An Automata-based Algorithm for Description Logics around $\mathcal{SRIQ}$ ⋆

Magdalena Ortiz

Institute of Information Systems, Vienna University of Technology
ortiz@kr.tuwien.ac.at

Abstract. In this paper we use automata-theoretic techniques to tightly characterize the worst-case complexity of the knowledge base satisfiability problem for the very expressive Description Logics (DLs) $\mathcal{ALCQIb}^+_{reg}$ and $\mathcal{SRIQ}$. The logic $\mathcal{ALCQIb}^+_{reg}$ extends $\mathcal{ALC}$ with qualified number restrictions, inverse roles, safe Boolean role expressions, regular expressions over roles, and concepts of the form $\exists P.\mathsf{Self}$ in the style of $\mathcal{SRIQ}$, a well known DL closely related to the new Semantic Web Standard OWL 2. By reducing its knowledge base satisfiability problem to the emptiness test of an automaton on infinite trees, we show that all these additions do not increase the worst case complexity of $\mathcal{ALC}$ and provide a decision procedure for one of the most expressive DLs that have been shown to be decidable in exponential time. We also close the open question of the precise complexity of reasoning in $\mathcal{SRIQ}$, exploiting a reduction from the $\mathcal{SRIQ}$ knowledge base satisfiability problem into the $\mathcal{ALCQIb}^+_{reg}$ one.

1 Introduction

Description Logics [1] are a well-established branch of logics for knowledge representation and reasoning, and the premier formalisms for modeling ontologies that describe application domains in terms of *concepts* (classes of objects) and *roles* (binary relationships between classes). They have gained increasing attention in areas like data and information integration, peer-to-peer data management and ontology-based data access, as well as in the Semantic Web, where they provide the basis for the standard Web Ontology Languages (OWL) [6].

Recent research in DLs has usually focused on the logics of the so-called $\mathcal{SH}$ family. In particular, the DL $\mathcal{SHIQ}$ is closely related to OWL-Lite and extends the 'basic' $\mathcal{ALC}$ (the minimal propositionally closed DL capable of capturing EXPTIME-hard problems) with *inverse roles* and *number restrictions* (counting), as well as with *role inclusions* and *transitive roles*. The DL known as $\mathcal{SHOIQ}$, underlying OWL-DL, further extends $\mathcal{SHIQ}$ with *nominals*. $\mathcal{SHIQ}$ and $\mathcal{SHOIQ}$ are EXPTIME and NEXPTIME complete, respectively.

Recently, $\mathcal{SHIQ}$ and $\mathcal{SHOIQ}$ were enhanced with *regular role hierarchies* in which the composition of a chain of roles may imply another role. This and other smaller features were included in their extensions known as $\mathcal{SRIQ}$ and $\mathcal{SROIQ}$ respectively; the latter underlies the new OWL 2 standard. For reasoning in them, adaptations of the tableaux algorithms for $\mathcal{SHIQ}$ and $\mathcal{SHOIQ}$ were proposed [7, 8], where in a pre-processing stage, the implications between roles given by the role hierarchy are captured in a set of finite state automata, at the price of exponential blow-up in the size of the representation of the KB. It was recently shown that this exponential blow-up is unavoidable, and that $\mathcal{SRIQ}$ and $\mathcal{SROIQ}$ are 2EXPTIME and 2NEXPTIME hard respectively. Even in the light of these results, no tight upper bounds emerged from [7, 8], as they build on tableaux algorithms that are known not to be optimal in the worst case. For example, the algorithm in [10] and its extension in [7] may require

⋆ This work was partially supported by the Austrian Science Fund (FWF) grant P20840 and the Mexican National Council for Science and Technology (CONACYT) grant 187697.

non-deterministic double exponential time even in the restricted case of $\mathcal{SHIQ}$ KBs. For $\mathcal{SROIQ}$, an optimal 2NEXPTIME upper bound could be established by reducing it (with an exponential blow-up) to the two variable fragment of First Order Logic with counting, but the tight complexity of $\mathcal{SRIQ}$ remained open [11].

In this paper, we introduce an extension of the logic $\mathcal{ALCQI}b_{reg}$ previously considered in [2, 4]. Like $\mathcal{SHIQ}$, $\mathcal{SRIQ}$ and other 'main stream' DLs, $\mathcal{ALCQI}b_{reg}$ extends $\mathcal{ALC}$ with inverses and counting. The main difference is that, instead of role inclusions and other axioms asserting properties of roles, $\mathcal{ALCQI}b_{reg}$ supports regular expressions over roles. They allow $\mathcal{ALCQI}b_{reg}$ to simulate the role hierarchies of $\mathcal{SHIQ}$ and $\mathcal{SRIQ}$ and, together with the presence of (safe) Boolean role constructors, capture most of the features of $\mathcal{SRIQ}$. Since all remaining features are related to the $\exists R.\mathsf{Self}$ constructor, we add it to $\mathcal{ALCQI}b_{reg}$ and obtain its extension $\mathcal{ALCQI}b^+_{reg}$, which is (at least) as expressive as $\mathcal{SRIQ}$ (although less succinct).

As we show in this paper, the $\mathcal{ALCQI}b^+_{reg}$ knowledge base satisfiability problem allows for an elegant reduction to the emptiness test of an automaton on infinite trees. The reduction builds on [4] and yields a worst-case optimal, single exponential time decision procedure. We also present a reduction of the $\mathcal{SRIQ}$ knowledge base satisfiability problem to $\mathcal{ALCQI}b^+_{reg}$ that exponentially increases the size of the input knowledge base; this yields the first 2EXP-TIME decision procedure for $\mathcal{SRIQ}$ and a tight characterization of its worst case complexity.

The paper is organized as follows. In Section 2, after introducing the syntax and semantics of $\mathcal{ALCQI}b^+_{reg}$ knowledge bases, we define their syntactic closure, introduce a normal form for them and a canonical form for their models. These allow us to extend the automata algorithm of [4] to $\mathcal{ALCQI}b^+_{reg}$ in Section 3. In section 4 we show how $\mathcal{SRIQ}$ can be reduced to $\mathcal{ALCQI}b^+_{reg}$. Final discussion and conclusions are given in Section 5.

2 The DL $\mathcal{ALCQI}b^+_{reg}$

We introduce the syntax and semantics of the DL $\mathcal{ALCQI}b^+_{reg}$, a natural extension of $\mathcal{ALCQI}b_{reg}$ [2, 4] with concepts of the form $\exists S.\mathsf{Self}$ and Boolean role inclusion axioms.

Definition 1 ($\mathcal{ALCQI}b^+_{reg}$ concepts and roles). *We consider fixed, countably infinite alphabets of* concept names $\mathbf{C}$ *(also called* atomic concepts*),* role names $\mathbf{R}$ *and individual names* $\mathbf{I}$. *We assume that the set* $\mathbf{C}$ *contains the special concepts* $\top$ *(top) and* $\bot$ *(bottom), while* $\mathbf{R}$ *contains the top (universal) role* T *and the bottom (empty) role* B.

According to the following syntax, we define ($\mathcal{ALCQI}b^+_{reg}$*)* concepts C, C', atomic roles Q, simple roles S, S', *and* roles R, R', *where* $A \in \mathbf{C}$, $P \in \mathbf{R}$ *and* $P \neq \mathsf{T}$.

$$C, C' \longrightarrow A \mid \neg C \mid C \sqcap C' \mid C \sqcup C' \mid \forall R.C \mid \exists R.C \mid \geqslant n\, S.C \mid \leqslant n\, S.C \mid \exists S.\mathit{Self}$$
$$Q \longrightarrow P \mid P^-$$
$$S, S' \longrightarrow Q \mid S \cap S' \mid S \cup S' \mid S \cap \neg S'$$
$$R, R' \longrightarrow \mathsf{T} \mid S \mid R \cup R' \mid R \circ R' \mid R^* \mid id(C)$$

We use $S \setminus S'$ *as a shortcut for* $S \cap \neg S'$. *An* $\mathcal{ALCQI}b^+_{reg}$ expression *is a concept or a role.* Subconcepts, subroles *and* subexpressions *are defined in the natural way.*

Definition 2 (Knowledge Bases). *An (*$\mathcal{ALCQI}b^+_{reg}$*)* assertion *is of the form* $C(a)$, $S(a, b)$, *or* $a \neq b$, *where* C *is a concept,* S *a simple role and* $a, b \in \mathbf{I}$. *An (*$\mathcal{ALCQI}b^+_{reg}$*)* ABox *is a set of assertions. An (*$\mathcal{ALCQI}b^+_{reg}$*)* concept inclusion axiom (CIA) *is an expression* $C \sqsubseteq C'$ *for arbitrary concepts* C *and* C'. *An* Boolean role inclusion axiom (BRIA) *is an expression* $S \sqsubseteq S'$ *for simple roles* S *and* S'. *An* $\mathcal{ALCQI}b^+_{reg}$ TBox *is a set of CIAs and BRIAs.*

An ($\mathcal{ALCQI}b^+_{reg}$) knowledge base *(KB) is a pair* $\mathcal{K} = \langle \mathcal{A}, \mathcal{T} \rangle$ *where* $\mathcal{T}$ *is an* $\mathcal{ALCQI}b^+_{reg}$ *TBox and* $\mathcal{A}$ *is a non-empty* $\mathcal{ALCQI}b^+_{reg}$ *ABox.We denote by* $\mathbf{C}_\mathcal{K}$ *the set of atomic concepts occurring in* $\mathcal{K}$*, by* $\mathbf{R}_\mathcal{K}$ *the set of roles names occurring in* $\mathcal{K}$*, by* $\overline{\mathbf{R}}_\mathcal{K}$ *the set* $\mathbf{R}_\mathcal{K} \cup \{P^- \mid P \in \mathbf{R}_\mathcal{K}\}$*, and by* $\mathbf{I}_K$ *the individuals in* $\mathcal{K}$*.*

