All Elephants are Bigger than All Mice
Technical Report

Sebastian Rudolph and Markus Krötzsch and Pascal Hitzler

Institut AIFB, Universität Karlsruhe, Germany
{sru,mak,phi}@aifb.uni-karlsruhe.de

Abstract. We introduce the *concept product* as a new expressive feature to description logics (DLs). While this construct allows us to express an arguably very common and natural type of statement, it can be simulated only by the very expressive DL $\mathcal{SROIQ}$ for which no tight worst-case complexity is known. However, we show that concept products can also be added to the DLs $\mathcal{SHOIQ}$ and $\mathcal{SHOI}$, and to the tractable DL $\mathcal{EL}^{++}$ without increasing the worst-case complexities in any of those cases. We therefore argue that concept products provide practically relevant expressivity at little cost, making them a good candidate for future extensions of the DL-based ontology language OWL.

1 Introduction

The development of description logics (DLs) has been driven by the desire to push the expressivity bounds of these knowledge representation formalisms while still maintaining decidability and implementability. This has lead to very expressive DLs such as $\mathcal{SHOIN}$, the logic underlying the Web Ontology Language OWL DL, $\mathcal{SHOIQ}$, and more recently $\mathcal{SROIQ}$ [1] which is the basis for the ongoing standardisation of OWL2[1] as the next version of the Web Ontology Language. On the other hand, more light-weight DLs for which most common reasoning problems can be implemented in (sub)polynomial time have also been sought, leading, e.g., to the tractable DL $\mathcal{EL}^{++}$ [2].

In this work, we continue these lines of research by introducing a new expressive feature – the *concept product* – to various well-known DLs, showing that this added expressivity does not increase worst-case complexities in any of these cases. Intuitively, the concept product allows us to define a role that connects every instance in one class with every instance in another class. An example is given in the title: Given the class of all elephants, and the class of all mice, we wish to specify a DL knowledge base that allows us to conclude that any individual elephant is bigger than any individual mouse, or, stated more formally:

$$\forall(x).\forall y.\mathsf{Elephant}(x) \land \mathsf{Mouse}(y) \to \mathsf{biggerThan}(x, y)$$

Using common DL syntax, one could also write $\mathsf{Elephant}^{\mathcal{I}} \times \mathsf{Mouse}^{\mathcal{I}} \subseteq \mathsf{biggerThan}^{\mathcal{I}}$, which explains the name “concept product” and will also motivate our DL syntax.

[1] http://www.w3.org/2007/OWL

Maybe surprisingly, this semantic relationship cannot be specified in any but the most expressive DLs today. Using quantifiers, one can only state that any elephant is bigger than *some* mouse, or that elephants are bigger than nothing but mice. Nominals also allow us to state that some particular elephant is bigger than all mice, and with DL-safe rules [3], one might say that all *named* elephants are bigger than all *named* mice. Yet, none of these formalisations captures the true intention of the informal statement.

Now one could hope that this kind of statement would be rarely needed in practical applications, but in fact it represents a very common modelling problem of relating two individuals based on their (inferred) properties. Natural and life sciences provide a wealth of typical examples, for example:

- Alkaline solutions neutralise acid solutions.
- Antihistamines alleviate allergies.
- Oppositely charged bodies attract each other.

Reasoning about such relations qualitatively is important for example in the context of the HALO project[2], which sets out to develop reasoning systems for solving complex examination questions from physics, biology, and chemistry. Qualitative reasoning about a given scenario is often required before any concrete arithmetic processing steps can be invoked.

Another particularly interesting example is the task of developing a knowledge base capturing our current insights about DL complexities and available reasoning implementations. It should entail statements like

- Any reasoner that can handle $\mathcal{SHIQ}$ can deal with every DLP-ontology.
- Any problem within ExpTime can be polynomially reduced to any ExpTime-complete problem.
- In any description logic containing nominals, inverses and number restrictions, satisfiability checking is hard for any complexity below or equal ExpTime.

All of those can easily be cast into concept products. An interesting aspect of reasoning about complexities is that it involves upper *and* lower bounds, and thus also escapes from most other modelling attempts (e.g. using classes instead of instances to represent concrete DLs). This might be a reason that the DL complexity navigator[3] is based on JavaScript rather than on any of the more advanced DL knowledge representation technologies.

In this paper, we show that it is in fact not so difficult to extend a broad array of existing description logics with enough additional modelling power to capture all of the above, while still retaining their known upper complexity bounds. We start with the short preliminary Section 2 to recall the definition of the DL $\mathcal{SROIQ}$, and then proceed by introducing the concept product formally in Section 3. Concept products there can indeed be simulated by existing constructs and thus are recognised as syntactic sugar. This is quite different for the tractable DL $\mathcal{EL}^{++}$ investigated in Section 4. Yet, we will see that polynomial reasoning in $\mathcal{EL}^{++}$ with concept products is possible, thus further

[2] `http://www.projecthalo.com/`
[3] `http://www.cs.man.ac.uk/~ezolin/dl/`

pushing the $\mathcal{EL}$ envelope. In the subsequent Section 5, we show that $\mathcal{SHOIQ}$ with concept products is still NExpTime-complete, thus obtaining tight complexity bounds for a very expressive DL as well. Finally, we establish a similar result for $\mathcal{SHOI}$ and ExpTime-completeness in Section 6, and then provide an outlook on upcoming work in Section 7.

2 Preliminaries: the DL $\mathcal{SROIQ}$

In this section, we recall the definition of the expressive description logic $\mathcal{SROIQ}$ [1]. We assume that the reader is familiar with description logics [4].

As usual, the DLs considered in this paper are based on three disjoint sets of *individual names* N_I, *concept names* N_C, and *role names* N_R containing the *universal role* $U \in \mathsf{N}_R$.

Definition 1. *A $\mathcal{SROIQ}$ Rbox for N_R is based on a set* $\mathbf{R}$ *of* atomic roles *defined as* $\mathbf{R} := \mathsf{N}_R \cup \{R^- \mid R \in \mathsf{N}_R\}$, *where we set* $\mathrm{Inv}(R) := R^-$ *and* $\mathrm{Inv}(R^-) := R$ *to simplify notation. In the sequel, we will use the symbols R, S, possibly with subscripts, to denote atomic roles.*

A generalised role inclusion axiom *(RIA) is a statement of the form* $S_1 \circ \ldots \circ S_n \sqsubseteq R$, *and a set of such RIAs is a generalised* role hierarchy. *A role hierarchy is* regular *if there is a strict partial order $\prec$ on* $\mathbf{R}$ *such that*

- $S \prec R$ *iff* $\mathrm{Inv}(S) \prec R$, *and*
- *every RIA is of one of the forms:*

  $$R \circ R \sqsubseteq R, \quad R^- \sqsubseteq R, \quad S_1 \circ \ldots \circ S_n \sqsubseteq R, \quad R \circ S_1 \circ \ldots \circ S_n \sqsubseteq R, \quad S_1 \circ \ldots \circ S_n \circ R \sqsubseteq R$$

  such that $R \in \mathsf{N}_R$ is a (non-inverse) role name, and $S_i \prec R$ for $i = 1, \ldots, n$.

The set of simple *roles for some role hierarchy is defined inductively as follows:*

- *If a role R occurs only on the right-hand-side of RIAs of the form $S \sqsubseteq R$ such that S is simple, then R is also simple.*
- *The inverse of a simple role is simple.*

A role assertion *is a statement of the form* $\mathsf{Ref}(R)$ *(*reflexivity*)*, $\mathsf{Asy}(S)$ *(*asymmetry*), or* $\mathsf{Dis}(S, S')$ *(*role disjointness*), where S and S' are simple. A $\mathcal{SROIQ}$* Rbox *is the union of a set of role assertions together and a role hierarchy. A $\mathcal{SROIQ}$ Rbox is regular if its role hierarchy is regular.*

Definition 2. *Given a $\mathcal{SROIQ}$ Rbox $\mathcal{R}$, the set of* concept expressions $\mathbf{C}$ *is defined as follows:*

- $\mathsf{N}_C \subseteq \mathbf{C}$, $\top \in \mathbf{C}$, $\bot \in \mathbf{C}$,
- *if $C, D \in \mathbf{C}$, $R \in \mathbf{R}$, $S \in \mathbf{R}$ a simple role, $a \in \mathsf{N}_I$, and n a non-negative integer, then* $\neg C$, $C \sqcap D$, $C \sqcup D$, $\{a\}$, $\forall R.C$, $\exists R.C$, $\exists S.\mathsf{Self}$, ${\leq}n\,S.C$, *and* ${\geq}n\,S.C$ *are also concept expressions.*

Table 1. Semantics of concept constructors in $\mathcal{SROIQ}$ for an interpretation $\mathcal{I}$ with domain $\Delta^{\mathcal{I}}$.

Name	Syntax	Semantics
inverse role	R^-	$\{\langle x, y\rangle \in \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \mid \langle y, x\rangle \in R^{\mathcal{I}}\}$
universal role	U	$\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$
top	$\top$	$\Delta^{\mathcal{I}}$
bottom	$\bot$	$\emptyset$
negation	$\neg C$	$\Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$
conjunction	$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
disjunction	$C \sqcup D$	$C^{\mathcal{I}} \cup D^{\mathcal{I}}$
nominals	$\{a\}$	$\{a^{\mathcal{I}}\}$
univ. restriction	$\forall R.C$	$\{x \in \Delta^{\mathcal{I}} \mid \langle x, y\rangle \in R^{\mathcal{I}}$ implies $y \in C^{\mathcal{I}}\}$
exist. restriction	$\exists R.C$	$\{x \in \Delta^{\mathcal{I}} \mid$ for some $y \in \Delta^{\mathcal{I}}, \langle x, y\rangle \in R^{\mathcal{I}}$ and $y \in C^{\mathcal{I}}\}$
Self concept	$\exists S.\mathsf{Self}$	$\{x \in \Delta^{\mathcal{I}} \mid \langle x, x\rangle \in S^{\mathcal{I}}\}$
qualified number	$\leq n\, S.C$	$\{x \in \Delta^{\mathcal{I}} \mid \#\{y \in \Delta^{\mathcal{I}} \mid \langle x, y\rangle \in S^{\mathcal{I}}$ and $y \in C^{\mathcal{I}}\} \leq n\}$
restriction	$\geq n\, S.C$	$\{x \in \Delta^{\mathcal{I}} \mid \#\{y \in \Delta^{\mathcal{I}} \mid \langle x, y\rangle \in S^{\mathcal{I}}$ and $y \in C^{\mathcal{I}}\} \geq n\}$

Throughout this paper, the symbols C, D will be used to denote concept expressions. A $\mathcal{SROIQ}$ Tbox *is a set of* general concept inclusion axioms *(GCIs) of the form $C \sqsubseteq D$.*

An individual assertion *can have any of the following forms: $C(a)$, $R(a, b)$, $\neg R(a, b)$, $a \not\approx b$, with $a, b \in \mathsf{N}_I$ individual names, $C \in \mathbf{C}$ a concept expression, and $R, S \in \mathbf{R}$ roles with S simple. A* $\mathcal{SROIQ}$ Abox *is a set of individual assertions.*

A $\mathcal{SROIQ}$ knowledge base KB *is the union of a regular Rbox $\mathcal{R}$, and an Abox $\mathcal{A}$ and Tbox $\mathcal{T}$ for $\mathcal{R}$.*

We further recall the semantics of $\mathcal{SROIQ}$ knowledge bases.

