24th European Conference on Artificial Intelligence - ECAI 2020
Santiago de Compostela, Spain

Satisfiability and Query Answering in Description Logics with Global and Local Cardinality Constraints

Franz Baader[1] and **Bartosz Bednarczyk**[1,2] and **Sebastian Rudolph**[1]

Abstract. We introduce and investigate the expressive description logic (DL) $\mathcal{ALCSCC}^{++}$, in which the global and local cardinality constraints introduced in previous papers can be mixed. We prove that the added expressivity does not increase the complexity of satisfiability checking and other standard inference problems. However, reasoning in $\mathcal{ALCSCC}^{++}$ becomes undecidable if inverse roles are added or conjunctive query entailment is considered. We prove that decidability of querying can be regained if global and local constraints are not mixed and the global constraints are appropriately restricted. In this setting, query entailment can be shown to be EXPTIME-complete and hence not harder than reasoning in $\mathcal{ALC}$.

1 Introduction

Description Logics (DLs) [7] are a well-investigated family of logic-based knowledge representation languages, enjoying widespread adoption for formalizing ontologies in various application domains such as biology and medicine [11]. To define the important notions of such an application domain, DLs allow for stating necessary and sufficient conditions for individuals to belong to a formal concept. These conditions can be (Boolean combinations of) atomic properties of the individual in question (expressed by concept names) or properties that depend on the individual's relationships with other individuals and their properties (expressed as role restrictions). Using an example from [8], the concept of a motor vehicle can be formalized by the concept description *Vehicle* $\sqcap$ $\exists$*part.Motor*, which uses the concept names *Vehicle* and *Motor* and the role name *part* as well as the concept constructors conjunction ($\sqcap$) and existential restriction ($\exists r.C$). The concept inclusion (CI) *Motor-vehicle* $\sqsubseteq$ *Vehicle* $\sqcap$ $\exists$*part.Motor* then states that every motor vehicle needs to belong to this concept description. Numerical constraints on the number of role successors (so-called number restrictions) have been used early on in DLs [10, 13, 12]. For example, using number restrictions, motorcycles can be constrained to being motor vehicles with exactly two wheels:

$$\textit{Motorcycle} \sqsubseteq \textit{Motor-vehicle} \sqcap (\leqslant 2\, \textit{part.Wheel}) \sqcap (\geqslant 2\, \textit{part.Wheel}).$$

The exact complexity of reasoning in $\mathcal{ALCQ}$, the DL that has all Boolean operations and number restrictions of the form $(\leqslant n\, r.C)$ and $(\geqslant n\, r.C)$ as concept constructors, was determined by Stephan Tobies [21, 23]: it is PSPACE-complete without CIs and EXPTIME-complete in the presence of CIs, independently of whether the numbers occurring in the number restrictions are encoded in unary or binary. Note that, using unary coding of numbers, the number n is assumed to contribute n to the size of the input, whereas with binary coding the size of the number n is $\log n$. Thus, for large numbers, using binary coding is more realistic.

Whereas number restrictions are local in the sense that they consider role successors of an individual under consideration (e.g. the wheels that are part of a particular motor vehicle), cardinality restrictions on concepts (CRs) [6, 22] are global, i.e., they consider all individuals in an interpretation. For example, the cardinality restriction $(\leqslant 45\,000\,000\; (\textit{Car} \sqcap \exists \textit{registered-in.German-district}))$ states that at most 45 million cars are registered all over Germany. Such cardinality restrictions can be seen as quantitative generalizations of CIs since a CI of the form $C \sqsubseteq D$ can be expressed by the CR $(\leqslant 0\, (C \sqcap \neg D))$. The availability of CRs increases the complexity of reasoning: as mentioned above, consistency in $\mathcal{ALCQ}$ w.r.t. CIs is EXPTIME-complete, but consistency w.r.t. CRs is NEXPTIME-complete if the numbers occurring in the CRs are assumed to be encoded in binary [22]. With unary coding of numbers, consistency stays EXPTIME-complete even w.r.t. CRs [22]. However, as the above example considering 45 million cars indicates, unary coding does not yield a realistic measure for the input size if numbers with large values are employed.

In two previous publications we have, on the one hand, extended the DL $\mathcal{ALCQ}$ by more expressive number restrictions using cardinality and set constraints expressed in the quantifier-free fragment of Boolean Algebra with Presburger Arithmetic (QFBAPA) [14]. In the resulting DL $\mathcal{ALCSCC}$, which was introduced and investigated in [1], cardinality and set constraints are applied locally, i.e., they refer to the role successors of an individual under consideration. For example, we can state that the number of cylinders of a motor must coincide with the number of spark plugs in this motor, without fixing what this number actually is, using the following $\mathcal{ALCSCC}$ CI:

$$\textit{Motor} \sqsubseteq \textit{succ}(|\textit{part} \cap \textit{Cylinder}| = |\textit{part} \cap \textit{SparkPlug}|).$$

It was shown in [1] that pure concept satisfiability in $\mathcal{ALCSCC}$ is a PSPACE-complete problem, and concept satisfiability w.r.t. a TBox (i.e. a finite set of CIs) is EXPTIME-complete. This shows that the more expressive number restrictions do not increase the complexity of reasoning since reasoning in $\mathcal{ALCQ}$ has the same complexity.

On the other hand, we have extended the terminological formalism of the well-known description logic $\mathcal{ALC}$[2] from CIs not only to CRs, but to more general cardinality constraints expressed in QFBAPA [8], which we called extended cardinality constraints (ECBoxes). These constraints are global since they refer to all individuals in the interpretation domain. An example of such a constraint, which is not expressible using CRs, states that, in Germany, cars running on petrol outnumber cars running on diesel by a factor of at least two:

$$\begin{aligned} &2 \cdot |\textit{Car} \sqcap \exists \textit{registered-in.German-district} \sqcap \exists \textit{fuel.Diesel}| \\ &\leq |\textit{Car} \sqcap \exists \textit{registered-in.German-district} \sqcap \exists \textit{fuel.Petrol}|. \end{aligned}$$

[1] TU Dresden, Germany [2] University of Wrocław, Poland

[2] The DL $\mathcal{ALC}$ is the fragment of $\mathcal{ALCQ}$ in which only number restrictions $(\leqslant 0\, r.\neg C)$ (written $\forall r.C$) and $(\geqslant 1\, r.C)$ (written $\exists r.C$) are available.

It was shown in [8] that reasoning w.r.t. ECBoxes is still in NEXPTIME even if the numbers occurring in the constraints are encoded in binary. The NEXPTIME lower bound follows from the result of Tobies [22] for CRs mentioned above. This complexity can be lowered to EXPTIME if a restricted form of cardinality constraints (RCBoxes) is used. Such RCBoxes are still powerful enough to express statistical knowledge bases [17].

An obvious way to generalize these two approaches is to combine the two extensions, i.e., to consider extended cardinality constraints, but now on $\mathcal{ALCSCC}$ concepts rather than just $\mathcal{ALC}$ concepts. This combination was investigated in [2, 3], where a NEXPTIME upper bound was established for reasoning in $\mathcal{ALCSCC}$ w.r.t. ECBoxes. It is also shown in [2, 3] that reasoning w.r.t. RCBoxes stays in EXPTIME also for $\mathcal{ALCSCC}$. Here we go one step further by allowing for a tighter integration of global and local constraints. The resulting logic, which we call $\mathcal{ALCSCC}^{++}$, allows, for example, to relate the number of role successors of a given individual with the overall number of elements of a certain concept. For example, the $\mathcal{ALCSCC}^{++}$ concept description[3]

$$sat(|likes \cap Car| = |Car|)$$

describes car lovers, i.e., individuals that like *all* cars, independently of whether these cars are related to them by some role or not. More generally, DLs that can express both local cardinality constraints (i.e., constraints concerning the role successors of specific individuals) and global cardinality constraints (i.e., constraints on the overall cardinality of concepts) can, for instance, be used to check the correctness of statistical statements. For example, if a German car company claims that they have produced more than N cars in a certain year, and $P\%$ of the tires used for their cars were produced by Betteryear, this may be contradictory to a statement of Betteryear that they have sold less than M tires in Germany. In addition to the global statistical constraints, such an inconsistency may also depend on local constraints such as the fact that each car has at least four tires. Such numerical information may, of course, also influence the answers to queries. If we know that the car company VMW uses only tires from Betteryear or Badmonth, but the statistical information allows us to conclude that another car company has actually bought all the tires sold by Betteryear, then we know that the cars sold by VMW all have tires produced by Badmonth. This motivates investigating DLs with expressive cardinality constraints, and to consider not just standard inferences such as satisfiability checking for these DLs, but also query answering.

