Generalized Satisfiability for the Description Logic $\mathcal{ALC}$

Arne Meier

Leibniz Universität Hannover, Institut für Theoretische Informatik, Appelstr. 4, 30167 Hannover, Germany

Thomas Schneider*

Universität Bremen, Fachbereich 3, Postfach 330440, 28334 Bremen, Germany

Abstract

The standard reasoning problem, concept satisfiability, in the basic description logic $\mathcal{ALC}$ is PSPACE-complete, and it is EXPTIME-complete in the presence of general concept inclusions. Several fragments of $\mathcal{ALC}$, notably logics in the $\mathcal{FL}$, $\mathcal{EL}$, and DL-Lite families, have an easier satisfiability problem; for some of these logics, satisfiability can be decided in polynomial time. We classify the complexity of the standard variants of the satisfiability problem for all possible Boolean and quantifier fragments of $\mathcal{ALC}$ with and without general concept inclusions.

Keywords: Satisfiability, Ontologies, $\mathcal{ALC}$, Complexity, Post's lattice, Description logic

1. Introduction

Standard reasoning problems of description logics, such as satisfiability or subsumption, have been studied extensively. Depending on the expressivity of the logic, the complexity of reasoning for DLs between fragments of the basic DL $\mathcal{ALC}$ and the OWL 2 standard $\mathcal{SROIQ}$ is between trivial and N2EXPTIME.

*Corresponding author

Email addresses: meier@thi.uni-hannover.de (Arne Meier), tschneider@informatik.uni-bremen.de (Thomas Schneider)

For $\mathcal{ALC}$, concept satisfiability is PSpace-complete [37]. The complexity is the same for concept satisfiability with respect to theories that are acyclic terminologies [10; 16]. Such theories consist of concept inclusions (CIs) where the left-hand side is atomic, representing partial definitions of that term, and no term is allowed to be defined in the theory, directly or indirectly, in terms of itself. In the presence of theories with general CIs, concept satisfiability is ExpTime-complete: the upper bound is due to the correspondence with propositional dynamic logic [22; 34; 39], and the lower bound was proven by Schild [36]. Since the standard reasoning tasks are interreducible, subsumption has the same complexity.

Several fragments of $\mathcal{ALC}$, such as logics in the $\mathcal{FL}$, $\mathcal{EL}$ or DL-Lite families, are well-understood. They usually restrict the use of Boolean operators and of quantifiers, and it is known that their reasoning problems are often easier than for $\mathcal{ALC}$. We now need to distinguish between satisfiability and subsumption because they may be no longer interreducible if certain Boolean operators are missing. Concept subsumption with respect to acyclic and cyclic terminologies, and even with general CIs, is tractable in the logic $\mathcal{EL}$, which allows only conjunctions and existential restrictions [5; 14], and it remains tractable under a variety of extensions such as nominals, concrete domains, role chain inclusions, and domain and range restrictions [6; 8]. Satisfiability for $\mathcal{EL}$, in contrast, is trivial, i.e., every $\mathcal{EL}$-ontology is satisfiable. However, the presence of universal quantifiers usually breaks tractability: subsumption in $\mathcal{FL}_0$, which allows only conjunction and universal restrictions, is coNP-complete [31] and is PSpace-complete with respect to cyclic terminologies [4; 26] and up to ExpTime-complete with general CIs [6; 25]. In [20; 21], concept satisfiability and subsumption for several logics below and above $\mathcal{ALC}$ that extend $\mathcal{FL}_0$ with disjunction, negation and existential restrictions and other features, is shown to be tractable, NP-complete, coNP-complete or PSpace-complete. Subsumption in the presence of general CIs is ExpTime-hard already in fragments of $\mathcal{ALC}$ containing both existential and universal restrictions plus conjunction or disjunction [23], as well as in $\mathcal{AL}$, where only conjunction, universal restrictions and unqualified existential restrictions are allowed [19]. In the original logic DL-Lite, where unqualified existential restrictions, atomic negation on the right-hand side of concept inclusions, as well as inverse and functional roles are allowed, satisfiability of ontologies is tractable [17]. Several extensions of DL-Lite are shown to have tractable, NP-complete, or ExpTime-complete satisfiability problems in [1–3]. The logics in the $\mathcal{EL}$ and DL-Lite families are so important for (medical and database) applications that OWL 2 has two profiles that correspond to logics in these families.