Definition 3. *An interpretation $\mathcal{I}$ consists of a set $\Delta^{\mathcal{I}}$ called* domain *(the elements of it being called* individuals*) together with a function $\cdot^{\mathcal{I}}$ mapping*

- *individual names to elements of $\Delta^{\mathcal{I}}$,*
- *concept names to subsets of $\Delta^{\mathcal{I}}$, and*
- *role names to subsets of $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$.*

The function $\cdot^{\mathcal{I}}$ is inductively extended to role and concept expressions as shown in Table 1. An interpretation $\mathcal{I}$ satisfies *an axiom φ if we find that $\mathcal{I} \models \varphi$:*

- *$\mathcal{I} \models S \sqsubseteq R$ if $S^{\mathcal{I}} \subseteq R^{\mathcal{I}}$,*
- *$\mathcal{I} \models S_1 \circ \ldots \circ S_n \sqsubseteq R$ if $S_1^{\mathcal{I}} \circ \ldots \circ S_n^{\mathcal{I}} \sqsubseteq R^{\mathcal{I}}$ ($\circ$ being overloaded to denote the standard composition of binary relations here),*
- *$\mathcal{I} \models \mathsf{Ref}(R)$ if $R^{\mathcal{I}}$ is a reflexive relation,*
- *$\mathcal{I} \models \mathsf{Asy}(R)$ if $R^{\mathcal{I}}$ is antisymmetric and irreflexive,*
- *$\mathcal{I} \models \mathsf{Dis}(R, S)$ if $R^{\mathcal{I}}$ and $S^{\mathcal{I}}$ are disjoint,*
- *$\mathcal{I} \models C \sqsubseteq D$ if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$.*

An interpretation $\mathcal{I}$ satisfies *a knowledge base* KB *(we then also say that $\mathcal{I}$ is a* model *of* KB *and write $\mathcal{I} \models$* KB*) if it satisfies all axioms of* KB*. A knowledge base* KB *is* satisfiable *if it has a model. Two knowledge bases are* equivalent *if they have exactly the same models, and they are* equisatisfiable *if either both are unsatisfiable or both are satisfiable.*

Further details on $\mathcal{SROIQ}$ can be found in [1]. We have omitted here several syntactic constructs that can be expressed indirectly, especially role assertions for transitivity, reflexivity of simple roles, and symmetry.

3 Simulating Concept Products in $\mathcal{SROIQ}$

We now formally introduce the DL *concept product* as a new constructor in description logic knowledge bases. The DL $\mathcal{SROIQ}$ extended with this constructor will be denoted $\mathcal{SROIQ}^{\times}$. It will turn out that concept products appear as *syntactic sugar* in $\mathcal{SROIQ}^{\times}$ since they can be represented by combining nominals, inverse roles, and complex role inclusion axioms. On the other hand, the universal role is recognised as a special case of concept product, though, as we will discuss below, our simulation method imposes some additional restrictions on simplicity of roles.

Definition 4. *A* concept product inclusion *is a statement of the form $C \times D \sqsubseteq R$ where $C, D \in \mathbf{C}$ are $\mathcal{SROIQ}$ concepts, and R is an atomic $\mathcal{SROIQ}$ role.*

A $\mathcal{SROIQ}^{\times}$ Rbox is the union of a $\mathcal{SROIQ}$ Rbox with a set of concept product inclusions based on roles and concepts for that Rbox. Simplicity of roles is defined as in $\mathcal{SROIQ}$ where concept product axioms are considered as additional kinds of RIAs. Especially, any role R occuring in such a statement is not simple in $\mathcal{SROIQ}^{\times}$.

A $\mathcal{SROIQ}^{\times}$ knowledge base KB *is the union of a $\mathcal{SROIQ}^{\times}$ Rbox $\mathcal{R}$, and a $\mathcal{SROIQ}$ Abox $\mathcal{A}$ and $\mathcal{SROIQ}$ Tbox $\mathcal{T}$ (for $\mathcal{R}$).*

The model theoretic semantics of $\mathcal{SROIQ}$ is extended to $\mathcal{SROIQ}^{\times}$ by setting

$$\mathcal{I} \models C \times D \sqsubseteq R \text{ iff } C^{\mathcal{I}} \times D^{\mathcal{I}} \subseteq R^{\mathcal{I}}$$

for any interpretation $\mathcal{I}$.

In the remainder of this section, we discuss some basic formal properties of the concept product. We immediately observe that $\times$ generalises the universal role, which can now be defined by the axiom $\top \times \top \sqsubseteq U$. However, our extension of the notion of simplicity of roles would then cause U to become non-simple, which is not needed. In fact, we conjecture that one can generally consider the concept product to have no impact on simplicity of roles, but our below approach of simulating concept products in $\mathcal{SROIQ}$ requires us to impose that restriction. We leave it to future work to conceive a modified tableau procedure for $\mathcal{SROIQ}^{\times}$ that directly takes the cross product into account – our subsequent results for $\mathcal{SHOIQ}^{\times}$ show that this extended version of simplicity does not impose any problems there.

We now find that $\times$ itself can be expressed by existing constructs of $\mathcal{SROIQ}$:

Lemma 5. *Consider a $\mathcal{SROIQ}^{\times}$ knowledge base* KB *with some concept product axiom $C \times D \sqsubseteq R$, and let* KB′ *be the knowledge base obtained from* KB *as follows:*

- *delete the Rbox axiom $C \times D \sqsubseteq R$,*
- *add a new RIA $R_1 \circ R_2 \sqsubseteq R$, where R_1, R_2 are fresh role names,*
- *introduce fresh nominal $\{a\}$, and add the Tbox axioms $C \sqsubseteq \exists R_1.\{a\}$ and $D \sqsubseteq \exists R_2^-.\{a\}$.*

Then KB *and* KB′ *are equisatisfiable.*

Proof. First note that the introduced axioms are indeed admissible for $\mathcal{SROIQ}$, and that regularity of the Rbox is not endangered.

Now we show that for any model $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ of KB we can construct a model $\mathcal{J} = (\Delta^{\mathcal{J}}, \cdot^{\mathcal{J}})$ of KB′ as follows:

- $\Delta^{\mathcal{J}} := \Delta^{\mathcal{I}}$,
- for all $i \in \mathsf{N}_I \setminus \{a\}$, let $i^{\mathcal{J}} := i^{\mathcal{I}}$
- let $a^{\mathcal{J}} = \delta_a$ for an arbitrary but fixed $\delta_a \in \Delta^{\mathcal{J}}$
- for all $A \in \mathsf{N}_C$, let $A^{\mathcal{J}} := A^{\mathcal{I}}$
- for all $T \in \mathsf{N}_R \setminus \{R_1, R_2\}$, let $R^{\mathcal{J}} := R^{\mathcal{I}}$
- let $R_1^{\mathcal{J}} = C^{\mathcal{I}} \times \{\delta_o\}$
- let $R_2^{\mathcal{J}} = D^{\mathcal{I}} \times \{\delta_o\}$

Then by construction, the new KB-axioms $C \sqsubseteq \exists R_1.\{a\}$ and $D \sqsubseteq \exists R_2.\{a\}$ are satisfied in $\mathcal{J}$. Next note that for all concept expressions E not containing R_1, R_2 or a, we have $E^{\mathcal{I}} = E^{\mathcal{J}}$ which follows by an easy structural induction from the fact that the interpretation of the previously present roles, atomic concepts and individuals which coincides in $\mathcal{I}$ and $\mathcal{J}$.

Thereby we obtain that all Tbox axioms from KB ∩ KB′ are valid in $\mathcal{J}$. Moreover, the construction of $\mathcal{J}$ and the validity of $C \times D \sqsubseteq R$ in $\mathcal{I}$ together assure $R_1^{\mathcal{J}} \circ R_2^{-\mathcal{J}} \subseteq A^{\mathcal{J}} \times \subseteq R^{\mathcal{J}}$ therefore also the newly introduced RIA $R_1 \circ R_2 \sqsubseteq R$ is satisfied in $\mathcal{J}$.

Finally, we observe that any model $\mathcal{I}$ of KB′ is a model of KB: from $C \sqsubseteq \exists R_1.\{a\}$ and $D \sqsubseteq \exists R_2.\{a\}$ follows $C^{\mathcal{I}} \times \{a^{\mathcal{I}}\} \subseteq R_1^{\mathcal{I}}$ as well as $D^{\mathcal{I}} \times \{a^{\mathcal{I}}\} \subseteq R_2^{\mathcal{I}}$ the latter being equivalent to $\{a^{\mathcal{I}}\} \times D^{\mathcal{I}} \subseteq R_2{}^{-\mathcal{I}}$. Hence, we can conclude $C^{\mathcal{I}} \times D^{\mathcal{I}} \subseteq R_1^{\mathcal{I}} \circ R_2^{-\mathcal{I}}$. Now due to the RIA $R_1 \circ R_2 \sqsubseteq R$ being satisfied in $\mathcal{I}$ as well we know that $R_1^{\mathcal{I}} \circ R_2^{-\mathcal{I}} \subseteq R^{\mathcal{I}}$, and can conclude $C^{\mathcal{I}} \times D^{\mathcal{I}} \subseteq R^{\mathcal{I}}$. Hence also the cross product inclusion $C \times D \sqsubseteq R$ is satisfied in $\mathcal{I}$. All other axioms of KB are present in KB′ as well and therefore satisfied anyway. □

Clearly, the elimination step from the above lemma can be applied recursively to eliminate all concept products. A simple induction thus yields the following result:

Proposition 6. *Every $\mathcal{SROIQ}^{\times}$ knowledge base can be reduced to an equisatisfiable $\mathcal{SROIQ}$ knowledge base in polynomial time. In particular, satisfiability of $\mathcal{SROIQ}^{\times}$ knowledge bases is decidable.*

Decidability of $\mathcal{SROIQ}$ was shown in [1]. Since $\mathcal{SROIQ}$ is already NExpTime-hard, this also suffices to conclude that the (currently unknown) worst-case complexities of $\mathcal{SROIQ}^{\times}$ and $\mathcal{SROIQ}$ coincide.

