Available online at www.sciencedirect.com
SCIENCE DIRECT®

Artificial Intelligence 160 (2004) 79–104

Artificial Intelligence

www.elsevier.com/locate/artint

Decidability of $\mathcal{SHIQ}$ with complex role inclusion axioms ☆

Ian Horrocks *, Ulrike Sattler

Department of Computer Science, University of Manchester, Kilburn Building, Manchester M13 9PL, UK

Received 2 January 2004; accepted 17 June 2004

Abstract

Motivated by medical terminology applications, we investigate the decidability of an expressive and prominent description logic (DL), $\mathcal{SHIQ}$, extended with role inclusion axioms of the form $R \circ S \dot{\sqsubseteq} T$. It is well known that a naive such extension leads to undecidability, and thus we restrict our attention to axioms of the form $R \circ S \dot{\sqsubseteq} R$ or $S \circ R \dot{\sqsubseteq} R$, which is the most important form of axioms in the applications that motivated this extension. Surprisingly, this extension is still undecidable. However, it turns out that by restricting our attention further to acyclic sets of such axioms, we regain decidability. We present a tableau-based decision procedure for this DL and report on its implementation, which promises to behave well in practice and provides important additional functionality in a medical terminology application.

Keywords: Description logics; Tableau algorithm; Automated reasoning

☆ This is an extended version of the conference paper [I. Horrocks, U. Sattler, in: Proc. of the 17th Int. Joint Conf. on Artificial Intelligence, IJCAI-03, Morgan Kaufmann, Los Altos, 2003, a long version is available as technical report LTCS 02-06 at http://lat.inf.tu-dresden.de/research/reports.html].

* Corresponding author.
E-mail address: horrocks@cs.man.ac.uk (I. Horrocks).

0004-3702/$ – see front matter
doi:10.1016/j.artint.2004.06.002

1. Motivation

The description logic (DL) $\mathcal{SHIQ}$ [14,17] is an expressive knowledge representation formalism that extends $\mathcal{ALC}$ [29] with qualifying number restrictions, inverse roles, role inclusion axioms, and transitive roles. The development of $\mathcal{SHIQ}$ was motivated and inspired by several applications, one of which was the representation of knowledge about complex physically structured domains found, e.g., in chemical engineering [26] and medical terminology [25].

For example, in $\mathcal{SHIQ}$, we can describe fractures of the femur by the following concept which, intuitively, denotes fractures that are located in the femur or the neck of the femur:

$$\mathtt{FemurFracture} \doteq \mathtt{Fracture} \sqcap \exists \mathtt{hasLocation}.(\mathtt{Femur} \sqcup \mathtt{FemurNeck}).$$

To make this definition work, we also should describe the neck of the femur, e.g., as follows:

$$\mathtt{FemurNeck} \doteq \mathtt{BodyPart} \sqcap \mathtt{Proxima} \sqcap \exists \mathtt{isDivisionOf}.\mathtt{Femur}.$$

$\mathcal{SHIQ}$ allows many important properties of application domains to be captured: e.g., we can state that `hasLocation` is transitive, and that `LocatedIn` is the inverse of `hasLocation`. However, there is one extremely useful feature that $\mathcal{SHIQ}$ cannot express, namely the “propagation” of one property along another property [21,23,31]. Coming back to our example above, to capture that also a fracture of the shaft of the femur is a fracture of the femur, we need to add this information explicitly the definition of `FemurFracture`. As such, this is easily feasible. A more elegant approach would be to change our definition to

$$\mathtt{FemurFracture} \doteq \mathtt{Fracture} \sqcap \exists \mathtt{hasLocation} \\ .(\mathtt{Femur} \sqcup \exists \mathtt{isDivisionOf}.\mathtt{Femur}).$$

Still, we have to have a similar disjunction in the definition of a fracture of the tibia, and all other fractures. Thus, it would be useful if we could express, in general, the fact that certain locative properties are transfered across certain partonomic properties so that a fracture or trauma located in a part of a body structure is recognised as being located in the body structure as a whole. This would yield highly desirable inferences such as a fracture of the shaft of the femur being inferred to be a kind of fracture of the femur, or an ulcer located in the gastric mucosa being inferred to be a kind of stomach ulcer—without the necessity to repeat this statement in the definition of every single such concept.

The importance of these kinds of inferences, particularly in medical terminology applications, is illustrated by the fact that three different such applications provide means to express propagation. The Grail DL [24], which was specifically designed for use with medical terminology, is able to represent these kinds of propagation (although it is quite weak in other respects). In another medical terminology application using the comparatively inexpressive DL $\mathcal{ALC}$, a rather complex “work around” is performed in order to represent similar propagations [30]: so-called *SEP-triplets* are used both to compensate for the absence of transitive roles in $\mathcal{ALC}$, and to express the propagation of properties across a distinguished “part-of” role. In a third application, use is made of so-called *right-identities*, which correspond to our complex role inclusion axioms [31]. Finally, similar expressiveness was also provided in the CycL language by the `transfersThro` statement [19]. To the best of our knowledge, however, there is no proof of the correct treatment of propagation in any of these applications.

It is quite straightforward to extend $\mathcal{SHIQ}$ so that this kind of propagation can be expressed: simply allow for role inclusion axioms (RIAs) of the form $R \circ S \dot{\sqsubseteq} P$, which then forces all models $\mathcal{I}$ to interpret the composition of $R^{\mathcal{I}}$ with $S^{\mathcal{I}}$ as a sub-relation of $P^{\mathcal{I}}$. E.g., the above examples translate into

$$\texttt{hasLocation} \circ \texttt{isDivisionOf} \dot{\sqsubseteq} \texttt{hasLocation},$$

which implies that

$$\texttt{Fracture} \sqcap \exists \texttt{hasLocation}.(\texttt{Neck} \sqcap \exists \texttt{isDivisionOf}.\texttt{Femur}),$$

i.e., a concept describing fractures of the neck of the femur, is indeed subsumed by (is a specialisation of)

$$\texttt{Fracture} \sqcap \exists \texttt{hasLocation}.\texttt{Femur},$$

i.e., a concept describing fractures of the femur.

Unfortunately, this extension leads to the undecidability of interesting inference problems such as concept satisfiability and subsumption [33]. This undecidability is not surprising once we observe the close relationship between RIAs, *Grammar Logics* [3,4,8], and *role value maps* [6,28]. This relationship is discussed in more detail in Section 2.1. Here, it should suffice to mention that a RIA $R\,S \dot{\sqsubseteq} T$ can be viewed as a notational variant of the production rule $T \rightarrow R\,S$ of Grammar Logics or the concept inclusion $\top \dot{\sqsubseteq} (R\,S \tilde{\sqsubseteq} T)$ of a description logic allowing for role value maps.

On closer inspection of our motivating examples, we observe that only RIAs of the form $R\,S \dot{\sqsubseteq} S$ or $S\,R \dot{\sqsubseteq} S$ are required in order to express propagation. To the best of our knowledge, no (un)decidability results are known for similar restrictions of the above mentioned Grammar Logics or DLs with role value maps. In this paper, we will show that $\mathcal{SHIQ}$ extended with this restricted form of RIAs is still undecidable. Due to the syntactic restrictions imposed on RIAs, we cannot re-use techniques employed to prove undecidability of Grammar Logics or DLs with role value maps. Instead, our proof is by reduction of the undecidable domino problem [5], and uses a rather special technique to ensure a grid structure.

Decidability can be regained, however, by further restricting the set of RIAs to be *regular*, and the logic obtained by restricting RIAs to regular ones is called $\mathcal{RIQ}$. From a practical point of view, the restrictions imposed by regularity do not seem to be severe: regular RIAs should suffice for many applications, and non-regular RIAs may even be an indicator of modelling flaws [23].

We prove the decidability of $\mathcal{SHIQ}$ with regular RIAs via a tableau-based decision procedure for the satisfiability of concepts. We first translate regular RIAs into non-deterministic automata, and then use these automata in the tableau algorithm. More precisely, the tableau algorithm replaces concepts of the form $\forall R.C$ (where R is a role) with expressions of the form $\forall \mathcal{B}_R.C$, where $\mathcal{B}_R$ is a non-deterministic finite automaton (NFA) capturing exactly the restrictions imposed on R by RIAs. Using these expressions, we ensure that the concept C is indeed "pushed" to all those nodes it has to be pushed to, even if they are far away from a node that has to satisfy $\forall R\,.\,C$. The algorithm is of the same complexity as the one for $\mathcal{SHIQ}$—in the size of $\mathcal{B}_R$ and the length of the input concept—but, unfortunately, $\mathcal{B}_R$ can be exponential in the "depth" of $\mathcal{R}$, i.e., in the length of chains of roles depending on each other. We also present a syntactic restriction that avoids this blow-up; investigating whether this blow-up can be avoided in general will be part of future work.

As we have discussed above, the interaction between roles in regular RIAs can be captured by NFAs, but we have not yet explained which RIAs are regular. This is so because, in the presence of inverse roles, the definition of regularity becomes slightly tricky: each "left-linear" RIA of the form $R\,S \dot\sqsubseteq S$ is equivalent to a "right-linear" RIA $S^-\,R^- \dot\sqsubseteq S^-$. Thus each left-linear RIA has consequences that are inherently a mixture of right- and left-linear RIAs. Now it is well known that grammars with a such a linear mixture are stronger than right-linear grammars or left-linear grammars [11], and this is true also for RIAs, as our undecidability result shows. Thus, to enable the transformation into an automaton, we impose an additional restriction, which we have chosen to be *acyclicity* in a rather loose sense, i.e., we still allow for RIAs $S\,S \dot\sqsubseteq S$, $R\,S \dot\sqsubseteq S$, and $S\,R \dot\sqsubseteq S$, but we do not allow for combinations of RIAs such as $R\,S \dot\sqsubseteq S$ and $S\,R \dot\sqsubseteq R$.

Finally, in order to evaluate the practicability of this algorithm, we have extended the DL system FaCT [12] to deal with $\mathcal{RIQ}$. We discuss how the properties of NFAs are exploited in the implementation, and we present some preliminary results showing that the performance of the extended system is comparable with that of the original, and that it is able to compute inferences of the kind mentioned above w.r.t. the well-known Galen medical terminology knowledge base [12,25].

2. Preliminaries

In this section, we introduce the DL $\mathcal{SH}^+\mathcal{IQ}$. This includes the definition of syntax, semantics, and inference problems.

Definition 1. Let **C** be a set of *concept names* and **R** a set of *role names*. The set of *roles* is $\mathbf{R} \cup \{R^- \mid R \in \mathbf{R}\}$. A *role inclusion axiom* is an expression of one of the following forms:

$$R_1 \dot\sqsubseteq R_2, \qquad R_1 R_2 \dot\sqsubseteq R_1, \quad \text{or} \quad R_1 R_2 \dot\sqsubseteq R_2,$$

for roles R_i (each of which can be inverse). A *generalised role hierarchy* is a set of role inclusion axioms.

An interpretation $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$ associates, with each role name R, a binary relation $R^\mathcal{I} \subseteq \Delta^\mathcal{I} \times \Delta^\mathcal{I}$. Inverse roles are interpreted as usual, i.e.,

$$(R^-)^\mathcal{I} = \left\{\langle y, x\rangle \mid \langle x, y\rangle \in R^\mathcal{I}\right\} \quad \text{for each role } R \in \mathbf{R}.$$

An interpretation $\mathcal{I}$ is a *model* of a generalised role hierarchy $\mathcal{R}$ if it satisfies each inclusion assertion in $\mathcal{R}$, i.e., if

$$R_1^\mathcal{I} \subseteq R_2^\mathcal{I} \quad \text{for each } R_1 \dot\sqsubseteq R_2 \in \mathcal{R}, \quad \text{and}$$
$$R_1^\mathcal{I} \circ R_2^\mathcal{I} \subseteq R_3^\mathcal{I} \quad \text{for each } R_1 R_2 \dot\sqsubseteq R_3 \in \mathcal{R},$$

where $\circ$ stands for the composition of binary relations.

Note that we did not introduce *transitive role names* since adding $RR \dot{\sqsubseteq} R$ to the generalised role hierarchy is equivalent to saying that R is a transitive role.

To avoid considering roles such as R^{--}, we define a function Inv on roles such that $\mathsf{Inv}(R) = R^-$ if R is a role name, and $\mathsf{Inv}(R) = S$ if $R = S^-$.

Since we will often work with a string of roles, it is convenient to extend both $\cdot^{\mathcal{I}}$ and $\mathsf{Inv}(\cdot)$ to such strings: if $w = R_1 \ldots R_n$ for R_i roles, then $w^{\mathcal{I}} = R_1^{\mathcal{I}} \circ \cdots \circ R_n^{\mathcal{I}}$ and $\mathsf{Inv}(w) = \mathsf{Inv}(R_n) \ldots \mathsf{Inv}(R_1)$. It follows immediately from the definition of the semantics that

$$\langle x, y \rangle \in w^{\mathcal{I}} \quad \text{iff} \quad \langle y, x \rangle \in \mathsf{Inv}(w)^{\mathcal{I}}.$$

Next, since each model satisfying $w \dot{\sqsubseteq} S$ also satisfies $\mathsf{Inv}(w) \dot{\sqsubseteq} \mathsf{Inv}(S)$ (and vice versa), we can restrict generalised role hierarchies to those with role *names* on their right-hand side without any effect on the expressivity. For better readability, we will not do this in the undecidability proof of $\mathcal{SH}^+\mathcal{IQ}$, but we will do it for the decidable logic $\mathcal{RIQ}$ since it makes the construction in the proofs easier.

