# Not too Big, Not too Small...
# Complexities of Fixed-Domain Reasoning in First-Order and Description Logics

Sebastian Rudolph and Lukas Schweizer
firstname.lastname@tu-dresden.de

Technische Universität Dresden, Computational Logic Group

**Abstract.** We consider reasoning problems in description logics and variants of first-order logic under the fixed-domain semantics, where the model size is finite and explicitly given. It follows from previous results that standard reasoning is NP-complete for a very wide range of logics, if the domain size is given in unary encoding. In this paper, we complete the complexity overview for unary encoding and investigate the effects of binary encoding with partially surprising results. Most notably, fixed-domain standard reasoning becomes NEXPTIME for the rather low-level description logics $\mathcal{ELI}$ and $\mathcal{ELF}$ (as opposed to EXPTIME when no domain size is given). On the other hand, fixed-domain reasoning remains NEXPTIME even for first-order logic, which is undecidable under the unconstrained semantics. For less expressive logics, we establish a generic criterion ensuring NP-completeness of fixed-domain reasoning. Amongst other logics, this criterion captures all the tractable profiles of OWL 2.

## 1 Introduction

Description logics [2, 14] and other fragments of first-order logic are popular knowledge representation (KR) formalisms. Traditionally, the semantics underlying these formalisms do not constrain the number of elements of the described domain of interest; classical KR even admits models of infinite size. In many realistic knowledge representation scenarios, however, it is known that the domain must be finite, or even a bound on its size is given. In such scenarios, the classical semantics allowing for models of arbitrary (even infinite) size does not adequately reflect the reasoning requirements. Consequently, KR research has recently started to consider alternative semantics, leading to numerous novel (un)decidability and complexity results. The *finite model semantics*, inspired by database theory, requires models to be finite (but of arbirary size). Finite satisfiability checking and finite query entailment have been considered for a variety of description logics [11, 5, 13, 15]. Still, in some situations just requiring finiteness of the domain is not restrictive enough; sometimes the complete set of domain elements (or at least their precise number) are known, leading to the notion of the *fixed-domain semantics* [8]. First complexity investigations have shown that standard reasoning tasks are NP-complete for a wide range of DLs, when the elements of the fixed domain are explicitly enumerated. These results directly carry over to the case where only the number of domain elements is provided but this number is given in unary encoding.

Previous work has left open (or has been too unspecific on) important questions regarding fixed-domain reasoning. First, it was not explicitly investigated, if and how the complexity results would be affected if the domain size would be considered as a fixed parameter rather than a variable part of the input. Second, reasoning complexities of (fragments of) first-order logic (which are undecidable under classical and finite-model semantics but become decidable when fixing the domain) have not been investigated thoroughly. Third, the effect of representing the size of the domain in binary encoding has not been considered at all. In this paper, we clarify the complexity landscape for fixed-domain standard reasoning in first-order logic and description logics by making the following contributions.

- We show that for unrestricted first-order logic, standard reasoning is NExpTime, no matter if the domain size is a fixed parameter or given in unary or binary as part of the input. For fixed size or unary encoding, the complexity drops to PSpace when bounding the predicate arity, and to NP when the number of variables is bounded.
- We show that for the binary encoding, the complexity remains NExpTime, even when the logic is drastically restricted. In particular, we show corresponding hardness results for $\mathcal{ELI}$ and $\mathcal{ELF}$ terminologies.
- Finally, we establish NP-completeness for the binary encoding case for a wide range of lightweight logics by introducing a model-theoretic property shared by these logics and showing that it warrants NP-membership.

## 2 Preliminaries

### 2.1 KR Formalisms

We briefly recap syntax and semantics of the description logic $\mathcal{SROIQ}$ and some of its fragments [9].

Let $N_I$, $N_C$, and $N_R$ be finite, disjoint sets called *individual names*, *concept names* and *role names* respectively. These atomic entities can be used to form complex ones as displayed in Table 1. A $\mathcal{SROIQ}$ *knowledge base* $\mathcal{K}$ is a tuple $(\mathcal{A}, \mathcal{T}, \mathcal{R})$ where $\mathcal{A}$ is an ABox, $\mathcal{T}$ is a TBox and $\mathcal{R}$ is a RBox. Table 1 presents the axiom types available in the three parts.[1]

The semantics of $\mathcal{SROIQ}$ is defined via interpretations $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$ composed of a non-empty set $\Delta^\mathcal{I}$ called the *domain of* $\mathcal{I}$ and 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 $\Delta^\mathcal{I} \times \Delta^\mathcal{I}$. This mapping is extended to complex role and concept expressions and finally used to define satisfaction of axioms (see Table 1). We say that $\mathcal{I}$ *satisfies* a knowledge base $\mathcal{K} = (\mathcal{A}, \mathcal{T}, \mathcal{R})$ (or $\mathcal{I}$ is a *model* of $\mathcal{K}$, written: $\mathcal{I} \models \mathcal{K}$) if it satisfies all axioms of $\mathcal{A}$, $\mathcal{T}$, and $\mathcal{R}$. We say that a knowledge base $\mathcal{K}$ *entails* an axiom $\alpha$ (written $\mathcal{K} \models \alpha$) if all models of $\mathcal{K}$ are models of $\alpha$. If $\alpha$ is a GCI, i.e., an axiom of the form $C \sqsubseteq D$ the entailment problem is also referred to as *subsumption*.

[1] For brevity, we omit the global restrictions of $\mathcal{SROIQ}$ as they are irrelevant in our setting.