This article revisits restrictions to the Boolean operators in $\mathcal{ALC}$. Instead of looking at one particular subset of $\{\sqcap, \sqcup, \neg\}$, we are considering all possible sets of Boolean operators, and therefore our analysis includes less commonly used operators such as the binary exclusive *or* $\oplus$. Our aim is to find for *every* possible combination of Boolean operators whether it makes satisfiability of the corresponding restriction of $\mathcal{ALC}$ hard or easy. Since each Boolean operator corresponds to a Boolean function—i.e., an n-ary function whose arguments and values are in $\{\mathtt{0}, \mathtt{1}\}$—there are infinitely many sets of Boolean operators that determine fragments of $\mathcal{ALC}$. The complexity classification of the corresponding concept satisfiability problems without theories is known: satisfiability of the basic modal logic, of which $\mathcal{ALC}$ is a notational variant, has been classified in [24] between being PSPACE-complete, coNP-complete, tractable and trivial for all combinations of Boolean operators and quantifiers.

In this article, we classify the concept satisfiability problems with respect to theories for $\mathcal{ALC}$ fragments obtained by arbitrary sets of Boolean operators and quantifiers. We separate these problems into EXPTIME-complete, NP-complete, P-complete and NL-complete. Although this complete classification is a purely theoretical study, it will provide an insight into the use of sub-Boolean $\mathcal{ALC}$ fragments for knowledge representation. This insight is rather negative but partly confirms the use of current sub-Boolean DLs: our analysis will yield that the only two ways of achieving a tractable but non-trivial satisfiability problem with respect to theories are (a) to separate quantifiers *and* conjunction and disjunction at the same time, or (b) to use only negation and no quantifiers. More precisely, our results will imply that the sets $\{\top, \bot, \sqcap, \exists\}$, $\{\top, \bot, \sqcup, \forall\}$, and $\{\top, \bot, \neg\}$ are the maximal sets of operators for which the satisfiability problem with respect to theories is tractable and non-trivial. These sets represent (a well-studied extension of) the logic $\mathcal{EL}$, its dual, and a very restricted Boolean DL. Of these three, $\mathcal{EL}$ is well-established in knowledge representation, and none of the other two operator sets can reasonably be expected to be of real use to ontology modelling. Our study can therefore be seen as a systematic underpinning of the folklore knowledge that logics in the $\mathcal{EL}$ and DL-Lite families are the only reasonably useful sub-Boolean $\mathcal{ALC}$-fragments[1] for which satisfiability in the presence of general

[1]Here, "sub-Boolean $\mathcal{ALC}$-fragment" means "fragment obtained from $\mathcal{ALC}$ by restricting the use of Boolean operators and/or quantifiers". This does not include the limitation to

CIs is tractable. This is a more general statement than that obtained from the results in [6] concerning intractable extensions of $\mathcal{EL}$.

For subsumption, which is not interreducible with satisfiability under restricted Boolean operators, the tractable cases are essentially the same [28].

The tool used in [24] for classifying the infinitely many satisfiability problems of modal logic was Post's lattice [33], which consists of all sets of Boolean functions closed under superposition. These sets directly correspond to all sets of Boolean operators closed under composition. Similar classifications exist for satisfiability for classical propositional logic [27], Linear Temporal Logic [11], hybrid logic [29], and for constraint satisfaction problems [35; 38].

This study extends our previous work in [30] by matching upper and lower bounds and considering restricted use of quantifiers.