4 Polynomial Reasoning with Concept Products in $\mathcal{EL}^{++}$

In this section, we investigate the use of concepts products in the DL $\mathcal{EL}^{++}$ [2], for which many typical inference problems can be solved in polynomial time. $\mathcal{EL}^{++}$ cannot

Table 2. Normal form transformation for $\mathcal{EL}^{++\times}$. A, B, C, $\hat{A}$, $\hat{C}$, and D are concept expressions, where $\hat{A}$ and $\hat{C}$ are neither concept names nor nominals, and D is a fresh concept name. R_i, S, and T are role names, where T is fresh. Commutativity of $\sqcap$ is assumed to simplify the rule set.

P1:	$R_1 \circ \ldots \circ R_{n-1} \circ R_n \sqsubseteq S$	$\mapsto$	$\{R_1 \circ \ldots \circ R_{n-1} \sqsubseteq T, T \circ R_n \sqsubseteq S\}$
	$\hat{A} \times B \sqsubseteq R$	$\mapsto$	$\{\hat{A} \sqsubseteq D, D \times B \sqsubseteq R\}$
	$A \times \hat{B} \sqsubseteq R$	$\mapsto$	$\{\hat{B} \sqsubseteq D, A \times D \sqsubseteq R\}$
	$B \sqcap \hat{A} \sqsubseteq C$	$\mapsto$	$\{\hat{A} \sqsubseteq D, D \sqcap B \sqsubseteq C\}$
	$\exists R.\hat{A} \sqsubseteq B$	$\mapsto$	$\{\hat{A} \sqsubseteq D, \exists R.D \sqsubseteq B\}$
	$\bot \sqsubseteq C$	$\mapsto$	$\emptyset$
P2:	$A \sqsubseteq B \sqcap C$	$\mapsto$	$\{A \sqsubseteq B, A \sqsubseteq C\}$
	$\hat{A} \sqsubseteq \hat{C}$	$\mapsto$	$\{\hat{A} \sqsubseteq D, D \sqsubseteq \hat{C}\}$
	$A \sqsubseteq \exists R.\hat{C}$	$\mapsto$	$\{A \sqsubseteq \exists R.D, D \sqsubseteq \hat{C}\}$
	$A \sqsubseteq \top$	$\mapsto$	$\emptyset$

simulate concept products as it does support nominals and RIAs, but no inverse roles. While it is known that the addition of inverses makes satisfiability checking ExpTime-complete [5], we show that sound and complete reasoning with the concept product is still tractable. We simplify our presentation by omitting concrete domains from $\mathcal{EL}^{++}$ – they are not affected by our extension and can be treated as shown in [2].

Definition 7. *An $\mathcal{EL}^{++\times}$ knowledge base* KB *is a $\mathcal{SROIQ}^{\times}$ knowledge base that contains only constructors $\top$, $\bot$, $\sqcap$, $\exists R$ for some (non-inverse) role name $R \in \mathsf{N}_R$, and $\{a\}$ for some individual name $a \in \mathsf{N}_I$, possibly with a non-regular role box.*

A polynomial algorithm for checking class subsumptions in $\mathcal{EL}^{++}$ has been given in [2], and it was shown that other standard inference problems can easily be reduced to that problem. We now present a modified subsumption checking algorithm for $\mathcal{EL}^{++\times}$ – also using some modified notation – and show its correctness for this extended DL.

Without loss of generality, we assume that all Abox axioms in $\mathcal{EL}^{++\times}$ are expressed by equivalent Tbox axioms using nominals. We can further restrict our attention to $\mathcal{EL}^{++\times}$ knowledge bases in a certain normal form:

Definition 8. *An $\mathcal{EL}^{++\times}$ knowledge base* KB *is in* normal form *if it contains only axioms of one of the following forms:*

$$A \sqsubseteq C \qquad A \sqcap B \sqsubseteq C \qquad R \sqsubseteq T \qquad A \times B \sqsubseteq T$$
$$\exists R.A \sqsubseteq B \qquad A \sqsubseteq \exists R.B \qquad R \circ S \sqsubseteq T$$

where $A, B \in \mathsf{N}_C \cup \{\{a\} \mid a \in \mathsf{N}_I\} \cup \{\top\}$, $C \in \mathsf{N}_C \cup \{\{a\} \mid a \in \mathsf{N}_I\} \cup \{\bot\}$, and $R, S, T \in \mathsf{N}_R$.

Proposition 9. *Any $\mathcal{EL}^{++\times}$ knowledge base can be transformed into an equisatisfiable $\mathcal{EL}^{++\times}$ knowledge base in normal form. The transformation can be done in linear time.*

Proof. The transformation is accomplished by the rules of Table 2, where each rule describes the replacement of some axiom by one or more alternative axioms. In a first step, the rules (P1) are applied exhaustively, and afterwards the rules (P2) are applied exhaustively to the knowledge base. We omit the easy proof as the result is very similar to the normal form transformation given in [2]. □

Table 3. Completion rules for reasoning in $\mathcal{EL}^{++\times}$. Symbols C, D, possibly with subscripts or primes, denote elements of $\mathcal{B}$, whereas E might be any element of $\mathcal{B} \cup \{\exists R.C \mid C \in \mathcal{B}\}$.

(R1) If $D \sqsubseteq E \in$ KB and $C \sqsubseteq D \in \mathcal{S}$ then $\mathcal{S} := \mathcal{S} \cup \{C \sqsubseteq E\}$.
(R2) If $C_1 \sqcap C_2 \sqsubseteq D \in$ KB and $\{C \sqsubseteq C_1, C \sqsubseteq C_2\} \subseteq \mathcal{S}$ then $\mathcal{S} := \mathcal{S} \cup \{C \sqsubseteq D\}$.
(R3) If $\exists R.C \sqsubseteq D \in$ KB and $\{C_1 \sqsubseteq \exists R.C_2, C_2 \sqsubseteq C\} \subseteq \mathcal{S}$ then $\mathcal{S} := \mathcal{S} \cup \{C_1 \sqsubseteq D\}$.
(R4) If $\{C \sqsubseteq \exists R.D, D \sqsubseteq \bot\} \subseteq \mathcal{S}$ then $\mathcal{S} := \mathcal{S} \cup \{C \sqsubseteq \bot\}$.
(R5) If $\{C \sqsubseteq \{a\}, D \sqsubseteq \{a\}, D \sqsubseteq E\} \subseteq \mathcal{S}$ and $C \rightsquigarrow D$ then $\mathcal{S} := \mathcal{S} \cup \{C \sqsubseteq E\}$.
(R6) If $R \sqsubseteq S \in$ KB and $C \sqsubseteq \exists R.D \in \mathcal{S}$ then $\mathcal{S} := \mathcal{S} \cup \{C \sqsubseteq \exists S.D\}$.
(R7) If $R \circ S \sqsubseteq T \in$ KB and $\{C_1 \sqsubseteq \exists R.C_2, C_2 \sqsubseteq \exists S.C_3\} \subseteq \mathcal{S}$ then $\mathcal{S} := \mathcal{S} \cup \{C_1 \sqsubseteq \exists T.C_3\}$.
(R8) If $C \times D \sqsubseteq R \in$ KB, $D' \sqsubseteq D \in \mathcal{S}$, and $C \rightsquigarrow D'$ then $\mathcal{S} := \mathcal{S} \cup \{C \sqsubseteq \exists R.D'\}$.

It is easy to see that the above transformation to normal form does not change the relative subsumption hierarchy between classes in the original knowledge base. Hence, subsumption testing can equivalently be performed on the normalised knowledge base.

We now provide an algorithm that checks whether a subsumption $A \sqsubseteq B$ between concept names is entailed by some normalised $\mathcal{EL}^{++\times}$ knowledge base KB. As discussed in [2], this is sufficient to solve arbitrary subsumption problems, and to decide knowledge base consistency and instance classification. The algorithm proceeds by computing a set $\mathcal{S}$ of inclusion axioms that are entailed by KB, and it turns out we only need to consider very simple axioms of the forms $C \sqsubseteq D$ and $C \sqsubseteq \exists R.D$, where C, D are elements of the set $\mathcal{B} := \mathsf{N}_C \cup \{\{a\} \mid a \in \mathsf{N}_I\} \cup \{\top, \bot\}$.

The set $\mathcal{S}$ is initialised by setting $\mathcal{S} := \{C \sqsubseteq C \mid C \in \mathcal{B}\} \cup \{C \sqsubseteq \top \mid C \in \mathcal{B}\}$. The algorithm then proceeds by applying the rules in Table 3 until no possible rule application further modifies the set $\mathcal{S}$. The rules refer to a binary relation $\rightsquigarrow \subseteq \mathcal{B} \times \mathcal{B}$ that is defined based on the current content of $\mathcal{S}$. Namely, $C \rightsquigarrow D$ holds whenever there are $C_1, \ldots, C_k \in \mathcal{B}$ such that

- C_1 is equal to one of the following: C, $\top$, $\{a\}$ (for some individual $a \in \mathsf{N}_I$), or A (where the subsumption $A \sqsubseteq B$ is to be checked),
- $C_i \sqsubseteq \exists R.C_{i+1} \in \mathcal{S}$ for some $R \in \mathsf{N}_R$ ($i = 1, \ldots, k-1$), and
- $C_k = D$.