In the present paper, we show that, from a worst-case complexity point of view, the extended expressivity of $\mathcal{ALCSCC}^{++}$ comes for free if we consider classical reasoning problems. Concept satisfiability in $\mathcal{ALCSCC}^{++}$ has the same complexity as in $\mathcal{ALC}$ and $\mathcal{ALCSCC}$ with global cardinality constraints: it is NEXPTIME-complete. However, if we add inverse roles, then concept satisfiability becomes undecidable. In addition, for effective conjunctive query answering this logic turns out to be too expressive. We show that conjunctive query entailment w.r.t. $\mathcal{ALCSCC}^{++}$ knowledge bases is, in fact, undecidable. In contrast, we can show that conjunctive query entailment w.r.t. (an extension of) $\mathcal{ALCSCC}$ RCBoxes is decidable and, in fact, only EXPTIME-complete.

We assume the reader to be sufficiently familiar with all the standard notions of description logics [7, 9, 20]. More details and full proofs are available in the extended version of the paper [5].

[3] To distinguish between constraints in $\mathcal{ALCSCC}$ and in $\mathcal{ALCSCC}^{++}$, which have a different semantics, we use different keywords for them.

2 The logic $\mathcal{ALCSCC}^{++}$

As in [1, 8], we use the quantifier-free fragment of Boolean Algebra with Presburger Arithmetic (QFBAPA) [14] to express our constraints. In this logic, one can build *set terms* by applying Boolean operations (intersection $\cap$, union $\cup$, and complement $\cdot^c$) to set variables as well as the constants $\emptyset$ and $\mathcal{U}$. Set terms s, t can then be used to state *set constraints*, which are equality and inclusion constraints of the form $s = t, s \subseteq t$, where s, t are set terms. *Presburger Arithmetic (PA) expressions* are built from integer constants and set cardinalities $|s|$ using addition as well as multiplication with an integer constant. They can be used to form *cardinality constraints* of the form $k = \ell, k < \ell, N \operatorname{dvd} \ell$, where k, ℓ are PA expressions, N is an integer constant, and dvd stands for divisibility. A *QFBAPA formula* is a Boolean combination of set and cardinality constraints. A *solution* σ of a QFBAPA formula ϕ is a substitution that assigns a finite set $\sigma(\mathcal{U})$ to $\mathcal{U}$ and subsets of $\sigma(\mathcal{U})$ to set variables such that ϕ is satisfied by this assignment (see [1, 5, 14] for more details). A QFBAPA formula ϕ is *satisfiable* if it has a solution. In [14] it is shown that the satisfiability problem for QFBAPA formulae is NP-complete.

We are now ready to introduce our new logic, which we call $\mathcal{ALCSCC}^{++}$ to indicate that it is an extension of the logic $\mathcal{ALCSCC}$ introduced in [1] (see [5] for a more detailed definition). When defining the semantics of $\mathcal{ALCSCC}^{++}$, we restrict the attention to *finite* interpretations to ensure that cardinalities of concept descriptions are always well-defined non-negative integers.

Definition 1 *Given disjoint* finite *sets N_C and N_R of* concept names *and* role names*, respectively,* $\mathcal{ALCSCC}^{++}$ concept descriptions *(short:* concepts*) are Boolean combinations of concept names and* constraint expressions*, where a constraint expression is of the form $sat(Con)$ for a set constraint or a cardinality constraint Con that uses role names and $\mathcal{ALCSCC}^{++}$ concept descriptions in place of set variables. As usual, we use $\top$ (*top*) and $\bot$ (*bottom*) as abbreviations for $A \sqcup \neg A$ and $A \sqcap \neg A$, respectively.*

A finite interpretation *of N_C and N_R consists of a finite, non-empty set $\Delta^\mathcal{I}$ and a mapping $\cdot^\mathcal{I}$ that maps every concept name $A \in N_C$ to a subset $A^\mathcal{I}$ of $\Delta^\mathcal{I}$ and every role name $r \in N_R$ to a binary relation $r^\mathcal{I}$ over $\Delta^\mathcal{I}$. For a given element $d \in \Delta^\mathcal{I}$ we define $r^\mathcal{I}(d) := \{e \in \Delta^\mathcal{I} \mid (d,e) \in r^\mathcal{I}\}$. The interpretation function $\cdot^\mathcal{I}$ is inductively extended to $\mathcal{ALCSCC}^{++}$ concept descriptions by interpreting the Boolean operators as usual, and the constraint expressions as follows: $sat(Con)^\mathcal{I} := \{d \in \Delta^\mathcal{I} \mid$ the substitution $\sigma_d^\mathcal{I}$ satisfies $Con\}$, where $\sigma_d^\mathcal{I}$ maps*
- *$\emptyset$ to the empty set and $\mathcal{U}$ to $\Delta^\mathcal{I}$,*
- *the $\mathcal{ALCSCC}^{++}$ concepts C occurring in Con to $C^\mathcal{I}$,*
- *and the role names r occurring in Con to $r^\mathcal{I}(d)$.*

The $\mathcal{ALCSCC}^{++}$ concept description C is satisfiable *if there is a finite interpretation $\mathcal{I}$ such that $C^\mathcal{I} \neq \emptyset$.*

Note that the interpretation of concepts as set variables in $\mathcal{ALCSCC}^{++}$ is global in the sense that it does not depend on d, i.e., $\sigma_d^\mathcal{I}(C) = C^\mathcal{I} = \sigma_e^\mathcal{I}(C)$ for all $d, e \in \Delta^\mathcal{I}$. In contrast, the interpretation of role names r as set variables is local since only the r-successors of d are considered by $\sigma_d^\mathcal{I}(r)$. In $\mathcal{ALCSCC}$, also the interpretation of concepts as set variables is local since in the semantics of $\mathcal{ALCSCC}$ the substitution $\sigma_d^\mathcal{I}$ considers only the elements of $C^\mathcal{I}$ that are role successors of d for some role name in N_R (see [1] and [5] for details). Thus, the local successor constraints $succ(c)$ of $\mathcal{ALCSCC}$ can be simulated in $\mathcal{ALCSCC}^{++}$ by using $C \cap (\bigcup_{r \in N_R} r)$ instead of just the concept C when formulating the constraints. This shows that $\mathcal{ALCSCC}$ concepts can be expressed by $\mathcal{ALCSCC}^{++}$ concepts. In addition, extended cardinality constraints (ECBoxes), as introduced in [8], are expressible within $\mathcal{ALCSCC}^{++}$ concept descriptions, as are nominals, the universal role, and role negation (see [5] for detailed definitions and a proof of the following proposition).

Proposition 2 *$\mathcal{ALCSCC}^{++}$ concepts can polynomially express nominals, role conjunctions, and $\mathcal{ALCSCC}$ ECBoxes, and thus also ABoxes, $\mathcal{ALC}$ ECBoxes and $\mathcal{ALCSCC}$ TBoxes. In addition, they have the same expressivity as concepts of $\mathcal{ALCSCC}$ extended with the universal role or with role negation, whereas both of these features are not expressible in plain $\mathcal{ALCSCC}$.*

Though TBoxes, ABoxes, and ECBoxes can be expressed by $\mathcal{ALCSCC}^{++}$ concepts, and TBoxes and ABoxes in $\mathcal{ALCSCC}$ can be expressed by $\mathcal{ALCSCC}$ ECBoxes, we will nevertheless sometimes consider *knowledge bases* $\mathcal{K} = (\mathcal{A}, \mathcal{T}, \mathcal{E})$ consisting of an ABox $\mathcal{A}$, a TBox $\mathcal{T}$, and a (possibly restricted) ECBox $\mathcal{E}$.