**Table 1.** Syntax and semantics of $\mathcal{SROIQ}$ role and concept constructors, where $a_1, \ldots a_n$ are individual names, $s$ a role name, $r$ a role expression and $C$ and $D$ concept expressions.

| Name | Syntax | Semantics | |
|---|---|---|---|
| inverse role | $s^-$ | $\{(x,y) \in \Delta^\mathcal{I} \times \Delta^\mathcal{I} \mid (y,x) \in s^\mathcal{I}\}$ | |
| 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 | $\{a_1, \ldots, a_n\}$ | $\{a_1^\mathcal{I}, \ldots, a_n^\mathcal{I}\}$ | |
| univ. restriction | $\forall r.C$ | $\{x \mid \forall y.(x,y) \in r^\mathcal{I} \rightarrow y \in C^\mathcal{I}\}$ | |
| exist. restriction | $\exists r.C$ | $\{x \mid \exists y.(x,y) \in r^\mathcal{I} \wedge y \in C^\mathcal{I}\}$ | |
| *Self* concept | $\exists r.\mathit{Self}$ | $\{x \mid (x,x) \in r^\mathcal{I}\}$ | |
| qualified number | $\leqslant n\, r.C$ | $\{x \mid \#\{y \in C^\mathcal{I} \mid (x,y) \in r^\mathcal{I}\} \leq n\}$ | |
| restriction | $\geqslant n\, r.C$ | $\{x \mid \#\{y \in C^\mathcal{I} \mid (x,y) \in r^\mathcal{I}\} \geq n\}$ | |
| | Axiom $\alpha$ | $\mathcal{I} \models \alpha$, if | |
| role chains | $r_1 \circ \cdots \circ r_n \sqsubseteq r$ | $r_1^\mathcal{I} \circ \cdots \circ r_n^\mathcal{I} \subseteq r^\mathcal{I}$ | RBox $\mathcal{R}$ |
| role disjointness | $\mathsf{Dis}(s,r)$ | $s^\mathcal{I} \cap r^\mathcal{I} = \emptyset$ | |
| subsumption | $C \sqsubseteq D$ | $C^\mathcal{I} \subseteq D^\mathcal{I}$ | TBox $\mathcal{T}$ |
| concept assertion | $C(a)$ | $a^\mathcal{I} \in C^\mathcal{I}$ | ABox $\mathcal{A}$ |
| role assertion | $r(a,b)$ | $(a^\mathcal{I}, b^\mathcal{I}) \in r^\mathcal{I}$ | |
| same individual | $a \doteq b$ | $a^\mathcal{I} = b^\mathcal{I}$ | |
| different individual | $a \neq b$ | $a^\mathcal{I} \neq b^\mathcal{I}$ | |

*$\mathcal{EL}$ / $\mathcal{ELI}$ / $\mathcal{ELF}$ Terminologies* $\mathcal{EL}$ terminologies are knowledge bases with only axioms of the form $C \sqsubseteq D$, where $C, D$ are concept expressions built from concept and role names using only top, conjunction, and existential quantification. We consider two extensions: $\mathcal{ELI}$ terminologies additionally allow the usage of role inverses, while $\mathcal{ELF}$ terminologies admit role functionality axioms of the shape $\top \sqsubseteq \leqslant 1\, r.C$.

$\mathsf{DL}_\mathsf{min}$ *Knowledge Bases* With $\mathsf{DL}_\mathsf{min}$, we refer to a minimalistic description logic that merely allows TBox axioms of the form $A \sqsubseteq \neg B$, with $A, B \in \mathsf{N}_C$. Moreover, only atomic assertions of the form $A(a)$ are admitted. It is immediate that (finite) satisfiability checking in $\mathsf{DL}_\mathsf{min}$ is in $\mathrm{AC}^0$.

*First-Order Logic* We assume the reader to be familiar with syntax and semantics of first-order predicate logic (FOL). By default, we assume that the only functions occurring are of arity zero (i.e., constants). We use $\mathrm{FOL}^=$ to denote FOL with equality. By *bounded-arity FOL*$^{(=)}$ we denote $\mathrm{FOL}^{(=)}$ using only predicates of arity smaller or equal to a given bound. By *bounded-variable FOL*$^{(=)}$ we denote $\mathrm{FOL}^{(=)}$ using only a bounded number of variables. For uniformity, we use the typical DL notation also for first-order interpretations and we refer to $\mathrm{FOL}^{(=)}$ sentences as ($\mathrm{FOL}^{(=)}$) axioms and to $\mathrm{FOL}^{(=)}$ as ($\mathrm{FOL}^{(=)}$) knowledge bases. We recall that virtually all mainstream DLs (including $\mathcal{SROIQ}$) can be expressed in bounded-arity $\mathrm{FOL}^=$. We also recall that Datalog denotes the function-free first-order Horn clauses.

## 3 Fixed-Domain Semantics

In this paper, we investigate the effects of fixing the domain size of models. Given some $s \in \mathbb{N}$, we say a knowledge base $\mathcal{K}$ is *$s$-satisfiable*, if it has a model $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ with $|\Delta^{\mathcal{I}}| = s$ (referred to as an *$s$-model*) and an axiom $\varphi$ is *$s$-entailed* by $\mathcal{K}$ (written: $\mathcal{K} \models_s \varphi$) if every $s$-model of $\mathcal{K}$ is a model of $\varphi$. We define *$s$-subsumption* accordingly.

Depending on how we treat the size parameter $s$, we distinguish three versions of the fixed-domain entailment decision problem (where, the function $||\cdot||$ determines the size of a knowledge base or axiom according to some usual encoding):

- assuming $s$ fixed, the input is $(\mathcal{K}, \varphi)$; the problem size is $||\mathcal{K}||+||\varphi||$
- the input is $(\mathcal{K}, \varphi, s)$ with $s$ in unary; the problem size is $||\mathcal{K}||+||\varphi||+s$
- the input is $(\mathcal{K}, \varphi, s)$ with $s$ in binary; the problem size is $||\mathcal{K}||+||\varphi||+\lceil \log_2 s \rceil$

In all cases, the output is "yes" if $\mathcal{K} \models_s \varphi$ and "no" otherwise. We define the three versions of the fixed-domain satisfiability problem accordingly.

## 4 Fixed and Unary Encoding

### 4.1 Variants of First-Order Logic

We first consider the case of unrestricted first-order logic, rectifying an incorrect result from the literature. Without giving a proof, Gaggl et al. [8] claim that this problem is PSPACE-complete, sketching an argument based on the assumption that any FOL model with polynomially many domain elements can be represented in polynomial space, which, however, only holds when the maximum predicate arity is bounded (see our results below). In fact, we can show that even for domain size 2, the problem is NEXPTIME-hard (even when no constants are used). We use a reduction from the TILING-problem, as it can be found in [12], which is known to be NEXPTIME-hard.