Finally, for a generalised role hierarchy $\mathcal{R}$, we define the relation $\underline{\circledast}$ to be the transitive–reflexive closure of $\dot{\sqsubseteq}$ over $\{R \dot{\sqsubseteq} S, \mathsf{Inv}(R) \dot{\sqsubseteq} \mathsf{Inv}(S) \mid R, S \text{ roles and } R \dot{\sqsubseteq} S \in \mathcal{R}\}$. A role R is called a *sub-role* (respectively super-role) of a role S if $R \underline{\circledast} S$ (respectively $S \underline{\circledast} R$). Two roles R and S are *equivalent* ($R \equiv S$) if $R \underline{\circledast} S$ and $S \underline{\circledast} R$.

Now we are ready to define the syntax and semantics of $\mathcal{SH}^+\mathcal{IQ}$-concepts.

Definition 2. Let $\mathcal{R}$ be a generalised role hierarchy. A role R is *simple in* $\mathcal{R}$ if, for each $R' \underline{\circledast} R$, $\mathcal{R}$ contains no RIA of the form $R_1 R_2 \dot{\sqsubseteq} R'$ or $R_1 R_2 \dot{\sqsubseteq} \mathsf{Inv}(R')$. If $\mathcal{R}$ is clear from the context, we often use "simple" instead of "simple in $\mathcal{R}$".

The set of *$\mathcal{SH}^+\mathcal{IQ}$-concepts* is the smallest set such that

- every concept name and $\top$, $\bot$ are concepts, and,
- if C, D are concepts, R is a role (possibly inverse), S is a simple role (possibly inverse), and n is a non-negative integer, then $C \sqcap D$, $C \sqcup D$, $\neg C$, $\forall R \,.\, C$, $\exists R \,.\, C$, $(\geqslant nS \,.\, C)$, and $(\leqslant nS \,.\, C)$ are also concepts.

A *general concept inclusion axiom* (GCI) is an expression of the form $C \dot{\sqsubseteq} D$ for two $\mathcal{SH}^+\mathcal{IQ}$-concepts C and D. A *terminology* is a set of GCIs.

An *interpretation* $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ consists of a set $\Delta^{\mathcal{I}}$, called the *domain* of $\mathcal{I}$, and a *valuation* $\cdot^{\mathcal{I}}$ which maps every concept to a subset of $\Delta^{\mathcal{I}}$ and every role to a subset of $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ such that, for all concepts C, D, roles R, S, and non-negative integers n, the following equations are satisfied, where $\sharp M$ denotes the cardinality of a set M:

$$\top^{\mathcal{I}} = \Delta^{\mathcal{I}} \qquad \bot^{\mathcal{I}} = \emptyset \qquad \text{(top and bottom)},$$
$$(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}} \qquad \text{(conjunction)},$$
$$(C \sqcup D)^{\mathcal{I}} = C^{\mathcal{I}} \cup D^{\mathcal{I}} \qquad \text{(disjunction)},$$
$$(\neg C)^{\mathcal{I}} = \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}} \qquad \text{(negation)},$$
$$(\exists R \,.\, C)^{\mathcal{I}} = \{x \mid \exists y \,.\, \langle x, y \rangle \in R^{\mathcal{I}} \text{ and } y \in C^{\mathcal{I}}\} \qquad \text{(exists restriction)},$$
$$(\forall R \,.\, C)^{\mathcal{I}} = \{x \mid \forall y \,.\, \langle x, y \rangle \in R^{\mathcal{I}} \text{ implies } y \in C^{\mathcal{I}}\} \qquad \text{(value restriction)},$$

$$(\geqslant nR\,.\,C)^{\mathcal{I}} = \big\{x \mid \sharp\big\{y\,.\,\langle x,y\rangle \in R^{\mathcal{I}} \text{ and } y \in C^{\mathcal{I}}\big\} \geqslant n\big\} \quad \text{(at least restriction)},$$
$$(\leqslant nR\,.\,C)^{\mathcal{I}} = \big\{x \mid \sharp\big\{y\,.\,\langle x,y\rangle \in R^{\mathcal{I}} \text{ and } y \in C^{\mathcal{I}}\big\} \leqslant n\big\} \quad \text{(at most restriction)}.$$

An interpretation $\mathcal{I}$ is a *model* of a terminology $\mathcal{T}$ (written $\mathcal{I} \models \mathcal{T}$) iff $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ for each GCI $C \dot{\sqsubseteq} D$ in $\mathcal{T}$.

A concept C is called *satisfiable* iff there is an interpretation $\mathcal{I}$ with $C^{\mathcal{I}} \neq \emptyset$. A concept D *subsumes* a concept C (written $C \sqsubseteq D$) iff $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ holds for each interpretation. Two concepts are *equivalent* (written $C \equiv D$) if they are mutually subsuming. The above inference problems can be defined w.r.t. a generalised role hierarchy $\mathcal{R}$ and/or a terminology $\mathcal{T}$ in the usual way, i.e., by replacing *interpretation* with *model of* $\mathcal{R}$ and/or $\mathcal{T}$.

For an interpretation $\mathcal{I}$, an element $x \in \Delta^{\mathcal{I}}$ is called an *instance* of a concept C iff $x \in C^{\mathcal{I}}$.

Please note that number restrictions $(\geqslant nR\,.\,C)$ and $(\leqslant nR\,.\,C)$ are restricted to *simple* roles. Intuitively, these are (possibly inverse) roles that are not implied by the composition of other roles. The reason for this restriction is that, without it, satisfiability of $\mathcal{SHIQ}$-concepts is undecidable [16], even for a logic without inverse roles and with only *unqualifying* number restrictions (these are number restrictions of the form $(\geqslant nR\,.\,\top)$ and $(\leqslant nR\,.\,\top)$).

For DLs that are closed under negation, subsumption and (un)satisfiability can be mutually reduced: $C \sqsubseteq D$ iff $C \sqcap \neg D$ is unsatisfiable, and C is unsatisfiable iff $C \sqsubseteq \bot$. It is straightforward to extend these reductions to generalised role hierarchies and terminologies. In contrast, the reduction of inference problems w.r.t. a terminology to pure concept inference problems (possibly w.r.t. a role hierarchy), deserves special care: in [1,2,27], the *internalisation* of GCIs is introduced, a technique that realises exactly this reduction. For $\mathcal{SH}^+\mathcal{IQ}$, this technique only needs to be slightly modified. The following lemma shows how general concept inclusion axioms can be *internalised* using a "universal" role U, that is, a transitive super-role of all roles occurring in $\mathcal{T}$ or $\mathcal{R}$ and their respective inverses.

Lemma 3. *Let C, D be concepts, $\mathcal{T}$ a terminology, and $\mathcal{R}$ a generalised role hierarchy. We define*

$$C_{\mathcal{T}} := \prod_{C_i \dot{\sqsubseteq} D_i \in \mathcal{T}} \neg C_i \sqcup D_i.$$

Let U be a role that does not occur in $\mathcal{T}$, C, D, or $\mathcal{R}$. We set

$$\mathcal{R}_U := \mathcal{R} \cup \{UU \dot{\sqsubseteq} U\} \cup \big\{R \dot{\sqsubseteq} U, \mathsf{Inv}(R) \dot{\sqsubseteq} U \mid R \textit{ occurs in } \mathcal{T}, C, D, \textit{ or } \mathcal{R}\big\}.$$

- *C is satisfiable w.r.t. $\mathcal{T}$ and $\mathcal{R}$ iff $C \sqcap C_{\mathcal{T}} \sqcap \forall U\,.\,C_{\mathcal{T}}$ is satisfiable w.r.t. $\mathcal{R}_U$.*
- *D subsumes C with respect to $\mathcal{T}$ and $\mathcal{R}$ iff $C \sqcap \neg D \sqcap C_{\mathcal{T}} \sqcap \forall U\,.\,C_{\mathcal{T}}$ is unsatisfiable w.r.t. $\mathcal{R}_U$.*

The proof of Lemma 3 is similar to the ones that can be found in [1,27]. Most importantly, it must be shown that, (a) if a $\mathcal{SH}^+\mathcal{IQ}$-concept C is satisfiable with respect to a terminology $\mathcal{T}$ and a generalised role hierarchy $\mathcal{R}$, then $C, \mathcal{T}$ have a *connected* model, i.e., a model where any two elements are connect by a role path over those roles occurring in C and $\mathcal{T}$, and (b) if y is reachable from x via a role path (possibly involving inverse roles), then $\langle x, y \rangle \in U^{\mathcal{I}}$. These are easy consequences of the semantics and the definition of U.

Theorem 4. *Satisfiability and subsumption of $\mathcal{SH}^{+}\mathcal{IQ}$-concepts w.r.t. terminologies and generalised role hierarchies are polynomially reducible to* (*un*)*satisfiability of $\mathcal{SH}^{+}\mathcal{IQ}$-concepts w.r.t. generalised role hierarchies.*

2.1. Relationship with grammar logics

It is well known that description and modal logics are closely related: for example, $\mathcal{ALC}$ can be viewed as a notational variant of the multi modal logic **K** [7,27]. Related to the logics investigated here are *grammar logics* [9], a class of propositional multi modal logics where the accessibility relations are "axiomatised" through a grammar. More precisely, for σ_i, τ_j modal parameters, the production rule $\sigma_1 \ldots \sigma_m \to \tau_1 \ldots \tau_n$ can be viewed as an abbreviation for the axioms

$$[\sigma_1] \ldots [\sigma_m] p \;\Rightarrow\; [\tau_1] \ldots [\tau_n] p,$$

or as being a notational variant for the role inclusion axiom

$$\tau_1 \ldots \tau_n \mathrel{\dot{\sqsubseteq}} \sigma_1 \ldots \sigma_m.$$

Analogously to the description logic case, the semantics of a grammar logic is defined by taking into account only those frames/relational structures that "satisfy the grammar".

Grammars are traditionally organised in (refinements of) the Chomsky hierarchy (see any textbook on formal languages, e.g., [11]), which also induces classes of grammar logics. For example, the class of *context free* grammar logics is the class of those propositional multi modal logics where the accessibility relations are axiomatised through a *context free* grammar. Unsurprisingly, the expressiveness of the grammars influences the expressiveness of the corresponding grammar logics. It was shown that satisfiability of *regular* grammar logics is ExpTime-complete [8], whereas this problem is undecidable for context free grammar logics [3,4]. The latter result is closely related to the undecidability proof in [33]. In this paper, we are concerned with

- grammars that are *not* regular, but we do not allow for arbitrary context-free grammars (or any known normal forms thereof), and
- multi modal logics that provide a converse operator on modal parameters. That is, for σ a modal parameter, both $[\sigma]\varphi$ and $[\sigma^{-}]\varphi$ are formulae of our logic, and we allow mixtures of converse and atomic modal parameters in the rules of the grammar. Moreover, $\mathcal{SH}^{+}\mathcal{IQ}$ provides *graded* modalities that restrict the number of accessible worlds, see, e.g., [18,32].

As a consequence of the first point, we could not re-use the technique from [3,4] for our undecidability proof: we could not reduce the emptiness problem for the intersection of context-free grammars to the satisfiability of $\mathcal{SH}^+\mathcal{IQ}$-concepts because $\mathcal{SH}^+\mathcal{IQ}$'s syntactic restriction on role inclusion axioms means that we cannot capture all context-free grammars. However, we can capture "some" context-freeness: our undecidability proof in Section 3 is by a reduction of the undecidable domino problem [5], and is heavily based on the language $\{(ab)^n(cd)^n \mid n \geqslant 0\}$ to enforce a model with a "grid" structure. Although we were not able to construct a grammar for this language directly using only productions of the form $R \to RS$ or $R \to SR$, we used a grammar G such that the language generated by G, when intersected with $(ab)^*(cd)^*$, equals $\{(ab)^n(cd)^n \mid n \geqslant 0\}$. This grammar G contains the four production rules

$$D \to AD,$$
$$A \to AC,$$
$$C \to BC,$$
$$B \to BD, \qquad A \to a, \ldots, D \to d,$$

and can be found in four versions as the last axioms of $\mathcal{R}_\mathcal{D}$ in Fig. 2, where we use x_i, y_i, and their inverses instead of $A, \ldots, B$.