3 Satisfiability of $\mathcal{ALCSCC}^{++}$ concept descriptions

In the following we consider an $\mathcal{ALCSCC}^{++}$ concept description E and show how to test E for satisfiability by reducing this problem to the problem of testing satisfiability of QFBAPA formulae. Since the reduction is exponential and satisfiability in QFBAPA is in NP, this yields a NExpTime upper bound for satisfiability of $\mathcal{ALCSCC}^{++}$ concept descriptions. This bound is optimal since consistency of extended cardinality constraints in $\mathcal{ALC}$, as introduced in [8], is already NExpTime hard, and can be expressed as an $\mathcal{ALCSCC}^{++}$ satisfiability problem by Proposition 2.

Our NExpTime algorithm combines ideas from the satisfiability algorithm for $\mathcal{ALCSCC}$ concept descriptions [1] and the consistency procedure for $\mathcal{ALC}$ ECBoxes [8]. In particular, we use the notion of a type, as introduced in [8]. This notion is also similar to the Venn regions employed in [1]. We assume in the following that E is an arbitrary, but fixed $\mathcal{ALCSCC}^{++}$ concept and $\mathcal{M}_E$ consists of *all subdescriptions* of the concept description E as well as the *negations of these subdescriptions*.

Definition 3 *A set $t \subseteq \mathcal{M}_E$ is a* type *for E if it satisfies:*
(i) if $\neg C \in \mathcal{M}_E$, then either C or $\neg C$ belongs to t;
(ii) if $C \sqcap D \in \mathcal{M}_E$, then $C \sqcap D \in t$ iff $C \in t$ and $D \in t$;
(iii) if $C \sqcup D \in \mathcal{M}_E$, then $C \sqcup D \in t$ iff $C \in t$ or $D \in t$.
We denote the set of all types for E with types(E). Given an interpretation $\mathcal{I}$ and a domain element $d \in \Delta^{\mathcal{I}}$, the type of d w.r.t. E *is the set $t^E_{\mathcal{I}}(d) := \{C \in \mathcal{M}_E \mid d \in C^{\mathcal{I}}\}$.*

It is easy to show that the type of an individual really satisfies the conditions stated in the definition of a type, i.e., $t^E_{\mathcal{I}}(d) \in \text{types}(E)$.

Any type $t \in \text{types}(E)$ yields a concept description C_t, which is the conjunction of all the elements of t. Due to Condition (i) in the definition of types, concept descriptions $C_t, C_{t'}$ induced by different types $t \neq t'$ are disjoint, and all concept descriptions in $\mathcal{M}_E$ can be obtained as the union of the concept descriptions induced by the types containing them, i.e., we have

$$C^{\mathcal{I}} = \bigcup_{t \in \text{types}(E) \text{ s.t. } C \in t} C^{\mathcal{I}}_t$$

for all $C \in \mathcal{M}_E$ and finite interpretations $\mathcal{I}$. Since the type concepts are disjoint, the following holds for all finite interpretations $\mathcal{I}$:

$$|C^{\mathcal{I}}| = \sum_{t \in \text{types}(E) \text{ s.t. } C \in t} |C^{\mathcal{I}}_t| \quad \text{and} \quad |C^{\mathcal{I}}_t| = |\bigcap_{C \in t} C^{\mathcal{I}}|,$$

where the latter identity is an immediate consequence of the definition of C_t as the conjunction of all the elements of t.

Given a type $t \in \mathcal{M}_E$, the constraints occurring in t induce a QFBAPA formula ψ_t, in which the concepts C and roles r occurring in these constraints are replaced by set variables X_C and X^t_r, respectively. For example, if $t = \{sat(|A| \geq 4), sat(A \subseteq r), A\}$, then $\psi_t = |X_A| \geq 4 \wedge X_A \subseteq X^t_r$. Note that set variables corresponding to concepts are independent of the type t, i.e., they are shared by all types, whereas the set variables corresponding to roles are different for different types. This corresponds to the fact that roles are evaluated locally, but concepts are evaluated globally in the semantics of $\mathcal{ALCSCC}^{++}$. In order to ensure that the Boolean structure of concepts is respected by the set variables, we introduce the formula

$$\beta \;=\; \bigwedge_{C \sqcap D \in \mathcal{M}_E} X_{C \sqcap D} = X_C \cap X_D \wedge \bigwedge_{C \sqcup D \in \mathcal{M}_E} X_{C \sqcup D} = X_C \cup X_D \wedge \bigwedge_{\neg C \in \mathcal{M}_E} X_{\neg C} = (X_C)^c.$$

Overall, we translate the $\mathcal{ALCSCC}^{++}$ concept E into the QFBAPA formula

$$\delta_E := (|X_E| \geq 1) \wedge \beta \wedge \bigwedge_{t \in \text{types}(E)} (|\bigcap_{C \in t} X_C| = 0) \vee \psi_t.$$

Intuitively, to satisfy E, we need to have at least one element in it, which explains the first conjunct. The third conjunct together with β ensures that, for any type that is realized (i.e., has elements), the constraints of this type are satisfied.

The following lemma, whose proof can be found in [5], states that solvability of δ_E and satisfiability of E are indeed equivalent.

Lemma 4 *The $\mathcal{ALCSCC}^{++}$ concept description E is satisfiable iff the QFBAPA formula δ_E is satisfiable.*

Since it is easy to see that the size of δ_E is exponential in E, and satisfiability of QFBAPA formulae can be decided within NP even for binary coding of numbers [14], this lemma shows that satisfiability of $\mathcal{ALCSCC}^{++}$ concept descriptions can be decided within NExpTime. Together with the known NExpTime lower bound for consistency of $\mathcal{ALC}$ ECBoxes in [8], this yields:

Theorem 5 *Satisfiability of $\mathcal{ALCSCC}^{++}$ concept descriptions is* NExpTime*-complete independently of whether the numbers occurring in these descriptions are encoded in unary or binary.*

Thanks to Proposition 2, the NExpTime upper bound carries over to satisfiability of $\mathcal{ALCSCC}^{++}$ knowledge bases, which may feature an ABox, a TBox and an ECBox.

4 Restricted Cardinality Constraints and ABoxes in $\mathcal{ALCSCC}$

Recall that $\mathcal{ALCSCC}$ is the restriction of $\mathcal{ALCSCC}^{++}$ where concepts C in constraint expressions occur only in the form $C \cap (\bigcup_{r \in N_R} r)$. In the syntax of $\mathcal{ALCSCC}$, we dispense with writing the intersection with $(\bigcup_{r \in N_R} r)$ explicitly, and then realize the restriction to the role successors of the individual in question by defining the semantics of set variables corresponding to concepts in the constraint expressions accordingly. Syntactically, we write $succ(c)$ instead of $sat(c)$ to make clear that the constraint is to be interpreted locally by considering only the role successors of the given individual (see [1, 5] for a detailed introduction of the syntax and semantics of $\mathcal{ALCSCC}$). ECBoxes for $\mathcal{ALCSCC}$ are basically $\mathcal{ALCSCC}^{++}$ concept descriptions that are Boolean combinations of constraint expressions $sat(c)$ where c contains only $\mathcal{ALCSCC}$ concept descriptions as set variables, but now such expressions are not viewed as concept constructors, but as terminological statements that may be true or false in an interpretation, corresponding to the respective settings where the concept description contains all individuals (true) or no individual (false) (see [8, 2, 5] for a detailed introduction of the syntax and semantics of ECBoxes in $\mathcal{ALC}$ and $\mathcal{ALCSCC}$).

