyclic finite state automata.

cyclic TBoxes, we make the easier reduction from subsumption in $\mathcal{EL}^f$ to subsumption in $\mathcal{ELF}$.

Lemma 28 *Let $\mathcal{T} = \{A_1 \equiv C_1, \ldots, A_n \equiv C_n\}$ be an $\mathcal{EL}^f$ TBox in normal form in which m different role names $r_1, \ldots, r_m$ occur. Let $\mathcal{T}'$ be the $\mathcal{ELF}$ TBox obtained from $\mathcal{T}$ by adding the concept definition*

$$F \equiv \bigsqcap_{1 \leq i \leq m} \leq 1 r_i$$

and replacing every $A \equiv \exists r.B \in \mathcal{T}$ by $A \equiv \exists r.(F \sqcap B)$. Then, for the defined concepts $A, B \in N_{def}(\mathcal{T})$, the following are equivalent:

1. *$A \sqsubseteq_{\mathcal{T}} B$*
2. *$A \sqcap F \sqsubseteq_{\mathcal{T}'} B$*

Proof. We show the contrapositive in both directions.

(1 ⇒ 2) Let $\mathcal{I}$ be a model of $\mathcal{T}'$ such that $x \in (A \sqcap F)^{\mathcal{I}} \setminus B^{\mathcal{I}}$. In order to obtain a model $\mathcal{I}'$ of $\mathcal{T}$, we need to strip $\mathcal{I}$. For a relation $R \subseteq M \times M$, we redefine $R^1 := R^0 \cup R$ and set

$$\begin{aligned} \Delta^{\mathcal{I}'_i} &:= (\bigcup_{1 \leq i \leq n} r_i^{\mathcal{I}})^i(x) \cap F^{\mathcal{I}} \\ r_j^{\mathcal{I}'_i} &:= r_j^{\mathcal{I}} \cap (\Delta^{\mathcal{I}'_i} \times \Delta^{\mathcal{I}'_i}), 1 \leq j \leq m \\ A^{\mathcal{I}'_i} &:= A^{\mathcal{I}} \cap \Delta^{\mathcal{I}'_i}, A \in N_C. \end{aligned}$$

Then,

$$\Delta^{\mathcal{I}'} := \bigcup_{i \geq 0} \Delta^{\mathcal{I}'_i}, \quad r_j^{\mathcal{I}'} := \bigcup_{i \geq 0} r_j^{\mathcal{I}'_i}, 1 \leq j \leq m, \quad A_j^{\mathcal{I}'} := \bigcup_{i \geq 0}, A_j^{\mathcal{I}'_i}, 1 \leq j \leq n.$$

In order to prove that $\mathcal{I}'$ is a model of $\mathcal{I}$, we prove by induction on m the following statement, where $A_j \equiv C_j \in \mathcal{T}, 1 \leq j \leq n$:

$$A_j^{\mathcal{I}'_m} \subseteq C_j^{\mathcal{I}'_{m+1}}$$

It suffices to only consider the case $C = \exists r.B$ in the induction step. First of all, it is easily seen that $\Delta^{\mathcal{I}'_0} \subseteq \Delta^{\mathcal{I}'_1} \subseteq \ldots$. Let $y \in A^{\mathcal{I}'_{m+1}} \setminus A^{\mathcal{I}'_m}$. Since $y \in (\exists r.(B \sqcap F))^{\mathcal{I}}$, there is a $z \in (B \sqcap F)^{\mathcal{I}}$ such that $(y, z) \in r^{\mathcal{I}}$. Consequently, $z \in \Delta^{\mathcal{I}'_{m+2}}$ and $(y, z) \in r^{\mathcal{I}'_{m+2}}$. Hence, $y \in (\exists r.B)^{\mathcal{I}'_{m+2}}$. It is not hard to see that $C^{\mathcal{I}'} \subseteq A^{\mathcal{I}'}$. Moreover, $\Delta^{\mathcal{I}'} \subseteq \Delta^{\mathcal{I}} \cap F^{\mathcal{I}}$. Thus, we have that $\mathcal{I}'$ is a model of $\mathcal{T}$ and $x \in A^{\mathcal{I}'} \setminus B^{\mathcal{I}'}$.