2.2. Role value maps

The role inclusion axioms we investigate here are closely related to *role value maps* [6,28], i.e., concepts of the form $R_1 \ldots R_m \mathrel{\tilde{\sqsubseteq}} S_1 \ldots S_n$ for R_i, S_i roles. The semantics of these concepts is defined as follows:

$$(R_1 \ldots R_m \mathrel{\tilde{\sqsubseteq}} S_1 \ldots S_n)^\mathcal{I} = \{x \in \Delta^\mathcal{I} \mid (R_1 \ldots R_m)^\mathcal{I}(x) \subseteq (S_1 \ldots S_n)^\mathcal{I}(x)\},$$

where $(R_1 \ldots R_m)^\mathcal{I}(x)$ denotes the set of those $y \in \Delta^\mathcal{I}$ that are reachable from x via $R_1^\mathcal{I} \circ \cdots \circ R_m^\mathcal{I}$.

Thus the role inclusion axiom $RS \mathrel{\dot{\sqsubseteq}} T$ is equivalent to the general concept inclusion axiom $\top \mathrel{\dot{\sqsubseteq}} (RS \mathrel{\tilde{\sqsubseteq}} T)$, i.e., both axioms have the same models. The role value maps used to show the undecidability of KL-ONE [28] are of a more general form than $(RS \mathrel{\tilde{\sqsubseteq}} T)$, i.e., they use role chains of unbounded length on both sides of $\tilde{\sqsubseteq}$, and there is no direct translation of the undecidability proof in [28] to our logic.

3. $\mathcal{SH}^+\mathcal{IQ}$ is undecidable

Due to the syntactic restriction on role inclusion axioms, neither the undecidability proof for $\mathcal{ALC}$ with context-free or linear grammars in [3,4,8] nor the one for $\mathcal{ALC}$ with role boxes [33] can be adapted to prove undecidability of $\mathcal{SH}^+\mathcal{IQ}$ satisfiability. In the following, we reduce the (undecidable) domino problem [5] to $\mathcal{SH}^+\mathcal{IQ}$ satisfiability. This problem asks whether, for a set of domino types, there exists a *tiling* of an $\mathbb{N}^2$ grid such that each point of the grid is covered with exactly one of the domino types, and adjacent dominoes are "compatible" with respect to some predefined criteria.

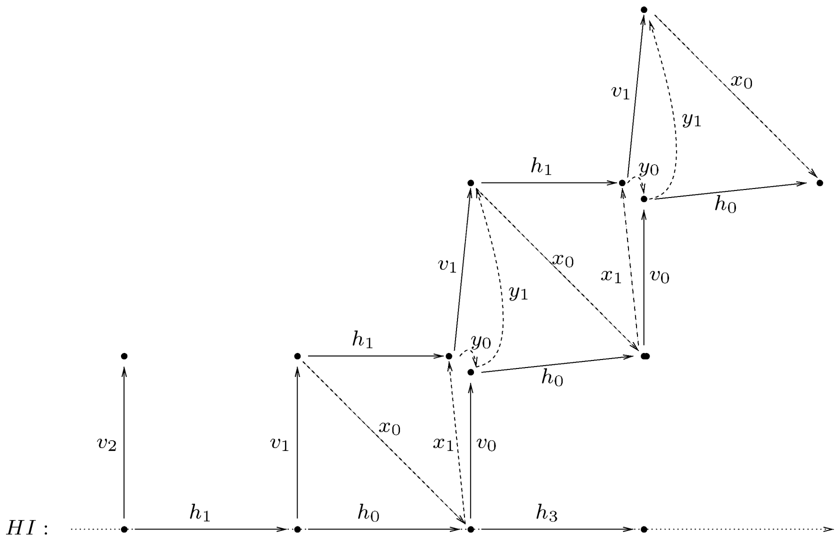

Fig. 1. A staircase model and the implications of the last group of axioms in $\mathcal{R}_\mathcal{D}$.

Definition 5. A domino system $\mathcal{D} = (D, H, V)$ consists of a non-empty set of domino types $D = \{D_1, \ldots, D_n\}$, and of sets of horizontally and vertically matching pairs $H \subseteq D \times D$ and $V \subseteq D \times D$. The problem is to determine if, for a given $\mathcal{D}$, there exists a *tiling* of an $\mathbb{N} \times \mathbb{N}$ grid such that each point of the grid is covered with a domino type in D and all horizontally and vertically adjacent pairs of domino types are in H and V, respectively, i.e., a mapping

$$
\begin{aligned}
&t : \mathbb{N} \times \mathbb{N} \to D \quad \text{such that, for all } m, n \in \mathbb{N}, \\
&\quad \langle t(m,n), t(m+1,n) \rangle \in H \quad \text{and} \\
&\quad \langle t(m,n), t(m,n+1) \rangle \in V.
\end{aligned}
$$

Given a domino system $\mathcal{D}$, the problem of determining if there exists a tiling for $\mathcal{D}$ is known to be undecidable [5].

In Fig. 2, for a domino system $\mathcal{D}$, we define a $\mathcal{SH}^+\mathcal{IQ}$-concept $C_\mathcal{D}$, a terminology $\mathcal{T}_D$ (that can be internalised, see Theorem 4), and a generalised role hierarchy $\mathcal{R}_D$ such that $\mathcal{D}$ has a tiling iff $C_\mathcal{D}$ is satisfiable w.r.t. $\mathcal{R}_\mathcal{D}$ and $\mathcal{T}_\mathcal{D}$. For better readability, we use $C \Rightarrow D$ as an abbreviation for $\neg C \sqcup D$.

Ensuring that a point is associated with exactly one domino type, that it has at most one vertical and at most one horizontal successor, and that these successors satisfy the horizontal and vertical matching conditions induced by H and V is standard and is done in the first GCI of $\mathcal{T}_\mathcal{D}$.

The next step is rather special: we do not force a grid structure, but a structure with “staircases”, which is illustrated in Fig. 1. To this purpose, we introduce four sub-roles $v_0, \ldots, v_3$ of v and four sub-roles $h_0, \ldots, h_3$ of h (see first line of $\mathcal{R}_\mathcal{D}$), and ensure that we only have “staircases”. For each $i \in \{0, \ldots, 3\}$, an i-staircase is an alternating chain of

$$
\begin{aligned}
\mathcal{T}_{\mathcal{D}} := \{ \top \doteq\ & (\bigsqcup_{1\leqslant i\leqslant n} D_i) \sqcap (\bigsqcap_{1\leqslant i<j\leqslant n} \neg(D_i \sqcap D_j)) \sqcap \\
& \bigsqcap_{1\leqslant i\leqslant n} D_i \Rightarrow ((\leqslant 1v\,.\,\top) \sqcap (\forall v\,.\bigsqcup_{(D_i,D_j)\in V} D_j)) \sqcap \\
& \bigsqcap_{1\leqslant i\leqslant n} D_i \Rightarrow ((\leqslant 1h\,.\,\top) \sqcap (\forall h\,.\bigsqcup_{(D_i,D_j)\in H} D_j), \\
I \doteq\ & HI \sqcup VI, \\
\top \doteq\ & \bigsqcap_{0\leqslant i\leqslant 3} (\exists v_i^-\,.\,\top \sqcap \neg I) \Rightarrow \big(\exists h_i\,.\,\neg I \sqcap \bigsqcap_j \forall v_j\,.\,\bot \sqcap \bigsqcap_{j\neq i} \forall h_j\,.\,\bot)\big) \sqcap \\
& (\exists h_i^-\,.\,\top \sqcap \neg I) \Rightarrow \big(\exists v_i\,.\,\neg I \sqcap \bigsqcap_{j\neq i} \forall v_j\,.\,\bot \sqcap \bigsqcap_j \forall h_j\,.\,\bot)\big) \sqcap \\
& (\exists h_i^-\,.\,\top \sqcap HI) \Rightarrow \big(\exists v_i\,.\,\neg I \sqcap \exists h_{i\ominus 1}\,.\,HI \sqcap \\
& \qquad \bigsqcap_{j\neq i\ominus 1} \forall h_j\,.\,\bot \sqcap \bigsqcap_{j\neq i} \forall v_j\,.\,\bot\big) \sqcap \\
& (\exists v_i^-\,.\,\top \sqcap VI) \Rightarrow \big(\exists h_i\,.\,\neg I \sqcap \exists v_{i\oplus 1}\,.\,VI \sqcap \\
& \qquad \bigsqcap_{j\neq i\oplus 1} \forall v_j\,.\,\bot \sqcap \bigsqcap_{j\neq i} \forall h_j\,.\,\bot\big), \\
\top \doteq\ & \bigsqcap_{0\leqslant i\leqslant 3} \bigsqcap_{1\leqslant j\leqslant n} \exists x_{i\oplus 1}^-\,.\,\top \Rightarrow (D_j \Rightarrow \forall y_i\,.\,D_j)\} \\
C_{\mathcal{D}} :=\ & HI \sqcap VI \sqcap \exists h_0.HI \sqcap \exists v_1.VI \\
\mathcal{R}_{\mathcal{D}} :=\ & \{v_i \dot\sqsubseteq v,\ h_i \dot\sqsubseteq h,\ v_i \dot\sqsubseteq y_i,\ h_i \dot\sqsubseteq x_i \mid 0 \leqslant i \leqslant 3\} \cup \\
& \{x_{i\oplus 1}^- y_i \ \dot\sqsubseteq\ y_i, \\
& \ x_{i\oplus 1}^- x_i \ \dot\sqsubseteq\ x_{i\oplus 1}^-, \\
& \ y_{i\oplus 1}^- x_i \ \dot\sqsubseteq\ x_i, \\
& \ y_{i\oplus 1}^- y_i \ \dot\sqsubseteq\ y_{i\oplus 1}^- \mid 0 \leqslant i \leqslant 3\}
\end{aligned}
$$

Fig. 2. Reduction terminology, generalised role hierarchy, and concept.

v_i and h_i edges, without any other v_j- or h_j-successors. We use concepts *HI* and *VI* for points on the x-axis and y-axis respectively. At each point on the x-axis, two staircases start that need not meet again, one i-staircase starting with v_i and one $i \ominus 1$-staircase starting with $h_{i\ominus 1}$ (we use $\oplus$ and $\ominus$ to denote addition and subtraction modulo four); points on the y-axis exhibit a symmetrical behaviour. The second GCI in $\mathcal{T}_{\mathcal{D}}$ introduces the concept I for all "initial" points, and then the third GCI ensures the staircase structure. It contains four implications: one for the vertical and one for the horizontal successorships, and these two implications once for the "non-initial" points (i.e., instances of $\neg I$), and once for the "initial points" (i.e., instances of *HI* or *VI*).

It remains to make sure that two elements b, b' representing the same point in the grid have the same domino type associated with them, where b and b' "represent the same point" if there is an n and an instance a of I such that each of them is reachable following a staircase starting at a for n steps, i.e., if there is

- a $v_i h_i$-path (respectively $h_i v_i$-path) of length $2n$ from a to b, and
- a $h_{i\ominus 1} v_{i\ominus 1}$-path (respectively $v_{i\oplus 1} h_{i\oplus 1}$-path) of length $2n$ from a to b'.

To this purpose, we add super roles x_i of h_i and y_i of v_i (for which we use dashed arrows in Fig. 1), and the last group of role inclusion axioms in $\mathcal{R}_{\mathcal{D}}$. These role inclusion axioms ensure appropriate, additional role successorships between elements, and we use the additional roles x_i and y_i since we only want to have at most one v_i or h_i-successor.

For each 2 staircases starting at the same element on one of the axes, these role inclusions ensure that each pair of elements representing the same point is related by y_i. That is, each element on an $i \oplus 1$-staircase that is an $x_{i\oplus 1}$-successor is related via y_i to the element on the i-staircase (which is a v_i-successor) representing the same point (see Fig. 1).

To see this, start by considering the consequences of the role inclusion axioms for elements representing the four points $(1,0)$, $(2,0)$, $(1,1)$ and $(2,1)$. The elements representing $(1,0)$ and $(2,1)$ are related via $h_3 v_3$ and $v_0 h_0$, and as we cannot force these two paths to end in the same element, we might have two elements representing $(2,1)$. From the axioms $h_3 \dot{\sqsubseteq} x_3$, $v_3 \dot{\sqsubseteq} y_3$, $v_0 \dot{\sqsubseteq} y_0$ and $h_0 \dot{\sqsubseteq} x_0$, we see that $(1,0)$ and $(2,1)$ are also related via $x_3 y_3$ and $y_0 x_0$. Using the axiom $y_0^- x_3 \dot{\sqsubseteq} x_3$ first, then $x_0^- x_3 \dot{\sqsubseteq} x_0^-$, and finally $x_0^- y_3 \dot{\sqsubseteq} y_3$, we also see that, if there are two elements representing the point $(2,1)$, then they are related via y_3. Next, consider elements representing the four points $(2,1)$, $(2,2)$, $(3,1)$ and $(3,2)$, start with the axiom $y_0^- y_3 \dot{\sqsubseteq} y_0^-$, and then continue to work through the same role inclusion axioms as above. Repeating this argumentation, all elements on these two staircases that represent the same point can be seen to be related via the relation y_3. From an analogous argumentation for other pairs of staircases, using corresponding sets of role inclusion axioms, it follows that the last GCI in $\mathcal{T}_\mathcal{D}$ ensures that two elements representing the same point in the grid do indeed have the same domino type associated with them.