For the sub-logic $\mathcal{ALC}$ of $\mathcal{ALCSCC}$, a restricted notion of cardinality boxes, called RCBoxes, was introduced in [8], and it was proved that this restriction lowers the complexity of the consistency problem from NExpTime to ExpTime. In [2, 3] it was shown that the same is true for $\mathcal{ALCSCC}$. Here we demonstrate that this result can be extended to consistency of $\mathcal{ALCSCC}$ ABoxes w.r.t. $\mathcal{ALCSCC}$ RCBoxes. In the presence of ECBoxes, this extension is irrelevant since ECBoxes can express nominals, and thus also ABoxes. However, this is not the case for RCBoxes. Here, we actually consider an extension of RCBoxes, which were called ERCBoxes in [19].

Definition 6 Semi-restricted $\mathcal{ALCSCC}$ cardinality constraints *are of the form* $N_1|C_1| + \cdots + N_k|C_k| + M \leq N_{k+1}|C_{k+1}| + \cdots + N_{k+\ell}|C_{k+\ell}|$*, where* C_i *are* $\mathcal{ALCSCC}$ *concept descriptions,* N_i *are integer constants for* $1 \leq i \leq k + \ell$*, and* M *is a non-negative integer constant. An* extended restricted $\mathcal{ALCSCC}$ cardinality box (ERCBox) *is a positive Boolean combination of semi-restricted* $\mathcal{ALCSCC}$ *cardinality constraints.*

An $\mathcal{ALCSCC}$ ABox *is a finite set of concept assertions of the form* $C(a)$ *and role assertions* $r(a, b)$*, where* C *is an* $\mathcal{ALCSCC}$ *concept description,* r *is a role name, and* a, b *are individual names from a set* N_I *of such names, which is disjoint with* N_C *and* N_R*. The set of all individual names occurring in an ABox* $\mathcal{A}$ *is denoted as* $\mathsf{Ind}_{\mathcal{A}}$*.*

An interpretation $\mathcal{I}$ *is a* model *of a semi-restricted* $\mathcal{ALCSCC}$ *cardinality constraint of the form introduce above if* $N_1|C_1^{\mathcal{I}}| + \cdots + N_k|C_k^{\mathcal{I}}| + B \leq N_{k+1}|C_{k+1}^{\mathcal{I}}| + \cdots + N_{k+\ell}|C_{k+\ell}^{\mathcal{I}}|$*. The notion of a model is extended to ERCBoxes using the usual interpretation of conjunction and disjunction in propositional logic.*

In the presence of an ABox, an interpretation additionally assigns elements $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$ *to individual names* a*. The interpretation* $\mathcal{I}$ *is a* model of an $\mathcal{ALCSCC}$ ABox $\mathcal{A}$ w.r.t. an $\mathcal{ALCSCC}$ ERCBox $\mathcal{R}$ *if it is a model of* $\mathcal{R}$ *that additionally satisfies* $a^{\mathcal{I}} \in C^{\mathcal{I}}$ *for all concept assertions* $C(a) \in \mathcal{A}$ *and* $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$ *for all role assertions* $r(a, b) \in \mathcal{A}$*.*

Note that $\mathcal{ALCSCC}$ ECBoxes can express both ERCBoxes and ABoxes, which yields a NExpTime-upper bound for the consistency problem of $\mathcal{ALCSCC}$ ABoxes w.r.t. $\mathcal{ALCSCC}$ ERCBoxes [2, 3]. Since ExpTime-hardness already holds for consistency of restricted cardinality boxes (RCBoxes) in $\mathcal{ALCSCC}$ without an ABox [8, 2, 3], we also obtain an ExpTime complexity lower bound. Actually, the hardness proof in [8] does not require large numbers, and thus ExpTime-hardness even holds for unary coding of numbers.

One important contribution of the present paper is to close this complexity gap, by lowering the upper bound to ExpTime.

Theorem 7 *Consistency of* $\mathcal{ALCSCC}$ *ABoxes w.r.t.* $\mathcal{ALCSCC}$ *ERCBoxes is an* ExpTime*-complete problem.*

Since the proof of this theorem is quite long and technical, we cannot give it here. A detailed proof can be found in [5]. It uses an extension of the approach employed in [2, 3] to decide consistency of $\mathcal{ALCSCC}$ RCBoxes, but the presence of the ABox requires substantial changes. Basically, this approach is based on type elimination, but instead of the simple types employed in the previous section, so-called *augmented types* [1] are used. An augmented type (t, V) consists of a type t together with a polynomially large set of Venn regions V, such that the QFBAPA formula ψ_t induced by the successor constraints contained in t can be satisfied by a solution in which only these Venn regions are non-empty. The ABox individuals are taken into account by allowing them to occur in types and Venn regions. Locally, we can actually express in QFBAPA that set variables X_b corresponding to individuals b must have cardinality ≤ 1. In addition, if t contains the individual a and $\mathcal{A}$ contains $r(a, b)$, we can add the constraint $|X_b \cap X_r| \geq 1$ to the QFBAPA formula ψ_t induced by t. Globally, constraints that ensure that individuals occur only once cannot be expressed by ERCBoxes, but this is taken care of by the second elimination step below.

In principle, *type elimination* now proceeds on input $\mathcal{A}, \mathcal{R}$ (where without loss of generality $\mathcal{A} \neq \emptyset$ and $\mathcal{R}$ is a conjunction of semi-restricted constraints) as follows:

1. The set of all augmented types for $\mathcal{A}$ and $\mathcal{R}$ are computed. There are exponentially many such types and they can be computed in exponential time using QFBAPA reasoning.
2. Starting with the set of all augmented types, all maximal subsets are computed such that (i) for every individual b there is exactly one augmented type (t, V) with $b \in t$ in this set, and (ii) the concept assertions in $\mathcal{A}$ are respected, i.e., $b \in t$ and $C(b) \in \mathcal{A}$ implies $C \in t$ for all augmented types (t, V) in the set. It can be shown that there are exponentially many such maximal sets, and they can be computed in exponential time (see [5]). For each of these maximal sets, the procedure continues with the next step.
3. Remove augmented types (t, V) for which the successors required by its Venn regions are not realized by an augmented type in the set. Continue with this step until no more augmented types are removed. If there is an individual b in $\mathcal{A}$ such that there is no augmented type $(t, \cdot)$ with $b \in t$ in the set, then terminate with failure. Otherwise, continue with the next step.
4. Remove augmented types $(t, \cdot)$ that are forced to be empty by the ERCBox, i.e., where there is no solution of the corresponding system of linear inequations in which the variable v_t standing for the cardinality of the type concept C_t has a value different from 0. Continue with the previous step if such a type is removed. If no augmented type is removed in this step, then the algorithm terminates with success.

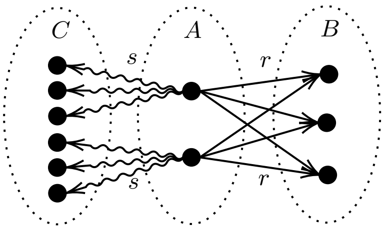
In [5] we show that this elimination algorithm is sound and complete, and runs in exponential time. Given an arbitrary, not necessarily conjunctive ERCBox $\mathcal{R}$, we can reduce testing consistency of $\mathcal{A}$ w.r.t. $\mathcal{R}$ to testing consistency of $\mathcal{A}$ w.r.t. $\mathcal{R}_\rho$ for exponentially many conjunctive ERCBoxes $\mathcal{R}_\rho$ of a size that is linearly bounded by the size of $\mathcal{R}$. Here $\mathcal{A}$ is consistent w.r.t. $\mathcal{R}$ iff $\mathcal{A}$ is consistent w.r.t. $\mathcal{R}_\rho$ for one of these conjunctive ERCBoxes $\mathcal{R}_\rho$. This yields the ExpTime upper bound stated in Theorem 7.