(2 ⇒ 1) This direction is trivial, since any model $\mathcal{I}$ of $\mathcal{T}$ witnessing $A \not\sqsubseteq_{\mathcal{T}} B$ is also a witness for $A \sqcap F \not\sqsubseteq_{\mathcal{T}'} B$, when we interpret $F^{\mathcal{I}} := \Delta^{\mathcal{I}}$. □

Theorem 18 *Subsumption in $\mathcal{ELF}$ w.r.t. cyclic TBoxes is* PSPACE*-hard.*

The upper bounds of the complexity of subsumption in $\mathcal{ELF}$ w.r.t. acyclic and cyclic TBoxes remain an open question of this thesis. The best known upper bounds come from $\mathcal{ELF}$ being a notational fragment of $\mathcal{ALCN}$. Thus, subsumption in $\mathcal{ELF}$ w.r.t. acyclic TBoxes is in PSPACE and in EXPTIME w.r.t. cyclic TBoxes (Baader et al. 2003). However, without going into further details, the author's speculation is that in both cases, the lower bounds are tight.

4.7 Concrete Domains with Abstract Feature Chains

Last but not least, we will examine concept-satisfiability and subsumption in $\mathcal{EL}$ extended by concrete domains with abstract feature chains. For the tractable extension $\mathcal{EL}^{\sqcup,\sqcap,\neg}(\mathcal{D})$ in Chapter 3, we did not allow abstract feature chains and in the following we will see the reason for that.

Firstly, we introduce a popular family of concrete domains, so-called *arithmetic concrete domains*, which have been introduced in (Lutz 2002). They allow to represent the natural numbers and operations thereon.

Definition 26 A concrete domain $\mathcal{D}$ is called *arithmetic* iff

1. $\Delta_{\mathcal{D}}$ contains the natural numbers and
2. $\Phi_{\mathcal{D}}$ contains
   - unary predicates for equality and inequality with zero
   - binary predicates for equality and inequality
   - a binary predicate expressing addition with 1, and
   - a ternary predicate expressing addition ◊

It has been shown by Lutz in (Lutz 2002) that subsumption and satisfiability in $\mathcal{EL}$ extended by arithmetic concrete domains become undecidable in the presence of general TBoxes[2]. We show in the following that these undecidability results carry over to satisfiability and subsumption in $\mathcal{EL}$ extended by arithmetic concrete domains w.r.t. cyclic TBoxes by a reduction of the Post Correspondence Problem.

Definition 27 (PCP) An *instance* P of the *Post Correspondence Problem* is given by a finite, non-empty list $(\ell_1, r_1), \ldots, (\ell_k, r_k)$ of pairs of words over some alphabet Σ. A sequence of integers $i_1, \ldots, i_m$ with $m \geq 1$ is a *solution to P* iff

$$\ell_{i_1} \ldots \ell_{i_m} \quad = \quad r_{i_1} \ldots r_{i_m}$$

[2]Strictly speaking, in (Lutz 2002) this has only been shown for $\mathcal{ALC}(\mathcal{D})$ with abstract feature chains and general TBoxes. However, the reductions in the proof for undecidability only use $\mathcal{EL}$ syntax.

$$C_P := =_\epsilon(\ell) \sqcap =_\epsilon(r) \sqcap \bigsqcap_{(\ell_i,r_i)\in P} (conc_{\ell_i}(\ell, f_i\ell) \sqcap conc_{r_i}(r, f_ir)) \sqcap \bigsqcap_{1\leq i\leq k} \exists f_i.L$$

$$\mathcal{T}_P := \left\{ L \equiv \bigsqcap_{(\ell_i,r_i)\in P} (conc_{\ell_i}(\ell, f_i\ell) \sqcap conc_{r_i}(r, f_ir)) \sqcap {\neq}(\ell, r) \sqcap \bigsqcap_{1\leq i\leq k} \exists f_i.L \right\}$$