**Definition 1** **($n \times n$ Tiling Problem).** *Given a set of square tile types $T = \{t_0, \ldots, t_k\}$, together with two relations $H, V \subseteq T \times T$ (horizontal and vertical compatibility, respectively), as well as an integer $n$ in binary. An $n \times n$ tiling is a function $f : \{1, \ldots, n\} \times \{1, \ldots, n\} \mapsto T$ such that (a) $f(1,1) = t_0$, and (b) for all $i, j$ $(f(i,j), f(i+1,j)) \in H$, and $(f(i,j), f(i,j+1)) \in V$.* TILING *is the problem of deciding, given $T, H, V$, and $n$, whether an $n \times n$ tiling exists. We refer to given $T$,$H$,$V$, and $n$ as tiling system $\mathfrak{T} = (T, H, V, n)$.*

**Theorem 1.** *The 2-satisfiability problem for constant-free FOL is* NEXPTIME*-hard.*

*Proof.* We provide a reduction from TILING. Let $n$ be the size of the grid. W.l.o.g. we assume $n = 2^m$. We now provide a FOL knowledge base (of size polynomial in $m$) which is satisfiable iff a tiling exists. In the following, $\vec{x}$ stands for the sequence

$x_0, \ldots, x_{m-1}$ of variables (likewise $\vec{y}$ and $\vec{z}$).

$$\exists x, y.1(x) \wedge 0(y) \qquad \forall x.1(x) \leftrightarrow \neg 0(x) \tag{1}$$

$$\forall x, y.same(x, y) \leftrightarrow (1(x) \leftrightarrow 1(y)) \tag{2}$$

$$\forall \vec{x}, \vec{y}.flip_i(\vec{x}, \vec{y}) \leftrightarrow \bigwedge_{j \in \{0 \ldots i-1\}} (1(x_j) \wedge 0(y_j)) \wedge 0(x_i) \wedge 1(y_i) \wedge \bigwedge_{j \in \{i+1 \ldots m-1\}} same(x_j, y_j) \tag{3}$$

$$\forall \vec{x}, \vec{y}.next(\vec{x}, \vec{y}) \leftrightarrow \bigvee_{i \in \{0 \ldots m-1\}} flip_i(\vec{x}, \vec{y}) \tag{4}$$

$$\forall \vec{x}, \vec{y}.\neg(t_i(\vec{x}, \vec{y}) \wedge t_j(\vec{x}, \vec{y})) \tag{5}$$

$$\forall \vec{x}, \vec{y}. \bigwedge_{i \in \{0 \ldots m-1\}} (0(x_i) \wedge 0(y_i)) \rightarrow t_0(\vec{x}, \vec{y}) \tag{6}$$

$$\forall \vec{x}, \vec{y}, \vec{z}.next(\vec{x}, \vec{y}) \rightarrow \bigvee_{(t_i, t_j) \in H} t_i(\vec{x}, \vec{z}) \wedge t_j(\vec{y}, \vec{z}) \wedge \bigvee_{(t_i, t_j) \in V} t_i(\vec{z}, \vec{x}) \wedge t_j(\vec{z}, \vec{y}) \tag{7}$$

With this knowledge base and a domain of size two, we use the two elements as $0$ and $1$ to encode the coordinates in the grid $(0 \ldots 2^m - 1)$ in binary, using $m$ positions in the predicate for each coordinate. With this encoding, the $next$ predicate is axiomatized to contain any pair of consecutive $m$-bit numbers. Then the $t_i$ predicates indicate the coordinate pairs of grid positions where the tile $t_i$ is positioned. □

A matching upper bound will be provided through Theorem 4. The proof of the preceding theorem suggests that an unbounded predicate arity is essential for the result. Indeed, when bounding the maximal arity, the complexity can be shown to be PSPACE-complete.

**Theorem 2.** *The fixed-domain satisfiability problem for bounded-arity FOL$^{(=)}$ is* PSPACE*-complete when the domain size is fixed or given unary.*

*Proof.* We show PSPACE-hardness of FOL satisfiability for domain size 2 by providing a reduction from the validity problem of quantified Boolean formulae (QBFs). We recap that for any QBF, it is possible to construct in polynomial time an equivalent QBF that has the specific shape $\psi = Q_1 x_1 Q_2 x_2 \ldots Q_n x_n \varphi$ with $Q_1, \ldots Q_n \in \{\exists, \forall\}$ and $\varphi$ being a propositional formula over the propositional variables $x_1, \ldots, x_n$. Let the first-order sentence $\psi'$ be obtained from $\psi$ by replacing every occurrence of a propositional variable $x_i$ by $true(x_i)$ (thus reinterpreting the propositional variables as first-order variables and introducing $true$ as the only unary predicate). It is now easy to see that $\psi$ is true iff $\psi' \wedge \exists x.true(x) \wedge \exists x.\neg true(x)$ has a model with two elements.

We show PSPACE membership of FOL$^{(=)}$ satisfiability checking for domain size given in unary encoding by providing a PSPACE decision procedure for a given $\mathcal{K}$ and domain size $s$. Let $k = ||\mathcal{K}||$ and $\ell = k + s$. Let $a$ be the upper bound on the arity. There can be at most $k$ predicates, hence the size needed to represent a model is upper-bounded by $k \cdot s^a \leq \ell \cdot \ell^a$, i.e., polynomial. Then we guess a polynomial size model representation and verify it in PSPACE [17]. This gives an NPSPACE algorithm which by Savitch's theorem [16] can be turned into a PSPACE one. □

Again, inspecting the previous proof, it seems to be essential that the number of used variables is unbounded. The subsequent theorem confirms that bounding the number of variables will lead to NP-membership.

**Theorem 3.** *The fixed-domain satisfiability problem for bounded-variable FOL$^{=}$ is in* NP *when the domain size is given in unary encoding.*

*Proof.* Let $v$ be the upper bound on the number of variables. Let $\mathcal{K}$ be a FOL knowledge base with at most $v$ variables, let $s$ be the prescribed domain size and let $\ell = ||\mathcal{K}|| + s$. We now describe a nondeterministic polytime procedure for checking $s$-satisfiability of $\mathcal{K}$. Let $\varphi = \bigwedge_{\alpha \in \mathcal{K}} \alpha$ and let $\mathcal{I}$ be a model of $\varphi$ (and, hence, of $\mathcal{K}$) with $|\Delta^{\mathcal{I}}| = s$. We guess $c^{\mathcal{I}}$ for every free constant $c$ occurring in $\mathcal{K}$. We also guess for every subformula $\psi$ of $\varphi$ the set $\mathcal{Z}_\psi$ of variable assignments $\nu : \mathsf{freevars}(\varphi) \to \Delta^{\mathcal{I}}$ for which $\mathcal{I}, \nu \models \psi$. We determine an upper bound for the size of the guessed information: $\varphi$ has not more than $2\ell$ subformulae each of which has maximally $v$ free variables, hence the size to store all $\mathcal{Z}_\psi$ is bounded by $2\ell \cdot s^v$ and hence by $2\ell \cdot \ell^v$, i.e., polynomial in the input.