One might ask whether the approach used here to deal with individuals in ABoxes could also be used to treat nominals in concept descriptions, where a nominal is a concept that must be interpreted as a singleton set. The answer to the above question is, unfortunately, negative. In fact, using a reduction from [23], it is easy to see that adding nominals increases the complexity of ERCBox consistency from ExpTime to NExpTime even for $\mathcal{ALC}$, i.e. consistency of conjunctive $\mathcal{ALCO}$ ERCBoxes is NExpTime-complete (see [5] for details).

5 Undecidable Extensions of $\mathcal{ALCSCC}^{++}$

It turns out that a seemingly harmless extension of $\mathcal{ALCSCC}^{++}$ makes the satisfiability problem undecidable. $\mathcal{ALCISCC}^{++}$ is obtained from $\mathcal{ALCSCC}^{++}$ by allowing *role inverses*, i.e. expressions of the form r^- for any $r \in N_R$, to occur in the places of role names. The semantics of the expression r^- is defined by $(r^-)^\mathcal{I} = \{(e,d) \mid (d,e) \in r^\mathcal{I}\}$. The key insight for showing our undecidability result is that adding this feature enables us to encode multiplication of concept extensions, allowing for a reduction from Hilbert's tenth problem [16]. We provide an example illustrating how "class extension multiplication" can be expressed.

Example 8 *In order to express that the cardinality of a concept C coincides with the product of the cardinalities of concepts A and B, we employ two auxiliary roles r and s. The following figure illustrates the construction we are aiming at.*



We first enforce that role r connects precisely each member of A with every member of B:

$$A \equiv \exists r.\top \quad B \equiv \exists r^-.\top \quad A \sqsubseteq sat(B = r) \quad B \sqsubseteq sat(A = r^-)$$

Next, we make sure that (i) every domain element has precisely as many outgoing r edges as outgoing s edges; (ii) the elements with incoming s edges are precisely the instances of concept C; and (iii) no element can have more than one incoming s edge (in other words, s is inverse functional):

$$\top \sqsubseteq sat(|r| = |s|) \qquad C \equiv \exists s^-.\top \qquad \top \sqsubseteq sat(|s^-| \leq 1)$$

A construction very much along the lines of the given example allows us to express the undecidable Hilbert's tenth problem as an $\mathcal{ALCISCC}^{++}$ concept satisfiability problem, and hence establish undecidability of the latter.

Theorem 9 *Satisfiability of $\mathcal{ALCISCC}^{++}$ concept descriptions is undecidable.*

Our other undecidability result for $\mathcal{ALCSCC}^{++}$ concerns querying. As usual, we focus on Boolean conjunctive queries, since general query answering can be reduced to it.

Definition 10 *In queries, we use* variables *from a countably infinite set V. A Boolean* conjunctive query *(CQ) q is a finite set of atoms of the form $r(x,y)$ or $C(z)$, where r is a role, C is concept, and $x, y, z \in V$. A CQ q is* satisfied *by $\mathcal{I}$ (written: $\mathcal{I} \models q$) if there is a* variable assignment *$\pi : V \to \Delta^\mathcal{I}$ (called* match*) such that $(\pi(x), \pi(y)) \in r^\mathcal{I}$ for every $r(x,y) \in q$ and $\pi(z) \in C^\mathcal{I}$ for every $C(z) \in q$. A CQ q is* (finitely) entailed *by a knowledge base $\mathcal{K}$ (written: $\mathcal{K} \models_{(\mathrm{fin})} q$) if every (finite) model $\mathcal{I}$ of $\mathcal{K}$ satisfies q.*

We actually show undecidability of CQ entailment for a much weaker logic, thereby providing a very restricted fragment of constant-free and equality-free two-variable first-order logic for which finite CQ entailment is already undecidable, significantly strengthening and solidifying earlier results along those lines [18].

We show our undecidability result for the DL $\mathcal{ALC}^{\mathrm{cov}}$, a slight extension of $\mathcal{ALC}$ by *role cover axioms* of the form $\mathrm{cov}(r,s)$ for role names r and s. An interpretation $\mathcal{I}$ satisfies $\mathrm{cov}(r,s)$ if $r^\mathcal{I} \cup s^\mathcal{I} = \Delta^\mathcal{I} \times \Delta^\mathcal{I}$. Role cover axioms can be expressed in $\mathcal{ALCSCC}^{++}$ via $sat(\top \subseteq sat(|r \cup s| = |\mathcal{U}|))$, hence $\mathcal{ALC}^{\mathrm{cov}}$ is a sub-logic of $\mathcal{ALCSCC}^{++}$.

The undecidability of finite CQ entailment from $\mathcal{ALC}^{\mathrm{cov}}$ TBoxes is established by a reduction from the undecidable problem of determining if a Turing machine (TM) is looping. As usual, the key ingredient for this is to enforce a grid structure on which the TM's tape configurations are represented horizontally (connected via a role named h) while the vertical direction (role v) corresponds to the subsequent time steps.

Intuitively, we use the query to catch the unwanted situation of the grid not being "closed", i.e., two corresponding tape cells of time-consecutive configurations are v-connected, while the cells to their right are not:

$$q = \exists x, y, x', y'.v(x,y) \land h(x,x') \land h(y,y') \land \overline{v}(x',y')$$

Now, the covering axiom $\mathrm{cov}(v, \overline{v})$ ensures that, whenever two elements are not v-connected, they must be $\overline{v}$-connected. This is needed to enable the above query to catch the described problem.

Beyond this key ingredient, expressing the progression (i.e., computing successor configurations) of the TM by $\mathcal{ALC}$ TBox axioms is straightforward. Note however, that the requirement of model-finiteness forces the grid to "loop back" at some stage, thereby only allowing to represent TM runs that become repetitive after some time – which is the reason for our reduction from the looping rather than the halting problem.

Hence we obtain an $\mathcal{ALC}^{\mathrm{cov}}$ TBox $\mathcal{T}$ and CQ q such that $\mathcal{T} \models_{\mathrm{fin}} q$ if and only if the underlying TM is not looping.

Theorem 11 *Finite conjunctive query entailment by $\mathcal{ALC}^{\mathrm{cov}}$ TBoxes is undecidable.*

Finally, taking into account that $\mathcal{ALCSCC}^{++}$subsumes $\mathcal{ALC}^{\mathrm{cov}}$ and only allows for finite models, we obtain the announced result.

Corollary 12 *CQ entailment for $\mathcal{ALCSCC}^{++}$is undecidable.*

6 Decidable querying for $\mathcal{ALCSCC}$

In stark contrast to the undecidability result just presented, we prove that conjunctive query entailment by $\mathcal{ALCSCC}$ ABoxes w.r.t. $\mathcal{ALCSCC}$ ERCBoxes is only ExpTime-complete, and thus not harder than deciding knowledge base consistency for plain $\mathcal{ALC}$.

Our result employs a construction by Lutz [15], but careful and non-trivial argumentation is needed to show that the idea, conceived for arbitrary models, carries over to our finite-model case. The approach reduces entailment of some CQ q to an exponential number of inconsistency checks, which are in ExpTime by Theorem 7, resulting in an overall ExpTime procedure. In their entirety, these mentioned checks verify if some model exists that does not admit any matches of q having a specific, forest-like shape.

It remains to argue that these specific, forest-shaped query matches of q are the only ones that matter for checking entailment. To this end, we show that all other matches can be "removed" by a model transformation consisting of the following three consecutive steps:

(i) forward-unraveling, resulting in possibly-infinite structures, then (ii) cautious collapsing to regain finiteness while keeping the model "forest-like enough" for small conjunctive queries to match only in a tree-shaped way, and finally (iii) enriching the model by copies of domain elements to again satisfy the global counting constraints which had possibly become violated in the course of the previous steps. Due to lack of space, we will focus on sketching the mentioned model transformation constructions. Technical details as well as proofs are available in the technical report [5].