The above observations imply that the concept $C_\mathcal{D}$ is satisfiable w.r.t. $\mathcal{T}_\mathcal{D}$ and $\mathcal{R}_\mathcal{D}$ iff $\mathcal{D}$ has a solution. Hence, together with Theorem 4, we have the following:

Theorem 6. *Satisfiability of $\mathcal{SH}^+\mathcal{IQ}$-concepts w.r.t. generalized role hierarchies is undecidable.*

As mentioned above, the usage of inverse roles on the right-hand side in RIAs of $\mathcal{R}_\mathcal{D}$ is of no importance: we can replace these RIAs with equivalent ones with role names on their right-hand side, e.g., we can replace $x_{i\oplus 1}^- x_i \dot{\sqsubseteq} x_{i\oplus 1}^-$ with $x_i^- x_{i\oplus 1} \dot{\sqsubseteq} x_{i\oplus 1}$. However, we have chosen the representation in Fig. 2 to make the relationship with the grammar from Section 2.1 more clear.

4. $\mathcal{RIQ}$ is decidable

In this section, we show that $\mathcal{SHIQ}$ with *regular* role hierarchies is decidable, where “regular” is both a restriction and a generalisation of “generalised”. On the one hand, we restrict role hierarchies to be *acyclic*, where acyclic role hierarchies still allow for RIAs of the form $RS \dot{\sqsubseteq} S$, $SR \dot{\sqsubseteq} S$, $SS \dot{\sqsubseteq} S$, and $R^- \dot{\sqsubseteq} R$. Moreover, for convenience of proofs, we restrict our attention to RIAs with a role *name* on their right-hand side. As mentioned above, this is of no importance. On the other hand, we also allow for axioms of the form $R_1 \ldots R_n S \dot{\sqsubseteq} S$ and $S R_1 \ldots R_n \dot{\sqsubseteq} S$ (for $\mathcal{SH}^+\mathcal{IQ}$, we restricted n to be 1). Finally, we also allow for statements that force roles to be *symmetric*, i.e., in contrast to the decidable case in [15], regularity also allows for RIAs of the form $\mathsf{Inv}(S) \dot{\sqsubseteq} S$.

We present a tableau-based algorithm that decides satisfiability of $\mathcal{RIQ}$-concepts w.r.t. regular role hierarchies, and therefore also subsumption in $\mathcal{RIQ}$ and, with Theorem 4, both inferences w.r.t. terminologies. The FaCT system [12] was extended to use the algorithm presented in this section, and the empirical results are reported in Section 5.

The algorithm tries to construct, for a $\mathcal{RIQ}$-concept C, a *tableau* for C, that is, an abstraction of a model of C. Given the appropriate notion of a tableau, it is then quite straightforward to prove that the algorithm is a decision procedure for $\mathcal{RIQ}$-satisfiability. Before specifying this algorithm, we translate the role hierarchy into non-deterministic automata which are used both in the definition of a tableau and in the tableau algorithm. Intuitively, an automaton is used to memorise the path between an object x that has to satisfy a concept of the form $\forall R . C$ and other objects, and then to determine which of these objects must satisfy C.[1]

In the following definition of general role hierarchies, we use a strict partial order $\prec$ (irreflexive, transitive, and antisymmetric) on roles to ensure acyclicity.

Definition 7. Let $\prec$ be a strict partial order on role names. A RIA $w \,\dot{\sqsubseteq}\, R$ is *$\prec$-regular* if

- R is a role name,
- $w = RR$,
- $w = R^-$,
- $w = S_1 \ldots S_n$ and $S_i \prec R$, for all $1 \leqslant i \leqslant n$,
- $w = RS_1 \ldots S_n$ and $S_i \prec R$, for all $1 \leqslant i \leqslant n$, or
- $w = S_1 \ldots S_n R$ and $S_i \prec R$, for all $1 \leqslant i \leqslant n$.

A role hierarchy $\mathcal{R}$ is *regular* if there exists a strict partial order $\prec$ such that each RIA in $\mathcal{R}$ is *$\prec$-regular*. The semantics is defined analogously to the semantics of generalised role hierarchies, i.e., $\mathcal{I}$ satisfies a RIA $w \,\dot{\sqsubseteq}\, R$ if $w^{\mathcal{I}} \subseteq R^{\mathcal{I}}$.

$\mathcal{RIQ}$ is obtained from $\mathcal{SH}^+\mathcal{IQ}$ by replacing generalised role hierarchies with regular role hierarchies, where *simple role names* are inductively defined as follows:[2]

- every role name that does not occur on the right-hand side of a RIA is simple,
- a role name S is simple if, for each $w \,\dot{\sqsubseteq}\, S \in \mathcal{R}$, $w = R$ for R a simple role or the inverse of a simple role.

An inverse role S^- is simple if S is simple.

Please note that, due to the third restriction in the definition of R-compatibility, we also restrict $\underline{\boxast}$ to be acyclic. However, this is not a serious restriction since, for $\mathcal{R}$ containing $\underline{\boxast}$ cycles, we can simply choose one role R from each cycle and replace all other roles on this cycle with R, both in the input role hierarchy and the input concept.

For the following considerations, it is worthwhile to recall that, for $w = R_1 \ldots R_m$ and R_i roles, $\mathsf{Inv}(w) = \mathsf{Inv}(R_m) \ldots \mathsf{Inv}(R_1)$. The following lemma is a direct consequence of the definition of the semantics.

---

[1] This technique together with the relationship between automata and regular languages is the reason why we called these role hierarchies "regular".

[2] We need to re-define "simple" roles because of the more general form of RIAs.

Lemma 8. *If $\mathcal{I}$ is a model of $\mathcal{R}$ with $S^- \dot{\sqsubseteq} S \in \mathcal{R}$ and $w \dot{\sqsubseteq} S \in \mathcal{R}$, then $\mathsf{Inv}(w)^{\mathcal{I}} \subseteq S^{\mathcal{I}}$.*

4.1. Translating RIAs into automata

Next, we will define, for a regular role hierarchy $\mathcal{R}$ and a (possibly inverse) role S occurring in $\mathcal{R}$, a non-deterministic finite automaton (NFA) $\mathcal{B}_S$ which captures all implications between (paths of) roles and S that are consequences of $\mathcal{R}$. To make this clear, before we define $\mathcal{B}_S$, we formulate the lemma which we are going to prove for it.

Proposition 9. *$\mathcal{I}$ is a model of $\mathcal{R}$ if and only if, for each (possibly inverse) role S occurring in $\mathcal{R}$, each word $w \in L(\mathcal{B}_S)$, and each $\langle x, y\rangle \in w^{\mathcal{I}}$, we have $\langle x, y\rangle \in S^{\mathcal{I}}$.*

In [15], to construct a similar automaton for a more restricted logic, we first unfolded $\mathcal{R}$ into a set of implications between regular expressions, and then constructed the automata from these implications. Here, we show how to build these automata directly, which yields an easier construction.

In the following, we use NFAs with ε-transitions in a rather informal way (see, e.g., [11] for more details), e.g., we use $p \xrightarrow{R} q$ to denote that there is a transition from a state p to a state q with the letter R instead of introducing transition relations formally. The automata $\mathcal{B}_S$ are defined in three steps.

Definition 10. Let C_0 be a $\mathcal{RIQ}$-concept and $\mathcal{R}$ a regular role hierarchy.

For each role name R occurring in $\mathcal{R}$ or C_0, we first define the NFA $\mathcal{A}_R$ as follows: $\mathcal{A}_R$ contains a state i_R and a state f_R with the transition $i_R \xrightarrow{R} f_R$. The state i_R is the only initial state and f_R is the only final state. Moreover, for each $w \dot{\sqsubseteq} R \in \mathcal{R}$, $\mathcal{A}_R$ contains the following states and transitions:

(1) if $w = RR$, then $\mathcal{A}_R$ contains $f_R \xrightarrow{\varepsilon} i_R$, and
(2) if $w = R_1 \cdots R_n$ and $R_1 \neq R \neq R_n$, then $\mathcal{A}_R$ contains

$$i_R \xrightarrow{\varepsilon} i_w \xrightarrow{R_1} f_w^1 \xrightarrow{R_2} f_w^2 \xrightarrow{R_3} \cdots \xrightarrow{R_n} f_w^n \xrightarrow{\varepsilon} f_R,$$

(3) if $w = RR_2 \cdots R_n$, then $\mathcal{A}_R$ contains

$$f_R \xrightarrow{\varepsilon} i_w \xrightarrow{R_2} f_w^2 \xrightarrow{R_3} f_w^3 \xrightarrow{R_4} \cdots \xrightarrow{R_n} f_w^n \xrightarrow{\varepsilon} f_R,$$

(4) if $w = R_1 \cdots R_{n-1}R$, then $\mathcal{A}_R$ contains

$$i_R \xrightarrow{\varepsilon} i_w \xrightarrow{R_1} f_w^1 \xrightarrow{R_2} f_w^2 \xrightarrow{R_3} \cdots \xrightarrow{R_{n-1}} f_w^{n-1} \xrightarrow{\varepsilon} i_R,$$

where all f_w^i, i_w are assumed to be distinct.

In the next step, we use a *mirrored copy* of NFAs: this is a copy of an NFA in which we have carried out the following modifications: we

- make final states to non-final but initial states,
- make initial states to non-initial but final states,

- replace each transition $p \xrightarrow{S} q$ for S a (possibly inverse) role S with $q \xrightarrow{\mathsf{Inv}(S)} p$, and
- replace each transition $p \xrightarrow{\varepsilon} q$ with $q \xrightarrow{\varepsilon} p$.

Secondly, we define the NFAs $\hat{\mathcal{A}}_R$ as follows:

- if $R^- \dot{\sqsubseteq} R \notin \mathcal{R}$, then $\hat{\mathcal{A}}_R := \mathcal{A}_R$,
- if $R^- \dot{\sqsubseteq} R \in \mathcal{R}$, then $\hat{\mathcal{A}}_R$ is obtained as follows: first, take the disjoint union[3] of $\mathcal{A}_S$ with a mirrored copy of $\mathcal{A}_S$. Secondly, make i_R the only initial state, f_R the only final state. Finally, for f'_R the copy of f_R and i'_R the copy of i_R, add transitions $i_R \xrightarrow{\varepsilon} f'_R$, $f'_R \xrightarrow{\varepsilon} i_R$, $i'_R \xrightarrow{\varepsilon} f_R$, and $f_R \xrightarrow{\varepsilon} i'_R$.

Thirdly, the NFAs $\mathcal{B}_R$ are defined inductively over $\prec$:

- if R is minimal w.r.t. $\prec$ (i.e., there is no R' with $R' \prec R$), we set $\mathcal{B}_R := \hat{\mathcal{A}}_R$,
- otherwise, $\mathcal{B}_R$ is the disjoint union of $\hat{A}_R$ with a copy $\mathcal{B}'_S$ of $\mathcal{B}_S$ for each transition $p \xrightarrow{S} q$ in $\hat{A}_R$ with $S \neq R$. Moreover, for each such transition, we add ε-transitions from p to the initial state in $\mathcal{B}'_S$ and from the final state in $\mathcal{B}'_S$ to q, and we make i_R the only initial state and f_R the only final state in $\mathcal{B}_R$.

Finally, the automaton $\mathcal{B}_{R^-}$ is a mirrored copy of $\mathcal{B}_R$.

Please note that the inductive definition $\mathcal{B}_R$ is well-defined since the acyclic relation $\prec$ is used to restrict the dependencies between roles.

We have kept the construction of $\mathcal{B}_S$ as simple as possible. If one wants to construct an equivalent NFA without ε-transitions or which is deterministic, then there are well-known techniques to do this [11]. Recall that elimination of ε-transitions can be carried out without increasing the number of an automaton's states, whereas determinisation might yield an exponential blow-up.

Lemma 11. *For R a role, the size of $\mathcal{B}_R$ is bounded exponentially in the depth*

$$d_{\mathcal{R}} := \max\{n \mid \textit{there are } S_1 \prec \cdots \prec S_n, u_i, v_i \textit{ with } u_i S_{i-1} v_i \dot{\sqsubseteq} S_i \in \mathcal{R}\}$$

and thus in the size of $\mathcal{R}$. Moreover, there are $\mathcal{R}$ and R such that the number of states in $\mathcal{B}_R$ is $2^{d_{\mathcal{R}}}$.

Proof. Obviously, the size of $\mathcal{A}_R$ and $\hat{\mathcal{A}}_R$ is linear in

$$b_{\mathcal{R}} = \max\bigl\{|w_1| + \cdots + |w_k| \mid \text{there is } S \text{ with } w_i \dot{\sqsubseteq} S \in \mathcal{R} \text{ for all } 1 \leqslant i \leqslant n\bigr\}.$$

Each automaton $\mathcal{B}_R$ is a "tree" of automata $\mathcal{A}_S$ whose

- outdegree is bounded by $b_{\mathcal{R}}$ and
- whose depth is bounded by $d_{\mathcal{R}}$.