Table 4.6: The $\mathcal{EL}(\mathsf{W})$-concept description C_P and $\mathcal{EL}(\mathsf{W})$-TBox $\mathcal{T}_P$ for the reduction from an instance $P = (\ell_1, r_1), \ldots, (\ell_k, r_k)$ of PCP to unsatisfiability in $\mathcal{EL}(\mathsf{W})$.

The *Post Correspondence Problem (PCP)* is to decide for a given instance P, whether P has a solution. $\Diamond$

The Post Correspondence Problem is known to be undecidable (Post 1947). As in (Lutz 2002), we employ the concrete domain W for the reduction. For Σ being an alphabet, W is defined by setting $\Delta_\mathsf{W} := \Sigma^*$ and Φ_W to be the smallest set containing:

- unary predicates $=_\epsilon$ and $\neq_\epsilon$ with $=_\epsilon^\mathsf{W} = \{\epsilon\}$ and $\neq_\epsilon^\mathsf{W} = \Sigma^+$
- a binary equality predicate $=$ and a binary inequality predicate $\neq$ with the obvious interpretation
- for each $w \in \Sigma^+$, a binary predicate conc_w with

$$\mathsf{conc}_w^\mathsf{W} = \{(u, v) \mid v = uw\}$$

The concrete domain W is p-admissible (Lutz 2002). In the following, let $\mathcal{EL}(\mathsf{W})$ denote $\mathcal{EL}$ extended by the concrete domain W with abstract feature chains. Table 4.7 shows the definitions of the concept description C_P and the cyclic TBox $\mathcal{T}_P$ that we will use for the reduction.

Lemma 29 *Let $P = (\ell_1, r_1), \ldots, (\ell_k, r_k)$ be a PCP. Then, the following are equivalent:*

1. *P has a solution*
2. *C_P is unsatisfiable w.r.t. $\mathcal{T}_P$.*

Proof. The proof is similar to the proof of undecidability of satisfiability in $\mathcal{ALC}(\mathsf{W})$ w.r.t. general TBoxes, as presented in (Lutz 2002).

(1 $\Rightarrow$ 2) Let $i_1, \ldots, i_m$ be a sequence of integers such that $\ell_{i_1} \ldots \ell_{i_m} = r_{i_1} \ldots r_{i_m}$. Assume there exists a model $\mathcal{I}$ of $\mathcal{T}_P$, let $x \in C_P^\mathcal{I}$ and $y = (f_1 \circ \ldots \circ f_{i_m})^\mathcal{I}(x) \in L^\mathcal{I}$.

However, $\ell^{\mathcal{I}}(y) = r^{\mathcal{I}}(y)$, in contradiction to $y \in L^{\mathcal{I}}$. Thus, $\mathcal{I}$ is not a model of $\mathcal{T}_P$. Hence C_P is unsatisfiable.

(2 $\Rightarrow$ 1) We show the contrapositive and assume that P has no solution. We construct a model $\mathcal{I}$ of $\mathcal{T}_P$ such that C_P holds in $\mathcal{I}$. Let $w = i_1 \dots i_n$ be a sequence of indices. We denote by *leftconc*(w) the concatenation of $\ell_{i_1} \dots \ell_{i_n}$, and by *rightconc*$(w)$ the concatenation of $r_{i_1} \dots r_{i_n}$. Then, $\mathcal{I}$ is defined as follows:

$$\begin{array}{rcl} \Delta^{\mathcal{I}} & := & \{i_1 \dots i_n \mid n \geq 0, 1 \leq i_j \leq k, 1 \leq j \leq n\} \\ f_i^{\mathcal{I}}(w) & := & wi \text{ for } 1 \leq i \leq k \\ \ell^{\mathcal{I}}(w) & := & \mathit{leftconc}(w) \\ r^{\mathcal{I}}(w) & := & \mathit{rightconc}(w) \\ L^{\mathcal{I}} & := & \Delta^{\mathcal{I}} \setminus \{i_\epsilon\} \text{ where } i_\epsilon \text{ denotes the empty sequence} \end{array}$$