The construction of sufficiently tree-like models

Let $\mathcal{K} = (\mathcal{A}, \mathcal{T}, \mathcal{R})$ be an $\mathcal{ALCSCC}$ knowledge base, where $\mathcal{A}$ is an ABox, $\mathcal{T}$ is a TBox and $\mathcal{R}$ is an ERCBox. By routine transformations, we can ensure that $\mathcal{K}$ is in a *normalized* form where all concepts appearing in $\mathcal{A}$ and $\mathcal{R}$ are concept names and all concepts in $\mathcal{T}$ are of depth at most one.

Let $\mathcal{I}$ be a finite model of $\mathcal{K}$. We denote with $\Delta^{\mathcal{I}}_{\text{named}}$ the set of all *named individuals*, i.e., elements $d \in \Delta^{\mathcal{I}}$ for which $a^{\mathcal{I}} = d$ holds for some individual name $a \in \mathsf{Ind}_{\mathcal{A}}$.

Definition 13 *The* forward unraveling *of an interpretation $\mathcal{I}$ is (a potentially infinite) interpretation $\mathcal{I}^{\rightarrow} = (\Delta^{\mathcal{I}^{\rightarrow}}, \cdot^{\mathcal{I}^{\rightarrow}})$ defined by*

- $\Delta^{\mathcal{I}^{\rightarrow}} = (\Delta^{\mathcal{I}})^{+} \setminus \big(\Delta^{\mathcal{I}}_{\text{named}} \cdot \Delta^{\mathcal{I}}_{\text{named}} \cdot (\Delta^{\mathcal{I}})^{*}\big)$*, in words: $\Delta^{\mathcal{I}^{\rightarrow}}$ consists of all nonempty sequences of elements from $\Delta^{\mathcal{I}}$ except those, where the first two elements are named individuals.*
- *For any $a \in \mathsf{Ind}_{\mathcal{A}}$, let $a^{\mathcal{I}^{\rightarrow}} = a^{\mathcal{I}}$, i.e. a is interpreted by the one-element sequence consisting of the named element $a^{\mathcal{I}}$ from $\mathcal{I}$.*[4]
- *For concept names $A \in N_C$ and role names $r \in N_R$, we let*

$$A^{\mathcal{I}^{\rightarrow}} = \{w \mid \mathsf{last}(w) \in A^{\mathcal{I}}\} \text{ and}$$

$$r^{\mathcal{I}^{\rightarrow}} = r^{\mathcal{I}} \cap (\Delta^{\mathcal{I}}_{\text{named}} \times \Delta^{\mathcal{I}}_{\text{named}}) \cup \{(w, wd) \mid (\mathsf{last}(w), d) \in r^{\mathcal{I}}\},$$

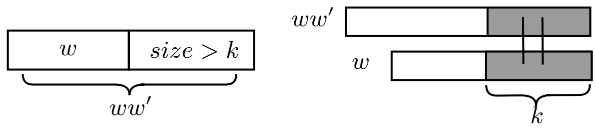
where $\mathsf{last}(w)$ denotes the last element of the sequence w.

The notion of forward-unravelings differs only slightly from the classical notion of unraveling [9]. The only difference is that the sequences starting from two named individuals are excluded from the domain and that roles linking named invividuals are assigned manually by the last item from Definition 13. It is not surprising that forward-unravellings preserve satisfaction of $\mathcal{ALCSCC}$ ABoxes and TBoxes as well as conjunctive query non-entailment. The proof is standard and hinges on the fact that $w \in \Delta^{\mathcal{I}^{\rightarrow}}$ and $\mathsf{last}(w) \in \Delta^{\mathcal{I}}$ satisfy the same $\mathcal{ALCSCC}$ concepts. For CQ non-entailment it is enough to see that $\mathsf{last}(\cdot)$ is a homomorphism from $\mathcal{I}^{\rightarrow}$ to $\mathcal{I}$.

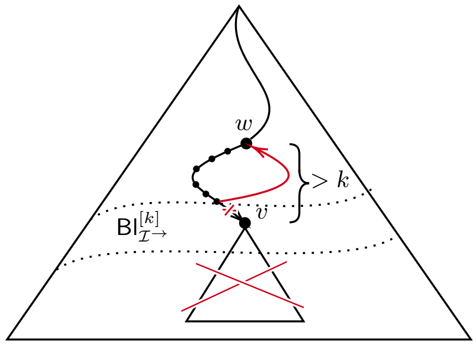
Lemma 14 *If $\mathcal{I} \models (\mathcal{A}, \mathcal{T})$ then $\mathcal{I}^{\rightarrow} \models (\mathcal{A}, \mathcal{T})$. Moreover, for any conjunctive query q, if $\mathcal{I} \not\models q$ then $\mathcal{I}^{\rightarrow} \not\models q$.*

Unraveling removes non-forest-shaped query matches. However, $\mathcal{I}^{\rightarrow}$ does not need to be finite even if $\mathcal{I}$ is. To regain finiteness without re-introducing unwanted query matches, we are going to introduce the notion of a *k-loosening*, which depends on k–blocking. An element $u \in \Delta^{\mathcal{I}^{\rightarrow}}$ is *k–blocked* by its prefix w, if $u = ww'$ for some w' of length longer than k, and w's and u's suffixes of length k coincide (see the illustrating picture). We say that u is *minimally k–blocked* if it is k–blocked (by some prefix), but none of its prefixes is k–blocked. With $\mathsf{Bl}^{[k]}_{\mathcal{I}^{\rightarrow}}$ we denote the set of minimally k–blocked elements in $\mathcal{I}^{\rightarrow}$.

[4] We will not syntactically distinguish elements from $\Delta^{\mathcal{I}}$ and one-element sequences from $\Delta^{\mathcal{I}^{\rightarrow}}$; in particular this means $\Delta^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}^{\rightarrow}}$.



Definition 15 *The k-loosening $\mathcal{I}^{[k]}$ of $\mathcal{I}$ is obtained from $\mathcal{I}^{\rightarrow}$ by exhaustively selecting minimally k–blocked elements v from $\mathsf{Bl}^{[k]}_{\mathcal{I}^{\rightarrow}}$ (k–blocked by some w), removing all descendants of v and identifying the nodes v and w.*



The above sketch illustrates a single step in the construction of $\mathcal{I}^{[k]}$. Note that the k–loosening of a finite $\mathcal{I}$ is finite since $\mathcal{I}^{\rightarrow}$ is finitely branching (due to the finiteness of $\mathcal{I}$) and has finite depth (since blocking eventually occurs on every branch due to the pigeonhole principle). Thus, from König's lemma we can conclude that $\mathcal{I}^{[k]}$ is indeed finite.

Like unravelings, k-loosenings preserve satisfaction of normalized ABoxes and TBoxes, as well as CQ non-entailment. However, ERCBoxes might become violated in the construction.

Lemma 16 *For any positive k, $\mathcal{I} \models (\mathcal{A}, \mathcal{T})$ implies $\mathcal{I}^{[k]} \models (\mathcal{A}, \mathcal{T})$, and $\mathcal{I} \not\models q$ implies $\mathcal{I}^{[k]} \not\models q$.*

For a given interpretation $\mathcal{J}$, an *anonymous cycle* is simply a word $w \in (\Delta^{\mathcal{J}})^{+} \cdot (\Delta^{\mathcal{J}} \setminus \Delta^{\mathcal{J}}_{\text{named}}) \cdot (\Delta^{\mathcal{J}})^{+}$, where the first and the last element are the same, and for any two consecutive elements d_i, d_{i+1} of w there exists a role r witnessing $(d_i, d_{i+1}) \in r^{\mathcal{J}}$. The *girth* of $\mathcal{J}$ is the length of the smallest anonymous cycle in $\mathcal{J}$ if such a cycle exists or ∞ otherwise. The main feature of the k–loosening $\mathcal{I}^{[k]}$ is that the girth of $\mathcal{I}^{[k]}$ is at least k.