Definition 3 (Semantics). *An* interpretation $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$ *consists of a non-empty* domain $\Delta^\mathcal{I}$ *and a* valuation function $\cdot^\mathcal{I}$ *that maps each individual* $a \in \mathbf{I}$ *to an element* $a^\mathcal{I} \in \Delta^\mathcal{I}$*, each concept name* $A \in \mathbf{C}$ *to a set* $A^\mathcal{I} \subseteq \Delta^\mathcal{I}$*, and each role name* $P \in \mathbf{R}$ *to a set of pairs* $R^\mathcal{I} \subseteq \Delta^\mathcal{I} \times \Delta^\mathcal{I}$*, in such a way that* $\top^\mathcal{I} = \Delta^\mathcal{I}$*,* $\bot^\mathcal{I} = \emptyset$*,* $\mathsf{T}^\mathcal{I} = \Delta^\mathcal{I} \times \Delta^\mathcal{I}$ *and* $\mathsf{B}^\mathcal{I} = \emptyset$*. The function* $\cdot^\mathcal{I}$ *is inductively extended to all concepts and roles as follows:*

$$
\begin{array}{rlrl}
(\neg C)^\mathcal{I} &= \Delta^\mathcal{I} \setminus C^\mathcal{I} & (P^-)^\mathcal{I} &= \{(y,x) \mid (x,y) \in P^\mathcal{I}\} \\
(C \sqcap C')^\mathcal{I} &= C^\mathcal{I} \cap C'^\mathcal{I} & (R \cap R')^\mathcal{I} &= R^\mathcal{I} \cap R'^\mathcal{I} \\
(C \sqcup C')^\mathcal{I} &= C^\mathcal{I} \cup C'^\mathcal{I} & (R \cup R')^\mathcal{I} &= R^\mathcal{I} \cup R'^\mathcal{I} \\
(\forall R.C)^\mathcal{I} &= \{x \mid \forall y.(x,y) \in R^\mathcal{I} \to y \in C^\mathcal{I}\} & (\neg R)^\mathcal{I} &= (\Delta^\mathcal{I} \times \Delta^\mathcal{I}) \setminus R^\mathcal{I} \\
(\exists R.C)^\mathcal{I} &= \{x \mid \exists y.(x,y) \in R^\mathcal{I} \wedge y \in C^\mathcal{I}\} & (R \circ R')^\mathcal{I} &= R^\mathcal{I} \circ R'^\mathcal{I} \\
(\geqslant n\, S.C)^\mathcal{I} &= \{x \mid |\{y \mid (x,y) \in S^\mathcal{I} \wedge y \in C^\mathcal{I}\}| \geq n\} & (R^*)^\mathcal{I} &= (R^\mathcal{I})^* \\
(\leqslant n\, S.C)^\mathcal{I} &= \{x \mid |\{y \mid (x,y) \in S^\mathcal{I} \wedge y \in C^\mathcal{I}\}| \leq n\} & (id(C))^\mathcal{I} &= \{(x,x) \mid x \in C^\mathcal{I}\} \\
(\exists S.\mathsf{Self})^\mathcal{I} &= \{x \mid (x,x) \in S^\mathcal{I}\} & &
\end{array}
$$

where $\cap$*,* $\cup$ *and* $\setminus$ *are overridden to denote the standard set-theoretic operations,* $\circ$ *to denote composition and* $\cdot^*$ *to denote the reflexive transitive closure of a binary relation.*

$\mathcal{I}$ satisfies *an assertion* α*, denoted* $\mathcal{I} \models \alpha$*, if* $a^\mathcal{I} \in A^\mathcal{I}$ *when* $\alpha = A(a)$*;* $\langle a^\mathcal{I}, b^\mathcal{I} \rangle \in P^\mathcal{I}$ *when* $\alpha = P(a,b)$*; and* $a^\mathcal{I} \neq b^\mathcal{I}$ *when* $\alpha = a \not\approx b$*.* $\mathcal{I}$ is a model of an ABox $\mathcal{A}$*, denoted* $\mathcal{I} \models \mathcal{A}$ *if it satisfies every assertion in* $\mathcal{A}$*.*$\mathcal{I}$ *satisfies a CIA or BRIA* $E \sqsubseteq E'$ *if* $E^\mathcal{I} \subseteq E'^\mathcal{I}$*.* $\mathcal{I}$ is a model of a TBox $\mathcal{T}$*, denoted* $\mathcal{I} \models \mathcal{T}$ *if it satisfies every CIA and BRIA in* $\mathcal{T}$*. Finally,* $\mathcal{I}$ *is a* model *of* $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$*, denoted* $\mathcal{I} \models K$*, if* $\mathcal{I} \models \mathcal{T}$ *and* $\mathcal{I} \models \mathcal{A}$*. Given a KB* $\mathcal{K}$*,* Knowledge base satisfiability *is the problem of deciding whether there exists an* $\mathcal{I}$ *such that* $\mathcal{I} \models \mathcal{K}$*.*

2.1 Syntactic Closure

We define the *(syntactic) closure* of a concept, which contains all the concepts and simple roles that are relevant for deciding its satisfiability. It is analogous to the well-known Fischer-Ladner closure of PDL, and it is exploited by the automata construction in Section 3.

In this section, we consider concepts and roles in a DL that we call $\mathcal{ALCQIB}^+_{reg}$; it is very similar to $\mathcal{ALCQI}b^+_{reg}$ but supports arbitrary role negation $\neg S$ instead of role difference $S \cap \neg S'$. If an $\mathcal{ALCQIB}^+_{reg}$ expression is equivalent to an $\mathcal{ALCQI}b^+_{reg}$ one, we call it *safe*. Intuitively, since $\mathcal{ALCQIB}^+_{reg}$ does not impose safety, it allows for a more flexible manipulation of Boolean role expressions and a simpler notion of syntactic closure.

In what follows, we use $\gtreqless$ to denote either $\geqslant$ or $\leqslant$. For a role name $P \in \mathbf{C}$, we call P^- the *inverse of* P and P the inverse of P^-; the inverse of an atomic role Q is denoted $\mathsf{Inv}(Q)$. For a simple role S, $\mathsf{Inv}(S)$ denotes the role obtained by replacing each atomic role Q occurring in S by its inverse $\mathsf{Inv}(Q)$. For brevity, we assume in this subsection that $\sqcup$ and $\forall$ are expressed by means $\sqcap$, $\exists$ and using $\neg$, and that $\setminus$ and $\cup$ are expressed using $\cap$ and $\neg$. As usual, C and C', S and S' and R and R' respectively stand for concepts, simple roles and arbitrary roles.

Definition 4. *The closure* $Cl(D)$ *of an* $\mathcal{ALCQIB}^+_{reg}$ *concept* D *is defined as the smallest set of* $\mathcal{ALCQIB}^+_{reg}$ *expressions such that* $D \in Cl(D)$ *and:*

if	$C \in Cl(D)$	*then*	$\neg C \in Cl(D)$	*(if* C *is not of the form* $\neg C'$*)*
if	$\neg C \in Cl(D)$	*then*	$C \in Cl(D)$	
if	$C \sqcap C' \in Cl(D)$	*then*	$C, C' \in Cl(D)$	
if	$\exists R.C \in Cl(D)$	*then*	$C \in Cl(D)$	

if	$\exists(R \cup R').C \in Cl(D)$	*then*	$\exists R.C,\ \exists R'.C \in Cl(D)$
if	$\exists(R \circ R').C \in Cl(D)$	*then*	$\exists R.\exists R'.C \in Cl(D)$
if	$\exists R^*.C \in Cl(D)$	*then*	$\exists R.\exists R^*.C \in Cl(D)$
if	$\exists id(C).C' \in Cl(D)$	*then*	$C,\ C' \in Cl(D)$
if	$\exists S.C \in Cl(D)$	*then*	$S \in Cl(D)$
if	$\geqslant n\, S.C \in Cl(D)$	*then*	$S,\ C \in Cl(D)$
if	$\exists S.\mathsf{Self} \in Cl(D)$	*then*	$S \in Cl(D)$
if	$S \cap S' \in Cl(D)$	*then*	$S,\ S' \in Cl(D)$
if	$\neg S \in Cl(D)$	*then*	$S \in Cl(D)$
if	$S \in Cl(D)$	*then*	$\neg S \in Cl(D)$ *(if S is not of the form $\neg S'$)*
if	$S \in Cl(D)$	*then*	$\mathsf{Inv}(S) \in Cl(D)$

Note that $|Cl(D)|$ is linear in the length of D and that $Cl(D)$ may contain non safe $\mathcal{ALCQIB}^+_{reg}$ expressions even when D is an $\mathcal{ALCQI}b^+_{reg}$ concept.

2.2 Normalizing Knowledge Bases

We present now some simple reductions that allow us to transform a KB into an equivalent one with a more restricted syntactic structure. We consider $\mathcal{ALCQI}b^+_{reg}$ KBs as well as $\mathcal{ALCQIB}^+_{reg}$ ones, which are defined in the natural way.

First of all, we consider the special expressions $\top$, $\bot$, B and T, and show that they can be expressed without the need of special symbols. Then we show how every KB can be transformed into an *extensionally reduced* one, where all terminological information is expressed by CIAs. Finally, we consider the well known negation normal form of KBs.

Universal role and special expressions. The special concepts $\top$ and $\bot$ can be simulated via any concept names $C_\top$ and $C_\bot$ not occurring in $\mathcal{K}$ by adding, e.g., CIAs $C \sqcup \neg C \sqsubseteq C_\top$, $\neg C_\top \sqsubseteq C_\bot$ and $\neg C_\bot \sqsubseteq C_\top$ for any concept C. Further, using $\top$ and $\bot$, the empty role B can also be easily simulated by a fresh role name P_B by adding an axiom $\top \sqsubseteq \forall P_\mathsf{B}.\bot$. Note that the above holds for every DL containing $\mathcal{ALC}$.

The universal role can be expressed in $\mathcal{ALCQIB}^+_{reg}$ as $\neg\mathsf{B}$, but this is not a safe role and is hence disallowed in $\mathcal{ALCQI}b^+_{reg}$ KBs. In fact, in the absence of the T symbol, there is no $\mathcal{ALCQI}b^+_{reg}$ role expression that is always equivalent to T. However, for each input KB $\mathcal{K}$, there exists a role expression U such that the resulting KB has a model $\mathcal{I}$ with $U^\mathcal{I} = \Delta^\mathcal{I} \times \Delta^\mathcal{I}$ whenever $\mathcal{K}$ is satisfiable. This is similar to what occurs in other expressive DLs (e.g., $\mathcal{SHIQ}$) and sufficient for the problems we consider.

Indeed, since every satisfiable $\mathcal{ALCQI}b^+_{reg}$ concept C has a *connected* model, we can transform C into an equisatisfiable concept replacing each occurrence of T by $U = (\bigcup_{Q \in \overline{\mathbf{R}}_C} Q)^*$, where $\overline{\mathbf{R}}_C$ denotes the set of all atomic roles occurring in C. In the presence of ABoxes, we additionally need to ensure that all pairs of ABox individuals are in the extension of the role expression simulating T. Hence, given a KB $\mathcal{K} = \langle \mathcal{A}, \mathcal{T} \rangle$, we can eliminate the universal role T as follows:

- For a fresh role name A_U, an assertion $A_U(a, b)$ is added to $\mathcal{A}$ for every pair $a, b \in \mathbf{I}_\mathcal{K}$.
- Each occurrence of T in $\mathcal{T}$ is replaced by the role $U = (A_U \cup \bigcup_{Q \in \overline{\mathbf{R}}_\mathcal{K}} Q)^*$.