P has no solution, and thus it is readily checked that $\mathcal{I}$ is a model of $\mathcal{T}_P$ and $i_\epsilon \in C_P^{\mathcal{I}}$. Hence, C_P is satisfiable w.r.t. $\mathcal{T}_P$. □

We thus have shown that $\mathcal{EL}(\mathsf{W})$-concept satisfiability is undecidable. Moreover, C_P is satisfiable w.r.t. $\mathcal{T}_P$ iff $C_P \not\sqsubseteq_{\mathcal{T}_P} \bot$ iff $C_P \not\sqsubseteq_{\mathcal{T}_P} {\neq}(\ell, \ell)$.

Theorem 19 *$\mathcal{EL}(\mathsf{W})$-concept satisfiability and subsumption w.r.t. cyclic TBoxes is undecidable.*

The main causes for undecidability come from the expressiveness of concrete domains and the functionality constraints on abstract features together with the fact that cyclic TBoxes allow to "reach" every domain element in connected models.

For $\mathcal{ALC}(\mathcal{D})$, it has been shown in (Lutz 2002) that W can be replaced by any arithmetic concrete domain. The reduction only uses $\mathcal{EL}$ syntax, and for that reason this result carries over to $\mathcal{EL}(\mathcal{D})$.

Corollary 1 *Let $\mathcal{D}$ be an arithmetic concrete domain. Then, $\mathcal{EL}(\mathcal{D})$-concept satisfiability and subsumption w.r.t. cyclic TBoxes and abstract feature chains is undecidable.*

4.8 Inverse Roles

Until now, we have seen in this chapter quite a number of extensions of $\mathcal{EL}$ that lead to intractability of the subsumption problem w.r.t. cyclic TBoxes. However, as an open question remains $\mathcal{ELI}$, which extends $\mathcal{EL}$ by inverse roles. In (Baader, Molitor & Tobies 1999), the description logic $\mathcal{ELIRO}^1$ has been studied, which extends $\mathcal{EL}$ by inverse roles, conjunction of roles and constants. Using a homomorphism approach, it has been shown that subsumption in $\mathcal{ELIRO}^1$ can be decided in polynomial time, and thus subsumption in $\mathcal{ELI}$ is polynomial.

However, the complexity of subsumption w.r.t. standard TBoxes remains open. For general TBoxes, it is known that subsumption is ExpTime-complete[3]. For the moment, the only known upper bounds for subsumption in $\mathcal{ELI}$ come from $\mathcal{ELI}$ being a notational fragment of $\mathcal{ALCQI}_{\text{reg}}$, for which subsumption is ExpTime-complete w.r.t. TBoxes (Baader et al. 2003).

[3]This result has not been published yet. For a proof of a PSpace-lower bound, see (Baader, Brandt & Lutz 2005a).

Chapter 5

Conclusion

The main goal of this thesis was to investigate which further extensions of $\mathcal{EL}$ can be made so that subsumption remains tractable without or w.r.t. standard TBoxes.