Verifying that the guessed information indeed satisfies the abovementioned property requires four checks: first, $\mathcal{Z}_\varphi$ must contain the empty function. Second, every $\mathcal{Z}_\psi$ must be compatible with the variable assignments of $\psi$'s subformulae in the following way (where $\delta$ ranges over $\Delta^{\mathcal{I}}$):

$$\begin{aligned}
\mathcal{Z}_{\neg\psi} &= (\Delta^{\mathcal{I}})^{\mathsf{freevars}(\psi)} \setminus \mathcal{Z}_\psi \\
\mathcal{Z}_{\psi_1 \wedge \psi_2} &= \{\nu \in (\Delta^{\mathcal{I}})^{\mathsf{freevars}(\psi_1 \wedge \psi_2)} \mid \mathcal{Z}_{\psi_1} \subseteq \nu \text{ and } \mathcal{Z}_{\psi_2} \subseteq \nu\} \\
\mathcal{Z}_{\psi_1 \vee \psi_2} &= \{\nu \in (\Delta^{\mathcal{I}})^{\mathsf{freevars}(\psi_1 \vee \psi_2)} \mid \mathcal{Z}_{\psi_1} \subseteq \nu \text{ or } \mathcal{Z}_{\psi_2} \subseteq \nu\} \\
\mathcal{Z}_{\exists x.\psi} &= \{\nu \in (\Delta^{\mathcal{I}})^{\mathsf{freevars}(\exists x.\psi)} \mid \exists \delta : \nu \cup \{(x, \delta)\} \in \mathcal{Z}_\psi\} \\
\mathcal{Z}_{\forall x.\psi} &= \{\nu \in (\Delta^{\mathcal{I}})^{\mathsf{freevars}(\forall x.\psi)} \mid \forall \delta : \nu \cup \{(x, \delta)\} \in \mathcal{Z}_\psi\}
\end{aligned}$$

Third, for any atomic subformula $\psi$ which is an equality atom $t_1 \doteq t_2$, we need to check that $\nu \in \mathcal{Z}_\psi$ exactly if $\nu^*(t_1) = \nu^*(t_2)$ where, given a $\nu$, we let $\nu^*$ denote the extension of $\nu$ to arbitrary terms, mapping constants $c$ to $c^{\mathcal{I}}$ (as guessed before). Fourth, for any two atomic subformulae $\psi_1 = p(t_1, \ldots, t_k)$ and $\psi_2 = p(t'_1, \ldots, t'_k)$, referring to the same predicate $p$, we need to check that they do not contradict each other w.r.t. any $k$-tuple being or not being in $p^{\mathcal{I}}$. This is checked by verifying if $\{(\nu^*(t_1), \ldots, \nu^*(t_k) \mid \nu \in \mathcal{Z}_{\psi_1})\} \cap \{(\nu^*(t'_1), \ldots, \nu^*(t'_k) \mid \nu \in (\Delta^{\mathcal{I}})^{\mathsf{freevars}(\psi)} \setminus \mathcal{Z}_{\psi_2})\} = \emptyset$, with $\nu^*$ defined as before. Note that the sets in this condition do only have polynomially many elements. All these checks can be done in polynomial time. Hence we obtain an NP upper bound for checking satisfiability. □

### 4.2 Upper and Lower Bounds for DLs

We recall from Gaggl et al. [8] that fixed-domain standard reasoning in $\mathcal{SROIQ}$ with unary encoding is in NP. Note that this result is not subsumed by Theorem 3 since encoding number restrictions may require an unbounded number of variables. For one lower bound, we recall that 3-satisfiability is already NP-hard for $\mathsf{DL}_{\mathsf{min}}$ [8], a very minimalistic DL that is subsumed by all tractable profiles of OWL. To also provide a lower bound for logics without ABox, we add a hardness result for terminologies.

**Proposition 1.** *Deciding* 3*-subsumption in* $\mathcal{EL}$ *terminologies is* coNP*-hard.*

*Proof.* We provide a reduction from the 3-colorability problem to non-3-subsumption. For a graph $(V, E)$ with $V = \{v_1, \ldots, v_n\}$, we introduce a concept name $A_{v_i}$ for every vertex and one distinguished concept name *Clash*. Then we let the terminology $\mathcal{K}$ consist of the axioms $A_{v_i} \sqcap A_{v_j} \sqsubseteq \mathit{Clash}$ for every $\{v_i, v_j\} \in E$. Then the graph is not 3-colorable iff $\mathcal{K} \models_3 \exists r.A_{v_1} \sqcap ... \sqcap \exists r.A_{v_n} \sqsubseteq \exists r.\mathit{Clash}$. □

# 5 Binary Encoding

## 5.1 NExpTime Upper Bound for FOL

**Theorem 4.** *The fixed-domain satisfiability problem for FOL*$^=$ *with the domain size given in binary is in* NExpTime*.*

*Proof.* Let $\mathcal{K}$ be a FOL knowledge base, $s$ the prescribed domain size and let $\ell = ||\mathcal{K}|| + \lceil \log_2 s \rceil$. We now describe a nondeterministic exponential time procedure to check $s$-satisfiability of $\mathcal{K}$. Let $\varphi = \bigwedge_{\alpha \in \mathcal{K}} \alpha$. We guess a model $\mathcal{I}$ of $\varphi$ (and, hence, of $\mathcal{K}$) with $|\Delta^\mathcal{I}| = s$ and for every subformula $\psi$ of $\varphi$ we guess the set $\mathcal{Z}_\psi$ of variable assignments $\nu : \mathsf{freevars}(\varphi) \to \Delta^\mathcal{I}$ for which $\mathcal{I}, \nu \models \psi$.