Extensionally reduced KBs. Let KB $\mathcal{K} = \langle \mathcal{A}, \mathcal{T} \rangle$ be an $\mathcal{ALCQIB}^+_{reg}$ KB. Of the transformations below, the one w.r.t. to the ABox is well known. Here we use a similar technique to rewrite by means of CIAs all terminological information in the $\exists R.\mathsf{Self}$ concepts and BRIAs. Please note that all the transformations below involve only simple roles, hence the resulting KB complies to the allowed syntax. Also, since they do not introduce any non-safe role expressions, if $\mathcal{K}$ is an $\mathcal{ALCQI}b^+_{reg}$ KB, it will remain in $\mathcal{ALCQI}b^+_{reg}$ after the reduction.

– **ABox reduction.** $\mathcal{K}$ is *extensionally reduced w.r.t. the ABox* if only assertions of the forms $A(a)$, $P(a,b)$ and $a \neq b$ with $A \in \mathbf{C}$ and $P \in \mathbf{R}$ occur in $\mathcal{A}$.
  The *ABox reduction of* $\mathcal{K}$ is the KB $\Omega_{\text{ABox}}(\mathcal{K})$ obtained as follows:
  1. Each assertion of the form $C(a)$ in $\mathcal{A}$ with $C \notin \mathbf{C}$ is replaced with an assertion $A_C(a)$ for a fresh $A_C \in \mathbf{C}$ and an axiom $A_C \sqsubseteq C$ is added to $\mathcal{T}$.
  2. Each assertion $S(a,b)$ in $\mathcal{A}$ with $S \notin \mathbf{R}$ is replaced with an assertion $P_S(a,b)$ for a fresh $P_S \in \mathbf{R}$ and an axiom $P_S \sqsubseteq S$ is added to $\mathcal{T}$.
– **Self concepts reduction.** $\mathcal{K}$ is *extensionally reduced w.r.t. Self concepts* if $P \in \mathbf{R}$ for every concept of the form $\exists P.\mathsf{Self}$ occurring in it.
  The *Self concept reduction of* $\mathcal{K}$ is the KB $\Omega_{\text{Self}}(\mathcal{K})$ obtained by replacing each concept of the form $\exists S.\mathsf{Self}$ with $S \notin \mathbf{R}$ occurring in $\mathcal{T}$ with a concept $\exists P_S.\mathsf{Self}$ for a fresh $P_S \in \mathbf{R}$ and an adding an axiom $P_S \sqsubseteq S$ to $\mathcal{T}$.
– **BRIA reduction.** $\mathcal{K}$ is *extensionally reduced w.r.t. BRIAs* if all axioms in $\mathcal{T}$ are CIAs.
  The *BRIA reduction of* $\mathcal{K}$ is the KB $\Omega_{\text{BRIA}}(\mathcal{K})$ obtained by replacing in $\mathcal{T}$ each BRIA of the form $S \sqsubseteq S'$ by a CIA $\exists(S \setminus S').\top \sqsubseteq \bot$.
– **KB reduction.** $\mathcal{K}$ is *extensionally reduced* if it is extensionally reduced w.r.t. the ABox, Self concepts and BRIAs.
  The *reduction of* $\mathcal{K}$ is the KB $\Omega(\mathcal{K}) = \Omega_{\text{BRIA}}(\Omega_{\text{Self}}(\Omega_{\text{ABox}}(\mathcal{K})))$.

The above reductions preserve the semantics of the knowledge base $\mathcal{K}$, and $\Omega(\mathcal{K})$ additionally constrains the interpretation of some concept names not occurring in $\mathcal{K}$. If we consider the standard first order translation of $\mathcal{K}$ and $\Omega(\mathcal{K})$, then the latter is a conservative extension of the former. With this observation, the proof of the following proposition is straightforward:

Proposition 1. *For a given $\mathcal{ALCQIB}^+_{reg}$ KB $\mathcal{K}$, $\Omega(\mathcal{K})$ can be obtained in linear time. For every interpretation $\mathcal{I}$, $\mathcal{I} \models \Omega(\mathcal{K})$ implies $\mathcal{I} \models \mathcal{K}$ and $\mathcal{I} \models \mathcal{K}$ implies $\mathcal{I}' \models \Omega(\mathcal{K})$, where $\mathcal{I}'$ denotes the interpretation s.t. $E^{\mathcal{I}'} = E^{\mathcal{I}}$ for every $E \in \mathbf{C}_{\mathcal{K}} \cup \mathbf{R}_{\mathcal{K}}$, $(A_C)^{\mathcal{I}'} = C^{\mathcal{I}}$ for every $A_C \in \mathbf{C}_{\Omega(\mathcal{K})} \setminus \mathbf{C}_{\mathcal{K}}$, and $(P_S)^{\mathcal{I}'} = S^{\mathcal{I}}$ for every $P_S \in \mathbf{R}_{\Omega(\mathcal{K})} \setminus \mathbf{R}_{\mathcal{K}}$.*

Thus deciding the satisfiability of $\mathcal{K}$ can be reduced (in linear time) to used deciding the satisfiability of $\Omega(\mathcal{K})$, and we can restrict or attention to extensionally reduced knowledge bases. Note that we can also consider $\Omega(\mathcal{K})$ to decide the entailment of any sentence over the language of $\mathcal{K}$; this is useful, e.g., for *query answering* [3].

Negation Normal Form. Finally, we transform KBs to *negation normal form* (NNF).

An $\mathcal{ALCQIB}^+_{reg}$ role R is in NNF form if Q is atomic for every subrole $\neg Q$ of R. Similarly, a concept C is in *negation normal form* (NNF) if A is atomic for every subconcept $\neg A$ of C, and all roles occurring in C are in NNF. A knowledge base is in NNF if only expressions in NNF occur in it. For an expression E, $nnf(E)$ denotes the equivalent expression in NNF obtained from E using the standard transformations. For a concept D, we let $Cl^{nnf}(D) = \{nnf(E) \mid E \in Cl(D)\}$. For a KB $\mathcal{K}$, $nnf(\mathcal{K})$ denotes the KB obtained from $\mathcal{K}$ by replacing each expression E in $\mathcal{K}$ by $nnf(E)$.

Proposition 2. *For every $\mathcal{ALCQIB}^+_{reg}$ expression E, $nnf(E)$ can be obtained in linear time, and if E is an $\mathcal{ALCQI}b^+_{reg}$ expression, then so is $nnf(E)$. Further, for every KB $\mathcal{K}$ and every interpretation $\mathcal{I}$, $\mathcal{I} \models \mathcal{K}$ iff $\mathcal{I} \models nnf(\mathcal{K})$.*

An $\mathcal{ALCQI}b^+_{reg}$ KB $\mathcal{K} = \langle \mathcal{A}, \mathcal{T} \rangle$ is *normal* if it is extensionally reduced and in NNF, it does not contain $\top$, $\bot$, T and B, and each concept occurring in $\mathcal{A}$ also occurs in $\mathcal{T}$.[1]

[1] This is w.l.o.g. as we can add to $\mathcal{T}$, e.g., $A \sqsubseteq \top$ for each $A \in \mathbf{C}$ that occurs in $\mathcal{A}$ but not in $\mathcal{T}$.

2.3 Canonical Models and Trees

We have seen that, to decide KB satisfiability, we only need to consider KBs with a restricted syntax. Now we consider some semantic properties by which the shape of the considered interpretations can also be conveniently restricted.

In what follows, we consider only $\mathcal{ALCQI}b^+_{reg}$ KBs; the results of this section do not hold for $\mathcal{ALCQIB}^+_{reg}$ in general.[2] Like many DLs, $\mathcal{ALCQI}b^+_{reg}$ has some form of the *tree model property*: every satisfiable TBox $\mathcal{T}$ (or similarly, every satisfiable concept C) has a model that can be seen as a tree with possible additional loops at some nodes, say a 'quasi-tree'. A satisfiable $\mathcal{ALCQI}b^+_{reg}$ KB $\mathcal{K} = \langle \mathcal{A}, \mathcal{T} \rangle$ has a 'quasi-forest' shaped *canonical model*, in which each ABox individual is the root of a quasi-tree shaped model of $\mathcal{T}$.

Definition 5. *Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be an $\mathcal{ALCQI}b^+_{reg}$ KB, and let $1 \leq n \leq |\mathbf{I}_\mathcal{K}|$, $k \geq 0$. An interpretation $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$ for $\mathcal{K}$ is called a* canonical interpretation (with n roots and branching degree k) *if:*

(1) each $i{\cdot}x \in \Delta^\mathcal{I}$ has $i \in \{1, \ldots, n\}$ and $x \in \{1, \ldots, k\}^*$; if $x = \varepsilon$ then i is called a root.
(2) $\{\varepsilon\} \cup \Delta^\mathcal{I}$ is prefix closed, i.e., if $x{\cdot}c \in \Delta^\mathcal{I}$, then $x \in \{\varepsilon\} \cup \Delta^\mathcal{I}$.
(3) For each $a \in \mathbf{I}_\mathcal{K}$ there is exactly one root j such that $a^\mathcal{I} = j$.
(4) For each root j there is some $a \in \mathbf{I}_\mathcal{K}$ with $a^\mathcal{I} = j$.
(5) If $(i{\cdot}w, j{\cdot}w') \in P^\mathcal{I}$ for some atomic role P and two roots i, j, then either $w = w' = \varepsilon$, or $i = j$ and one of the following holds: (i) *$w = w'$,* (ii) *$w' = w{\cdot}l$ or* (iii) *$w = w'{\cdot}l$, for some $1 \leq l \leq k$.*

In $\mathcal{ALCQI}b^+_{reg}$, any TBox $\mathcal{T}$ can be 'internalized' into an equivalent concept $C_\mathcal{T}$, so that the satisfiability of $\mathcal{T}$ can be established by obtaining a model of $C_\mathcal{T}$ [14].

Proposition 3. *Consider an $\mathcal{ALCQI}b_{reg}$ knowledge base $\mathcal{K} = \langle \mathcal{A}, \mathcal{T} \rangle$, and let*

$$C_\mathcal{T} = \forall(\bigcup_{R \in \mathcal{R}_\mathcal{K}} R)^*.\bigsqcap_{C_1 \sqsubseteq C_2 \in \mathcal{T}} (\neg C_1 \sqcup C_2)$$

For every interpretation $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$, $\mathcal{I} \models \mathcal{T}$ iff $(C_\mathcal{T})^\mathcal{I} = \Delta^\mathcal{I}$. Furthermore, if $\mathcal{I}$ is a canonical interpretation, then $\mathcal{I} \models \mathcal{T}$ iff $i \in (C_\mathcal{T})^\mathcal{I}$ for each root i of $\mathcal{I}$.

Now we can establish the *canonical model property* of $\mathcal{ALCQI}b^+_{reg}$; it can be shown by adapting corresponding proofs for related logics [17, 15]. Roughly, any model of a satisfiable $\mathcal{ALCQI}b^+_{reg}$ concept D can be unraveled into a model of D that is a quasi-tree and that has branching degree $k_D = |Cl(D)| \times \mathbf{n}$, where $\mathbf{n}$ is the maximal n occurring in a concept of the form $\geqslant n\, S.C$ in $Cl^{nnf}(D)$, or 1 if there are no such concepts.