In Chapter 3, we have introduced two such extensions, $\mathcal{EL}^{\sqcup,\sqcap,\neg}(\mathcal{D})$ and $\mathcal{EL}^{\sqcup,\sqcap,\geq}$, for which subsumption w.r.t. cyclic TBoxes is tractable. We have shown in Chapter 4 that in a combination of both extensions subsumption is no longer tractable. Moreover, we have examined the complexity of subsumption of several other extensions, which all have in common that subsumption is intractable w.r.t. cyclic TBoxes. Thus, except for the open question of inverse roles, both $\mathcal{EL}^{\sqcup,\sqcap,\neg}(\mathcal{D})$ and $\mathcal{EL}^{\sqcup,\sqcap,\geq}$ are maximal in the sense that they cannot be further extended without losing tractability of subsumption w.r.t. cyclic TBoxes. We have noted as an interesting fact that, except for role conjunction, the sole addition of any of the extensions of $\mathcal{EL}^{\sqcup,\sqcap,\neg}(\mathcal{D})$ and $\mathcal{EL}^{\sqcup,\sqcap,\geq}$ to $\mathcal{EL}$ leads to ExpTime-completeness of subsumption w.r.t. general TBoxes. A summary of the complexity results of subsumption in extensions of $\mathcal{EL}$ obtained in this thesis can be found in Table 5.1.

There also remain some interesting open problems that have not been solved in this thesis. Most notably, the complexity of subsumption in $\mathcal{ELI}$ w.r.t. standard TBoxes. Furthermore, the exact complexity of subsumption w.r.t. standard TBoxes in $\mathcal{ELF}$ is still open. It is likely that the co-NP-lower bound for subsumption w.r.t. acyclic TBoxes is tight, as well as the PSpace-upper bound for subsumption w.r.t. cyclic TBoxes. Developing an algorithm that checks subsumption in $\mathcal{ELF}$ w.r.t. acyclic TBoxes in co-NP could result in a deeper general understanding of the complexity of subsumption in the $\mathcal{EL}$ family. Moreover, it would be interesting to see whether such an algorithm can also be employed for obtaining a co-NP-upper bound and a PSpace upper bound for subsumption in $\mathcal{ELI}$ w.r.t. acyclic and cyclic TBoxes respectively.

For future work, in order to make use of the tractable extension of $\mathcal{EL}$, the subsumption algorithms offer some potential of optimization and an ABox formalism has to be introduced. At the moment, the subsumption algorithms always naively check the all completion conditions. Introducing some heuristics for which and when to check completion conditions will likely increase the overall performance significantly. Concerning the introduction of an ABox formalism, it becomes problematic that instance checking in the presence of at-least restrictions as well as role disjunction is co-NP-complete w.r.t.

acyclic TBoxes (Krisnadhi 2007).

A further extension of $\mathcal{EL}$ that calls for investigation is the concept constructor introduced in (Baader, Lutz, Karabaev & Theißen 2005) that generalizes existential restrictions and qualified number restrictions. There, it has been shown that subsumption in $\mathcal{EL}$ extended by this new concept constructor is tractable w.r.t. acyclic TBoxes. It would be interesting to see whether this is also the case w.r.t. cyclic TBoxes, and whether this concept constructor can be combined with the tractable extensions of $\mathcal{EL}$ considered in this thesis.

Lastly, we briefly have attended to the correspondence of subsumption in $\mathcal{EL}$ and XPath query containment. Although for all extension of $\mathcal{EL}$ considered in Chapter 4 subsumption is intractable w.r.t. cyclic TBoxes, subsumption is tractable for some of them, e.g., for functionality. It would be interesting to identify a maximal extension of $\mathcal{EL}$ such that subsumption is tractable. These results could then be used to obtain a very expressive fragment of XPath for which query containment is tractable.

Extension	TBox	Complexity of subsumption	Page
$\mathcal{C}$	empty	PSPACE-complete	39
	acyclic		
	cyclic	EXPTIME-complete	
$\neg$	empty	in PTIME	24
	acyclic		
	cyclic		
$\mathcal{U}$	empty	CO-NP-complete	39
	acyclic	PSPACE-complete	42
	cyclic	EXPTIME-complete	44
$\mathcal{F}$	empty	in PTIME	60
	acyclic	CO-NP-hard	61
	cyclic	PSPACE-hard	63
$\geq$	empty	in PTIME	34
	acyclic		
	cyclic		
$(\mathcal{D})$[1]	empty	in PTIME	24
	acyclic		
	cyclic		
$\sqcup$	empty	in PTIME	24
	acyclic		
	cyclic		
$\sqcap$	empty	in PTIME	24
	acyclic		
	cyclic		
$+$	empty	CO-NP-complete	52
	acyclic	PSPACE-complete	55
	cyclic	EXPTIME-complete	54
$\mathcal{I}$	empty	in PTIME	66
	acyclic	in EXPTIME	
	cyclic		
f	empty	in PTIME	60
	acyclic	CO-NP-complete	57
	cyclic	PSPACE-complete	