[3] A disjoint union of two automata is the disjoint union of their states, transition relations, etc.

Hence the number of $\mathcal{B}_R$'s states is bounded exponentially in $d_{\mathcal{R}}$ and, since $d_{\mathcal{R}}$ is linear in the size of $\mathcal{R}$, also bounded exponentially in the size of $\mathcal{R}$.

Next, it is easily verified that, for the following regular role hierarchy $\mathcal{R}_n$, the automaton $\mathcal{B}_{S_n}$ has 2^{n+1} states and the size of $\mathcal{R}_n$ is linear in n:

$$\mathcal{R}_n = \{S_{i-1}S_i \dot{\sqsubseteq} S_i,\ S_iS_{i-1} \dot{\sqsubseteq} S_i \mid 1 \leqslant i \leqslant n\}. \qquad \square$$

We will consider ways to avoid this exponential blow-up in Section 4.4, and continue with the proof of Proposition 9. In this proof, we will use the following lemma, which is an immediate consequence of the definition of $\mathcal{B}_S$ and of mirrored copies of $\mathcal{B}_S$.

Lemma 12.

(1) *$S \in L(\mathcal{B}_S)$ and, if $w \dot{\sqsubseteq} S \in R$, then $w \in L(\mathcal{B}_S)$.*
(2) *If S is a simple role, then $L(\mathcal{B}_S) = \{R \mid R \underline{\sqsubseteq}_* S\}$.*
(3) *If $\overleftarrow{\mathcal{A}}$ is a mirrored copy of an NFA $\mathcal{A}$, then $L(\overleftarrow{\mathcal{A}}) = \{\mathsf{Inv}(w) \mid w \in L(\mathcal{A})\}$.*

Proof of Proposition 9. The "if" direction is easily proved by contraposition. If $\mathcal{I}$ is *not* a model of $\mathcal{R}$, then there is some RIA $w \dot{\sqsubseteq} S \in \mathcal{R}$ not satisfied by $\mathcal{I}$. Hence there are some x, y such that $\langle x, y\rangle \in w^{\mathcal{I}}$ but $\langle x, y\rangle \notin S^{\mathcal{I}}$. By Lemma 12.1, $w \in L(\mathcal{B}_S)$, and we are done.

For the "only-if" direction, let $\mathcal{I}$ be a model of $\mathcal{R}$, S a role, $w \in L(\mathcal{B}_S)$, and $\langle x, y\rangle \in w^{\mathcal{I}}$. We prove $\langle x, y\rangle \in S^{\mathcal{I}}$ by well-founded induction on $\prec$. Obviously, we can restrict our attention to a role name S due to Lemma 12.3 and since $\mathcal{B}_{S^-}$ is defined as a mirrored copy of $\mathcal{B}_S$.

First, we observe that $w \in L(\mathcal{B}_S)$ induces a decomposition $w = w_1 \dots w_k$ and word $\hat{w} = S_1 \dots S_k$ such that

- $S_i \prec S$ or $S_i = S$ for all $1 \leqslant i \leqslant k$,
- $\hat{w} \in L(\hat{A}_S)$, and
- $w_i \in L(\mathcal{B}_{S_i})$.

Next, $\langle x, y\rangle \in w^{\mathcal{I}}$ implies that there are x_i with $x = x_0$, $y = x_k$, and $\langle x_i, x_{i+1}\rangle \in w_{i+1}^{\mathcal{I}}$, for each $0 \leqslant i < k$. By induction, $\langle x_i, y_i\rangle \in S_i^{\mathcal{I}}$ and thus $\langle x, y\rangle \in \hat{w}^{\mathcal{I}}$.

(1) If $SS \dot{\sqsubseteq} S \notin \mathcal{R}$ and $S^- \dot{\sqsubseteq} S \notin \mathcal{R}$, then, by construction, $\hat{w}$ is of the form

$$\hat{w} = u_1 \dots u_m x v_1 \dots v_n \quad \text{and} \quad \begin{aligned} &u_i S \dot{\sqsubseteq} S \in \mathcal{R}, \text{ for each } 1 \leqslant i \leqslant m \\ &x \dot{\sqsubseteq} S \in \mathcal{R} \text{ or } x = S \\ &S v_j \dot{\sqsubseteq} S \in \mathcal{R}, \text{ for each } 1 \leqslant j \leqslant n \end{aligned}$$

Thus $\mathcal{I}$ being a model of $\mathcal{R}$ implies that $\langle x, y\rangle \in S^{\mathcal{I}}$.

(2) If $SS \dot{\sqsubseteq} S \in \mathcal{R}$ and $S^- \dot{\sqsubseteq} S \notin \mathcal{R}$, then, by construction, $\hat{w}$ is of the form

$$\hat{w} = \big(u_1^{(1)} \dots u_{m_1}^{(1)} x^{(1)} v_1^{(1)} \dots v_{n_1}^{(1)}\big) \dots \big(u_1^{(\ell)} \dots u_{m_\ell}^{(\ell)} x^{(\ell)} v_1^{(\ell)} \dots v_{n_\ell}^{(\ell)}\big) \quad \text{and}$$
$$u_i^{(k)} S \dot{\sqsubseteq} S \in \mathcal{R}, \quad \text{for each } 1 \leqslant i \leqslant m,\ 1 \leqslant k \leqslant \ell$$
$$x^{(k)} \dot{\sqsubseteq} S \in \mathcal{R} \text{ or } x^{(k)} = S, \quad \text{for each } 1 \leqslant k \leqslant \ell$$
$$S v_j^{(k)} \dot{\sqsubseteq} S \in \mathcal{R}, \quad \text{for each } 1 \leqslant j \leqslant n,\ 1 \leqslant k \leqslant \ell$$

Again, $\mathcal{I}$ being a model of $\mathcal{R}$ implies that $\langle x, y\rangle \in S^{\mathcal{I}}$.

(3) If $SS \dot{\sqsubseteq} S \notin \mathcal{R}$ and $S^- \dot{\sqsubseteq} S \in \mathcal{R}$, then $\mathcal{B}_S$ is the disjoint union of $\mathcal{A}_S$ with a mirrored copy of $\mathcal{A}_S$ and additional ε-transitions between the final and initial state and their copies. By construction, we have

$$\hat{w} = u_1 \dots u_m x v_1 \dots v_n \quad \text{and}$$
$$u_i S \dot{\sqsubseteq} S \in \mathcal{R} \text{ or } S\,\mathsf{Inv}(u_i) \dot{\sqsubseteq} S \in R \text{ for each } 1 \leqslant i \leqslant m$$
$$x \dot{\sqsubseteq} S \in \mathcal{R} \text{ or } \mathsf{Inv}(x) \dot{\sqsubseteq} S \in \mathcal{R} \text{ or } x = S \text{ or } x = S^-$$
$$S v_j \dot{\sqsubseteq} S \in \mathcal{R} \text{ or } \mathsf{Inv}(v_j) S \dot{\sqsubseteq} S \in R, \text{ for each } 1 \leqslant j \leqslant n$$

In both cases, $\mathcal{I}$ being a model of $\mathcal{R}$ implies that $\langle x, y\rangle \in S^{\mathcal{I}}$.

(4) If $SS \dot{\sqsubseteq} S \in \mathcal{R}$ and $S^- \dot{\sqsubseteq} S \in \mathcal{R}$, then we are in a mixture of the cases (2) and (3), i.e.,

$$\hat{w} = \hat{w}_1 \dots \hat{w}_r$$

and each $\hat{w}_i$ is accepted by a run through $\mathcal{B}_S$ which neither uses the ε-transition from f_S to i_S nor the corresponding one in the mirrored copy of $\hat{A}_S$. We can decompose each $\hat{w}_i$ as we have decomposed $\hat{w}$ in case (3), and conclude that $\mathcal{I}$ being a model of $\mathcal{R}$ implies that $\langle x, y\rangle \in S^{\mathcal{I}}$. □

4.2. A tableau for $\mathcal{RIQ}$

In the following, if not stated otherwise, C, D (possibly with subscripts) denote $\mathcal{RIQ}$-concepts, R, S (possibly with subscripts) roles, and $\mathcal{R}$ a regular role hierarchy.

We start by defining $\mathsf{fclos}(C_0, \mathcal{R})$, the *closure* of a concept C w.r.t. a regular role hierarchy $\mathcal{R}$. Intuitively, this contains all relevant sub-concepts of C together with universal value restrictions over sets of role paths described by an NFA. We use NFAs in universal value restrictions to memorise the path between an object that has to satisfy a value restriction and other objects. To do this, we "push" this NFA-value restriction along this path while the NFA gets "updated" with the path taken so far. For this "update", we use the following definition.

Definition 13. For $\mathcal{B}$ an NFA and q a state of $\mathcal{B}$, $\mathcal{B}(q)$ denotes the NFA obtained from $\mathcal{B}$ by making q the (only) initial state of $\mathcal{B}$, and we use $q \xrightarrow{S} q' \in \mathcal{B}$ to denote that $\mathcal{B}$ has a transition $q \xrightarrow{S} q'$.

Without loss of generality, we assume all concepts to be in NNF, that is, negation occurs in front of concept names only. Any $\mathcal{RIQ}$-concept can easily be transformed into an equivalent one in NNF by pushing negations inwards using a combination of DeMorgan's laws and the following equivalences:

$$\neg(\exists R\,.\,C) \equiv (\forall R\,.\,\neg C), \qquad \neg(\forall R\,.\,C) \equiv (\exists R\,.\,\neg C),$$
$$\neg(\leqslant nR\,.\,C) \equiv \big(\geqslant (n+1)R\,.\,C\big), \qquad \neg\big(\geqslant (n+1)R\,.\,C\big) \equiv (\leqslant nR\,.\,C),$$
$$\neg(\geqslant 0R\,.\,C) \equiv \bot.$$

We use $\dot{\neg}C$ for the NNF of $\neg C$. Obviously, the length of $\dot{\neg}C$ is linear in the length of C.

For a concept C_0, $\mathsf{clos}(C_0)$ is the smallest set that contains C_0 and that is closed under sub-concepts and $\dot{\neg}$. The set $\mathsf{fclos}(C_0, \mathcal{R})$ is then defined as follows:

$$\mathsf{fclos}(C_0, \mathcal{R}) := \mathsf{clos}(C_0) \cup \big\{\forall \mathcal{B}_S(q)\,.\,D \mid \forall S\,.\,D \in \mathsf{clos}(C_0) \text{ and } \mathcal{B}_S \text{ has a state } q\big\}.$$

It is not hard to show and well known that the size of $\mathsf{clos}(C_0)$ is linear in the size of C_0. For the size of $\mathsf{fclos}(C_0, \mathcal{R})$, we have seen in Lemma 11 that, for a role S, the size of $\mathcal{B}_S$ can be exponential in the depth of $\mathcal{R}$. Since there are at most linearly many concepts $\forall S\,.\,D$, this yields a bound for the cardinality of $\mathsf{fclos}(C_0, \mathcal{R})$ that is exponential in the depth of $\mathcal{R}$ and linear in the size of C_0. Investigating whether this exponential blow-up can be avoided will be part of future work. So far, we only define in Section 4.4 a further syntactic restriction which avoids this exponential blow-up.

We are now ready to define tableaux as a useful abstraction of models.

Definition 14. $T = (\mathbf{S}, \mathcal{L}, \mathcal{E})$ is a *tableau* for C_0 w.r.t. $\mathcal{R}$ iff

- $\mathbf{S}$ is a non-empty set,
- $\mathcal{L} : \mathbf{S} \to 2^{\mathsf{fclos}(C_0, \mathcal{R})}$ maps each element in $\mathbf{S}$ to a set of concepts, and
- $\mathcal{E} : \mathbf{R}_{C_0,\mathcal{R}} \to 2^{\mathbf{S}\times\mathbf{S}}$ maps each role to a set of pairs of elements in $\mathbf{S}$.