Intuitively, $C \rightsquigarrow D$ states that D cannot be interpreted as the empty set if we assume that C contains some element. The option $C_1 = A$ reflects the fact that we can base our conclusions on the assumption that A is not equivalent to $\bot$ either – if it is, the queried subsumption holds immediately, so we do not need to check this case.[4]

After terminating with the saturated set $\mathcal{S}$, the algorithm confirms the subsumption $A \sqsubseteq B$ iff one of the following conditions hold:

$$A \sqsubseteq B \in \mathcal{S} \quad \text{or} \quad A \sqsubseteq \bot \in \mathcal{S} \quad \text{or} \quad \{a\} \sqsubseteq \bot \in \mathcal{S} \text{ (for some } a \in \mathsf{N}_I) \quad \text{or} \quad \top \sqsubseteq \bot \in \mathcal{S}.$$

We will show below that this algorithm is indeed correct, and that it runs in polynomial time.

[4] This case is actually missing in [2], and it indeed needs to be added to obtain a complete algorithm.

Lemma 10. *The above algorithm for checking concept subsumption in $\mathcal{EL}^{++\times}$ terminates in polynomial time.*

Proof. The set $\mathcal{B}$ clearly is linear in the size of the knowledge base, and there are only $|\mathcal{B}|\times|\mathcal{B}|\times(1+|\mathsf{N}_R|)$ many possible elements in $\mathcal{S}$. At least one such element is computed in each step, so that the algorithm terminates after polynomially many steps.

In addition, applicability of each rule can be decided in polynomial time. In particular, the relation $\rightsquigarrow$ can be computed in polynomially many steps. □

Lemma 11. *Let $\mathcal{S}$ be the saturated set obtained by the subsumption checking algorithm for a normalised $\mathcal{EL}^{++\times}$ knowledge base* KB *and some queried subsumption $A \sqsubseteq B$. If* KB $\models A \sqsubseteq B$ *then one of the following holds:*

$$A \sqsubseteq B \in \mathcal{S} \quad \textit{or} \quad A \sqsubseteq \bot \in \mathcal{S} \quad \textit{or} \quad \{a\} \sqsubseteq \bot \in \mathcal{S} \textit{ (for some } a \in \mathsf{N}_I\textit{)} \quad \textit{or} \quad \top \sqsubseteq \bot \in \mathcal{S}.$$

Proof. We show the contrapositive: if none of the given conditions hold, then there is a model $\mathcal{I}$ for KB within which the subsumption $A \sqsubseteq B$ does not hold. The proof proceeds by constructing this model.

The domain $\Delta^\mathcal{I}$ of $\mathcal{I}$ is chosen to contain only one characteristic individual for all classes of KB that are necessarily non-empty, factorised to take inferred equalities into account. To this end, we first define a set of concept expressions $\mathcal{B}^- := \{C \in \mathcal{B} \mid A \rightsquigarrow C\}$. A binary relation $\sim$ on $\mathcal{B}^-$ that will serve us to represent inferred equalities is defined as follows:

$$C \sim D \quad \text{iff} \quad C = D \;\text{ or }\; \{C \sqsubseteq \{a\}, D \sqsubseteq \{a\}\} \subseteq \mathcal{S} \text{ for some } a \in \mathsf{N}_I.$$

We will see below that $\sim$ is an equivalence relation on $\mathcal{B}^-$. Reflexivity and symmetry are obvious. For transitivity, we first show that elements related by $\sim$ are subject to the same assertions in $\mathcal{S}$. Thus consider $C, C' \in \mathcal{B}^-$ such that $C \sim C'$. We claim that, for all concept expressions E, we find that $C \sqsubseteq E \in \mathcal{S}$ implies $C' \sqsubseteq E \in \mathcal{S}$ (Claim $*$). Assume $C \neq C'$ and $\{C \sqsubseteq \{a\}, C' \sqsubseteq \{a\}\} \subseteq \mathcal{S}$ – the other case is trivial. But by our definition of $\mathcal{B}^-$, we find that $C \rightsquigarrow C'$, and hence rule (R5) is applicable and establishes the required result.

This also yields transitivity of $\sim$, since $\{C_1 \sqsubseteq \{a\}, C_2 \sqsubseteq \{a\}\} \subseteq \mathcal{S}$ and $C_2 \sim C_3$ implies $C_3 \sqsubseteq \{a\} \in \mathcal{S}$ and thus $C_1 \sim C_3$. We use $[C]$ to denote the equivalence class of $C \in \mathcal{B}^-$ w.r.t. $\sim$.

These observations allow us to make the following definition of $\mathcal{I}$:

- $\Delta^\mathcal{I} := \{[C] \mid C \in \mathcal{B}^-\}$,
- $C^\mathcal{I} := \{[D] \in \Delta^\mathcal{I} \mid D \sqsubseteq C \in \mathcal{S}\}$ for all $C \in \mathsf{N}_C$,
- $a^\mathcal{I} := [\{a\}]$ for all $a \in \mathsf{N}_I$,
- $R^\mathcal{I} := \{\langle [C], [D] \rangle \in \Delta^\mathcal{I} \times \Delta^\mathcal{I} \mid C \sqsubseteq \exists R.D \in \mathcal{S}\}$ for all $R \in \mathsf{N}_R$.

Note that N_I was assumed to be fixed and finite, and that $\{a\} \in \mathcal{B}^-$ for all $a \in \mathsf{N}_I$ such that $[\{a\}]$ is well-defined. Roles and concepts not involved in $\mathcal{B}^-$ or $\mathcal{S}$ are automatically interpreted as the empty set by the above definition. The definitions of $C^\mathcal{I}$ and $R^\mathcal{I}$ are well-defined due to $(*)$ above.

We can now observe the following desired correspondence between $\mathcal{I}$ and $\mathcal{S}$: For any $C, D \in \mathcal{B}^-$, we find that $[C] \in D^\mathcal{I}$ iff $C \sqsubseteq D \in \mathcal{S}$ (Claim †). We distinguish various cases based on the structure of D:

– $D = \bot$. We can conclude $[C] \notin \bot^{\mathcal{I}}$ and $C \sqsubseteq \bot \notin \mathcal{S}$ by noting that, for any $E \in \mathcal{B}^-$ we have that $E \sqsubseteq \bot \notin \mathcal{S}$. To see that, suppose the contrary. By $A \rightsquigarrow E$ there is a chain $C_1, \ldots, C_k \in \mathcal{B}$ as in the definition of $\rightsquigarrow$ such that $C_k = E$. Using $C_{k-1} \sqsubseteq \exists R.E \in \mathcal{S}$ and rule (R4), we conclude that $C_{k-1} \sqsubseteq \bot \in \mathcal{S}$. Applying this reasoning inductively, we obtain $C_1 \sqsubseteq \bot \in \mathcal{S}$. But as C_1 is of the form A, $\{a\}$, or $\top$, this contradicts our initial assumptions.
– $D = \top$. By the initialisation of $\mathcal{S}$, $C \sqsubseteq \top \in \mathcal{S}$ and also $[C] \in \top^{\mathcal{I}}$.
– $D \in \mathsf{N}_C$. This case follows directly from the definition of $\mathcal{I}$.
– $D = \{a\}$ for some $a \in \mathsf{N}_I$. If $[C] \in \{a\}^{\mathcal{I}}$ then $[C] = [\{a\}]$, and hence $C \sim \{a\}$. Since $\{a\} \sqsubseteq \{a\} \in \mathcal{S}$, we obtain $C \sqsubseteq \{a\} \in \mathcal{S}$ from $(*)$.
  Conversely, if $C \sqsubseteq \{a\} \in \mathcal{S}$, then $C \sim \{a\}$ and hence $\{[C]\} = \{[\{a\}]\} = \{a\}^{\mathcal{I}}$ as required.

It is easy to see that $\mathcal{I} \not\models A \sqsubseteq B$: since $A \in \mathcal{B}^-$, we find that $[A] \in A^{\mathcal{I}}$ due to $A \sqsubseteq A \in \mathcal{S}$ by the initialisation of the algorithm. But since $A \sqsubseteq B \notin \mathcal{S}$, we have that $[A] \notin B^{\mathcal{I}}$ based on (†).

Finally, it only remains to show that $\mathcal{I}$ is indeed a model of KB. We argue that each axiom of KB is satisfied by $\mathcal{I}$ by considering the possible normal forms:

– $D \sqsubseteq E$ with $E \in \mathcal{B} \cup \{\exists R.E' \mid E' \in \mathcal{B}\}$. If $[C] \in D^{\mathcal{I}}$, then $C \sqsubseteq D \in \mathcal{S}$ by (†) and thus rule (R1) can be applied to yield $C \sqsubseteq E$. If $E \in \mathcal{B}$, the claim follows from (†). For $E = \exists R.E'$, we conclude that $C \rightsquigarrow E'$ and thus $E' \in \mathcal{B}^-$. By definition of $R^{\mathcal{I}}$, we find $\langle [C], [E'] \rangle \in R^{\mathcal{I}}$, and since $E' \sqsubseteq E' \in \mathcal{S}$ we can invoke (†) to obtain $[E'] \in E'^{\mathcal{I}}$ as required.
– $C_1 \sqcap C_2 \sqsubseteq D$. This case is treated similar to the above case, using rule (R2) and treating only the (simpler) case where $D \in \mathcal{B}$.
– $\exists R.D \sqsubseteq E$. If $[C] \in \exists R.D^{\mathcal{I}}$ then $\langle [C], [D'] \rangle \in R^{\mathcal{I}}$ for some $[D'] \in D^{\mathcal{I}}$. By the definition of $R^{\mathcal{I}}$ and $(*)$, there is some $D'' \in [D']$ such that $C \sqsubseteq \exists R.D'' \in \mathcal{S}$. Since $D'' \in \mathcal{B}$ and $[D''] = [D'] \in D^{\mathcal{I}}$, we can conclude $D'' \sqsubseteq D \in \mathcal{S}$ from (†). Thus rule (R3) implies that $C \sqsubseteq E$, and we obtain $[C] \in E^{\mathcal{I}}$ by invoking (†).
– $R \sqsubseteq S$. If $\langle [C], [D] \rangle \in R^{\mathcal{I}}$ then there is $C \sqsubseteq \exists R.D' \in \mathcal{S}$ with $[D'] = [D]$. Rule (R6) thus entailed $C \sqsubseteq \exists S.D' \in \mathcal{S}$, which yields $\langle [C], [D] \rangle \in S^{\mathcal{I}}$ again by definition of $S^{\mathcal{I}}$.
– $R \circ S \sqsubseteq T$. This case is treated similar to the previous case, using rule (R7) instead of rule (R6).
– $C \times D \sqsubseteq R$. If $[C'] \in C^{\mathcal{I}}$ and $[D'] \in D^{\mathcal{I}}$, we conclude $\{C' \sqsubseteq C, D' \sqsubseteq D\} \subseteq \mathcal{S}$ from (†). Since $D' \in \mathcal{B}^-$, we have $A \rightsquigarrow D'$ which clearly implies $C \rightsquigarrow D'$ by definition of $\rightsquigarrow$. Hence rule (R8) was applied to yield $C \sqsubseteq \exists R.D' \in \mathcal{S}$ and by rule (R1) we also obtain $C' \sqsubseteq \exists R.D' \in \mathcal{S}$. Now $\langle [C'], [D'] \rangle \in R^{\mathcal{I}}$ follows directly from the definition of $R^{\mathcal{I}}$.