Theorem 1. *Every satisfiable $\mathcal{ALCQI}b^+_{reg}$ KB $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ has a canonical model $\mathcal{I}$ such that $\mathcal{I}$ is a canonical interpretation for $\mathcal{K}$ with branching $k_{C_\mathcal{T}}$, $\mathcal{I} \models \mathcal{A}$ and $i \in (C_\mathcal{T})^\mathcal{I}$ for each root i of $\mathcal{I}$.*

By Theorem 1, which does not hold for $\mathcal{ALCQIB}^+_{reg}$ in general, deciding the satisfiability of an $\mathcal{ALCQI}b^+_{reg}$ KB boils down to deciding whether it has a canonical model. To decide the latter, we rely on a representation of canonical interpretations as infinite labeled trees and define an automaton that decides the existence of such trees.

[2] Theorem 1 fails already in the extension of $\mathcal{ALC}$ with non-safe Boolean role expressions, see [12].

Definition 6. *An* (infinite) tree *is a prefix-closed set T of words over the natural numbers $\mathbb{N}$. The elements of T are called* nodes, *the empty word ε is its* root. *For every $x \in T$, the nodes $x{\cdot}c$ with $c \in \mathbb{N}$ are the* successors *of x, and x is the* predecessor *of each $x{\cdot}c$. By convention, $x{\cdot}0 = x$, and $(x{\cdot}i){\cdot}{-1} = x$. The* branching degree *$d(x)$ of a node x is the number of its successors. If $d(x) \leq k$ for each node x of T, then T has* branching degree *k. An* infinite path *π of T is a prefix-closed set $\pi \subseteq T$ where for every $i \geq 0$ there exists a unique node $x \in P$ with $|x| = i$. A* labeled tree *over an alphabet Σ (or simply a Σ-labeled tree) is a pair (T, V), where T is a tree and $V : T \to \Sigma$ maps each node of T to an element of Σ.*

In the following, we assume a (fixed) given normal $\mathcal{ALCQIb}^+_{reg}$ KB $\mathcal{K} = \langle \mathcal{A}, \mathcal{T} \rangle$. We denote $\mathbf{I}_\mathcal{K}$ by $\mathcal{J}$ and assume an arbitrary, fixed, enumeration $a_1, \ldots, a_m$ of the elements of $\mathcal{J}$. We define *PI* $= \{Pij \mid a_i, a_j \in \mathcal{J}$ and $P \in \overline{\mathbf{R}}_\mathcal{K}\}$ and *PS* $= \{P_{\mathsf{Self}} \mid P \in \overline{\mathbf{R}}_\mathcal{K}\}$. We will define the *tree representation of a canonical interpretation* $\mathcal{I}$ for $\mathcal{K}$ as a labeled tree $\mathbf{T}_\mathcal{I}$ over the alphabet $\Sigma_\mathcal{K} = 2^{\mathbf{C}_\mathcal{K} \cup \overline{\mathbf{R}}_\mathcal{K} \cup \mathcal{J} \cup \{r\} \cup \{d\} \cup PI \cup PS}$.

To define $\mathbf{T}_\mathcal{I}$, we first note that the domain of a canonical interpretation is almost a tree; we only need to add a root ε to it. The extensions of the interpretations of individuals, concepts and roles can be represented as labels over the alphabet $\Sigma_\mathcal{K}$ in a straightforward way. Roughly, each element of $x \in \Delta^\mathcal{I}$, which is a node of the tree, is labeled with a set $V(x)$ that contains: (*i*) the atomic concepts A such that $x \in A^\mathcal{I}$; (*ii*) the atomic roles that connect the predecessor of x to x, and (*iii*) a special symbol P_{Self} for each P such that $x \in (\exists P.\mathsf{Self})^\mathcal{I}$. The root ε and the roots of $\mathcal{I}$ (which are at the first level of the tree $\mathbf{T}_\mathcal{I}$) are treated differently. The label of each root i of $\mathcal{I}$, apart from the atomic concepts to which i belongs, also contains the name of the individuals in $\mathcal{J}$ which it interprets, and it contains no basic roles. The relations between level one nodes are stored in the label of the root ε. The root ε does not represent any object in $\Delta^\mathcal{I}$ and is marked with a special identifier r and symbols of the form Pij indicating that the pair of individuals (a_i, a_j) belongs to the extension of the basic role P. For simplicity, if $\mathcal{I}$ has n roots and $|\mathcal{J}| > n$, $|\mathcal{J}| - n$ dummy children labeled $\{d\}$ are added to the root ε, ensuring that it has exactly $|\mathcal{J}|$ children.

Definition 7. *Let $\mathcal{I}$ be a canonical interpretation for $\mathcal{K}$ with n roots. The* tree representation of $\mathcal{I}$ *is the labeled tree $\mathbf{T}_\mathcal{I} = (T, V)$ over the alphabet $\Sigma_\mathcal{K} = 2^{\mathbf{C}_\mathcal{K} \cup \overline{\mathbf{R}}_\mathcal{K} \cup \mathcal{J} \cup \{r\} \cup \{d\} \cup PI \cup PS}$ defined as follows:*

- $T = \{\varepsilon\} \cup \Delta^\mathcal{I} \cup \{n+1, \ldots, m\}$.
- $V(\varepsilon) = \{r\} \cup \{Pij \mid a_i, a_j \in \mathcal{J}, P \in \mathbf{C}_\mathcal{K}$ *and* $\langle a_i^\mathcal{I}, a_j^\mathcal{I} \rangle \in P^\mathcal{I}\}$,
- *for each* $1 \leq i \leq n$, $V(i) = \{a_j \in \mathcal{J} \mid a_j^\mathcal{I} = i\} \cup \{A \in \mathbf{C}_\mathcal{K} \mid a_j^\mathcal{I} \in A^\mathcal{I}$ *and* $a_j^\mathcal{I} = i\}$,
- *for each* $n+1 \leq i \leq m$, $V(i) = \{d\}$,
- *for all other nodes* $i{\cdot}x$ *of* T, $V(i{\cdot}x) = \{A \in \mathbf{C}_\mathcal{K} \mid i{\cdot}x \in A^\mathcal{I}\} \cup \{Q \in \overline{\mathbf{R}}_\mathcal{K} \mid (i{\cdot}w, i{\cdot}x) \in P^\mathcal{I}$ *where* $x = w{\cdot}j$ *for some* j, $1 \leq j \leq k_{C_\mathcal{T}}\} \cup \{P_{\mathsf{Self}} \mid P \in \overline{\mathbf{R}}_\mathcal{K}$ *and* $i{\cdot}x \in (\exists P.\mathsf{Self})^\mathcal{I}\}$

Note that the branching degree of $\mathbf{T}_\mathcal{I}$ is bounded by $|\mathcal{J}|$ at the root and by $k_{C_\mathcal{T}}$ at all other levels, so the tree has branching degree $max(k_{C_\mathcal{T}}, |\mathcal{J}|)$.

3 An Automata Algorithm for $\mathcal{ALCQIb}^+_{reg}$

Automata on infinite trees allow for an elegant reduction of decision problems for temporal and program logics [5], and have been widely exploited for many variants of PDL, the μ-calculus and similar logics [16, 17]. In DLs they have been used for concept satisfiability [15] and, to a more limited extent, for KB satisfiability [4].

3.1 Preliminaries: Automata on Infinite Trees

We consider *two-way alternating tree automata* over infinite trees, introduced in [16]. Let $\mathcal{B}(I)$ be the set of positive Boolean formulas built inductively from **true**, **false**, and atoms from a given set I applying $\wedge$ and $\vee$. A set $J \subseteq I$ *satisfies* a formula $\varphi \in \mathcal{B}(I)$, if assigning **true** to the atoms in J and **false** to those in $I \setminus J$ makes φ true. A *two-way alternating tree automaton* (2ATA) running over infinite trees with branching degree k, is a tuple $\mathbf{A} = \langle \Sigma, Q, \delta, q_0, F \rangle$, where Σ is the input *alphabet*; Q is a finite set of *states*; $\delta : Q \times \Sigma \to \mathcal{B}([k] \times Q)$, where $[k] = \{-1, 0, 1, \ldots, k\}$, is the *transition function*; $q_0 \in Q$ is the *initial state*; and $F \subseteq Q$ is the *(Büchi) acceptance condition*.

The transition function δ maps a state $q \in Q$ and an input letter $\sigma \in \Sigma$ to a positive Boolean formula φ over atoms $[k] \times Q$. Intuitively, if $\delta(q, \sigma) = \varphi$, then each atom (c, q') in φ corresponds to a new copy of the automaton going in the direction given by c and starting in state q'. E.g., a transition of the form $\delta(q_1, \sigma) = (1, q_2) \wedge (1, q_3) \vee (-1, q_1) \wedge (0, q_3)$ indicates that if $\mathbf{A}$ is in the state q_1 and reads the node x labeled with σ, it proceeds by sending off either two copies, in the states q_2 and q_3 respectively, to the first successor of x, or one copy in the state q_1 to the predecessor of x and one copy in the state q_3 to x itself.

Informally, a run of a 2ATA $\mathbf{A}$ over a labeled tree (T, V) is a labeled tree (T_r, r) in which each node $y \in T_r$ is labeled by an element $r(y) = (x, q) \in T \times Q$ and describes a copy of $\mathbf{A}$ that is in the state q and reads the node x of T; the labels of adjacent nodes must satisfy the transition function of $\mathbf{A}$. The run is accepting if in every infinite path at least one state from F occurs infinitely often in the node labels.

Formally, given a 2ATA $\mathbf{A} = \langle \Sigma, Q, \delta, q_0, F \rangle$ and a Σ-labeled tree (T, V), a *run of* $\mathbf{A}$ *over* (T, V) is a $T \times Q$-labeled tree (T_r, r) satisfying:

- $\varepsilon \in T_r$ and $r(\varepsilon) = (\varepsilon, q_0)$.
- For each $y \in T_r$ with $r(y) = (x, q)$ there is a (possibly empty) set $S = \{(c_1, q_1), \ldots, (c_n, q_n)\} \subseteq [k] \times Q$ such that S satisfies $\delta(q, V(x))$ and for all $1 \leq i \leq n$, we have that $y \cdot i \in T_r$, $x \cdot c_i$ is defined and $r(y \cdot i) = (x \cdot c_i, q_i)$.

A run (T_r, r) over (T, V) is *accepting*, if for each infinite path π of T_r there is some state $q_f \in F$ such that the set $\{y \in \pi \mid r(y) = (x, q_f)$ for some $x \in T\}$ is infinite. A labeled tree (T, V) is *accepted* by $\mathbf{A}$ if there is an accepting run of $\mathbf{A}$ over (T, V), and $\mathcal{L}(\mathbf{A})$ denotes the set of Σ-labeled trees accepted by $\mathbf{A}$.