We determine an upper bound for the size of the guessed information: $\varphi$ can contain at most $\ell$ different predicates and $\ell$ is also an upper bound for the arity of the predicates used in $\varphi$. Therefore, the size to store $\mathcal{I}$ is bounded by $\ell \cdot s^\ell$ and hence by $\ell \cdot (2^\ell)^\ell = \ell \cdot 2^{\ell^2}$. $\varphi$ has not more than $2\ell$ subformulae each of which has maximally $\ell$ free variables, hence the size to store all $\mathcal{Z}_\psi$ is bounded by $2\ell \cdot s^\ell$ and hence by $2\ell \cdot (2^\ell)^\ell = 2\ell \cdot 2^{\ell^2}$. Verifying the claimed properties of $\mathcal{I}$ and all $\mathcal{Z}_\psi$ then can be done in polynomial time w.r.t. the exponential size input. Hence we obtain a NExpTime upper bound for checking satisfiability. □

This result subsumes NExpTime membership of fixed-domain satisfiability in all mainstream description logics. Also, by reducibility to FOL satisfiability checking, it follows that axiom entailment, conjunctive query entailment and even entailment of arbitrary Datalog queries (subsuming all kinds of navigational queries) is in co-NExpTime.

## 5.2 NExpTime Lower Bound for $\mathcal{ELI}$ and $\mathcal{ELF}$

We show co-NExpTime hardness for subsumption in $\mathcal{ELI}$ and $\mathcal{ELF}$ terminologies under the fixed-domain semantics in binary encoding. Note that for both logics, the problem is ExpTime-complete under the classical or finite-model semantics [10]. We show that constraining the domain size allows for encoding tiling problems. Similar constructions for much more expressive DLs have been described before [4, 18].

**Reducing Tiling Problems to $\mathcal{ELI}$ Subsumption** Given $\mathfrak{T} = (T, H, V, n)$ (cf. Section 4.1), we construct an $\mathcal{ELI}$ terminology $\mathcal{K}_\mathfrak{T}$ such that every model of $\mathcal{K}_\mathfrak{T}$ not satisfying a certain subsumption (called countermodel) represents a tiling. For simplicity, we assume $n = 2^m$. The countermodels we axiomatize shall consist of two types of domain elements: elements corresponding to grid positions and elements representing tile types. The former will be endowed with their x- and y-coordinates in binary representation, using concept names $X_i^z, Y_i^z$, with $(0 \leq i < m)$ and $z \in \{0,1\}$ to encode the each of the $m$ bits of each coordinate. By means of the following axioms, we axiomatize the $n \times n$ grid:

$$\exists h^-.(X_j^0 \sqcap X_i^0) \sqsubseteq X_i^0 \quad \exists h^-.(X_j^0 \sqcap X_i^1) \sqsubseteq X_i^1 (0 \leq j < i) \tag{8}$$

$$\exists h^-.((X_0^1 \sqcap \ldots \sqcap X_{i-1}^1) \sqcap X_i^0) \sqsubseteq X_i^1 \tag{9}$$

$$\exists h^-.((X_0^1 \sqcap \ldots \sqcap X_{i-1}^1) \sqcap X_i^1) \sqsubseteq X_i^0 \tag{10}$$

$$\exists v^-.X_i^z \sqsubseteq X_i^z \qquad X_i^z \sqsubseteq \mathit{Grid} \qquad X_i^0 \sqsubseteq \exists h.\mathit{Grid} \tag{11}$$

$$X_i^0 \sqcap X_i^1 \sqsubseteq C_\bot \qquad \exists h.C_\bot \sqsubseteq C_\bot \tag{12}$$

$$\mathit{Origin} \sqsubseteq X_0^0 \sqcap \ldots \sqcap X_{m-1}^0 \sqcap Y_0^0 \sqcap \ldots \sqcap Y_{m-1}^0 \tag{13}$$

with $0 \leq i < m$ and $z \in \{0,1\}$. Likewise, we let $\mathcal{K}_\mathfrak{T}$ contain axioms obtained from axioms (8–12) where the $X_i^z$ are replaced by $Y_i^z$ and the roles $v$ and $h$ are swapped. Axioms in (8) ensure that, the value of the $i^{\text{th}}$ bit of the x-coordinate does not change when going in horizontal direction, if some preceding bits are set to low. Correspondingly, Axioms (9–10) ensure, that the $i^{\text{th}}$ bit changes its value, if all preceding bits are set to high. The axioms in (11) enforce that there is an $h$-successor, as long as one of the $X_i^z$ bits is still set to low, thus stopping after $2^m$ consecutive $h$-successors. Naturally, a bit can only be set to one value which is reflected in the axioms in (12).[2] Then, instances of the first axiom in (11) merely ensure that $X_i^z$ bit values remain unchanged when moving vertically. Finally, in (13) we use $\mathit{Origin}$ to refer to the grid origin.

We now turn to the domain elements representing the tile types. We will make sure that every countermodel contains one element per tile type and that every grid element is associated with one tile type via the $\mathit{tiledBy}$ role. Regarding the tiling conditions in $T, H, V$, the following axioms are used:

$$\mathit{Tile}_i \sqsubseteq \mathit{Tile} \quad (0 \leq i \leq k) \quad \mathit{Tile}_i \sqcap \mathit{Tile}_j \sqsubseteq C_\bot \quad (0 \leq i < j \leq k) \tag{14}$$

$$\mathit{Origin} \sqsubseteq \exists \mathit{tiledBy}.\mathit{Tile}_0 \tag{15}$$

$$\mathit{Grid} \sqsubseteq \exists \mathit{tiledBy}.\mathit{Tile} \quad \mathit{Grid} \sqcap \mathit{Tile} \sqsubseteq C_\bot \tag{16}$$

$$\mathit{Origin} \sqsubseteq \exists \mathit{req}.\mathit{Tile}_1 \sqcap \ldots \sqcap \exists \mathit{req}.\mathit{Tile}_k \tag{17}$$

$$\exists \mathit{tiledBy}.\mathit{Tile}_i \sqcap \exists h.\exists \mathit{tiledBy}.\mathit{Tile}_j \sqsubseteq C_\bot \tag{18}$$

$$\exists \mathit{tiledBy}.\mathit{Tile}_i \sqcap \exists v.\exists \mathit{tiledBy}.\mathit{Tile}_j \sqsubseteq C_\bot \tag{19}$$

where for each $(t_i, t_j) \notin H$ and $(t_i, t_j) \notin V$ we find an instance of (18) or (19), respectively. Axiom (15) encodes the initial tiling condition, whereas (17) enforces the existence of $\mathit{Tile}_i$ instances whenever $\mathit{Origin}$ is nonempty.