Lemma 17 *For any $k \in \mathbb{N}$ the girth of $\mathcal{I}^{[k]}$ is at least k.*

Once k is greater than the number of atoms in q (denoted with $|q|$), the k–loosening of a model is still "locally acyclic enough" to ensure that the query matches only in a "forest-shaped" manner.

We next consider how to adjust a k-loosening such that it again satisfies the initial ERCBox. Since role inverses are not expressible in $\mathcal{ALCSCC}$, creating multiple copies of a single element, and forward-linking them to other elements precisely in the same way as the original element, can be done without any harm to modelhood nor query-non-entailment. We formalize this intuition below.

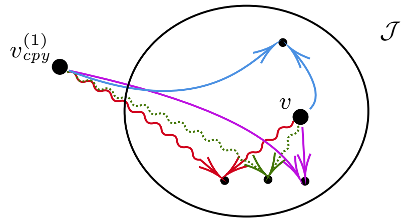
Definition 18 *For $\mathcal{J} \models (\mathcal{A}, \mathcal{T}, \mathcal{R})$ and $S \subseteq (\Delta^{\mathcal{I}} \times \mathbb{N}_{+})$ we define the S–*duplication *of $\mathcal{J}$ as the interpretation $\mathcal{J}_{+S} = (\Delta^{\mathcal{J}}_{+S}, \cdot^{\mathcal{J}_{+S}})$ with:*

- $\Delta^{\mathcal{I}}_{+S} = \Delta^{\mathcal{I}} \cup \bigcup_{(v,n) \in S} \{v^{(i)}_{cpy} \mid 1 \leq i \leq n\}$,
- *$a^{\mathcal{J}_{+S}} = a^{\mathcal{J}}$ for each individual name $a \in \mathsf{Ind}_{\mathcal{A}}$,*

- *For concept names $A \in N_C$ and role names $r \in N_R$ we set:*

$$A^{\mathcal{J}_{+S}} = A^{\mathcal{J}} \cup \bigcup_{(v,n)\in S} \left\{ v_{cpy}^{(i)} \mid 1 \le i \le n \wedge v \in A^{\mathcal{J}} \right\} \text{ and}$$

$$r^{\mathcal{J}_{+S}} = r^{\mathcal{J}} \cup \bigcup_{(v,n)\in S} \left\{ (v_{cpy}^{(i)}, w) \mid 1 \le i \le n \wedge (v,w) \in r^{\mathcal{J}} \right\}.$$



Once again, it can be shown that the S–duplication of $\mathcal{J}$ preserves satisfaction of ABoxes, TBoxes and query non-entailment.

Lemma 19 *For any interpretation $\mathcal{J}$, if $\mathcal{J} \models (\mathcal{A}, \mathcal{T})$, then for any $S \subseteq (\Delta^{\mathcal{I}} \times \mathbb{N}_+)$, we have $\mathcal{J}_{+S} \models (\mathcal{A}, \mathcal{T})$. Moreover if $\mathcal{J} \not\models q$ then also $\mathcal{J}_{+S} \not\models q$.*

The inequalities from $\mathcal{R}$ have the convenient property that, if a vector $\vec{x}$ containing the cardinalities of all atomic concepts' extensions is a solution to $\mathcal{R}$, then also a vector $c \cdot \vec{x}$, i.e., the vector obtained by multiplying each entry of $\vec{x}$ by a constant c is also a solution to $\mathcal{R}$. Thus there is a solution to $\mathcal{R}$ of the form $(1 + |\mathcal{I}^{[k]}|) \cdot \vec{x_{\mathcal{I}}}$, where $\vec{x_{\mathcal{I}}}$ is the solution to $\mathcal{R}$ describing the atomic concept extensions' cardinalities in $\mathcal{I}$. Since $\mathcal{I}^{[k]}$ preserves (non-)emptiness of all concepts from $\mathcal{I}$, we can simply copy an appropriate number of elements from $\mathcal{I}^{[k]}$, to make the ERCBox $\mathcal{R}$ satisfied again.

Lemma 20 *For any finite interpretation $\mathcal{I} \models (\mathcal{A}, \mathcal{T}, \mathcal{R})$, there exists a finite set $S \subseteq (\Delta^{\mathcal{I}} \times \mathbb{N}_+)$ such that $\mathcal{I}_{+S}^{[k]} \models (\mathcal{A}, \mathcal{T}, \mathcal{R})$.*

This concludes our construction, the core result of which can be informally stated as follows: *For any $\mathcal{ALCSCC}$ knowledge base $\mathcal{K}$ and every CQ q the following holds: if $\mathcal{K} \models q$ then there is a forest-shaped query match of q into every model of $\mathcal{K}$.* This follows from the fact that any model of $\mathcal{K}$ not admitting such a match would allow us to construct a model without any query matches, contradicting the assumption.

Deciding CQ entailment in exponential time

Now we are ready to employ the announced exponential time method for deciding conjunctive query entailment from [15]. For a given $\mathcal{K} = (\mathcal{A}, \mathcal{T}, \mathcal{R})$ and a query q, we enumerate a set of $\mathcal{ALCH}^{\cap}$ knowledge bases $\mathcal{K}_s = (\mathcal{A}', \mathcal{T}')$ called *spoilers*, and check whether $\mathcal{K} \cup \mathcal{K}_s$ is consistent. Spoilers are modeled to prevent forest-shaped query matches. They are constructed by, on the one hand, rolling-up tree-shaped partial query matches into concepts and forbidding existence of such concept in a model and, on the other hand, forbidding certain behaviour of the ABox part of a model. Lutz [15] shows that one can restrict ones attention to exponentially many spoilers and that the size of each such spoiler is only polynomial in $|\mathcal{K}|$ and $|q|$.

The algorithm for CQ entailment is then obtained by simply replacing Lutz's satisfiability algorithm for $\mathcal{ALCH}^{\cap}$ knowledge bases[5] by our finite satisfiability algorithm for $\mathcal{ALCSCC}$ knowledge bases from Section 4. We derive correctness of the procedure as follows: $\mathcal{K} \cup \mathcal{K}_s$ is satisfiable for some spoiler $\mathcal{K}_s$ exactly if there is a model of $\mathcal{K}$ without forest-shaped matches of q and hence – thanks to our above argument – there is a model without any match of q. We conclude:

Theorem 21 *Conjunctive query entailment by $\mathcal{ALCSCC}$ ABoxes w.r.t. $\mathcal{ALCSCC}$ ERCBoxes is* ExpTime*-complete.*

It is worth pointing out that our result also implies the exact upper bound for finite CQ entailment by $\mathcal{ALCHQ}$ knowledge-bases, for which only a doubly-exponential upper bound was known (see e.g. [18]). This result was somehow missing in the literature, probably due to the fact that $\mathcal{ALCHQ}$ is not known to be finitely-controllable.

Corollary 22 *Finite conjunctive query entailment by $\mathcal{ALCHQ}$ ABoxes w.r.t. $\mathcal{ALCHQ}$ TBoxes is* ExpTime*-complete.*

7 Conclusion

We have introduced the DL $\mathcal{ALCSCC}^{++}$, which allows for mixing local and global cardinality constraints. Though being considerably more expressive than previously investigated DLs with cardinality constraints, reasoning in $\mathcal{ALCSCC}^{++}$ has turned out to be not harder that reasoning in $\mathcal{ALC}$ with very simple cardinality restrictions. However, extending $\mathcal{ALCSCC}^{++}$ with inverse roles causes undecidability for the standard inference satisfiability, as does considering the non-standard inference of query entailment in $\mathcal{ALCSCC}^{++}$. We were able to show that decidability of query entailment can be regained by considering restricted cardinality constraints (ERCBoxes) in the sub-logic $\mathcal{ALCSCC}$ of $\mathcal{ALCSCC}^{++}$. The ExpTime upper bound proved for this task depends on the ExpTime upper bound for ABox consistency in $\mathcal{ALCSCC}$ w.r.t. ERCBoxes shown for the first time in the present paper.