□

Lemma 12. *Let $\mathcal{S}$ be the saturated set obtained by the subsumption checking algorithm for a normalised $\mathcal{EL}^{++\times}$ knowledge base* KB *and some queried subsumption $A \sqsubseteq B$. Then, for each model $\mathcal{I}$ of* KB*, one of the following holds:*

– *$A^{\mathcal{I}} = \emptyset$, or*

– $\mathcal{I} \models \mathcal{S}$

Especially, if $A \sqsubseteq B \in \mathcal{S}$*,* $A \sqsubseteq \bot \in \mathcal{S}$*,* $\{a\} \sqsubseteq \bot \in \mathcal{S}$ *(for some* $a \in \mathsf{N}_I$*), or* $\top \sqsubseteq \bot \in \mathcal{S}$*, then* $\mathrm{KB} \models A \sqsubseteq B$*.*

Proof. First note that the second part of the statement indeed is a consequence of the first: all models $\mathcal{I}$ with $A^\mathcal{I} = \emptyset$ certainly satisfy $A \sqsubseteq B$, and all other models need to satisfy the detected axioms in $\mathcal{S}$, which either shows the claim (if $A \sqsubseteq B \in \mathcal{S}$ or $A \sqsubseteq \bot \in \mathcal{S}$) or demonstrates that such models cannot exist (if $\{a\} \sqsubseteq \bot \in \mathcal{S}$ or $\top \sqsubseteq \bot \in \mathcal{S}$).

To show the first part of the claimed statement, consider any model $\mathcal{I}$ of KB such that $A^\mathcal{I} \neq \emptyset$. A simple induction on the processing steps of the algorithm shows that all elements of $\mathcal{S}$ are satisfied by $\mathcal{I}$. The base case is obvious, since all formulae $C \sqsubseteq C$ and $C \sqsubseteq \top$ are satisfied by any interpretation. for the induction step, assume that $\mathcal{S}$ in the current stage of computation is such that $\mathcal{I} \models \mathcal{S}$. We show that any rule of the algorithm only adds formulae to $\mathcal{S}$ that are also satisfied by $\mathcal{I}$:

- For rules (R1)–(R4), (R6), (R7) this is very easy to see. Indeed, any interpretation that satisfies the requirements of the respective rule applications clearly must also satisfy the resulting conclusions.
- For rules (R5) and (R8), we first show that $C \rightsquigarrow D$ entails that $C^\mathcal{I} \neq \emptyset$ implies $D^\mathcal{I} \neq \emptyset$. Indeed, if $C \rightsquigarrow D$ then there is an according chain $C_1, \ldots, C_k$ with $C_k = D$ such that, for any $i = 1, \ldots, k-1$, $\mathcal{I} \models C_i \sqsubseteq \exists R.C_{i+1}$ for some $R \in \mathsf{N}_R$. Hence, if $C_1^\mathcal{I} \neq \emptyset$ then also $D^\mathcal{I} \neq \emptyset$. The claim thus follows from the definition of $\rightsquigarrow$, together with our assumption that $A^\mathcal{I} \neq \emptyset$.
  Based on that observation, it is again easy to see that (R8) does yield sound results. For (R5), note that the preconditions do indeed imply that $C \equiv \{a\}$ or $C \equiv \bot$, and that the conclusion is satisfied in both cases.

This finishes the proof. □

Combining the results of Proposition 9, Lemma 10, Lemma 11, and Lemma 12, we obtain the main result of this section, where the lower bound (hardness) follows from the known hardness of $\mathcal{EL}^{++}$ [2].

Theorem 13. *The problem of checking concept subsumptions in* $\mathcal{EL}^{++\times}$ *is* P*-complete.*

Finally, one might ask how concept products affect other reasoning tasks, such as conjunctive query answering in $\mathcal{EL}^{++\times}$. As we have extended the original $\mathcal{EL}^{++}$ algorithm in a rather natural way, one would assume that related reasoning procedures for $\mathcal{EL}^{++}$ could similarly be extended. Indeed, we expect that the automata-based algorithm for conjunctive query answering that was presented in [6] can readily be modified to cover $\mathcal{EL}^{++\times}$, so that the same complexity results for conjunctive querying could be obtained.

5 The Concept Product in $\mathcal{SHOIQ}$

Below, we investigate the use of concept products in $\mathcal{SHOIQ}$, the description logic underlying OWL DL. Since $\mathcal{SHOIQ}$ does not support generalised role inclusion axioms, concept products can not be simulated by means of other axioms. Yet, we will see below that the addition of concept products does not increase the worst-case complexity of $\mathcal{SHOIQ}$ which is still NExpTime even for binary encoding of numbers. Moreover, the proof shows that roles occurring in concept product inclusions can still be considered simple without impairing this result.

Definition 14. *A $\mathcal{SHOIQ}^\times$ knowledge base* KB *is a $\mathcal{SROIQ}^\times$ knowledge base such that*

- *all Rbox axioms of* KB *are of the form $S \sqsubseteq R$, $R \circ R \sqsubseteq R$, or $C \times D \sqsubseteq R$ for $R \in \mathsf{N}_R$ a role name, $S \in \mathbf{R}$ an atomic role, and $C, D \in \mathbf{C}$ concept expressions,*
- KB *does not contain the universal role U or expressions of the form $\exists R.\mathsf{Self}$.*

Based on a fixed knowledge base KB*, we define $\sqsubseteq^*$ as the smallest binary relation on $\mathbf{R}$ such that:*

- *$R \sqsubseteq^* R$ for every atomic role R,*
- *$R \sqsubseteq^* S$ and $\mathrm{Inv}(R) \sqsubseteq^* \mathrm{Inv}(S)$ for every Rbox axiom $R \sqsubseteq S$, and*
- *$R \sqsubseteq^* T$ whenever $R \sqsubseteq^* S$ and $S \sqsubseteq^* T$.*

Given an atomic role R, we write $\mathsf{Trans}(R) \in$ KB as an abbreviation for: $R \circ R \sqsubseteq R \in$ KB or $\mathrm{Inv}(R) \circ \mathrm{Inv}(R) \sqsubseteq \mathrm{Inv}(R) \in$ KB.

A $\mathcal{SHOIQ}^\times$ knowledge base can be further normalised. Firstly, whenever we find that $R \sqsubseteq^* S$ and $S \sqsubseteq^* R$, the roles R and S are obviously interpreted identically in any model of KB. Hence in this case, one could syntactically substitute one of them by the other, which allows us to assume that all knowledge bases considered below have an acyclic Rbox (i.e., $\sqsubseteq^*$ is a partial order). Moreover, we assume that for all concept product inclusions $A \times B \sqsubseteq R$, both A and B are atomic concepts. Obviously, this restriction does not affect expressivity, as complex concepts in such axioms can be moved into the Tbox.

Secondly, given a knowledge base KB, we obtain its negation normal form $\mathsf{NNF}(\mathrm{KB})$ by converting every Tbox concept into its negation normal form in the usual way:

$$
\begin{array}{lcl}
\mathsf{NNF}(\neg\top) & := & \bot \\
\mathsf{NNF}(\neg\bot) & := & \top \\
\mathsf{NNF}(C) & := & C \text{ if } C \in \{A, \neg A, \{a\}, \neg\{a\}, \top, \bot\} \\
\mathsf{NNF}(\neg\neg C) & := & \mathsf{NNF}(C) \\
\mathsf{NNF}(C \sqcap D) & := & \mathsf{NNF}(C) \sqcap \mathsf{NNF}(D) \\
\mathsf{NNF}(\neg(C \sqcap D)) & := & \mathsf{NNF}(\neg C) \sqcup \mathsf{NNF}(\neg D) \\
\mathsf{NNF}(C \sqcup D) & := & \mathsf{NNF}(C) \sqcup \mathsf{NNF}(D) \\
\mathsf{NNF}(\neg(C \sqcup D)) & := & \mathsf{NNF}(\neg C) \sqcap \mathsf{NNF}(\neg D) \\
\mathsf{NNF}(\forall R.C) & := & \forall R.\mathsf{NNF}(C) \\
\mathsf{NNF}(\neg\forall R.C) & := & \exists R.\mathsf{NNF}(\neg C) \\
\mathsf{NNF}(\exists R.C) & := & \exists R.\mathsf{NNF}(C) \\
\mathsf{NNF}(\neg\exists R.C) & := & \forall R.\mathsf{NNF}(\neg C) \\
\mathsf{NNF}(\leq n\, R.C) & := & \leq n\, R.\mathsf{NNF}(C) \\
\mathsf{NNF}(\neg \leq n\, R.C) & := & \geq (n+1)\, R.\mathsf{NNF}(C) \\
\mathsf{NNF}(\geq n\, R.C) & := & \geq n\, R.\mathsf{NNF}(C) \\
\mathsf{NNF}(\neg \geq n\, R.C) & := & \leq (n-1)\, R.\mathsf{NNF}(C)
\end{array}
$$

It is well-known that KB and $\mathsf{NNF}(\mathrm{KB})$ are semantically equivalent.

Slightly generalising according results from [3], we show that any $\mathcal{SHOIQ}^\times$ knowledge base can be transformed into an equisatisfiable knowledge base not containing transitivity statements.