[1]when $\mathcal{D}$ is a p-admissible concrete domain

Table 5.1: Complexity results of subsumption in extensions of $\mathcal{EL}$ considered in this thesis.

Bibliography

Baader, F. (2003). Terminological cycles in a description logic with existential restrictions, *in* G. Gottlob & T. Walsh (eds), *Proceedings of the 18th International Joint Conference on Artificial Intelligence*, Morgan Kaufmann, pp. 325–330.

Baader, F., Brandt, S. & Lutz, C. (2005a). Pushing the $\mathcal{EL}$ envelope, *Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence IJCAI-05*, Morgan-Kaufmann Publishers, Edinburgh, UK.

Baader, F., Brandt, S. & Lutz, C. (2005b). Pushing the $\mathcal{EL}$ envelope, *LTCS-Report LTCS-05-01*, Chair for Automata Theory, Institute for Theoretical Computer Science, Dresden University of Technology, Germany.
URL: *http://lat.inf.tu-dresden.de/research/reports.html.*

Baader, F., Calvanese, D., McGuinness, D. L., Nardi, D. & Patel-Schneider, P. F. (eds) (2003). *The Description Logic Handbook: Theory, Implementation, and Applications*, Cambridge University Press.

Baader, F. & Hanschke, P. (1991). A scheme for integrating concrete domains into concept languages, *Proceedings of the 12th International Joint Conference on Artificial Intelligence, IJCAI-91*, Sydney (Australia), pp. 452–457.

Baader, F., Horrocks, I. & Sattler, U. (2005). Description logics as ontology languages for the semantic web, *in* D. Hutter & W. Stephan (eds), *Mechanizing Mathematical Reasoning: Essays in Honor of Jörg H. Siekmann on the Occasion of His 60th Birthday*, Vol. 2605 of *Lecture Notes in Artificial Intelligence*, Springer-Verlag, pp. 228–248.

Baader, F., Lutz, C., Karabaev, E. & Theißen, M. (2005). A new n-ary existential quantifier in description logics, *Proceedings of the 28th Annual German Conference on Artificial Intelligence, KI 2005*, Lecture Notes in Artificial Intelligence, Springer-Verlag.

Baader, F., Lutz, C. & Suntisrivaraporn, B. (2007). Is tractable reasoning in extensions of the description logic $\mathcal{EL}$ useful in practice?, *Journal of Logic, Language and Information, Special Issue on Method for Modality (M4M).* To appear.

Baader, F., Molitor, R. & Tobies, S. (1999). Tractable and decidable fragments of conceptual graphs, *in* W. Cyre & W. Tepfenhart (eds), *Proceedings of the Seventh International Conference on Conceptual Structures (ICCS'99)*, number 1640 in *Lecture Notes in Computer Science*, Springer Verlag, pp. 480–493.

Brachman, R. J., McGuiness, D. L., Patel-Schneider, P. F. & Resnick, L. A. (1990). Living with CLASSIC: when and how to use a KL-ONE-like language, *in* J. Sowa (ed.), *Principles of semantic networks*, Morgan Kaufmann, San Mateo, US.
URL: *http://citeseer.ist.psu.edu/brachman91living.html*

Brandt, S. (2004). Polynomial time reasoning in a description logic with existential restrictions, GCI axioms, and—what else?, *in* R. L. de Mantáras & L. Saitta (eds), *Proceedings of the 16th European Conference on Artificial Intelligence (ECAI-2004)*, IOS Press, pp. 298–302.