Some of the results presented here have already been sketched in a paper at the DL workshop [4]. However, there the positive result for query entailment was restricted to a setting without ABox since we did not yet have the result for ABox consistency, and only a 2ExpTime upper bound for the complexity was shown. In addition, the undecidability result for $\mathcal{ALCISCC}^{++}$ is also not contained in [4].

Regarding future work, it would be interesting to investigate the impact that adding inverse roles has on reasoning in $\mathcal{ALCSCC}$ w.r.t. different kinds of terminological boxes (TBox, ERCBox, ECBox), though this will probably be a very hard task. From an application point of view, as a first step towards a more practical query answering algorithm, we intend to investigate the ABox consistency problem in $\mathcal{ALCSCC}$ w.r.t. ERCBoxes. Since type elimination algorithms are not only worst-case, but also best-case exponential, we will try to devise a tableau-based algorithm for this problem, which may use numerical algorithms and satisfiability checkers for QFBAPA as subprocedures.

Acknowledgements

Franz Baader was supported by DFG within the Research Unit 1513 Hybris and grant 389792660 as part of TRR 248; Bartosz Bednarczyk by the ERC through the Consolidator Grant 771779 (DeciGUT) and the Polish Ministry of Science and Higher Education program "Diamentowy Grant" no. DI2017 006447; Sebastian Rudolph by the ERC through the Consolidator Grant 771779 (DeciGUT).

[5] Note that $\mathcal{ALCH}^{\cap}$ is a sub-logic of $\mathcal{ALCSCC}$.

REFERENCES

[1] Franz Baader, 'A new description logic with set constraints and cardinality constraints on role successors', in *Proceedings of the 11th International Symposium on Frontiers of Combining Systems (FroCoS'17)*, eds., Clare Dixon and Marcelo Finger, volume 10483 of *Lecture Notes in Computer Science*, pp. 43–59, Brasília, Brazil, (2017). Springer-Verlag.

[2] Franz Baader, 'Expressive cardinality constraints on $\mathcal{ALCSCC}$ concepts', in *Proceedings of the 34th ACM/SIGAPP Symposium On Applied Computing (SAC'19)*. ACM, (2019).

[3] Franz Baader, 'Expressive cardinality restrictions on concepts in a description logic with expressive number restrictions', *ACM SIGAPP Applied Computing Review*, **19**, 5–17, (2019).

[4] Franz Baader, Bartosz Bednarczyk, and Sebastian Rudolph, 'Satisfiability checking and conjunctive query answering in description logics with global and local cardinality constraints', in *Proc. of the 2019 Description Logic Workshop (DL 2019)*, eds., Mantas Simkus and Grant E. Weddell, volume 2373 of *CEUR Workshop Proceedings*. CEUR-WS.org, (2019).

[5] Franz Baader, Bartosz Bednarczyk, and Sebastian Rudolph, 'Satisfiability and Query Answering in Description Logics with Global and Local Cardinality Constraints', Technical report, (2020). Available at https://arxiv.org/abs/2002.06072.

[6] Franz Baader, Martin Buchheit, and Bernhard Hollunder, 'Cardinality restrictions on concepts', *Artificial Intelligence*, **88**(1–2), 195–213, (1996).

[7] *The Description Logic Handbook: Theory, Implementation, and Applications*, eds., Franz Baader, Diego Calvanese, Deborah McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, Cambridge University Press, 2003.

[8] Franz Baader and Andreas Ecke, 'Extending the description logic alc with more expressive cardinality constraints on concepts', in *GCAI 2017. 3rd Global Conference on Artificial Intelligence*, volume 50 of *EPiC Series in Computing*, pp. 6–19. EasyChair, (2017).

[9] Franz Baader, Ian Horrocks, Carsten Lutz, and Ulrike Sattler, *An Introduction to Description Logic*, Cambridge University Press, 2017.

[10] Alexander Borgida, Ronald J. Brachman, Deborah L. McGuinness, and Lori Alperin Resnick, 'CLASSIC: A structural data model for objects', in *Proc. of the ACM SIGMOD Int. Conf. on Management of Data*, pp. 59–67, (1989).

[11] Robert Hoehndorf, Paul N. Schofield, and Georgios V. Gkoutos, 'The role of ontologies in biological and biomedical research: A functional perspective', *Brief. Bioinform.*, **16**(6), 1069–1080, (2015).

[12] Bernhard Hollunder and Franz Baader, 'Qualifying number restrictions in concept languages', in *Proc. of the 2nd Int. Conf. on the Principles of Knowledge Representation and Reasoning (KR'91)*, pp. 335–346, (1991).

[13] Bernhard Hollunder, Werner Nutt, and Manfred Schmidt-Schauß, 'Subsumption algorithms for concept description languages', in *Proc. of the 9th Eur. Conf. on Artificial Intelligence (ECAI'90)*, pp. 348–353, London (United Kingdom), (1990). Pitman.

[14] Viktor Kuncak and Martin C. Rinard, 'Towards efficient satisfiability checking for Boolean algebra with Presburger arithmetic', in *Proc. of the 21st Int. Conf. on Automated Deduction (CADE-07)*, ed., Frank Pfenning, volume 4603 of *Lecture Notes in Computer Science*, pp. 215–230. Springer, (2007).

[15] Carsten Lutz, 'The complexity of conjunctive query answering in expressive description logics', in *Proc. of the Int. Joint Conf. on Automated Reasoning (IJCAR 2008)*, eds., Alessandro Armando, Peter Baumgartner, and Gilles Dowek, Lecture Notes in Artificial Intelligence, pp. 179–193. Springer-Verlag, (2008).

[16] Yuri V. Matiyasevich, *Hilbert's Tenth Problem*, MIT Press, 1993.

[17] Rafael Peñaloza and Nico Potyka, 'Towards statistical reasoning in description logics over finite domains', in *Proc. of the 11th Int. Conf. on Scalable Uncertainty Management (SUM 2017)*, eds., Serafin Moral and Olivier Pivert, volume 10564 of *Lecture Notes in Computer Science*. Springer-Verlag, (2017).

[18] Ian Pratt-Hartmann, 'Data-complexity of the two-variable fragment with counting quantifiers', *Inf. Comput.*, **207**(8), 867–888, (2009).

[19] "Johann" Sebastian Rudolph, 'Presburger concept cardinality constraints in very expressive description logics – allegro sexagenarioso ma non ritardando', in *Description Logic, Theory Combination, and All That – Essays Dedicated to Franz Baader on the Occasion of His 60th Birthday*, eds., Carsten Lutz, Uli Sattler, Cesare Tinelli, Anni-Yasmin Turhan, and Frank Wolter, volume 11560 of *Lecture Notes in Computer Science*, pp. 542–561. Springer, (2019).

[20] Sebastian Rudolph, 'Foundations of description logics', in *Reasoning Web. Semantic Technologies for the Web of Data – 7th International Summer School 2011*, eds., Axel Polleres, Claudia d'Amato, Marcelo Arenas, Siegfried Handschuh, Paula Kroner, Sascha Ossowski, and Peter F. Patel-Schneider, volume 6848 of *LNCS*, 76–136, Springer, (2011).

[21] Stephan Tobies, 'A PSPACE algorithm for graded modal logic', in *Proc. of the 16th Int. Conf. on Automated Deduction (CADE'99)*, ed., Harald Ganzinger, volume 1632 of *Lecture Notes in Artificial Intelligence*, pp. 52–66. Springer-Verlag, (1999).

[22] Stephan Tobies, 'The complexity of reasoning with cardinality restrictions and nominals in expressive description logics', *J. of Artificial Intelligence Research*, **12**, 199–217, (2000).

[23] Stephan Tobies, *Complexity Results and Practical Algorithms for Logics in Knowledge Representation*, Ph.D. dissertation, LuFG Theoretical Computer Science, RWTH-Aachen, Germany, 2001.