Definition 15. *Given a $\mathcal{SHOIQ}^\times$ knowledge base* KB, *let* $\mathsf{clos}(\mathrm{KB})$ *denote the smallest set of concept expressions where*

- $\mathsf{NNF}(\neg C \sqcup D) \in \mathsf{clos}(\mathrm{KB})$ *for any Tbox axiom* $C \sqsubseteq D$,
- $D \in \mathsf{clos}(\mathrm{KB})$ *for every subexpression* D *of some concept* $C \in \mathsf{clos}(\mathrm{KB})$,
- $\mathsf{NNF}(\neg C) \in \mathsf{clos}(\mathrm{KB})$ *for any* $\leq n\, R.C \in \mathsf{clos}(\mathrm{KB})$,
- $\forall S.C \in \mathsf{clos}(\mathrm{KB})$ *whenever* $\mathsf{Trans}(S) \in \mathrm{KB}$ *and* $S \sqsubseteq^* R$ *for a role* R *with* $\forall R.C \in \mathsf{clos}(\mathrm{KB})$.

Moreover, let $\Omega(\mathrm{KB})$ *denote the knowledge base obtained from* KB *by*

- *removing all transitivity axioms* $R \circ R \sqsubseteq R$ *and*
- *adding the axiom* $\forall R.C \sqsubseteq \forall S.(\forall S.C)$ *for every* $\forall R.C \in \mathsf{clos}(\mathrm{KB})$ *with* $\mathsf{Trans}(S) \in \mathrm{KB}$ *and* $S \sqsubseteq^* R$.

Proposition 16. KB *and* $\Omega(\mathrm{KB})$ *are equisatisfiable.*

Proof. Obviously we have that $\mathrm{KB} \models \Omega(\mathrm{KB})$, hence every model of KB is a model of $\Omega(\mathrm{KB})$ as well.

For the other direction, let $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$ be a model of $\Omega(\mathrm{KB})$. Then we define a new interpretation $\mathcal{J} = (\Delta^\mathcal{J}, \cdot^\mathcal{J})$ as follows:

- $\Delta^\mathcal{J} := \Delta^\mathcal{I}$
- $a^\mathcal{J} := a^\mathcal{I}$ for every $a \in \mathsf{N}_I$
- $A^\mathcal{J} := A^\mathcal{I}$ for every $A \in \mathsf{N}_C$
- $R^\mathcal{J}$ is set to the transitive closure of $R^\mathcal{I}$ if $\mathsf{Trans}(R) \in \mathrm{KB}$, otherwise $R^\mathcal{J} := R^\mathcal{I} \cup \bigcup_{S \sqsubseteq^* R, S \neq R} S^\mathcal{J}$

We now prove that $\mathcal{J}$ is a model of KB by considering all axioms starting with the Rbox: Firstly, every transitivity axiom of KB is obviously fulfilled by definition of $\mathcal{J}$. Secondly, every role inclusion $R_1 \sqsubseteq R_2$ axiom is fulfilled: in case R_2 is not transitive, this follows directly from the definition, otherwise we can conclude it from the fact that the transitive closure is a monotone operation w.r.t. set inclusion. Thirdly, we find that every cross product axiom $A \times B \sqsubseteq R$ is satisfied within $\mathcal{J}$, as the construction ensures $A^\mathcal{J} = A^\mathcal{I}$ and $B^\mathcal{J} = B^\mathcal{I}$ as well as $R^\mathcal{I} \subseteq R^\mathcal{J}$.

We proceed by examining the concept expressions $C \in \mathsf{clos}(\mathrm{KB})$ and show via structural induction that $C^\mathcal{I} \subseteq C^\mathcal{J}$. As base case, for every concept of the form $\{a\}$, $\neg\{a\}$, A, or $\neg A$ for $a \in \mathsf{N}_I$ resp. $A \in \mathsf{N}_C$ this claim follows directly from the definition of $\mathcal{J}$. Note also that nominal concepts are correctly mapped to singleton sets. We proceed with the induction steps for all possible forms of a complex concept C (mark that all $C \in \mathsf{clos}(\mathrm{KB})$ are in negation normal form):

- Clearly, if $D_1^\mathcal{I} \subseteq D_1^\mathcal{J}$ and $D_2^\mathcal{I} \subseteq D_2^\mathcal{J}$ by induction hypothesis, we can directly conclude $(D_1 \sqcap D_2)^\mathcal{I} \subseteq (D_1 \sqcap D_2)^\mathcal{J}$ as well as $(D_1 \sqcup D_2)^\mathcal{I} \subseteq (D_1 \sqcup D_2)^\mathcal{J}$.

– Likewise, as we have $R^{\mathcal{I}} \subseteq R^{\mathcal{J}}$ for all roles R and again $D^{\mathcal{I}} \subseteq D^{\mathcal{J}}$ due to the induction hypothesis, we can conclude $(\exists R.D)^{\mathcal{I}} \subseteq (\exists R.D)^{\mathcal{J}}$ as well as $({\geq}n\, R.D)^{\mathcal{I}} \subseteq ({\geq}n\, R.D)^{\mathcal{J}}$.
– Now, consider a $C = \forall R.D$ and assume $\delta \in (\forall R.D)^{\mathcal{I}}$. If there is no δ' with $(\delta, \delta') \in R^{\mathcal{J}}$, then $\delta \in (\forall R.D)^{\mathcal{J}}$ is trivially true. Now assume there are such δ'. For each of them, we can distinguish two cases:
  - $(\delta, \delta') \in R^{\mathcal{J}}$, implying $\delta' \in D^{\mathcal{I}}$ and, via the induction hypothesis, $\delta' \in D^{\mathcal{J}}$,
  - $(\delta, \delta') \notin R^{\mathcal{J}}$. Yet, by construction of $\mathcal{J}$, this means that there is a role S with $S \sqsubseteq^* R$ and $\mathsf{Trans}(S) \in$ KB and a sequence $\delta = \delta_0, \ldots, \delta_n = \delta'$ with $(\delta_k, \delta_{k+1}) \in S^{\mathcal{I}}$ for all $0 \leq k < n$. By definition of Ω, the knowledge base Ω(KB) contains the axiom $\forall R.D \sqsubseteq \forall S.(\forall S.D)$, hence we have $\delta \in \forall S.(\forall S.D)$ wherefrom a simple inductive argument ensures $\delta_k \in D^{\mathcal{I}}$ for all δ_k including $\delta_n = \delta'$.

  So we can conclude that for all such δ' we have $\delta' \in D^{\mathcal{I}}$. Via the induction hypothesis follows $\delta \in D^{\mathcal{J}}$ and hence we can conclude $\delta \in (\forall R.D)^{\mathcal{J}}$.
– Finally, consider $C = {\leq}n\, R.D$ and assume $\delta \in ({\leq}n\, R.D)^{\mathcal{I}}$. From the fact that R must be simple follows $R^{\mathcal{J}} = R^{\mathcal{I}}$. Moreover, since both D and $\mathsf{NNF}(\neg D)$ are contained in clos(KB) the induction hypothesis gives $D^{\mathcal{J}} = D^{\mathcal{I}}$. Those two facts together directly imply $\delta \in ({\leq}n\, R.D)^{\mathcal{I}}$.

Now considering an arbitrary KB Tbox axiom $C \sqsubseteq D$, we find $(\mathsf{NNF}(\neg C) \sqcup D)^{\mathcal{I}} = \Delta^{\mathcal{I}}$ as $\mathcal{I}$ is a model of KB. Moreover – by the correspondence just shown – we have $(\mathsf{NNF}(\neg C) \sqcup D)^{\mathcal{I}} \subseteq (\mathsf{NNF}(\neg C) \sqcup D)^{\mathcal{I}}$ and hence also $(\mathsf{NNF}(\neg C) \sqcup D)^{\mathcal{J}} = \Delta^{\mathcal{J}}$ making $C \sqsubseteq D$ an axiom satisfied in $\mathcal{J}$. This finishes the proof. □

Thus, we can reduce satisfiability checking in $\mathcal{SHOIQ}^{\times}$ to satisfiability checking in $\mathcal{ALCHOIQ}^{\times}$ – the fragment of $\mathcal{SHOIQ}^{\times}$ without transitivity axioms. Following the approach taken in [7], we can decide the latter problem by a reduction to C^2, the two-variable fragment of first-order logic with counting quantifiers for which this problem has been shown to be NExpTime-complete, even for binary coding of numbers [8]. Intuitively, C^2 admits all formulae of function-free first-order logic that contain at most two variable symbols, and which may additionally use the counting quantifiers $\exists_{\leq n}$, $\exists_{\geq n}$, and $\exists_{=n}$ for any number $n > 0$. Such quantifiers impose the obvious semantic restrictions on the number of individuals satisfying the quantified formula. Moreover, binary equality $\approx$ can easily be defined from those constructs. For formal details, see [8].

We transform $\mathcal{ALCHOIQ}^{\times}$ knowledge bases into C^2 by means of the recursive functions in Table 4. The transformation is a modification of the standard DL to FOL transformation given e.g. in [3], where further explanations can be found. Omitting the standard proof that π(KB) is indeed equisatisfiable to KB (cf. [3]), we obtain the following result:

Theorem 17. *The problem of checking knowledge base satisfiability for $\mathcal{SHOIQ}^{\times}$ is* NExpTime*-complete, even for binary encoding of numbers.*

Proof. Hardness follows from the according hardness result for $\mathcal{SHOIQ}$ [7]. The NExpTime upper bound is obtained by applying the above polynomial reductions to transform a $\mathcal{SHOIQ}^{\times}$ knowledge base KB into an equisatisfiable $\mathcal{ALCHOIQ}^{\times}$ knowledge base $\pi(\Omega(\text{KB}))$, the satisfiability of which can be checked in NExpTime (even for binary encoding of numbers) according to [8]. □

Table 4. Transformation from $\mathcal{ALCHOIQ}^\times$ to C^2. X is a meta-variable for representing various term symbols in the final translation. The transformations π_y are assumed to be analogous to the given transformations for π_x.