Furthermore, for all $s, t \in \mathbf{S}$, $C, C_1, C_2 \in \mathsf{fclos}(C_0, \mathcal{R})$, and $R, S \in \mathbf{R}_{C_0,\mathcal{R}}$, T satisfies:

(P0) there is some $s \in \mathbf{S}$ with $C_0 \in \mathcal{L}(s)$,
(P1) if $C \in \mathcal{L}(s)$, then $\neg C \notin \mathcal{L}(s)$,
(P2) if $C_1 \sqcap C_2 \in \mathcal{L}(s)$, then $C_1 \in \mathcal{L}(s)$ and $C_2 \in \mathcal{L}(s)$,
(P3) if $C_1 \sqcup C_2 \in \mathcal{L}(s)$, then $C_1 \in \mathcal{L}(s)$ or $C_2 \in \mathcal{L}(s)$,
(P4a) if $\forall \mathcal{B}(p)\,.\,C \in \mathcal{L}(s)$, $\langle s, t\rangle \in \mathcal{E}(S)$, and $p \xrightarrow{S} q \in \mathcal{B}(p)$, then $\forall \mathcal{B}(q)\,.\,C \in \mathcal{L}(t)$,
(P4b) if $\forall \mathcal{B}\,.\,C \in \mathcal{L}(s)$ and $\varepsilon \in L(\mathcal{B})$, then $C \in \mathcal{L}(s)$,
(P5) if $\exists S\,.\,C \in \mathcal{L}(s)$, then there is some t with $\langle s, t\rangle \in \mathcal{E}(S)$ and $C \in \mathcal{L}(t)$,
(P6) if $\forall S\,.\,C \in \mathcal{L}(s)$, then $\forall \mathcal{B}_S\,.\,C \in \mathcal{L}(s)$,
(P7) $\langle x, y\rangle \in \mathcal{E}(R)$ iff $\langle y, x\rangle \in \mathcal{E}(\mathsf{Inv}(R))$,
(P8) if $(\leqslant nS\,.\,C) \in \mathcal{L}(s)$, then $\sharp S^T(s, C) \leqslant n$,
(P9) if $(\geqslant nS\,.\,C) \in \mathcal{L}(s)$, then $\sharp S^T(s, C) \geqslant n$,
(P10) if $(\leqslant nS\,.\,C) \in \mathcal{L}(s)$ and $\langle s, t\rangle \in \mathcal{E}(S')$ for some $S' \in L(\mathcal{B}_S)$, then $C \in \mathcal{L}(t)$ or $\dot{\neg}C \in \mathcal{L}(t)$,

where $S^T(s, C) := \{t \in \mathbf{S} \mid \langle s, t\rangle \in \mathcal{E}(S') \text{ for some } S' \in L(\mathcal{B}_S) \text{ and } C \in \mathcal{L}(t)\}$.

Lemma 15. *A $\mathcal{RIQ}$-concept C_0 is satisfiable w.r.t. $\mathcal{R}$ iff there exists a tableau for C_0 w.r.t. $\mathcal{R}$.*

Proof. For the *if* direction, let $T = (\mathbf{S}, \mathcal{L}, \mathcal{E})$ be a tableau for C_0 w.r.t. $\mathcal{R}$. We extend the relational structure of T and then prove that this indeed gives a model. More precisely, a model $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ of D and $\mathcal{R}$ can be defined as follows: we set $\Delta^{\mathcal{I}} := \mathbf{S}$, $A^{\mathcal{I}} := \{s \mid A \in \mathcal{L}(s)\}$ for concept names A in $\mathsf{clos}(C_0)$, and for roles names R, we set

$$R^{\mathcal{I}} := \bigl\{\langle s_0, s_n\rangle \in \bigl(\Delta^{\mathcal{I}}\bigr)^2 \mid \text{ there are } s_1, \ldots, s_{n-1} \text{ with } \langle s_i, s_{i+1}\rangle \in \mathcal{E}(S_{i+1}) \text{ for } 0 \leqslant i \leqslant n-1 \text{ and } S_1 \cdots S_n \in L(\mathcal{B}_R)\bigr\}.$$

The semantics of complex concepts is given through the definition of the $\mathcal{RIQ}$ semantics. Due to Lemma 12.3 and (P7), the semantics of inverse roles can either be given directly as for role names, or by setting $(R^-)^{\mathcal{I}} = \{\langle y, x\rangle \mid \langle x, y\rangle \in R^{\mathcal{I}}\}$.

First, we show that $\mathcal{I}$ is a model of $\mathcal{R}$ and C_0. Due to Proposition 9, it suffices to prove that, for each (possibly inverse) role S, each word $w \in L(\mathcal{B}_S)$, and each $\langle x, y\rangle \in w^{\mathcal{I}}$, we have $\langle x, y\rangle \in S^{\mathcal{I}}$. Let $w \in L(\mathcal{B}_S)$ and $\langle x, y\rangle \in w^{\mathcal{I}}$. For $w = S_1 \ldots S_n$, this implies the existence of y_i such that $y_0 = x$, $y_n = y$, and $\langle y_{i-1}, y_i\rangle \in S_i^{\mathcal{I}}$ for each $1 \leqslant i \leqslant n$. For each i, we define a word w_i as follows:

- if $\langle y_{i-1}, y_i\rangle \in \mathcal{E}(S_i)$, then set $w_i := S_i$,
- otherwise, there is some $v_i = T_1^{(i)} \ldots T_{n_i}^{(i)} \in L(\mathcal{B}_{S_i})$ and there are $y_j^{(i)}$ such that $y_{i-1} = y_0^{(i)}$, $y_i = y_{n_i}^{(i)}$, and $\langle y_{j-1}^{(i)}, y_j^{(i)}\rangle \in \mathcal{E}(T_j^{(i)})$ for each $1 \leqslant j \leqslant n_i$. In this case, we set $w_i := v_i$.

Let $\hat{w} := w_1 \ldots w_n$. By construction of $\mathcal{B}_S$ from $\hat{\mathcal{A}}_S$, $w \in L(\mathcal{B}_S)$ implies that $\hat{w} \in L(\mathcal{B}_S)$. For $\hat{w} = U_1 \ldots U_{n'}$, we can thus re-name the y_i and $y_j^{(i)}$ to z_i such that we have $z_0 = x$, $z_n = y$, and $\langle z_{i-1}, z_i\rangle \in \mathcal{E}(U_i)$. Hence, by definition of $\cdot^{\mathcal{I}}$, we have $\langle x, y\rangle \in S^{\mathcal{I}}$.

Secondly, we prove that $\mathcal{I}$ is a model of C_0. We show that $C \in \mathcal{L}(s)$ implies $s \in C^{\mathcal{I}}$ for each $s \in \mathbf{S}$ and each $C \in \mathsf{clos}(C_0)$. Together with (P0), this implies that $\mathcal{I}$ is a model of C_0. This proof can be given by induction on the length of concepts, where we count neither negation nor integers in number restrictions. The only interesting cases are $C = (\leqslant nS \,.\, E)$ and $C = \forall S \,.\, E$ (for the other cases, see [14,17]):

- If $(\leqslant nS \,.\, E) \in \mathcal{L}(s)$, then (P8) implies that $\# S^T(s, E) \leqslant n$. Moreover, since S is simple, Lemma 12.2 implies that $L(\mathcal{B}_S) = \{S' \mid S' \,\underline{\circledast}\, S\}$, and thus (P10) implies that, for all t, if $\langle s, t\rangle \in S^{\mathcal{I}}$, then $E \in \mathcal{L}(t)$ or $\dot{\neg} E \in \mathcal{L}(t)$. By induction $E^{\mathcal{I}} = \{t \mid E \in \mathcal{L}(t)\}$, and thus $s \in (\leqslant nS \,.\, E)^{\mathcal{I}}$.
- Let $\forall S \,.\, E \in \mathcal{L}(s)$ and $\langle s, t\rangle \in S^{\mathcal{I}}$. From (P6) we have that $\forall \mathcal{B}_S \,.\, E \in \mathcal{L}(s)$. By definition of $S^{\mathcal{I}}$, there are $S_1 \ldots S_n \in L(\mathcal{B}_S)$ and s_i with $s = s_0$, $t = s_n$, and $\langle s_{i-1}, s_i\rangle \in \mathcal{E}(S_i)$. Applying (P4a) n times, this yields $\forall \mathcal{B}_S(q) \,.\, E \in \mathcal{L}(t)$ for q a final state of $\mathcal{B}_S$. Thus (P4b) implies that $E \in \mathcal{L}(t)$. By induction, $t \in E^{\mathcal{I}}$, and thus $s \in (\forall S \,.\, E)^{\mathcal{I}}$.

For the converse, for $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ a model of C_0 w.r.t. $\mathcal{R}$, we define a tableau $T = (\mathbf{S}, \mathcal{L}, \mathcal{E})$ for C_0 and $\mathcal{R}$ as follows:

$$\begin{aligned}
\mathbf{S} &:= \Delta^{\mathcal{I}},\\
\mathcal{E}(R) &:= R^{\mathcal{I}}, \text{ and}\\
\mathcal{L}(s) &:= \{C \in \mathsf{clos}(C_0) \mid s \in C^{\mathcal{I}}\}\\
&\quad \cup \{\forall \mathcal{B}_S . C \mid \forall S . C \in \mathsf{clos}(C_0) \text{ and } s \in (\forall S . C)^{\mathcal{I}}\}\\
&\quad \cup \{\forall \mathcal{B}_R(q) . C \in \mathsf{fclos}(C_0, \mathcal{R}) \mid \text{ for all } S_1 \ldots S_n \in L(\mathcal{B}_R(q)),\\
&\qquad\qquad s \in (\forall S_1 . \forall S_2 \ldots . \forall S_n . C)^{\mathcal{I}} \text{ and}\\
&\qquad\qquad \text{if } \varepsilon \in L(\mathcal{B}_R(q)), \text{ then } s \in C^{\mathcal{I}}\}
\end{aligned}$$

We have to show that T satisfies each (Pi). We restrict our attention to the only new cases (P4) and (P6).

For (P6), if $\forall S . C \in \mathcal{L}(s)$, then $s \in (\forall S . C)^{\mathcal{I}}$ and thus $\forall \mathcal{B}_S . C \in \mathcal{L}(s)$ by definition of T.

For (P4a), let $\forall \mathcal{B}(p) . C \in \mathcal{L}(s)$ and $\langle s, t\rangle \in \mathcal{E}(S) = S^{\mathcal{I}}$. Assume that there is a transition $p \xrightarrow{S} q$ in $\mathcal{B}(p)$ and $\forall \mathcal{B}(q) . C \notin \mathcal{L}(t)$. By definition of T, this can have two reasons:

- there is a word $S_2 \ldots S_n \in L(\mathcal{B}(q))$ and $t \notin (\forall S_2 . \ldots \forall S_n . C)^{\mathcal{I}}$. However, this implies that $SS_2 \ldots S_n \in L(\mathcal{B}(p))$ and thus that $s \in (\forall S . \forall S_2 . \ldots \forall S . C)^{\mathcal{I}}$, which contradicts, together with $\langle s, t\rangle \in S^{\mathcal{I}}$, the definition of the semantics of $\mathcal{RIQ}$ concepts.
- $\varepsilon \in L(\mathcal{B}(q))$ and $t \notin C^{\mathcal{I}}$. This implies that $S \in L(\mathcal{B}(p))$ and thus contradicts $s \in (\forall S . C)^{\mathcal{I}}$.

Hence $\forall \mathcal{B}(q) . C \notin \mathcal{L}(t)$.

For (P4b), $\varepsilon \in L(\mathcal{B}(p))$ implies $s \in C^{\mathcal{I}}$ by definition of T, and thus $C \in \mathcal{L}(s)$. □

4.3. The tableau algorithm

In this section, we present a tableau algorithm that tries to construct, for an input $\mathcal{RIQ}$-concept C_0 and a regular role hierarchy $\mathcal{R}$, a tableau for C_0 w.r.t. $\mathcal{R}$. We prove that this algorithm constructs a tableau for C_0 and $\mathcal{R}$ iff there exists a tableau for C_0 and $\mathcal{R}$, and thus decides satisfiability of $\mathcal{RIQ}$ concepts w.r.t. regular role hierarchies and, using Lemma 3, also w.r.t. terminologies.

This algorithm generates a *completion tree*, a structure that will be unravelled to an (infinite) tableau for the input concept. As usual, in the presence of transitive roles, *blocking* is employed to ensure termination of the algorithm. In the additional presence of inverse roles, blocking is *dynamic*, i.e., blocked nodes (and their sub-branches) can be un-blocked and blocked again later. In the further, additional presence of number restrictions, *pairs* of nodes are blocked rather than single nodes [17]. The blocking conditions as they are presented here are, clearly, too strict. As a consequence, blocking may occur later than necessary, and thus we end up with a search space that is larger than necessary. In [14], we have shown how to loosen the blocking condition for $\mathcal{SHIQ}$ while retaining correctness of the algorithm. Here, we focus on the decidability of $\mathcal{RIQ}$, and defer a similar loosening for $\mathcal{RIQ}$ to future work.

Definition 16. A *completion tree* $\mathbf{T}$ for a $\mathcal{RIQ}$ concept C_0 and a regular role hierarchy $\mathcal{R}$ is a tree, where each node x is labelled with a set $\mathcal{L}(x) \subseteq \mathsf{fclos}(C_0, \mathcal{R})$ and each edge $\langle x, y\rangle$ from a node x to its successor y is labelled with a non-empty set $\mathcal{L}(\langle x, y\rangle)$ of (possibly inverse) roles occurring in C_0 and $\mathcal{R}$. Finally, completion trees come with an explicit inequality relation $\not\doteq$ on nodes which is implicitly assumed to be symmetric.

If $R \in \mathcal{L}(\langle x, y\rangle)$ for a node x and its successor y, then y is called an *R-successor* of x and x is called an $\mathsf{Inv}(R)$*-predecessor* of y. If y is an R-successor or an $\mathsf{Inv}(R)$-predecessor of x, then y is called an R-neighbour of x. Finally, *ancestor* is the transitive closure of *predecessor* and *descendant* is the transitive closure of *successor*.