**Lemma 1.** *For given $\mathfrak{T} = (T, H, V, n)$, let $\mathcal{K}_\mathfrak{T}$ be the $\mathcal{ELI}$ terminology described above, and let $s = n^2 + k + 1$. Then $\mathcal{K}_\mathfrak{T} \not\models_s \mathit{Origin} \sqsubseteq C_\bot$ iff $\mathfrak{T}$ has a tiling.*

[2] Disjointness ($A \sqsubseteq \neg B$) of concepts $A, B$ are modeled in $\mathcal{ELI}$ as $A \sqcap B \sqsubseteq C_\bot$, where $C_\bot$ is a freshly introduced concept name that acts as the bottom concept in countermodels [1].

*Proof.* (⇒) Recall that $n = 2^m$. Let $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, with $|\Delta^{\mathcal{I}}| = 2^{2m} + k + 1 = s$, be the countermodel for the subsumption $\mathit{Origin} \sqsubseteq C_\perp$, i.e., there is a $\delta \in \Delta^{\mathcal{I}}$, such that $\delta \in \mathit{Origin}^{\mathcal{I}}$, but $\delta \notin {C_\perp}^{\mathcal{I}}$. Moreover, since $\mathcal{I} \models_s \mathcal{K}_{\mathfrak{T}}$, we know that there are elements $\tau_0, \ldots, \tau_k \in \Delta^{\mathcal{I}}$, with $\tau_i \in {\mathit{Tile}_i}^{\mathcal{I}}$, and in particular $(\delta, \tau_0) \in \mathit{tiledBy}^{\mathcal{I}}$ satisfying the initial tiling condition. Now given $x, y \in \{1, \ldots, 2^m\}$, let $x_0 x_2 \ldots x_{m-1}$ and $y_0 y_2 \ldots y_{m-1}$ be the binary representations of $x - 1$ and $y - 1$, respectively. Then we let $C_{x,y}$ denote the shorthand notation for the concept: $C_{x,y} \equiv X_0^{x_i} \sqcap \ldots \sqcap X_{m-1}^{x_i} \sqcap Y_0^{y_i} \sqcap \ldots \sqcap Y_{m-1}^{y_i}$. Axiom (13) then ensures $\delta \in C_{1,1}^{\mathcal{I}}$. It follows from Axiom (11) that, $(\delta, \delta') \in v^{\mathcal{I}}$, $(\delta, \delta'') \in h^{\mathcal{I}}$ for some $\delta', \delta'' \in \Delta^{\mathcal{I}}$, where $\delta' \in C_{1,2}^{\mathcal{I}}$ and $\delta'' \in C_{2,1}^{\mathcal{I}}$ due to axioms (8–10). Further, there must be $(\delta', \gamma') \in h^{\mathcal{I}}$ and $(\delta'', \gamma'') \in v^{\mathcal{I}}$, with $\gamma', \gamma'' \in C_{2,2}^{\mathcal{I}}$. In the same vein, by induction on $x$ and $y$ it follows that, for each possible $(x, y)$, $|C_{x,y}^{\mathcal{I}}| \geq 1$, and for every $x > 1$, at least one $\beta \in C_{x,y}^{\mathcal{I}}$ has an incoming $h$-role from some $\beta' \in C_{x-1,y}^{\mathcal{I}}$, just as for every $y > 1$, one $\beta \in C_{x,y}^{\mathcal{I}}$ has an incoming $v$-role from some $\beta' \in C_{x,y-1}^{\mathcal{I}}$. Note that no element on the binary tree thus created can be in $C_{x,y}^{\mathcal{I}}$ and $C_{x',y'}^{\mathcal{I}}$ at the same time for $(x, y) \neq (x', y')$ since any $\gamma \in (C_{x,y} \sqcap C_{x',y'})^{\mathcal{I}}$ would also satisfy $\gamma \in C_\perp^{\mathcal{I}}$ leading to $\delta \in C_\perp^{\mathcal{I}}$, contradicting our assumption.

Let now $s' = \sum_{x,y} |C_{x,y}^{\mathcal{I}}|$, and assume $|C_{x,y}^{\mathcal{I}}| > 1$, i.e., there are several elements carrying the same coordinate. Recall that $k + 1$ distinct domain elements are required for the tiles, but then $s - (k+1) = 2^{2m} < s'$. This contradicts the assumption, therefore $|C_{x,y}^{\mathcal{I}}| = 1$ for all $x$ and $y$, effectively leading to all elements of $\mathit{Grid}$ forming an $n \times n$ grid with $h$ and $v$ encoding horizontal and vertical neighbourhood, respectively. Axioms (16) and (18–19) then ensure that the assignment of tiles to grid positions satisfies the horizontal and vertical compatibility constraints of $H$ and $V$, respectively.

(⇐) By the arguments above it is immediate that from every correct tiling, a countermodel for the subsumption $\mathit{Origin} \sqsubseteq C_\perp$ can be extracted. □

We want to emphasize that the imposed domain size is crucial for a) enforcing a grid of exponential size, and b) for exploiting the non-deterministic choice in tile assignments.

**Theorem 5.** *Subsumption in $\mathcal{ELI}$ under the fixed-domain semantics with binary encoding is* CO-NEXPTIME*-hard.*

*Proof.* Note that for a given $\mathfrak{T} = (T, H, V, 2^m)$, the corresponding $\mathcal{ELI}$ terminology $\mathcal{K}_{\mathfrak{T}}$ is of polynomial size in $m$. From Lemma 1 it then follows that, subsumption in $\mathcal{ELI}$ is CO-NEXPTIME-hard. □

We finish the section by showing the same complexity for $\mathcal{ELF}$ by virtue of a small adaptation of the above argument.

**Theorem 6.** *Subsumption in $\mathcal{ELF}$ under the fixed-domain semantics with binary encoding is* CO-NEXPTIME*-hard.*

*Proof.* We reuse the construction made for $\mathcal{ELI}$ with the following modification: for $r \in \{h, v\}$ we add the axioms $\top \sqsubseteq \leqslant 1\, r.\top$ and we turn every axiom of the shape $\exists r^-.C_1 \sqsubseteq C_2$ into the axiom $C_1 \sqsubseteq \exists r.C_2$. It can be readily checked that the resulting knowledge base is an $\mathcal{ELF}$ terminology. Moreover, the countermodels obtained for $\mathcal{ELI}$ satisfy the functionality restriction imposed. Finally, in the presence of functionality of $r$, $C_1 \sqsubseteq \exists r.C_2$ entails $\exists r^-.C_1 \sqsubseteq C_2$, hence all the arguments in Lemma 1 carry over to this case. □

### 5.3 Logics Below NExpTime

We recall that even for a domain of fixed size and not part of the input, standard reasoning is already NP-hard for $\mathsf{DL_{min}}$ knowledge bases and $\mathcal{EL}$ terminologies (cf. Section 4.1, Theorem 1). Obviously these hardness results carry over to the unary and binary encoding case and to any logic subsuming any of the two. We now show that a generic property that is shared by many tractable DLs ensures NP-membership of standard reasoning tasks with domain size given in binary. We start with some model-theoretic considerations.