$\pi(C \sqsubseteq D)$	$:=$	$\forall x.\pi_y(\neg C \sqcup D, x)$
$\pi(R \sqsubseteq S)$	$:=$	$\forall x.\forall y.(\neg R(x,y) \vee S(x,y))$
$\pi(C \times D \sqsubseteq R)$	$:=$	$\forall x.\forall y.(\neg C(x) \vee \neg D(y) \vee R(x,y))$
$\pi(\mathrm{KB})$	$:=$	$\bigwedge_{\varphi\in\mathrm{KB}} \pi(\varphi)$
$\pi_x(\top, X)$	$:=$	$\top$
$\pi_x(\bot, X)$	$:=$	$\bot$
$\pi_x(A, X)$	$:=$	$A(X)$ for any concept name $A \in \mathsf{N}_C$
$\pi_x(\{a\}, X)$	$:=$	$a \approx X$ for any individual name $a \in \mathsf{N}_I$
$\pi_x(\neg C, X)$	$:=$	$\neg\pi_x(C, X)$
$\pi_x(C \sqcap D, X)$	$:=$	$\pi_x(C, X) \wedge \pi_x(D, X)$
$\pi_x(C \sqcup D, X)$	$:=$	$\pi_x(C, X) \vee \pi_x(D, X)$
$\pi_x(\forall R.C, X)$	$:=$	$\forall x.(R(X, x) \rightarrow \pi_y(C, x))$
$\pi_x(\exists R.C, X)$	$:=$	$\exists x.(R(X, x) \wedge \pi_y(C, x))$
$\pi_x(\geq n\, R.C, X)$	$:=$	$\exists_{\geq n} x.(R(X, x) \wedge \pi_y(C, x))$
$\pi_x(\leq n\, R.C, X)$	$:=$	$\exists_{\leq n} x.(R(X, x) \rightarrow \pi_y(C, x))$

6 The Concept Product in $\mathcal{SHOI}$

Following the common nomenclature, let $\mathcal{SHOI}^\times$ denote the description logic obtained from $\mathcal{SHOIQ}^\times$by disallowing all kinds of number restrictions. It is well-known that deciding satisfiability of $\mathcal{SHOI}$ knowledge bases is ExpTime-complete [9], and we will show that this worst case complexity is not increased when adding concept products.

As was shown by Property 16, transitivity axioms can be eliminated from $\mathcal{SHOIQ}^\times$ knowledge bases without affecting satisfiability. Obviously, if applied to a $\mathcal{SHOI}^\times$ knowledge base, the result of this transformation would be in $\mathcal{ALCHOI}^\times$. In this section, we provide a way of further reducing an $\mathcal{ALCHOI}^\times$ knowledge base to an equisatisfiable $\mathcal{ALCHOI}$ knowledge base in polynomial time.

In addition to the standard negation normal form, we now require another normalisation step that simplifies the structure of KB by *flattening* it to a knowledge base $\mathsf{FLAT}(\mathrm{KB})$. This is achieved by transforming KB into negation normal form and exhaustively applying the following transformation rules:

- Select an outermost occurrence of $\mathsf{Q}R.D$ in KB, such that $\mathsf{Q} \in \{\exists, \forall\}$ and D is a non-atomic concept.
- Substitute this occurrence with $\mathsf{Q}R.F$ where F is a fresh concept name (i.e. one not occurring in the knowledge base).
- Add $\neg F \sqcup D$ to the knowledge base.

Obviously, this procedure terminates yielding a flat knowledge base $\mathsf{FLAT}(\mathrm{KB})$ all Tbox axioms of which are Boolean expressions over formulae of the form $\top$, $\bot$, A, $\neg A$, or $\mathsf{Q}R.A$ with A an atomic concept name. Obviously, the flattening can be carried out in polynomial time.

Proposition 18. *Any* $\mathcal{ALCHOI}^\times$ *knowledge base* KB *is equisatisfiable to* $\mathsf{FLAT}(\mathrm{KB})$.

Proof. We first prove inductively that every model of $\mathsf{FLAT}(\mathrm{KB})$ is a model of KB. Let KB′ be an intermediate knowledge base and let KB″ be the result of applying one single substitution step to KB′ as described in the above procedure. We now show that any model $\mathcal{I}$ of KB″ is a model of KB′. Let $\mathsf{Q}R.D$ be the term substituted in KB′. Note that after every substitution step, the knowledge base is still in negation normal form. Thus, we see that $\mathsf{Q}R.D$ occurs outside the scope of any negation or quantifier in an KB′-axiom E', the same is the case for $\mathsf{Q}R.F$ in the respective KB″-axiom E'' obtained after the substitution. Hence, if we show that $(\mathsf{Q}R.F)^\mathcal{I} \subseteq (\mathsf{Q}R.D)^\mathcal{I}$, we can conclude that $E''^\mathcal{I} \subseteq E'^\mathcal{I}$. From $\mathcal{I}$ being a model of KB″ and therefore $E''^\mathcal{I} = \Delta^\mathcal{I}$, we would then easily derive that $E'^\mathcal{I} = \Delta^\mathcal{I}$ and hence find that $\mathcal{I} \models \mathrm{KB}'$, as all other axioms from KB′ are trivially satisfied due to their presence in KB″.

It remains to show $(\mathsf{Q}R.F)^\mathcal{I} \subseteq (\mathsf{Q}R.D)^\mathcal{I}$. We distinguish two cases:

– $\mathsf{Q} = \exists$
  Consider a $\delta \in (\exists R.F)^\mathcal{I}$. Then exists an individual $\delta' \in \Delta^\mathcal{I}$ with $\langle \delta, \delta' \rangle \in R^\mathcal{I}$ and $\delta' \in F^\mathcal{I}$. As a consequence of the KB″ axiom $\neg F \sqcup D$ (being equivalent to the GCI $F \sqsubseteq D$), we find that $\delta' \in D^\mathcal{I}$ as well, leading straightforwardly to the conclusion $\delta \in (\exists R.D)^\mathcal{I}$. Hence we have $(\exists R.F)^\mathcal{I} \subseteq (\exists R.D)^\mathcal{I}$.
– $\mathsf{Q} = \forall$
  Consider a $\delta \in (\forall R.F)^\mathcal{I}$. This implies for every individual $\delta' \in \Delta^\mathcal{I}$ with $\langle \delta, \delta' \rangle \in R^\mathcal{I}$ that $\delta' \in F^\mathcal{I}$. Again, the KB″ axiom $\neg F \sqcup D$ entails $\delta' \in D^\mathcal{I}$ for every such δ', leading to $\delta \in (\forall R.D)^\mathcal{I}$. Hence, we have $(\forall R.F)^\mathcal{I} \subseteq (\forall R.D)^\mathcal{I}$.

Every model $\mathcal{I}$ of KB can be transformed into a model $\mathcal{J}$ of $\mathsf{FLAT}(\mathrm{KB})$ by following the flattening process described above: Let KB″ result from KB′ by substituting $\mathsf{Q}R.D$ by $\mathsf{Q}R.F$ and adding the respective axiom. Furthermore, let $\mathcal{I}'$ be a model of KB′. Now we construct the interpretation $\mathcal{I}''$ as follows: $F^{\mathcal{I}''} := (\mathsf{Q}R.D)^{\mathcal{I}'}$ and for all other concept and role names N we set $N^{\mathcal{I}''} := N^{\mathcal{I}'}$. Then $\mathcal{I}''$ is a model of KB″. □

Lemma 19. *Consider a flattened $\mathcal{ALCHOI}^\times$ knowledge base* KB*. Let $C \times D \sqsubseteq R$ with $C, D \in \mathsf{N}_C$ be some concept product axiom contained in* KB *and let* KB′ *be the knowledge base obtained from* KB *as follows:*

– *delete the Rbox axiom $C \times D \sqsubseteq R$,*
– *add $C \sqsubseteq \exists S_1.\{o\}$ and $D \sqsubseteq \exists S_2.\{o\}$ where S_1, S_2 are fresh roles and o is a fresh individual name*
– *for every role T with $R \sqsubseteq^* T$ (including R itself)*
  • *substitute any occurrence of $\exists T.A$ by $\exists T.A \sqcup \exists S_1.\exists S_2^-.A$*
  • *substitute any occurrence of $\forall T.A$ by $\forall T.A \sqcap \forall S_1.\forall S_2^-.A$*

Then, KB *and* KB′ *are equisatisfiable.*

Proof. To show equisatisfiability, we prove, that any model $\mathcal{I}$ of KB can be converted into a model $\mathcal{J}$ of KB′ and vice versa.

First, let $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$ be a model of KB. We define the interpretation $\mathcal{J} = (\Delta^\mathcal{J}, \cdot^\mathcal{J})$ by choosing an arbitrary but fixed $\delta_o \in \Delta^\mathcal{I}$ and setting

– $\Delta^\mathcal{J} := \Delta^\mathcal{I}$,

- for all $i \in \mathsf{N}_I \setminus \{o\}$, let $i^{\mathcal{J}} := i^{\mathcal{I}}$
- let $o^{\mathcal{J}} = \delta_o$
- for all $B \in \mathsf{N}_C$, let $B^{\mathcal{J}} := B^{\mathcal{I}}$
- for all $T \in \mathsf{N}_R \setminus \{S_1, S_2\}$, let $T^{\mathcal{J}} := T^{\mathcal{I}}$
- let $S_1^{\mathcal{J}} = C^{\mathcal{I}} \times \{\delta_o\}$
- let $S_2^{\mathcal{J}} = D^{\mathcal{I}} \times \{\delta_o\}$

We now show that $\mathcal{J}$ is indeed a model of KB′. Clearly, the new axioms $C \sqsubseteq \exists S_1.\{o\}$ and $D \sqsubseteq \exists S_2.\{o\}$ are obviously fulfilled. Now note that the construction of $\mathcal{J}$ and the KB-Axiom $C \times D \sqsubseteq R$ assure $S_1^{\mathcal{J}} \circ S_2^{-\mathcal{J}} \subseteq R^{\mathcal{J}}$. Therefore, we have $(\exists S_1.\exists S_2^-.A)^{\mathcal{J}} \subseteq (\exists R.A)^{\mathcal{J}}$ implying $(\exists R.A \sqcup \exists S_1.\exists S_2^-.A)^{\mathcal{J}} = (\exists R.A)^{\mathcal{J}} \cup (\exists S_1.\exists S_2^-.A)^{\mathcal{J}} = (\exists R.A)^{\mathcal{J}}$, hence the KB′-axiom containing $\exists R.A \sqcup \exists S_1.\exists S_2^-.A$ is satisfied in $\mathcal{J}$ due to the $\mathcal{I}$-validity of the corresponding KB-axiom containing just $\exists R.A$ instead.