The *non-emptiness problem* for 2ATAs is to decide whether, given a 2ATA $\mathbf{A}$, the set $\mathcal{L}(\mathbf{A})$ is nonempty. It was shown in [16] that non-emptiness of $\mathbf{A} = \langle \Sigma, Q, \delta, q_0, F \rangle$ is decidable in time single exponential in $|Q|$ and polynomial in $|\Sigma|$.

3.2 Constructing the Automaton

In the following, we assume a given, fixed, normal $\mathcal{ALCQI}b^+_{reg}$ KB $\mathcal{K} = \langle \mathcal{A}, \mathcal{T} \rangle$ and provide a construction of a 2ATA $\mathbf{A}_\mathcal{K}$ from $\mathcal{K}$ that accepts a given tree $\mathbf{T}$ iff it is the tree representation of a canonical model of $\mathcal{K}$. Similar automata constructions have been given in the literature. In particular, this one is a simple extension of the one in [4] to handle the $\exists S.\mathsf{Self}$ concepts of $\mathcal{ALCQI}b^+_{reg}$. In turn, [4] extends to KB satisfiability a previous construction of an automaton that decides $\mathcal{ALCQI}b_{reg}$ concept satisfiability [2]. Due to space constraints, we briefly describe $\mathbf{A}_\mathcal{K} = \langle \Sigma_\mathcal{K}, Q_\mathcal{K}, \delta_\mathcal{K}, q_0, F_\mathcal{K} \rangle$ and focus on the extension from $\mathcal{ALCQI}b_{reg}$ to $\mathcal{ALCQI}b^+_{reg}$; we refer the reader to [3] for details.

The alphabet is $\Sigma_\mathcal{K} = 2^{\mathbf{C}_\mathcal{K} \cup \overline{\mathbf{R}}_\mathcal{K} \cup \mathcal{J} \cup \{r\} \cup \{d\} \cup PI \cup PS}$ as given in Definition 7; note that the only difference w.r.t. [3] is the presence of the set *PS*.

Like in [3], the 'basic' states of $\mathbf{A}_{\mathcal{K}}$ are the role and concept expressions in the closure of $C_{\mathcal{T}}$, which are used for checking whether a node of the tree is in the extension of an expression or marked with some symbol. There are also additional special states for verifying some more involved conditions like the satisfaction of the number restrictions, the correct representation of the individuals and the ABox assertions, etc.

More precisely, let $S = \mathcal{J} \cup PS \cup \{c, d\}$. There is a state for each element of the the set $Cl_{ext} = Cl^{nnf}(C_{\mathcal{T}}) \cup \{s, \neg s \mid s \in S\}$ which we call the S-extended closure of $C_{\mathcal{T}}$. The extended closure allows us to treat the individual names and special symbols in the node labels as well as the atomic expressions in a uniform way. Note that, in contrast to [3], Cl_{ext} may contain concepts of the form $\exists P.\mathsf{Self}$. The set of states is $S_{\mathcal{K}} = \{s_0, s_1\} \cup Cl_{ext} \cup S_{num} \cup S_{\mathcal{A}\_role} \cup S_{\mathcal{A}\_quant} \cup S_{\mathcal{A}\_num} \cup S_{\mathsf{Self}}$, where the sets $S_{\mathcal{A}\_quant}$ S_{num}, $S_{\mathcal{A}\_num}$ and $S_{\mathcal{A}\_role}$ are as in [3]. We use the states in $S_{\mathcal{A}\_quant}$ of the forms $\langle j, \exists S.C\rangle$ and $\langle j, \forall S.C\rangle$ to respectively check whether concepts of the form $\exists Q.C$ and $\forall Q.C$ are satisfied by a node representing some ABox individual a_j. For checking the satisfaction of number restrictions we need the states S_{num} of the form $\langle \gtrless n\, S.C, i, j\rangle$ that store two counters i and j, plus the states $S_{\mathcal{A}\_num}$ of the form $\langle m, \gtrless nQC, i, j\rangle$, which also store the value of two counters but for a particular individual a_m. For checking whether two ABox individuals a_i, a_j are related by a simple role S, we use states of the form Sij in $S_{\mathcal{A}\_role}$. The only additional states w.r.t. to [3] are the ones in the set $S_{\mathsf{Self}} = \{S_{\mathsf{Self}} \mid S$ a simple role in $Cl^{nnf}(C_{\mathcal{T}})\}$, which we use for decomposing a simple role and checking whether it links an individual to itself. Finally, there is the initial state s_0 and another special state s_1. As in [3], both are used to verify the satisfaction of general conditions necessary for the input tree to represent a canonical model of $\mathcal{K}$, e.g., that the root is the only node labeled r, that its label correctly stores the role assertions in the ABox and that the ABox individuals are correctly represented by the level one nodes.

As in [2–4], $F_{\mathcal{K}} = \{\forall R^*.C \mid \forall R^*.C \in Cl^{nnf}(C_{\mathcal{T}})\}$ is the acceptance condition.

The transition function $\delta : S_{\mathcal{K}} \times \Sigma_{\mathcal{K}} \rightarrow \mathcal{B}([k] \times S_{\mathcal{K}})$, where $k = max(k_{C_{\mathcal{T}}}, |\mathcal{J}|)$, is defined as follows. First, for each σ in $\sigma \in \Sigma_{\mathcal{K}}$ with $r \in \sigma$ we define a transition $\delta(s_0, \sigma)$ $= F_1 \wedge \cdots \wedge F_8$ from the initial state s_0, which verifies (*i*) that the root is the only node containing r and that the ABox individuals are properly represented by the level one nodes (F_1–F_4); (*ii*) that all ABox assertions are satisfied (F_5–F_7); and (*iii*) that every non-dummy node at level one is the root of a tree representing a model of $C_{\mathcal{T}}$ (F_8). F_4 requires a transition $\delta(s_1, \sigma)$ from the other special state s_1 for each $\sigma \in \Sigma_{\mathcal{K}}$, which is necessary for F_4 to correctly ensure that the symbol r only occurs at the root and the symbols in $\mathcal{J}$ only occur at the first level of the tree. These transitions are as in [3], to which the reader may refer for details.

The transitions that inductively decompose complex concepts while navigating the tree are as in [3]; so are the ones that decompose simple roles. Additionally, for each concept of the form $\exists P.\mathsf{Self}$ such that $\exists P.\mathsf{Self} \in S_{\mathcal{K}}$ and for each $\sigma \in \Sigma_{\mathcal{K}}$ there is a transition $\delta(\exists P.\mathsf{Self}, \sigma) = (0, P_{\mathsf{Self}})$. Special transitions are necessary to decompose the simple roles for each possible self loop and for each pair of ABox individuals. Thus we have for each $\sigma \in \Sigma_{\mathcal{K}}$ and for each $s \in S_{\mathsf{Self}} \cup S_{\mathcal{A}\_role}$ a transition $\delta(s, \sigma)$ as follows:

$$\begin{array}{ll}
\delta(S \cap S'_{\mathsf{Self}}, \sigma) = (0, S_{\mathsf{Self}}) \wedge (0, S'_{\mathsf{Self}}) & \delta(S \cap S'ij, \sigma) = (0, Sij) \wedge (0, S'ij) \\
\delta(S \cup S'_{\mathsf{Self}}, \sigma) = (0, S_{\mathsf{Self}}) \vee (0, S'_{\mathsf{Self}}) & \delta(S \cup S'ij, \sigma) = (0, Sij) \vee (0, S'ij) \\
\delta(Q \setminus Q'_{\mathsf{Self}}, \sigma) = (0, Q_{\mathsf{Self}}) \wedge (0, \neg Q'_{\mathsf{Self}}) & \delta(Q \setminus Q'ij, \sigma) = (0, Qij) \wedge (0, \neg Q'ij)
\end{array}$$

Informally, given a tree representing an interpretation, let us call *potential neighbors of* x all the nodes that could be related to a node x via a simple role. If x does not represent an ABox individual, then its potential neighbors are x itself, its predecessor and all its successors.

If x does represent an ABox individual, then instead of its predecessor (which is the dummy root), all the other nodes representing ABox individuals are potential neighbors of x.

Further transitions in [3] verify whether a node satisfies an universal restriction $\forall S.C$, an existential restriction $\exists S.C$ or a number restriction $\gtrless nS.C$, for a simple role S; they navigate all the potential neighbors of a node and test which are reachable via S and labelled C. We use similar transitions, but adapt them to take into account that, in an $\mathcal{ALCQIb}^+_{reg}$ interpretation, a node is a potential neighbor of itself. The transitions for the concepts of the form $\forall S.C$ and $\exists S.C$ are given below. The ones for the number restrictions (which use the states in S_{num} and $S_{\mathcal{A}\_num}$ to encode counters) are adapted similarly but omitted here due to space constraints.

Since the potential neighbors of a node x are different depending on whether (*i*) x is a level one node, or (*ii*) x is a node at any other level, the transitions must differ accordingly. Hence, for each concept of the form $\exists S.C$ or $\forall S.C$ in $Cl^{nnf}(C_{\mathcal{T}})$ with S simple and each $\sigma \in \Sigma_{\mathcal{K}}$, if $\sigma \cap (\mathcal{J} \cup \{d\}) = \emptyset$ we define:

$$\delta(\exists S.C, \sigma) = ((0, S_{\mathsf{Self}}) \wedge (0, C)) \ \vee \ ((0, \mathsf{Inv}(S)) \wedge (-1, C)) \ \vee \bigvee_{1 \leq i \leq k_{C_{\mathcal{T}}}} ((i, S) \wedge (i, C))$$

$$\delta(\forall S.C, \sigma) = ((0, nnf(\neg S_{\mathsf{Self}})) \vee (0, C)) \ \wedge \ ((0, nnf(\neg \mathsf{Inv}(S))) \vee (-1, C)) \ \wedge \bigwedge_{1 \leq i \leq k_{C_{\mathcal{T}}}} ((i, nnf(\neg S)) \vee (i, C))$$

Otherwise, if $\sigma \cap (\mathcal{J} \cup \{d\}) \neq \emptyset$, we have:

$$\delta(\exists S.C, \sigma) = ((0, S_{\mathsf{Self}}) \wedge (0, C)) \vee \bigvee_{a_j \in \sigma} (-1, \langle j, \exists S.C \rangle) \vee \bigvee_{1 \leq i \leq k_{C_{\mathcal{T}}}} ((i, S) \wedge (i, C))$$

$$\delta(\forall S.C, \sigma) = ((0, nnf(\neg S_{\mathsf{Self}})) \vee (0, C)) \ \wedge \ \bigwedge_{a_j \in \sigma} (-1, \langle j, \forall S.C \rangle) \ \wedge \bigwedge_{1 \leq i \leq k_{C_{\mathcal{T}}}} ((i, nnf(\neg S)) \vee (i, C))$$

The first disjunct/conjunct in these transitions, which verifies whether the current node reaches itself via S and is labelled C, is not present in the corresponding transitions from [3].