**Definition 2.** *We call a model* nontrivial *if its domain size is larger than* $1$. *A knowledge base is called* nontrivially satisfiable*, if it has a nontrivial model. A logic* $\mathcal{L}$ *has the* polynomial nontrivial model property *if there is a polynomial function* $p : \mathbb{N} \to \mathbb{N}$ *such that every nontrivially satisfiable* $\mathcal{L}$ *knowledge base of size* $k$ *has a nontrivial model with at most* $p(k)$ *elements.*

This property has been shown to hold for a variety of prominent tractable logics. Among those, the recently introduced role-safety-acyclic Horn-$\mathcal{SHOIQ}$ [6] is rather general and subsumes the tractable profiles OWL QL and OWL RL of the Web Ontology Language. Another logic satisfying this property is $\mathcal{EL}^{++}$, even the version extended by reflexive roles and range restrictions [3, 1] subsuming the third tractable OWL profile OWL EL. Finally, the property holds trivially for Datalog, since there is always a model containing only as many individuals as there are constants.

In the following, we will introduce two kinds of model transformations and state some logics for which modelhood is preserved under these operations.

**Definition 3.** *Let* $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ *and* $\mathcal{J} = (\Delta^{\mathcal{J}}, \cdot^{\mathcal{J}})$ *be interpretations. The product interpretation of* $\mathcal{I}$ *and* $\mathcal{J}$*, denoted* $\mathcal{I} \times \mathcal{J}$ *is the interpretation* $\mathcal{K}$ *with* $\Delta^{\mathcal{K}} = \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$, $a^{\mathcal{K}} = (a^{\mathcal{I}}, a^{\mathcal{J}})$ *for all* $a \in N_I$, $A^{\mathcal{K}} = A^{\mathcal{I}} \times A^{\mathcal{K}}$ *for all* $A \in N_C$*, and* $r^{\mathcal{K}} = \{((\delta, \delta'), (\epsilon, \epsilon')) \mid (\delta, \epsilon) \in r^{\mathcal{I}}, (\delta', \epsilon') \in r^{\mathcal{J}}\}$ *for all* $r \in N_R$.

A very helpful observation is that the classes of models of Horn (description) logics are closed under taking products [7]: given a Horn KB $\mathcal{K}$ and two interpretations $\mathcal{I}$ and $\mathcal{J}$ with $\mathcal{I} \models \mathcal{K}$ and $\mathcal{J} \models \mathcal{K}$, it follows that $\mathcal{I} \times \mathcal{J} \models \mathcal{K}$. The next model transformation that we describe consists in picking one element and "copying" it (as well as all its atomic class memberships and relation to other elements) $n$ times.

**Definition 4.** *Let* $n$ *be a natural number, let* $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ *be an interpretation and let* $\delta \in \Delta^{\mathcal{I}}$. *The* $n$*-fold duplication of* $\delta$ *in* $\mathcal{I}$ *creates an interpretation* $\mathsf{dup}^n(\mathcal{I}, \delta) = \mathcal{J}$ *with* $\Delta^{\mathcal{J}} = (\{0\} \times \Delta^{\mathcal{I}}) \cup (\{1 \dots n\} \times \{\delta\})$ *as well as* $a^{\mathcal{J}} = (0, a^{\mathcal{I}})$ *for all* $a \in N_I$ *and for every predicate* $p$ *of arity* $k$ *holds* $((n_1, \delta_1), \dots (n_k, \delta_k)) \in p^{\mathcal{J}}$ *if* $(\delta_1, \dots, \delta_k) \in p^{\mathcal{I}}$.

**Definition 5.** *We call a logic* $\mathcal{L}$ non-counting*, if modelhood is preserved under arbitrary duplication of anonymous elements (i.e., elements* $\delta \in \Delta^{\mathcal{I}}$ *with* $\delta \neq a^{\mathcal{I}}$ *for all* $a \in N_I$*).*

Note that FOL (without equality) is non-counting, and consequently all mainstream description logics without functionality and cardinality restrictions (that is, all DLs subsumed by $\mathcal{SROI}$) are non-counting as well. Subsequent finding allows us to conclude NP-membership of satisfiability checking for a wide variety of (description) logics.

**Theorem 7.** *Let $\mathcal{L}$ be a non-counting Horn logic with bounded maximal predicate arity satisfying the polynomial nontrivial model property. Then fixed-domain satisfiability checking of $\mathcal{L}$ knowledge bases is in* NP *when using binary encoding.*

*Proof.* We describe a guess-and-check procedure. Let $s$ be the prescribed domain cardinality and $k = ||\mathcal{K}||$ be the size of the knowledge base. Let $p$ be the polynomial as in the definition above. If $s \leq (p(k))^2$, we guess and polytime-verify a model of size $s$ (the guessed model takes polynomial space as $\mathcal{L}$ has bounded arity by assumption). Otherwise, we guess and polytime-verify a nontrivial model $\mathcal{I}$ of some cardinality $\tilde{s} \leq p(k)$.