For a role S, a concept C and a node x in $\mathbf{T}$ we define $S^{\mathbf{T}}(x, C)$ by

$$S^{\mathbf{T}}(x, C) := \big\{y \mid \text{for some } S' \mathrel{\underline{\circledast}} S,\ y \text{ is an } S'\text{-neighbour of } x \text{ and } C \in \mathcal{L}(y)\big\}.$$

A node is *blocked* iff it is either directly or indirectly blocked. A node x is *directly blocked* iff none of its ancestors are blocked, and it has ancestors x', y and y' such that

(1) x is a successor of x' and y is a successor of y' *and*
(2) $\mathcal{L}(x) = \mathcal{L}(y)$ and $\mathcal{L}(x') = \mathcal{L}(y')$ *and*
(3) $\mathcal{L}(\langle x', x\rangle) = \mathcal{L}(\langle y', y\rangle)$.

If there are no descendants x'', y'' of x' and y' with these properties, then we say that y *blocks* x.

A node y is *indirectly blocked* if one of its ancestors is blocked.

For a node x, $\mathcal{L}(x)$ is said to contain a *clash* if

- $\bot \in \mathcal{L}(x)$ or
- for some concept name A, $\{A, \neg A\} \subseteq \mathcal{L}(x)$ or
- there is some concept $(\leqslant n S . C) \in \mathcal{L}(x)$ and $\{y_0, \ldots, y_n\} \subseteq S^{\mathbf{T}}(x, C)$ with $y_i \not\doteq y_j$ for all $0 \leqslant i < j \leqslant n$.

A completion tree is *clash-free* if none of its nodes contains a clash, and it is *complete* if no rule from Fig. 3 can be applied to it.

Given C_0 (in NNF) and $\mathcal{R}$, the algorithm initialises a completion tree consisting only of a root node x_0 labelled with $\{C_0\}$. Then this tree is expanded by repeatedly applying the expansion rules from Fig. 3, stopping when a clash occurs. The algorithm answers "C_0 is satisfiable w.r.t. $\mathcal{R}$" iff the expansion rules can be applied in such a way that they yield a complete and clash-free completion tree, and "C_0 is unsatisfiable w.r.t. $\mathcal{R}$" otherwise.

All but the $\forall_i$-rules have been used before for fragments of $\mathcal{RIQ}$, e.g., $\mathcal{SHIQ}$ [14,16], and the three $\forall_i$-rules are the obvious counterparts to the tableau conditions (P4a), (P4b), and (P6).

As usual, we prove termination, soundness, and completeness of the tableau algorithm to show that it indeed decides satisfiability of $\mathcal{RIQ}$-concepts w.r.t. regular role hierarchies.

Lemma 17. *Let C_0 be a $\mathcal{RIQ}$-concept and $\mathcal{R}$ a regular role hierarchy. The tableau algorithm terminates when started for C_0 and $\mathcal{R}$.*

$\sqcap$-rule: if	$C_1 \sqcap C_2 \in \mathcal{L}(x)$, x is not indirectly blocked, and $\{C_1, C_2\} \not\subseteq \mathcal{L}(x)$
then	$\mathcal{L}(x) \to \mathcal{L}(x) \cup \{C_1, C_2\}$
$\sqcup$-rule: if	$C_1 \sqcup C_2 \in \mathcal{L}(x)$, x is not indirectly blocked, and $\{C_1, C_2\} \cap \mathcal{L}(x) = \emptyset$
then	$\mathcal{L}(x) \to \mathcal{L}(x) \cup \{E\}$ for some $E \in \{C_1, C_2\}$
$\exists$-rule: if	$\exists S . C \in \mathcal{L}(x)$, x is not blocked, and x has no S-neighbour y with $C \in \mathcal{L}(y)$
then	create a new node y with $\mathcal{L}(\langle x, y\rangle) := \{S\}$ and $\mathcal{L}(y) := \{C\}$
$\forall_1$-rule: if	$\forall S . C \in \mathcal{L}(x)$, x is not indirectly blocked, and $\forall \mathcal{B}_S . C \notin \mathcal{L}(x)$
then	$\mathcal{L}(x) \to \mathcal{L}(x) \cup \{\forall \mathcal{B}_S . C\}$
$\forall_2$-rule: if	$\forall \mathcal{B}(p) . C \in \mathcal{L}(x)$, x is not indirectly blocked, $p \xrightarrow{S} q$ in $\mathcal{B}(p)$, and there is an S-neighbour y of x with $\forall \mathcal{B}(q) . C \notin \mathcal{L}(y)$,
then	$\mathcal{L}(y) \to \mathcal{L}(y) \cup \{\forall \mathcal{B}(q) . C\}$
$\forall_3$-rule: if	$\forall \mathcal{B} . C \in \mathcal{L}(x)$, x is not indirectly blocked, $\varepsilon \in L(\mathcal{B})$, and $C \notin \mathcal{L}(x)$
then	$\mathcal{L}(x) \to \mathcal{L}(x) \cup \{C\}$
X-rule: if	$(\leqslant n S . C) \in \mathcal{L}(x)$, x is not indirectly blocked, and there is an S'-neighbour y of x with $S' \mathrel{\underline{\ast}} S$ and $\{C, \dot{\neg} C\} \cap \mathcal{L}(y) = \emptyset$
then	$\mathcal{L}(y) \to \mathcal{L}(y) \cup \{E\}$ for some $E \in \{C, \dot{\neg} C\}$
$\geqslant$-rule: if	$(\geqslant n S . C) \in \mathcal{L}(x)$, x is not blocked, and there are no $y_1, \ldots, y_n \in S^{\mathbf{T}}(x, C)$ with $y_i \not\doteq y_j$ for each $1 \leqslant i < j \leqslant n$
then	create n new nodes $y_1, \ldots, y_n$ with $\mathcal{L}(\langle x, y_i\rangle) = \{S\}$, $\mathcal{L}(y_i) = \{C\}$, and $y_i \not\doteq y_j$ for $1 \leqslant i < j \leqslant n$.
$\leqslant$-rule: if	$(\leqslant n S . C) \in \mathcal{L}(x)$, x is not indirectly blocked, and $\# S^{\mathbf{T}}(x, C) > n$, there are $y, z \in S^{\mathbf{T}}(x, C)$ with *not* $y \not\doteq z$ and y is not an ancestor of z,
then	1. $\mathcal{L}(z) \to \mathcal{L}(z) \cup \mathcal{L}(y)$ and 2. if z is an ancestor of x then $\mathcal{L}(\langle z, x\rangle) \to \mathcal{L}(\langle z, x\rangle) \cup \mathsf{Inv}(\mathcal{L}(\langle x, y\rangle))$ else $\mathcal{L}(\langle x, z\rangle) \to \mathcal{L}(\langle x, z\rangle) \cup \mathcal{L}(\langle x, y\rangle)$ 3. remove y and the sub-tree below y

Fig. 3. The expansion rules for the $\mathcal{RIQ}$ tableau algorithm.

Proof. Let $m = \sharp \mathsf{fclos}(C_0, \mathcal{R})$, n the number of roles occurring in C_0 and $\mathcal{R}$, and $n_{\max} := \max\{n \mid (\geqslant n R . C) \in \mathsf{clos}(C_0)\}$. Termination is a consequence of the following properties of the expansion rules:

(1) Nodes are labelled with subsets of $\mathsf{fclos}(C_0, \mathcal{R})$ and edges with sets of roles occurring in C_0 and $\mathcal{R}$, so there are at most 2^{2mn} different possible labellings for a pair of nodes and an edge. Therefore, if a path p is of length at least 2^{2mn}, the pair-wise blocking condition implies the existence of a node x on p such that x is blocked. Since a path on which nodes are blocked cannot become longer, paths are of length at most 2^{2mn}.

(2) The expansion rules never remove labels from nodes in the tree, and the only rule that removes a node from the tree is the $\leqslant$-rule.
(3) Only the $\exists$- or the $\geqslant$-rule generate new nodes, and each generation is triggered by a concept of the form $\exists R . C$ or $(\geqslant n R . C)$ in the label of a node x. Each of these concepts triggers at most once the generation of at most $n_{\max}$ R-successors y_i of x: note that if the $\leqslant$-rule subsequently causes an R-successor y_i of x to be removed, then x will have some R-neighbour z with $\mathcal{L}(z) \supseteq \mathcal{L}(y_i)$. This, together with the definition of a clash, implies that the rule application which led to the generation of y_i will not be repeated. Since $\mathsf{fclos}(C_0, \mathcal{R})$ contains a total of at most m $\exists R . C$, the out-degree of the tree is bounded by $mn_{\max}$. □

Lemma 18. *Let C_0 be a $\mathcal{RIQ}$-concept and $\mathcal{R}$ a regular role hierarchy. The expansion rules can be applied to C_0 and $\mathcal{R}$ such that they yield a complete and clash-free completion tree if and only if C_0 has a tableau w.r.t. $\mathcal{R}$.*

For the if direction, we can unravel a complete and clash-free completion tree $\mathbf{T}$ in a standard way into a tableau T, where the same technique as for $\mathcal{SHIQ}$ is used to make sure that (P9) is satisfied even if two "sibling" nodes are blocked by the same node. It is easily seen that the $\forall_i$ expansion rules make sure that the resulting structure indeed satisfies the new tableau condition (P4a), (P4b), and (P6).

For the only-if direction, we take a tableau $\mathcal{I}$ of C_0 and $\mathcal{R}$ and use it to steer the application of the non-deterministic rules, i.e., the $\sqcup$-, the X- and the $\leqslant$-rule. To do this, while building the completion tree, we define a mapping π from the nodes of the completion tree into the tableau which satisfies the following three conditions:

$$\left.\begin{array}{l} \mathcal{L}(x) \subseteq \mathcal{L}\big(\pi(x)\big), \\ \text{if } y \text{ is an } S\text{-neighbour of } x, \text{ then } \big\langle \pi(x), \pi(y) \big\rangle \in \mathcal{E}(S), \text{ and} \\ x \neq y \text{ implies } \pi(x) \neq \pi(y). \end{array}\right\} \qquad (*)$$

We start with π mapping the root node to some tableau element s_0 with C_0 in its label, and prove that, if an expansion rule is applicable to $\mathbf{T}$, then this rule can be applied in such a way that $(*)$ is preserved. As a consequence of this claim, (P1), (P8), and Lemma 17, we thus end with a complete and clash-free completion tree. For a full proof, see [13].

From Theorem 4, Lemmas 15, 17, and 18, we thus have the following theorem:

Theorem 19. *The tableau algorithm decides satisfiability and subsumption of $\mathcal{RIQ}$-concepts with respect to regular role hierarchies and terminologies.*

4.4. Avoiding the blow-up

In the previous section, we have presented an algorithm that decides satisfiability and subsumption of $\mathcal{RIQ}$-concepts with respect to regular role hierarchies and terminologies. Unfortunately, compared to similar algorithms that are implemented in state-of-the-art description logic reasoners [10,12,22] and behave well in many cases, we have here an exponential blow-up: the closure $\mathsf{fclos}(C_0, \mathcal{R})$ is exponential in the depth of $\mathcal{R}$ since we have "unfolded" the regular role hierarchy $\mathcal{R}$ into trees of NFAs. While investigating whether and how this exponential blow-up can be avoided, we observe that a further restriction of the syntax of regular role hierarchies avoids this blow-up:

A regular role hierarchy $\mathcal{R}$ is called *simple* when, for all S_i, T_i, n, m, $1 \leqslant i \leqslant n$, and $1 \leqslant j \leqslant m$, if

(1) $u_i S_i v_i \dot{\sqsubseteq} S_{i+1} \in \mathcal{R}$ and $u'_j T_j v'_j \dot{\sqsubseteq} T_{j+1} \in \mathcal{R}$,
(2) $S_i \neq S_{i+1}$ and $T_j \neq T_{j+1}$,
(3) $S_n = T_m$ and $u_n \neq u'_m$,

then $S_i \neq T_j$.

For a *simple* regular role hierarchy $\mathcal{R}$, the size of each NFA $\mathcal{B}_R$ is only polynomial in the size of $\mathcal{R}$ since each NFA $\mathcal{B}_S$ occurs at most once in $\mathcal{B}_R$.

Lemma 20. *For a $\mathcal{RIQ}$-concept C_0 and a simple regular role hierarchy $\mathcal{R}$, the size of* $\mathsf{fclos}(C_0, \mathcal{R})$ *is polynomial in the size of C_0 and $\mathcal{R}$.*

Thus, for simple role hierarchies, the tableau algorithm presented here is of the same worst case complexity as for $\mathcal{SHIQ}$, namely 2NExpTime. A detailed investigation of the exact complexity will be part of future work.

5. Evaluation of the $\mathcal{RIQ}$ algorithm in FaCT

In order to evaluate the practicability of the above algorithm, we have extended the DL system FaCT [12] to deal with $\mathcal{RIQ}$, and we have carried out a preliminary empirical evaluation.