In the last set of transitions (case (*i*) above, $\sigma \cap (\mathcal{J} \cup \{d\}) \neq \emptyset$), the second disjunct/conjunct is responsible for sending a copy of the automaton to the root of the tree and moving to the special states in $S_{\mathcal{A}\_quant}$, in order to traverse all the ABox individuals which are potential neighbors of the current node. The latter is achieved as in [4], i.e., for each $\sigma \in \Sigma_{\mathcal{K}}$ and each $\langle j, \exists S.C \rangle$ or $\langle j, \forall S.C \rangle$ in $S_{\mathcal{A}\_quant}$, there is a transition

$$\delta(\langle j, \exists S.C \rangle, \sigma) = \bigvee_{1 \leq i \leq |\mathcal{J}|} (\bigvee_{1 \leq k \leq |\mathcal{J}|} ((0, S_{jk}) \wedge (i, a_k) \wedge (i, C)))$$

$$\delta(\langle j, \forall S.C \rangle, \sigma) = \bigwedge_{1 \leq i \leq |\mathcal{J}|} (\bigwedge_{1 \leq k \leq |\mathcal{J}|} ((0, nnf(\neg Sjk)) \vee (i, \neg a_k) \vee (i, C)))$$

The above transitions decompose all concepts and roles until they reach states corresponding to atomic expressions or (possibly negated) special symbols in *PI* $\cup$ *PS*; it is then checked whether the expression is contained in the node label σ. The transitions form the states s_0 and s_1 may also move to states corresponding to possibly negated symbols in $\mathcal{J} \cup \{r, d\}$, which are similarly checked at the node labels. Thus, for each $\sigma \in \Sigma_{\mathcal{K}}$, each $s \in \mathbf{C}_{\mathcal{K}} \cup \overline{\mathbf{R}}_{\mathcal{K}} \cup \mathcal{J} \cup \{r, d\}$ and each $t \in$ *PI* $\cup$ *PS*, there are transitions:

$$\delta(s, \sigma) = \begin{cases} \textbf{true}, & \text{if } s \in \sigma \\ \textbf{false}, & \text{if } s \notin \sigma \end{cases} \qquad \delta(\neg s, \sigma) = \begin{cases} \textbf{true}, & \text{if } s \notin \sigma \\ \textbf{false}, & \text{if } s \in \sigma \end{cases}$$

$$\delta(t, \sigma) = \begin{cases} \textbf{true}, & \text{if } t \in \sigma \text{ or } \mathsf{Inv}(t) \in \sigma \\ \textbf{false}, & \text{otherwise} \end{cases} \qquad \delta(\neg t, \sigma) = \begin{cases} \textbf{true}, & \text{if } t \notin \sigma \text{ and } \mathsf{Inv}(t) \notin \sigma \\ \textbf{false}, & \text{otherwise} \end{cases}$$

where $\mathsf{Inv}(t) = \mathsf{Inv}(P)ji$ if $t = Pij$ and $\mathsf{Inv}(t) = \mathsf{Inv}(P)_{\mathsf{Self}}$ if $t = P_{\mathsf{Self}}$ for each $P \in \overline{\mathbf{R}}_{\mathcal{K}}$.

These transitions extend similar ones in [4] to include the symbols in *PS*.

By a simple adaptation of the proofs in [3], it can be easily verified that, given a canonical model $\mathcal{I}$ of $\mathcal{K}$, its tree representation $\mathbf{T}_{\mathcal{I}}$ is accepted by $\mathbf{A}_{\mathcal{K}}$. Furthermore, a model $\mathcal{I}_{\mathbf{T}}$ of $\mathcal{K}$ can be constructed from any labeled tree $\mathbf{T} = (T, V)$ accepted by $\mathbf{A}_{\mathcal{K}}$. The domain $\Delta^{\mathcal{I}_{\mathbf{T}}}$ is given by the nodes x in $\mathbf{T}$ with $a_i \in V(x)$ for some individual a_i, and the nodes in $\mathbf{T}$ that are reachable from any such x through the roles. The extensions of concepts and roles are determined by the labels of the nodes in $\mathbf{T}$. The only difference w.r.t. [4] is that we also add a pair (x, x) to $P^{\mathcal{I}_{\mathbf{T}}}$ for every x whose label contains P_{Self}.

This shows that the automaton $\mathbf{A}_{\mathcal{K}}$ can be used to decide the satisfiability of $\mathcal{K}$.

Theorem 2. *$\mathcal{K}$ is satisfiable iff the set of trees accepted by $\mathbf{A}_{\mathcal{K}}$ is nonempty.*

Under unary encoding of numbers in restrictions, the number of states of $\mathbf{A}_{\mathcal{K}}$ is polynomial in the size of $\mathcal{K}$. Since $\Sigma_{\mathcal{K}}$ is single exponential in the size of $\mathcal{K}$, we can exploit the results of [16] to obtain an EXPTIME upper bound for satisfiability of $\mathcal{ALCQI}b^{+}_{reg}$ KBs. This is optimal, as a matching lower bound is known for much weaker DLs [1].

Corollary 1. *Satisfiability of $\mathcal{ALCQI}b^{+}_{reg}$ knowledge bases is* EXPTIME*-complete.*

4 Reasoning in $\mathcal{SRIQ}$

The description logic $\mathcal{SRIQ}$ was introduced in [7] as an extension of the DL $\mathcal{RIQ}$ [9], which in turn extends the well known $\mathcal{SHIQ}$ [10] underlying OWL-Lite. It has gained considerable attention recently due to its close relationship to the logic $\mathcal{SROIQ}$ underlying the new OWL 2 standard. The most prominent feature of $\mathcal{SRIQ}$ are *complex role inclusion axioms* of the form $R_1 \circ \ldots \circ R_n \sqsubseteq R$. Additionally, in an $\mathcal{SRIQ}$ KB it is possible to explicitly state certain properties of roles like transitivity, (ir)reflexivity and disjointness. Some of these additions increase the expressivity of the logic, while other are just 'syntactic sugar' and are intended to be useful for ontology engineering. We now recall the definition of $\mathcal{SRIQ}$.

Definition 8 **($\mathcal{SRIQ}$ knowledge bases).** *Let $\overline{\mathbf{R}} = \mathbf{R} \cup \{R^- \mid R \in \mathbf{R}\}$. Each $R \in \overline{\mathbf{R}}$ is a* ($\mathcal{SRIQ}$) role. *As usual, P^- is the* inverse of P, *P the inverse of P^-, and $\mathsf{Inv}(R)$ denotes the inverse of the role R. A* ($\mathcal{SRIQ}$) role inclusion axiom (SRIA) *is an expression of the form $R_1 \circ \ldots \circ R_n \sqsubseteq R$, where $n \geq 1$ and $\{R_1, \ldots, R_n, R\} \subseteq \overline{\mathbf{R}}$.*

A set $\mathcal{R}_h$ of SRIAs is regular *if there exists a partial order $\prec$ on $\overline{\mathbf{R}}$ such that $\mathsf{Inv}(R) \prec R'$ iff $R \prec R'$ for every $R, R' \in \overline{\mathbf{R}}$, and such that every SRIA in $\mathcal{R}_h$ is of one of the following forms:*

- *(i)* $R \circ R \sqsubseteq R$, *or*
- *(ii)* $\mathsf{Inv}(R) \sqsubseteq R$, *or*
- *(iii)* $R_1 \circ \ldots \circ R_n \sqsubseteq R$ *and* $R_i \prec R$ *for each* $1 \leq i \leq n$, *or*
- *(iv)* $R \circ R_1 \circ \ldots \circ R_n \sqsubseteq R$ *and* $R_i \prec R$ *for each* $1 \leq i \leq n$, *or*
- *(v)* $R_1 \circ \ldots \circ R_n \circ R \sqsubseteq R$ *and* $R_i \prec R$ *for each* $1 \leq i \leq n$.

Given a set $\mathcal{R}_h$ of SRIAs, let $\overline{\mathbf{R}}_{\mathcal{R}_h}$ denote the set of roles occurring in $\mathcal{R}_h$. The set of simple roles *in $\mathcal{R}_h$ is the minimal set $\mathsf{SR}(\mathcal{R}_h) \subseteq \overline{\mathbf{R}}_{\mathcal{R}_h}$ containing each role R such that* (i) *there are no SRIAs of the form $R_1 \circ \ldots \circ R_n \sqsubseteq R$ in $\mathcal{R}_h$, and* (ii) *if $n = 1$ and $R_1 \in \mathsf{SR}(\mathcal{R}_h)$ for every SRIA of the form $R_1 \circ \ldots \circ R_n \sqsubseteq R$ in $\mathcal{R}_h$, then $R \in \mathsf{SR}(\mathcal{R}_h)$.*

A role assertion *is an expression of the form* $\mathsf{Sym}(R)$, $\mathsf{Ref}(R)$, $\mathsf{Irr}(R)$, *or* $\mathsf{Dis}(R, R')$, *where* $R, R' \in \overline{\mathbf{R}}$.[3] *An* RBox $\mathcal{R}$ *is a finite set of SRIAs and role assertions such that* $R \in \mathsf{SR}(\mathcal{R}_h)$ *for each* R *occurring in a role assertion of the form* $\mathsf{Ref}(R)$, $\mathsf{Irr}(R)$ *or* $\mathsf{Dis}(R, R')$ *in* $\mathcal{R}$, *where* $\mathcal{R}_h$ *denotes the set of SRIAs in* $\mathcal{R}$. *We say that* $\mathcal{R}$ *is* regular *if* $\mathcal{R}_h$ *is regular, and define* $\mathsf{SR}(\mathcal{R}) = \mathsf{SR}(\mathcal{R}_h)$.

Similarly as in $\mathcal{ALCQI}b^+_{reg}$, ($\mathcal{SRIQ}$) concepts C, C' *obey the following syntax*

$$C, C' \longrightarrow A \mid \neg C \mid C \sqcap C' \mid C \sqcup C' \mid \forall R.C \mid \exists R.C \mid \geqslant n\, S.C \mid \leqslant n\, S.C \mid \exists S.\mathit{Self},$$

where $A \in \mathbf{C}$ *and* $R, S \in \overline{\mathbf{R}}$. *A* ($\mathcal{SRIQ}$) concept inclusion axiom (SCIA) *is an expression* $C \sqsubseteq C'$ *for arbitrary* $\mathcal{SRIQ}$ *concepts* C *and* C'. *A* ($\mathcal{SRIQ}$) TBox *is a set of SCIAs. For an RBox* $\mathcal{R}$, *a concept* C *is* $\mathcal{R}$-simple *if* $S \in \mathsf{SR}(\mathcal{R}_h)$ *for each* S *occurring in a subconcept of* C *of the form* $\geqslant n\, S.C'$, $\leqslant n\, S.C'$ *or* $\exists S.\mathit{Self}$; *a TBox* $\mathcal{T}$ *is* $\mathcal{R}$-simple *if all concepts occurring in the SCIAs of* $\mathcal{T}$ *are* $\mathcal{R}$*-simple.*

An ($\mathcal{SRIQ}$) assertion *is an expression* $C(a)$, $R(a, b)$, $\neg S(a, b)$ *or* $a \neq b$, *where* C *is a* $\mathcal{SRIQ}$ *concept,* S, R *are* $\mathcal{SRIQ}$ *roles and* $a, b \in \mathbf{I}$. *An* ($\mathcal{SRIQ}$) ABox *is a set of* $\mathcal{SRIQ}$ *assertions. An ABox* $\mathcal{A}$ *is* $\mathcal{R}$-simple *for an RBox* $\mathcal{R}$ *if* $S \in \mathsf{SR}(\mathcal{R}_h)$ *for each* S *occurring in an assertion of the form* $\neg S(a, b)$.