It remains to show that the existence of $\mathcal{I}$ ensures the existence of a model $\mathcal{J}$ of cardinality $s$. Let $\mathcal{I}' = \mathcal{I} \times \mathcal{I}$. Obviously, $\mathcal{I}'$ has $\tilde{s}^2 \leq (p(k))^2$ elements and is again a model (since $\mathcal{L}$ is a Horn logic by assumption). Also by construction, $\mathcal{I}'$ contains anonymous individuals (namely all elements of the form $(\delta, \epsilon)$ with $\delta \neq \epsilon$, existence guaranteed due to $\mathcal{I}$ being nontrivial). Let $(\delta, \epsilon)$ be one such anonymous individual. We obtain $\mathcal{J}$ by $(s - \tilde{s}^2)$-fold duplication of $(\delta, \epsilon)$. Since $\mathcal{L}$ is non-counting, $\mathcal{J}$ is a model of the knowledge base. □

**Corollary 1.** *Fixed-domain satisfiability checking with binary encoding is in* NP *for the logics: bounded-arity Datalog, role-safety-acyclic (RSA) Horn-$\mathcal{SHOI}$, $\mathcal{EL}$++ with reflexivity and ranges, and all tractable profiles of OWL: OWL EL, OWL QL, and OWL RL.*

**Table 2.** Overview of fixed-domain standard reasoning complexities. Complexities marked with a star have been established in this paper for which pointers to the relevant theorems are given.

| | $s$ **fixed** | Ref. | **unary** | Ref. | **binary** | Ref. | finite |
|---|---|---|---|---|---|---|---|
| $\mathsf{DL}_{\mathsf{min}}$ | NP | [8] | NP | [8] | NP* | Cor. 1 | $AC^0$ |
| OWL QL | NP | [8] | NP | [8] | NP* | Cor. 1 | NL |
| OWL RL | NP | [8] | NP | [8] | NP* | Cor. 1 | P |
| $\mathcal{EL}$ terminologies | NP* | Prop. 1 | NP* | Prop. 1 | NP* | Cor. 1 | P |
| OWL EL | NP | [8] | NP | [8] | NP* | Cor. 1 | P |
| $\mathcal{EL}^{++}$ | NP | [8] | NP | [8] | NP* | Cor. 1 | P |
| RSA Horn-$\mathcal{SHOI}$ | NP | [8] | NP | [8] | NP* | Cor. 1 | P |
| $\mathcal{ELI}/\mathcal{ELF}$ terminologies | NP* | Prop. 1 | NP* | Prop. 1 | NExpTime* | Thm. 4, 5, 6 | ExpTime |
| $\mathcal{ALC}$ | NP | [8] | NP | [8] | NExpTime* | Thm. 4, 5 | ExpTime |
| $\mathcal{SHOIQ}$ | NP | [8] | NP | [8] | NExpTime* | Thm. 4, 5 | NExpTime |
| $\mathcal{SROIQ}$ | NP | [8] | NP | [8] | NExpTime* | Thm. 4, 5 | N2ExpTime |
| bounded-variable $FOL^{(=)}$ | NP* | Thm. 3 | NP* | Thm. 3 | NExpTime* | Thm. 4, 5 | undec. |
| bounded-arity $FOL^{(=)}$ | PSpace* | Thm. 2 | PSpace* | Thm. 2 | NExpTime* | Thm. 4, 5 | undec. |
| $FOL^{(=)}$ | NExpTime* | Thm. 1, 4 | NExpTime* | Thm. 1, 4 | NExpTime* | Thm. 4, 5 | undec. |

## 6 Conclusion

We investigated the complexities of standard reasoning under the fixed-domain semantics for first-order and a large range of description logics. We thereby specifically account for the encoding of the imposed domain size, and distinguish between fixed, unary, and binary. Table 2 summarizes our findings. We obtain quite uniform results of NP-completeness on the full range of description logics for the case of a fixed or unary encoded domain size. Contrariwise, in case of a binary encoding, little expressivity is needed to have standard reasoning jump to NExpTime where it remains for all formalisms subsumed by full first-order logic. Thus, regarding fixed-domain standard reasoning (i.e. satisfiability and non-entailment), we were able to complete the complexity landscape, and leave non-standard reasoning tasks, such as query answering as future work.

## Acknowledgements

This work is supported by DFG in the Research Training Group QuantLA (GRK 1763). We thank Franz Baader for asking the right questions, and are grateful for the valuable feedback from the anonymous reviewers, which helped greatly to improve this work.

## References

1. Baader, F., Brandt, S., Lutz, C.: Pushing the EL Envelope. In IJCAI, pp. 364–369. Professional Book Center (2005)
2. Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P.: The Description Logic Handbook: Theory, Implementation, and Applications. Cambridge University Press, 2nd edn. (2007)
3. Baader, F., Lutz, C., Brandt, S.: Pushing the EL Envelope Further. In OWLED. CEUR Workshop Proceedings, vol. 496. CEUR-WS.org (2008)
4. Baader, F., Sattler, U.: Expressive Number Restrictions in Description Logics. JLC 9(3), 319–350 (1999)
5. Calvanese, D.: Finite Model Reasoning in Description Logics. In DL, pp. 25–36. AAAI Press (1996)
6. Carral, D., Feier, C., Grau, B.C., Hitzler, P., Horrocks, I.: Pushing the boundaries of tractable ontology reasoning. In ISWC 2014, pp. 148–163. Springer (2014)
7. Chang, C., Keisler, H.J.: Model Theory, Studies in Logic and the Foundations of Mathematics, vol. 73. North Holland, 3rd edn. (1990)
8. Gaggl, S.A., Rudolph, S., Schweizer, L.: Fixed-Domain Reasoning for Description Logics. In ECAI, pp. 819 – 827. IOS Press (2016)
9. Horrocks, I., Kutz, O., Sattler, U.: The Even More Irresistible $\mathcal{SROIQ}$. In KR, pp. 57–67. AAAI Press (2006)
10. Krötzsch, M., Rudolph, S., Hitzler, P.: Complexities of Horn Description Logics. TOCL 14(1), 2:1–2:36 (2013).
11. Lutz, C., Sattler, U., Tendera, L.: The Complexity of Finite Model Reasoning in Description Logics. Information and Computation 199(1-2), pp. 132–171 (2005)
12. Papadimitriou, C.H.: Computational complexity. Addison-Wesley (1994)
13. Rosati, R.: Finite Model Reasoning in DL-Lite. In ESWC 2008, p. 215. Springer (2008)
14. Rudolph, S.: Foundations of Description Logics. In RW, pp. 76–136. Springer (2011)
15. Rudolph, S.: Undecidability Results for Database-Inspired Reasoning Problems in Very Expressive Description Logics. In KR, pp. 247–257. AAAI Press (2016)
16. Savitch, W.J.: Relationships Between Nondeterministic and Deterministic Tape Complexities. JCSS 4(2), pp. 177–192 (1970)
17. Stockmeyer, L.J.: The Complexity of Decision Problems in Automata Theory and Logic. Ph.D. thesis, MIT (1974)
18. Tobies, S.: The Complexity of Reasoning with Cardinality Restrictions and Nominals in Expressive Description Logics. JAIR 12, 199–217 (2000)