Brandt, S. (2006). *Standard and Non-standard Reasoning in Description Logics*, PhD thesis, TU Dresden.
URL: *http://lat.inf.tu-dresden.de/research/phd/Brandt-PhD-2006.pdf.*

Clark, J. & DeRose, S. (1999). Xml xpath language reference.
URL: *http://www.w3.org/TR/xpath*

Donini, F. M., Lenzerini, M., Nardi, D. & Nutt, W. (1991). Tractable concept languages, *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence (IJCAI'91)*, pp. 458–463. Best Paper Award.

Fischer, M. J. & Ladner, R. E. (1979). Propositional dynamic logic of regular programs, *Journal of Computer and System Sciences* **18**: 194–211.

Francesco M. Donini, Maurizio Lenzerini, D. N. & Nutt, W. (1997). The complexity of concept languages, *Information and Computation* **134**(1): 1–58.

Garey, M. R. & Johnson, D. S. (1990). *Computers and Intractability; A Guide to the Theory of NP-Completeness*, W. H. Freeman & Co., New York, NY, USA.

Greenlaw, R., Hoover, H. & Ruzzo, W. (1992). A compendium of problems complete for P.
URL: *http://citeseer.ist.psu.edu/greenlaw91compendium.html*

Krisnadhi, A. (2007). *Data complexity of instance checking in the $\mathcal{EL}$ family of description logics*, Master thesis, TU Dresden, Germany.

Küsters, R. (2001). *Non-Standard Inferences in Description Logics*, Vol. 2100 of *Lecture Notes in Artificial Intelligence*, Springer-Verlag. PhD thesis.

Lutz, C. (2002). *The Complexity of Description Logics with Concrete Domains*, PhD thesis, RWTH Aachen.
URL: *http://lat.inf.tu-dresden.de/research/phd/Lutz-PhD-2002.pdf.*

Lutz, C. (2003). Description logics with concrete domains—a survey, *Advances in Modal Logics Volume 4*, King's College Publications.

Miklau, G. & Suciu, D. (2002). Containment and equivalence for an xpath fragment, *PODS '02: Proceedings of the twenty-first ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, ACM Press, New York, NY, USA, pp. 65–76.

Nebel, B. (1990a). *Reasoning and revision in hybrid representation systems*, Springer-Verlag New York, Inc., New York, NY, USA.

Nebel, B. (1990b). Terminological reasoning is inherently intractable, *Artificial Intelligence* **43**: 235–249.

Nebel, B. (1991). Terminological cycles: Semantics and computational properties, *in* J. F. Sowa (ed.), *Principles of Semantic Networks: Explorations in the Representation of Knowledge*, Morgan Kaufmann Publishers, San Mateo (CA), USA, pp. 331–361.

Post, E. L. (1947). Recursive unsolvability of a problem of thue, *The Journal of Symbolic Logic* **12**(1): 1–11.

Schild, K. (1994). Terminological cycles and propositional μ-calculus, *in* J. Doyle, E. Sandewall & P. Torasso (eds), *Proceedings 4th Int. Conf. on Principles of Knowledge Representation and Reasoning, KR'94, Bonn, Germany, 24–27 May 1994*, Morgan-Kaufmann Publishers, San Francisco, CA, pp. 509–520.
URL: *ftp://ftp.dfki.uni-sb.de/pub/tacos/Papers/kr94-klaus.ps*

Schild, K. D. (1991). A correspondence theory for terminological logics: Preliminary report, *in* J. Mylopoulos & R. Reiter (eds), *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence (IJCAI-91)*, Morgan Kaufmann, Sydney, Australia, pp. 466–471.

Schmidt-Schauß, M. & Smolka, G. (1991). Attributive concept descriptions with complements, *Artif. Intell.* **48**(1): 1–26.