An ($\mathcal{SRIQ}$) knowledge base *is a tuple* $\mathcal{K} = \langle \mathcal{A}, \mathcal{T}, \mathcal{R} \rangle$ *where* $\mathcal{R}$ *is a regular* $\mathcal{SRIQ}$ *RBox,* $\mathcal{A}$ *a non-empty* $\mathcal{R}$*-simple* $\mathcal{SRIQ}$ *ABox and* $\mathcal{T}$ *an* $\mathcal{R}$*-simple* $\mathcal{SRIQ}$ *TBox.*

The semantics of $\mathcal{SRIQ}$ TBoxes and ABoxes is defined as for $\mathcal{ALCQI}b^+_{reg}$. An interpretation $\mathcal{I}$ satisfies a role assertion $\mathsf{Sym}(R)$, $\mathsf{Ref}(R)$ or $\mathsf{Irr}(R)$ if $R^{\mathcal{I}}$ is a symmetric, reflexive or irreflexive relation, respectively; $\mathcal{I}$ satisfies $\mathsf{Dis}(R, R')$ if the relations are disjoint, i.e., $R^{\mathcal{I}} \cap R'^{\mathcal{I}} = \emptyset$; $\mathcal{I}$ satisfies a SRIA $R_1 \circ \ldots \circ R_n \sqsubseteq R$ if $R_1^{\mathcal{I}} \circ \ldots \circ R_n^{\mathcal{I}} \subseteq R^{\mathcal{I}}$, where again we override the symbol $\circ$ and use it to denote binary role composition. An interpretation $\mathcal{I}$ is a model of an RBox, in symbols $\mathcal{I} \models \mathcal{R}$, if it satisfies all SRIAs and role assertions in $\mathcal{R}$. Modelhood of a KB is restricted in the natural way to the models of the RBox, i.e., $\mathcal{I} \models \langle \mathcal{A}, \mathcal{T}, \mathcal{R} \rangle$ iff $\mathcal{I} \models \mathcal{A}$, $\mathcal{I} \models \mathcal{T}$ and $\mathcal{I} \models \mathcal{R}$.

4.1 Reducing $\mathcal{SRIQ}$ to $\mathcal{ALCQI}b^+_{reg}$

The restriction to regular RBoxes, which is crucial for the decidability of $\mathcal{SRIQ}$, ensures that all implications between roles can be captured by a set of regular expressions. This will allow us to reduce reasoning in $\mathcal{SRIQ}$ to reasoning in $\mathcal{ALCQI}b^+_{reg}$, and to show that the algorithm we have presented is also an optimal decision procedure for $\mathcal{SRIQ}$ KB satisfiability.

If $\mathcal{R}$ is a regular RBox, then for every role $R \in \overline{\mathbf{R}}$ there is a regular expression ρ_R such that a word $w = R_1 \circ \ldots \circ R_n$ over $\overline{\mathbf{R}}$ is in the language of ρ_R iff $w^{\mathcal{I}} \subseteq R^{\mathcal{I}}$ in every model $\mathcal{I}$ of $\mathcal{R}$. This was shown in [9] for $\mathcal{RIQ}$ and its generalization to $\mathcal{SRIQ}$ is straightforward.[4]

Proposition 4. *Given a regular RBox* $\mathcal{R}$, *for each* $R \in \overline{\mathbf{R}}$ *there is a regular expression* ρ_R *such that* $R^{\mathcal{I}} = (\rho_R)^{\mathcal{I}}$ *for every model* $\mathcal{I}$ *of* $\mathcal{R}$.

Reducing TBoxes. Each regular expression ρ_R is an $\mathcal{ALCQI}b^+_{reg}$ role. Further, if a role S is simple in $\mathcal{R}$, then ρ_S is the regular expression $\bigcup_{(S' \sqsubseteq S) \in \mathcal{R}} S'$, which is in fact a simple $\mathcal{ALCQI}b^+_{reg}$ role. As a consequence, if we replace R by ρ_R is any $\mathcal{SRIQ}$ concept, the resulting expression is an $\mathcal{ALCQI}b^+_{reg}$ concept. This allows for a natural translation of $\mathcal{SRIQ}$ concepts and TBoxes into $\mathcal{ALCQI}b^+_{reg}$.

[3] In [7] also role assertions of the form $\mathsf{Tra}(R)$, asserting that R is transitive, are allowed. We omit them since this can be equivalently expressed with an SRIA $R \circ R \sqsubseteq R$.

[4] It follows also from [7] and the equivalence of finite state automata and regular expressions.

Definition 9. *Let $\mathcal{R}$ be a regular RBox. For each $R \in \overline{\mathbf{R}}$, ρ_R denotes a (fixed, arbitrary) regular expression such that $R^{\mathcal{I}} = (\rho_R)^{\mathcal{I}}$ for every model $\mathcal{I}$ of $\mathcal{R}$. For any $\mathcal{SRIQ}$ concept C, we denote by $C^{\mathcal{R}}$ the $\mathcal{ALCQI}b^+_{reg}$ concept that results from replacing each role R in C with ρ_R. Similarly, for any $\mathcal{SRIQ}$ TBox $\mathcal{T}$, we define $\mathcal{T}^{\mathcal{R}} = \{C^{\mathcal{R}} \sqsubseteq D^{\mathcal{R}} \mid C \sqsubseteq D \in \mathcal{T}\}$.*

Proposition 5. *Let $\mathcal{R}$ be a regular $\mathcal{SRIQ}$ RBox and $\mathcal{I}$ a model of $\mathcal{R}$. Then $R^{\mathcal{I}} = (\rho_R)^{\mathcal{I}}$ for every $\mathcal{SRIQ}$ role R and $C^{\mathcal{I}} = (C^{\mathcal{R}})^{\mathcal{I}}$ for every $\mathcal{SRIQ}$ concept C, and $\mathcal{I} \models \mathcal{T}$ iff $\mathcal{I} \models \mathcal{T}^{\mathcal{R}}$.*

Reducing ABoxes. Now we consider ABoxes. First, we remove the negative role assertions, which are not allowed in $\mathcal{ALCQI}b^+_{reg}$, as follows.

Definition 10. *Given an $\mathcal{SRIQ}$ ABox $\mathcal{A}$, let $\mathcal{A}^{neg}$ contain all negative role assertions $\neg S(a,b)$, and let $\mathcal{A}^{pos}$ be obtained by replacing each assertion $\neg S(a,b)$ in $\mathcal{A}^{neg}$ by $P_{\neg S}(a,b)$, where $P_{\neg S}$ is a fresh role name. Further, let $\mathcal{R}^{\mathcal{A}}$ be the set of $\mathcal{ALCQI}b^+_{reg}$ BRIAs containing $P_{\neg S} \cap S \sqsubseteq \mathsf{B}$ for each $\neg S(a,b)$ in $\mathcal{A}$.*

If an ABox contains an assertion in which a non-simple role R occurs, ρ_R is not a simple $\mathcal{ALCQI}b^+_{reg}$ role and hence we can not replace R with it. Instead, we explicitly add as assertions all relations between individuals that are implied by the RBox.

Definition 11. *Let $\mathcal{A}$ and $\mathcal{R}$ be a $\mathcal{SRIQ}$ ABox and a $\mathcal{SRIQ}$ RBox respectively. The $\mathcal{R}$-*extension of $\mathcal{A}$*, denoted $\mathcal{A}^{\mathcal{R}}$, is the minimal set of assertions containing $\mathcal{A}$ such that, if there are individuals $a_1, \ldots, a_{n+1} \in \mathbf{I}_{\mathcal{K}}$ and a SRIA $R_1 \circ \ldots \circ R_n \sqsubseteq R$ in $\mathcal{R}$ such that $R_i(a_i, a_{i+1}) \in \mathcal{A}^{\mathcal{R}}$ for each $1 \leq i \leq n$, then $R(a_1, a_{n+1}) \in \mathcal{A}^{\mathcal{R}}$.*

Reducing RBoxes. As for the RBox, the satisfaction of the SRIAs is ensured by the regular expressions added to the TBox and the assertions added to the ABox. We only have to ensure the satisfaction of the role assertions, which is easy in $\mathcal{ALCQI}b^+_{reg}$.

Definition 12. *Let $\mathcal{R}$ be a regular $\mathcal{SRIQ}$ RBox and let $\mathcal{R}'$ denote the set of role assertions in $\mathcal{R}$. We denote by $\mathcal{R}^B$ the set of BRIAs and CIAs that contains:*
- *$R \sqsubseteq R'$ for every SRIA $R \sqsubseteq R'$ in $\mathcal{R}$ such that $R' \in \mathsf{SR}(\mathcal{R})$,*
- *$R \sqsubseteq \mathsf{Inv}(R)$ and $\mathsf{Inv}(R) \sqsubseteq R$ for every assertion $\mathsf{Sym}(R)$ in $\mathcal{R}'$,*
- *$\top \sqsubseteq \exists R.\mathsf{Self}$ for every assertion $\mathsf{Ref}(R)$ in $\mathcal{R}'$,*
- *$\exists R.\mathsf{Self} \sqsubseteq \bot$ for every assertion $\mathsf{Irr}(R)$ in $\mathcal{R}'$, and*
- *$R \cap R' \sqsubseteq \mathsf{B}$ for every assertion $\mathsf{Dis}(R, R')$ in $\mathcal{R}'$.*

Reducing KBs. Now we can reduce any given $\mathcal{SRIQ}$ KB $\mathcal{K} = \langle \mathcal{A}, \mathcal{T}, \mathcal{R} \rangle$ into an equivalent $\mathcal{ALCQI}b^+_{reg}$ KB $\mathcal{K} = \langle \mathcal{A}, \mathcal{T} \rangle$ as follows.