From a practical point of view, one potential problem with the $\mathcal{RIQ}$ algorithm is that the number of states of automata, and hence the number of different $\forall\mathcal{B}.C$ concepts, could be very large. Moreover, many of these automata could be equivalent (i.e., accept the same languages). As blocking depends on finding ancestor nodes labelled with the same set of concepts, the discovery of blocks could be unnecessarily delayed, and this can lead to a serious degradation in performance [14].

The FaCT implementation addresses these possible problems by transforming all of the initial NFAs into minimal deterministic finite automata (DFAs), using the AT&T FSM Library™ for this purpose [20]. A minimal DFA is constructed for each role, the states in each DFA are uniquely numbered, and the implementation uses concepts of the form $\forall\mathcal{B}.C$, where $\mathcal{B}$ is the number of a state in one of the DFAs. Determinising the automata allows standard minimisation techniques to be used [11], and because the automata are minimal, if $\forall\mathcal{B}.C$ leads to the presence of $\forall\mathcal{B}'.C$ in some successor node (as a result of repeated applications of the $\forall_2$-rule), then $\forall\mathcal{B}.C$ is equivalent to $\forall\mathcal{B}'.C$ iff $\mathcal{B} = \mathcal{B}'$ (and as $\mathcal{B}$ and $\mathcal{B}'$ are numbers, such comparisons are very easy). Unnecessary blocking delays are thus avoided.

The implementation is still at the "beta" stage, but it has been possible to carry out some preliminary tests using the well-known Galen medical terminology KB [12,25]. This KB contains 2,740 named concepts and 413 roles, 26 of which are transitive. The roles are arranged in a relatively complex hierarchy with a maximum depth of 10. Classifying this KB using FaCT's $\mathcal{SHIQ}$ reasoner takes 116 s on an 800 MHz Pentium III equipped Linux PC. Classifying the same KB using the new $\mathcal{RIQ}$ reasoner took a total of 275 s on the same machine. This result is encouraging as it shows that, in the case of the Galen KB at least, using automata in $\forall\mathcal{B}.C$ concepts does not lead to a serious degradation in performance. Moreover, the time taken by the $\mathcal{RIQ}$ reasoner includes approximately 100 s to compute the minimal deterministic automata for the role box. This overhead could become important if optimisations of the $\mathcal{RIQ}$ reasoner result in even better performance, but it should be noted that (a) this is a preprocessing step that will not need to be repeated when the remainder of the KB is extended, modified or queried, and (b) compared to other KBs we have seen, the Galen KB involves an unusually large and complex role box.

The KB was then extended with several role inclusion axioms that express the propagation of location across various partonomic roles. These included

$$\texttt{hasLocation}\ \texttt{isSolidDivisionOf} \dot{\sqsubseteq} \texttt{hasLocation}$$

and

$$\texttt{hasLocation}\ \texttt{isLayerOf} \dot{\sqsubseteq} \texttt{hasLocation}.$$

Classifying the extended KB took 280 s, an increase of only 2% (3.5% if we exclude the NFA computation time). Subsumption queries w.r.t. this KB revealed that, e.g.,

$$\texttt{Fracture} \sqcap \exists \texttt{hasLocation}.\texttt{NeckOfFemur}$$

was implicitly a kind of

$$\texttt{Fracture} \sqcap \exists \texttt{hasLocation}.\texttt{Femur}$$

(`NeckOfFemur` is a solid division of `Femur`), and

$$\texttt{Ulcer} \sqcap \exists \texttt{hasLocation}.\texttt{GastricMucosa}$$

was implicitly a kind of

$$\texttt{Ulcer} \sqcap \exists \texttt{hasLocation}.\texttt{Stomach}$$

(`GastricMucosa` is a layer of `Stomach`). None of these subsumption relationships held w.r.t. the original KB. The times taken to compute these relationships w.r.t. the classified KB could not be measured accurately as they were of the same order as a system clock tick (10 ms).

6. Summary and outlook

Motivated (primarily) by medical terminology applications, we have investigated the decidability of the well-known expressive DL, $\mathcal{SHIQ}$, extended with RIAs of the form $RS \dot{\sqsubseteq} P$. We have shown that this extension is undecidable even when RIAs are restricted to the forms $RS \dot{\sqsubseteq} R$ or $SR \dot{\sqsubseteq} R$, but that decidability can be regained by further restricting sets of RIAs to *regular* ones. In the presence of inverse roles, this is slightly tricky, and is realised here using a partial order on role names to prevent cyclic dependencies between roles. The definition of regular sets of RIAs aimed at being as general as possible, and still allows for RIAs of the form $RS \dot{\sqsubseteq} S$, $SR \dot{\sqsubseteq} S$, $SS \dot{\sqsubseteq} S$, and $R^- \dot{\sqsubseteq} R$.

We have presented a tableau algorithm for this DL and reported on its implementation in the FaCT system. A preliminary evaluation suggests that the algorithm will perform well in realistic applications and demonstrates that it can provide important additional functionality in a medical terminology application.

Acknowledgements

We would like to thank Stephane Demri for useful suggestions and discussions.

References

[1] F. Baader, Augmenting concept languages by transitive closure of roles: An alternative to terminological cycles, in: Proc. of the 12th Int. Joint Conf. on Artificial Intelligence, IJCAI-91, Sydney, 1991.
[2] F. Baader, H.-J. Bürckert, B. Nebel, W. Nutt, G. Smolka, On the expressivity of feature logics with negation, functional uncertainty, and sort equations, J. Logic Language Inform. 2 (1993) 1–18.
[3] M. Baldoni, Normal multimodal logics: Automatic deduction and logic programming extension. Ph.D. Thesis, Dipartimento di Informatica, Università degli Studi di Torino, Italy, 1998.
[4] M. Baldoni, L. Giordano, A. Martelli, A tableau calculus for multimodal logics and some (un)decidability results, in: H.C.M de Swart (Ed.), Automated Reasoning with Analytic Tableaux and Related Methods, International Conference, Tableaux-98, Oisterwijk, The Netherlands, in: Lecture Notes in Artif. Intell., vol. 1397, Springer, Berlin, 1998, pp. 44–59.
[5] R. Berger, The Undecidability of the Dominoe Problem, Mem. Amer. Math. Soc., vol. 66, AMS, Providence, RI, 1966.
[6] R.J. Brachman, J. Schmolze, An overview of the KL-ONE knowledge representation system, Cognitive Science 9 (2) (1985) 171–216.
[7] G. De Giacomo, M. Lenzerini, Boosting the correspondence between description logics and propositional dynamic logics (extended abstract), in: Proc. of the 12th Nat. Conf. on Artificial Intelligence, AAAI-94, Seattle, WA, AAAI Press, 1994, pp. 205–212.
[8] S. Demri, The complexity of regularity in grammar logics and related modal logics, J. Logic Comput. 11 (6) (2001) 933–960.
[9] L. Farinãs del Cerro, M. Penttonen, Grammar logics, Logique et Analyse 121–122 (1988) 123–134.
[10] V. Haarslev, R. Möller, RACER system description, in: Proc. of the Int. Joint Conf. on Automated Reasoning, IJCAR-01, Siena, Italy, in: Lecture Notes in Artif. Intell., vol. 2083, Springer, Berlin, 2001, pp. 701–706.
[11] J.E. Hopcroft, J.D. Ullman, Introduction to Automata Theory, Languages, and Computation, Addison-Wesley, Reading, MA, 1997.
[12] I. Horrocks, Using an expressive description logic: FaCT or fiction?, in: Proc. of the 6th Int. Conf. on the Principles of Knowledge Representation and Reasoning, KR-98, Trento, Italy, Morgan Kaufmann, Los Altos, 1998, pp. 636–649.
[13] I. Horrocks, U. Sattler, Decidability of $\mathcal{SHIQ}$ with complex role inclusion axioms, Tech. Rep. LTCS-Report 02-06, TU-Dresden, Germany, 2002.
[14] I. Horrocks, U. Sattler, Optimised reasoning for $\mathcal{SHIQ}$, in: Proc. of the 15th European Conf. on Artificial Intelligence, Lyon, France, ECAI 2002, 2002, pp. 277–281.
[15] I. Horrocks, U. Sattler, Decidability of $\mathcal{SHIQ}$ with complex role inclusion axioms, in: Proc. of the 17th Int. Joint Conf. on Artificial Intelligence, IJCAI-03, Acapulco, Mexico, Morgan Kaufmann, Los Altos, 2003, pp. 343–348; a long version is available as technical report LTCS 02-06 at http://lat.inf.tu-dresden.de/research/reports.html.

[16] I. Horrocks, U. Sattler, S. Tobies, Practical reasoning for expressive description logics, in: H. Ganzinger, D. McAllester, A. Voronkov (Eds.), Proc. of the 6th Int. Conf. on Logic for Programming and Automated Reasoning (LPAR '99), in: Lecture Notes in Artif. Intell., vol. 1705, Springer, Berlin, 1999, pp. 161–180.

[17] I. Horrocks, U. Sattler, S. Tobies, Reasoning with individuals for the description logic SHIQ, in: D. MacAllester (Ed.), Proc. of the 17th Conf. on Automated Deduction, CADE-17, Pittsburgh, PA, in: Lecture Notes in Comput. Sci., vol. 1831, Springer, Berlin, 2000, pp. 482–496.

[18] O. Kupferman, U. Sattler, M.Y. Vardi, The complexity of the graded μ-calculus, in: Proc. of the 18th Conf. on Automated Deduction, CADE-18, Copenhagen, Denmark, in: Lecture Notes in Artif. Intell., vol. 2392, Springer, Berlin, 2002, pp. 423–437.

[19] D.B. Lenat, R.V. Guha, Building Large Knowledge-Based Systems, Addison-Wesley, Reading, MA, 1989.

[20] M. Mohri, F.C.N. Pereira, M. Riley, A Rational Design for a Weighted Finite-State Transducer Library, Lecture Notes in Comput. Sci., vol. 1436, Springer, Berlin, 1998.

[21] L. Padgham, P. Lambrix, A framework for part-of hierarchies in terminological logics, in: J. Doyle, E. Sandewall, P. Torasso (Eds.), Proc. of the 4th Int. Conf. on the Principles of Knowledge Representation and Reasoning, KR-94, Bonn,Germany, 1994, pp. 485–496.

[22] P.F. Patel-Schneider, I. Horrocks, DLP and FaCT, in: Proc. of the Int. Conf. on Automated Reasoning with Analytic Tableaux and Related Methods, Tableaux-99, Saratoga Springs, NY, in: Lecture Notes in Artif. Intell., vol. 1397, Springer, Berlin, 1999, pp. 19–23.

[23] A. Rector, Analysis of propagation along transitive roles: Formalisation of the galen experience with medical ontologies, in: Proc. of the 2001 Description Logic Workshop, DL 2002, CEUR, 2002, http://ceur-ws.org/.

[24] A. Rector, S. Bechhofer, C.A. Goble, I. Horrocks, W.A. Nowlan, W.D. Solomon, The GRAIL concept modelling language for medical terminology, AI in Medicine 9 (1997) 139–171.

[25] A. Rector, I. Horrocks, Experience building a large, re-usable medical ontology using a description logic with transitivity and concept inclusions, in: Proc. of the WS on Ontological Engineering, AAAI Spring Symposium, AAAI '97, AAAI Press, 1997.

[26] U. Sattler, Description logics for the representation of aggregated objects, in: W. Horn (Ed.), Proc. of the 14th European Conf. on Artificial Intelligence, Berlin, Germany, ECAI 2000, IOS Press, Amsterdam, 2000, pp. 239–243.

[27] K. Schild, A correspondence theory for terminological logics: Preliminary report, in: Proc. of the 12th Int. Joint Conf. on Artificial Intelligence, IJCAI-91, Sydney, 1991, pp. 466–471.

[28] M. Schmidt-Schauss, Subsumption in KL-ONE is undecidable, in: Proc. of the 1st Int. Conf. on the Principles of Knowledge Representation and Reasoning, KR-89, Toronto, ON, 1989, pp. 421–431.

[29] M. Schmidt-Schauß, G. Smolka, Attributive concept descriptions with complements, Artificial Intelligence 48 (1) (1991) 1–26.

[30] S. Schulz, U. Hahn, Parts, locations, and holes—formal reasoning about anatomical structures, in: Proc. of AIME 2001, Cascais, Portugal, in: Lecture Notes in Artif. Intell., vol. 2101, Springer, Berlin, 2001, pp. 293–303.

[31] K. Spackman, Managing clinical terminology hierarchies using algorithmic calculation of subsumption: Experience with SNOMED-RT, Journal of the American Medical Informatics Association (2000), Special Issue.

[32] S. Tobies, PSPACE reasoning for graded modal logics, J. Logic Comput. 11 (1) (2001) 85–106.

[33] M. Wessel, Obstacles on the way to qualitative spatial reasoning with description logics: Some undecidability results, in: Proc. of the 2001 Description Logic Workshop, DL 2001, CEUR, 2001, http://ceur-ws.org/.