Furthermore, we have $(\forall R.A)^{\mathcal{J}} \subseteq (\forall S_1.\forall S_2^-.A)^{\mathcal{J}}$ implying $(\forall R.A \sqcap \forall S_1.\forall S_2^-.A)^{\mathcal{J}} = (\forall R.A)^{\mathcal{J}} \cap (\forall S_1.\forall S_2^-.A)^{\mathcal{J}} = (\forall R.A)^{\mathcal{J}}$, so the same argumentation applies.

Second, assume $\mathcal{J} = (\Delta^{\mathcal{J}}, \cdot^{\mathcal{J}})$ is a model of KB′. Then let the interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ be defined by:

- $\Delta^{\mathcal{I}} := \Delta^{\mathcal{J}}$,
- for all $i \in \mathsf{N}_I$, let $i^{\mathcal{I}} := i^{\mathcal{J}}$
- for all $B \in \mathsf{N}_C$, let $B^{\mathcal{I}} := B^{\mathcal{J}}$
- for every $T \in \mathsf{N}_R$, if $R \sqsubseteq^* T$, let $T^{\mathcal{I}} := T^{\mathcal{J}} \cup (S_1^{\mathcal{J}} \circ S_2^{-\mathcal{J}})$, otherwise $T^{\mathcal{I}} := T^{\mathcal{J}}$

It remains to show that $\mathcal{I}$ is a model of KB. Obviously, all role inclusion statements from KB (coinciding with those from KB′) are valid in $\mathcal{I}$ ensured by the construction. Moreover, note that the axiom $C \sqsubseteq \exists S_1.\{o\}$ enforces $C^{\mathcal{J}} \times \{o^{\mathcal{J}}\} \subseteq S_1^{\mathcal{J}}$, likewise $C \sqsubseteq \exists S_1.\{o\}$ ensures $\{o^{\mathcal{J}}\} \times D^{\mathcal{J}} \subseteq S_2^{-\mathcal{J}}$. This allows to conclude $C^{\mathcal{I}} \times D^{\mathcal{I}} = C^{\mathcal{J}} \times D^{\mathcal{J}} \subseteq S_1^{\mathcal{J}} \circ S_2^{-\mathcal{J}}$. Now we see that by construction, the KB-axiom $C \times D \sqsubseteq R$ is satisfied in $\mathcal{I}$.

As to the Tbox axioms, we inductively show for every concept expression E occurring in KB and the respective (possibly substituted) KB′-concept expression E', that we have $E'^{\mathcal{J}} = E^{\mathcal{I}}$. Clearly, this entails for every KB′-Tbox axiom F' that the respective F is satisfied in $\mathcal{I}$ due to $\Delta^{\mathcal{I}} = \Delta^{\mathcal{J}} = F'^{\mathcal{J}} = F^{\mathcal{I}}$. For the base case, note that due to the construction of $\mathcal{I}$, all atomic concepts and nominals as well as their negations have the same extensions in $\mathcal{J}$ and $\mathcal{I}$. For the induction step, we find for all roles T with $R \not\sqsubseteq^* T$ that $(\exists T.A)^{\mathcal{J}} = (\exists T.A)^{\mathcal{I}}$ and $(\forall T.A)^{\mathcal{J}} = (\forall T.A)^{\mathcal{I}}$ due to $A^{\mathcal{J}} = A^{\mathcal{I}}$ as well as $T^{\mathcal{J}} = T^{\mathcal{I}}$. In the case $R \sqsubseteq^* T$, we would have $T^{\mathcal{I}} = T^{\mathcal{J}} \cup (S_1^{\mathcal{J}} \circ S_2^{-\mathcal{J}})$ which then yields us $(\exists T.A \sqcup \exists S_1.\exists S_2^-.A)^{\mathcal{J}} = (\exists T.A)^{\mathcal{I}}$ as well as $(\forall T.A \sqcap \forall S_1.\forall S_2^-.A)^{\mathcal{J}} = (\forall T.A)^{\mathcal{I}}$. Finally, invoking the induction hypothesis for concepts C_1, C_2, the claim trivially carries over to $C_1 \sqcap C_2$ and $C_1 \sqcup C_2$. □

Like for $\mathcal{SROIQ}^{\times}$, the elimination step from the above lemma can be applied iteratively to eliminate all concept products. Note that having a flat knowledge base is essential to ensure that the reduction can be done in polynomial time and space. So, a simple induction yields the following result:

Proposition 20. *Every $\mathcal{ALCHOI}^{\times}$ knowledge base can be reduced to an equisatisfiable $\mathcal{ALCHOI}$ knowledge base in polynomial time.*

$\mathcal{SHOI}$ has shown to be ExpTime-complete in [9]. Hence, we can eventually use the established results to prove ExpTime-completeness also for knowledge base satisfiability checking within $\mathcal{SHOI}^{\times}$.

Theorem 21. *The the problem of checking satisfiability for $\mathcal{SHOI}^{\times}$ knowledge bases is* ExpTime-*complete.*

Proof. As shown in Property 16, transitivity axioms can be eliminated from a $\mathcal{SHOIQ}^{\times}$ knowledge base while preserving satisfiability. Applied to a $\mathcal{SHOI}^{\times}$ knowledge base KB, this polynomial reduction yields an $\mathcal{ALCHOI}^{\times}$ knowledge base Ω(KB). Moreover, due to Proposition 18, Ω(KB) and its flattened version FLAT(Ω(KB)), which can again be computed in polynomial time, are equisatisfiable as well. Finally, Lemma 19 ensures that another polynomial time conversion transfers this knowledge base into an equisatisfiable $\mathcal{ALCHOI}$knowledge base KB′. Hence, satisfiability of KB can be checked by computing KB′ in polynomial time and then checking satisfiability of KB′. As $\mathcal{ALCHOI}$ is contained in $\mathcal{SHOI}$, which in turn is contained in $\mathcal{SHOI}^{\times}$, the ExpTime-completeness of $\mathcal{SHOI}$ shown in [9] carries over to $\mathcal{SHOI}^{\times}$. □

7 Conclusion

We have introduced the *concept product* as a new expressive feature for description logics. It allows statements of the form $C \times D \sqsubseteq R$, expressing that all instances of the class C are related to all instances of D by the role R. While this construct can be simulated in $\mathcal{SROIQ}$ with a combination of inverse roles, nominals, and role inclusion axioms, we have shown that it can also be added to many weaker DLs that do not support such simulation. In particular, we have investigated the extended DLs $\mathcal{EL}^{++\times}$, $\mathcal{SHOIQ}^{\times}$, and $\mathcal{SHOI}^{\times}$, showing that each of these preserves its known upper complexity bound P, NExpTime, and ExpTime. For the tractable logic $\mathcal{EL}^{++\times}$, we also provided a detailed algorithm that might serve as a basis for extending existing $\mathcal{EL}^{++}$ implementations with that new feature.

Our results indicate that concept products, even though they are hitherto only available in $\mathcal{SROIQ}$, do in fact not have a strong negative impact on the difficulty of reasoning in simpler DLs. In contrast, the features used to simulate concept products in $\mathcal{SROIQ}$ may have much more negative impact in general. Inverse roles, for example, are known to render $\mathcal{EL}^{++}$ ExpTime-complete [5]. Since concept products provide a valuable modelling tool that can be applied in many scenarios, they appear as a natural candidate for future extensions of the DL-based Web Ontology Language OWL, possibly even in the ongoing OWL2 effort.

Our results also entail a number of research questions for future works. First of all, one might ask what other features available (indirectly) in $\mathcal{SROIQ}$ could be easily ported back to less complex DLs. We are currently investigating a broad generalisation of concept products that appears to be rather promising in this respect.

But also the study of concept products as such bears various open problems. As remarked in Section 3, the simulation of concept products in $\mathcal{SROIQ}$ causes roles to be classified as non-simple. Yet, their use in number restrictions merely provides an alternative way of describing nominals, so that it might be conjectured that this restriction could be relaxed. Other obvious next steps are the investigation of concept products for $\mathcal{SHIQ}$ and $\mathcal{SHOQ}$, the direct treatment of concept products in further reasoning algorithms, and the possible augmentation of other popular tractable DLs with this feature. Moreover, implementations and concrete syntactical encodings for OWL would be important to make concept products usable in practice.

References

1. Horrocks, I., Kutz, O., Sattler, U.: The even more irresistible $\mathcal{SROIQ}$. In: Proc. of the 10th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR2006), AAAI Press (2006) 57–67
2. Baader, F., Brandt, S., Lutz, C.: Pushing the EL envelope. In: Proc. 19th Int. Joint Conf. on Artificial Intelligence (IJCAI-05), Edinburgh, UK, Morgan-Kaufmann Publishers (2005)
3. Motik, B.: Reasoning in Description Logics using Resolution and Deductive Databases. PhD thesis, Universität Karlsruhe (TH), Germany (2006)
4. Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P., eds.: The Description Logic Handbook: Theory, Implementation and Applications. Cambridge University Press (2007)
5. Krisnadhi, A., Lutz, C.: Data complexity in the $\mathcal{EL}$ family of description logics. In Dershowitz, N., Voronkov, A., eds.: Proceedings of the 14th International Conference on Logic for Programming, Artificial Intelligence, and Reasoning (LPAR2007). Volume 4790 of Lecture Notes in Artificial Intelligence., Springer-Verlag (2007) 333–347
6. Krötzsch, M., Rudolph, S., Hitzler, P.: Conjunctive queries for a tractable fragment of owl 1.1. In Aberer, K., Choi, K.S., Noy, N., Allemang, D., Lee, K.I., Nixon, L., Golbeck, J., Mika, P., Maynard, D., Mizoguchi, R., Schreiber, G., Cudré-Mauroux, P., eds.: Proc. 6th Int. Semantic Web Conference (ISWC 2007). Volume 4825 of LNCS., Springer (2007) 310–323
7. Tobies, S.: Complexity Results and Practical Algorithms for Logics in Knowledge Representation. PhD thesis, RWTH Aachen, Germany (2001)
8. Pratt-Hartmann, I.: Complexity of the two-variable fragment with counting quantifiers. Journal of Logic, Language and Information **14** (2005) 369–395
9. Hladik, J.: A tableau system for the description logic SHIO. In Sattler, U., ed.: IJCAR Doctoral Programme. Volume 106 of CEUR Workshop Proceedings., CEUR-WS.org (2004)