Definition 13. *Let $\mathcal{K} = \langle \mathcal{A}, \mathcal{T}, \mathcal{R} \rangle$ be a $\mathcal{SRIQ}$ knowledge base. We denote by $\mathcal{K}' = \langle \mathcal{A}', \mathcal{T}' \rangle$ the following $\mathcal{ALCQI}b^+_{reg}$ KB:*
- *The ABox $\mathcal{A}'$ is $\mathrm{A}^{\mathcal{R}}$, where $\mathrm{A} = (\mathcal{A} \setminus \mathcal{A}^{neg}) \cup \mathcal{A}^{pos}$.*
- *The TBox $\mathcal{T}'$ is $\mathcal{T}^{\mathcal{R}} \cup \mathcal{R}^{\mathcal{A}} \cup \mathcal{R}^B$.*

For any interpretation $\mathcal{I}$, $\mathcal{I} \models \mathcal{K}$ implies $\mathcal{I} \models \mathcal{K}'$. The converse holds in a slightly weaker form. While a model of $\mathcal{K}'$ need not be closed under the RBox and it may not be a model of $\mathcal{K}$, the models of $\mathcal{K}'$ and $\mathcal{K}$ may only differ in the interpretation of some 'implied' roles. Adding them to a model of $\mathcal{K}'$, we obtain a model of $\mathcal{K}$.

For a model $\mathcal{I}$ of $\mathcal{K}'$, the *extension of* $\mathcal{I}$ *to* $\mathcal{R}$ is the interpretation $\mathcal{I}^{\mathcal{R}}$ such that $\Delta^{\mathcal{I}^{\mathcal{R}}} = \Delta^{\mathcal{I}}$; $(A)^{\mathcal{I}^{\mathcal{R}}} = (A)^{\mathcal{I}}$ for every atomic concept A; and for every atomic role P occurring in $\mathcal{R}$, if $(x, y) \in (\rho_P)^{\mathcal{I}}$ or $(y, x) \in (\rho_{P^-})^{\mathcal{I}}$, then $(x, y) \in (P)^{\mathcal{I}^{\mathcal{R}}}$. It can be easily verified that if $\mathcal{I} \models \mathcal{K}'$, then $\mathcal{I}^{\mathcal{R}} \models \mathcal{K}$. Hence, we can reduce the satisfiability problem for any $\mathcal{SRIQ}$ knowledge bases to satisfiability of an $\mathcal{ALCQI}b^+_{reg}$ one.

Proposition 6. *Let $\mathcal{K} = \langle \mathcal{A}, \mathcal{T}, \mathcal{R} \rangle$ be a $\mathcal{SRIQ}$ knowledge base. Then $\mathcal{K}$ is satisfiable iff $\mathcal{K}'$ is satisfiable.*

As a consequence, the automata algorithm in Section 3 can be used to decide the satisfiability of $\mathcal{SRIQ}$ knowledge bases.

Furthermore, it is worst case optimal. All steps are clearly polynomial in the size of $\mathcal{A}$, $\mathcal{T}$ and $\mathsf{exp}(\mathcal{R})$. However, the size of $\mathsf{exp}(\mathcal{R})$ can be exponential in the size of $\mathcal{R}$ [9].

Theorem 3. *The satisfiability of an $\mathcal{SRIQ}$ KB $\mathcal{K} = \langle \mathcal{A}, \mathcal{T}, \mathcal{R} \rangle$ is decidable in time exponential in the combined sizes of $\mathcal{T}$, $\mathcal{A}$ and $\mathsf{exp}(\mathcal{R})$, and double exponential in the size of $\mathcal{K}$.*

It was recently shown that $\mathcal{SRIQ}$ is 2EXPTIME hard [11], hence our bound is optimal.

Corollary 2. *The satisfiability of $\mathcal{SRIQ}$ knowledge bases is* 2EXPTIME*-complete.*

Note that the blow-up in complexity w.r.t. $\mathcal{ALCQI}b^+_{reg}$ is due to the size of $\mathsf{exp}(\mathcal{R})$, and that the algorithm is single exponential whenever $\mathsf{exp}(\mathcal{R})$ is polynomial in $\mathcal{R}$; this includes, for example, the so-called *simple role hierarchies* defined in [9]. Our algorithm compares well to the $\mathcal{SRIQ}$ algorithm in [7], which may require non-deterministic double exponential time even for such restricted cases.

5 Conclusions

We have shown that the DL $\mathcal{ALCQI}b^+_{reg}$ fully captures the DL $\mathcal{SRIQ}$. Although the latter is exponentially more succinct, reasoning is also (provably) exponentially harder in the worst case. The syntax of $\mathcal{ALCQI}b^+_{reg}$ allows for additional constructions not supported by $\mathcal{SRIQ}$, like Boolean role expressions and inclusions between them. Since these additional features can be useful for modeling KBs and for ontology engineering [13], it seems reasonable to consider $\mathcal{ALCQI}b^+_{reg}$ as a suitable alternative for $\mathcal{SRIQ}$. It would even be possible to modify the syntax of $\mathcal{ALCQI}b^+_{reg}$ and make it more similar to $\mathcal{SRIQ}$ from the perspective of ontology engineering. For example, instead of using the regular expressions in concepts as in $\mathcal{ALCQI}b^+_{reg}$, a special set of defined role names r_R could be used, and assertions of the form $\rho_R \sqsubseteq r_R$ added to the KB. Provided that the defined roles r_R are not used as atomic roles in complex role expressions, reasoning could be done with the techniques described here. The resulting logic would be a very simple and natural extension of $\mathcal{SRIQ}$ and would avoid the effort of recognizing whether a given role hierarchy is regular and of transforming it into a set of finite automata or regular expressions. Note that the language of the regular expressions supported in (such a variation of) $\mathcal{ALCQI}b^+_{reg}$ could also be extended to be exponentially more succinct (e.g. by allowing squaring of expressions), resulting in a logic that may be comparable to $\mathcal{SRIQ}$ in terms of succinctness.

In order for $\mathcal{ALCQI}b^+_{reg}$ to become a suitable alternative to $\mathcal{SRIQ}$ in practice, the main challenge that remains ahead is to explore alternatives for practicable algorithms for this kind of logics, and to implement them in actual reasoners. Unfortunately, DLs that support transitive closure, or more generally, regular expressions over roles in the style of PDL, have largely been displaced by the better known alternative of role hierarchies on which implemented reasoners have focused. While it is often claimed that the development systems that support such logics is problematic, to our knowledge there have been no major attempts to do it. Regular expressions are nevertheless worth exploring, as they are well understood and widely used in computer science, and are likely to be useful for ontology engineering.

Another natural alternative is to extend $\mathcal{SRIQ}$ with more constructors from $\mathcal{ALCQI}b^{+}_{reg}$. In particular, one can consider $\mathcal{SRIQ}_b$, the extension of $\mathcal{SRIQ}$ with safe Boolean role expressions; similar extensions of closely related DLs like $\mathcal{SHIQ}$ and $\mathcal{SROIQ}$ were already considered in [13]. It is easy to verify that any $\mathcal{SRIQ}_b$ KB can be transformed into an $\mathcal{ALCQI}b^{+}_{reg}$ one with the reduction above, and the results given for $\mathcal{SRIQ}$ extend to $\mathcal{SRIQ}_b$ as well. Our results also show that $\mathcal{SRIQ}$ and $\mathcal{SRIQ}_b$ can be easily extended with a *universal* role (which is already supported in $\mathcal{SROIQ}$) as long as it is considered to be non-simple. Note that relaxing this restriction and allowing the universal role in the Boolean role expressions of $\mathcal{SRIQ}_b$ makes reasoning NExpTime hard, see [12].

Finally, we point out that with the techniques given here, the query answering problem for $\mathcal{SRIQ}$ can be reduced to $\mathcal{ALCQI}b^{+}_{reg}$, showing that answering *two-way positive regular path queries* as defined in [4] over $\mathcal{SRIQ}$ and $\mathcal{SRIQ}_b$ knowledge bases is decidable.

References

1. F. Baader, D. Calvanese, D. McGuinness, D. Nardi, and P. F. Patel-Schneider, editors. *The Description Logic Handbook*. Cambridge University Press, 2003.
2. D. Calvanese, G. De Giacomo, and M. Lenzerini. 2ATAs make DLs easy. In *Proc. of DL 2002*, pages 107–118. CEUR Electronic Workshop Proceedings, vol. 53, 2002.
3. D. Calvanese, T. Eiter, and M. Ortiz. Answering regular path queries in expressive description logics: An automata-theoretic approach. Technical Report INFSYS RR-1843-08-05.
4. D. Calvanese, T. Eiter, and M. Ortiz. Answering regular path queries in expressive description logics: An automata-theoretic approach. In *Proc. of AAAI'2007*, pages 391–396, 2007.
5. E. A. Emerson and C. S. Jutla. Tree automata, mu-calculus and determinacy. In *Proc. of FOCS'91*, pages 368–377, 1991.
6. J. Heflin and J. Hendler. A portrait of the Semantic Web in action. *IEEE Intelligent Systems*, 16(2):54–59, 2001.
7. I. Horrocks, O. Kutz, and U. Sattler. The irresistible $\mathcal{SRIQ}$. In *Proc. OWLED 2005*, 2005.
8. I. Horrocks, O. Kutz, and U. Sattler. The even more irresistible $\mathcal{SROIQ}$. In *Proc. KR 2006*, 2006.
9. I. Horrocks and U. Sattler. Decidability of $\mathcal{SHIQ}$ with complex role inclusion axioms. In *Proc. of IJCAI 2003*. Morgan-Kaufmann Publishers, 2003.
10. I. Horrocks, U. Sattler, and S. Tobies. Reasoning with individuals for the description logic $\mathcal{SHIQ}$. In *Proc. of CADE 2000*, pages 482–496, 2000.
11. Y. Kazakov. $\mathcal{RIQ}$ and $\mathcal{SROIQ}$ are harder than $\mathcal{SHOIQ}$. In *Proc. KR'08*, 2008.
12. C. Lutz and U. Sattler. The complexity of reasoning with Boolean modal logic. In *Proc. of Advances in Modal Logic 2000 (AiML 2000)*, 2000.
13. S. Rudolph, M. Krötzsch, and P. Hitzler. Cheap boolean role constructors for description logics. In *Proc. of JELIA 2008*, volume 5293 of *Lecture Notes in Computer Science*, pages 362–374, 2008.
14. K. Schild. A correspondence theory for terminological logics: Preliminary report. In *Proc. of IJCAI 1991*, pages 466–471, 1991.
15. S. Tobies. *Complexity Results and Practical Algorithms for Logics in Knowledge Representation*. PhD thesis, LuFG Theoretical Computer Science, RWTH-Aachen, Germany, 2001.
16. M. Y. Vardi. Reasoning about the past with two-way automata. In *Proc. of ICALP 2002*, volume 1443 of *LNCS*, pages 628–641. Springer, 1998.
17. M. Y. Vardi and P. Wolper. Automata-theoretic techniques for modal logics of programs. *J. Comput. Syst. Sci.*, 32:183–221, 1986.