2. Preliminaries

Complexity Theory. We assume familiarity with the standard notions of complexity theory as, e.g., defined in [32]. In particular, we will make use of the classes NL, P, NP, coNP, and ExpTime, as well as logspace reductions $\leq_{\mathrm{m}}^{\log}$.

Description Logic. We use the standard syntax and semantics of $\mathcal{ALC}$ [9], with the Boolean operators $\sqcap$, $\sqcup$, $\neg$, $\top$, $\bot$ replaced by arbitrary operators $\circ_f$ that correspond to Boolean functions $f : \{0, 1\}^n \to \{0, 1\}$ of arbitrary arity n. Let $\mathsf{N_C}$, $\mathsf{N_R}$ and $\mathsf{N_I}$ be sets of atomic concepts, roles and individuals. Then the set of *concept descriptions*, for short *concepts*, is defined by

$$C ::= A \mid \circ_f(C, \dots, C) \mid \exists R.C \mid \forall R.C,$$

where $A \in \mathsf{N_C}$, $R \in \mathsf{N_R}$, and $\circ_f$ is a Boolean operator. For a given set B of Boolean operators, a *B-concept* is a concept that uses only operators from B. A *general concept inclusion (GCI)* is an axiom of the form $C \sqsubseteq D$ where C, D are concepts. We use "$C \equiv D$" as the usual syntactic sugar for "$C \sqsubseteq D$ and $D \sqsubseteq C$". A *TBox* is a finite set of GCIs. An *ABox* is a finite set of axioms of the form $C(a)$ or $R(a, b)$, where C is a concept, $R \in \mathsf{N_R}$ and $a, b \in \mathsf{N_I}$. An *ontology* is the union of a TBox and an ABox. This simplified view suffices for our purposes.

---

unqualified existential restriction.

An *interpretation* is a pair $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$, where $\Delta^\mathcal{I}$ is a nonempty set and $\cdot^\mathcal{I}$ is a mapping from $\mathsf{N_C}$ to the power set of $\Delta^\mathcal{I}$, from $\mathsf{N_R}$ to the power set of $\Delta^\mathcal{I} \times \Delta^\mathcal{I}$ and from $\mathsf{N_I}$ to $\Delta^\mathcal{I}$. The mapping $\cdot^\mathcal{I}$ is extended to arbitrary concepts as follows:

$$\begin{aligned}\exists R.C^\mathcal{I} &:= \{x \in \Delta^\mathcal{I} \mid \{y \in C^\mathcal{I} \mid (x,y) \in R^\mathcal{I}\} \neq \emptyset\},\\ \forall R.C^\mathcal{I} &:= \{x \in \Delta^\mathcal{I} \mid \{y \in C^\mathcal{I} \mid (x,y) \notin R^\mathcal{I}\} = \emptyset\},\\ \circ_f(C_1,\ldots,C_n)^\mathcal{I} &:= \{x \in \Delta^\mathcal{I} \mid f(\|x \in C_1^\mathcal{I}\|,\ldots,\|x \in C_n^\mathcal{I}\|) = \mathtt{1}\},\end{aligned}$$

where $\|x \in C^\mathcal{I}\| := \mathtt{1}$ if $x \in C^\mathcal{I}$ and $\|x \in C^\mathcal{I}\| := \mathtt{0}$ if $x \notin C^\mathcal{I}$. Two concepts C_1 and C_2 are *equivalent* if $C_1^\mathcal{I} = C_2^\mathcal{I}$ for every interpretation $\mathcal{I}$.

An interpretation $\mathcal{I}$ *satisfies* the GCI $C \sqsubseteq D$, written $\mathcal{I} \models C \sqsubseteq D$, if $C^\mathcal{I} \subseteq D^\mathcal{I}$. Furthermore, $\mathcal{I}$ satisfies $C(a)$ or $R(a,b)$ if $a^\mathcal{I} \in C^\mathcal{I}$ or $(a^\mathcal{I}, b^\mathcal{I}) \in R^\mathcal{I}$. An interpretation $\mathcal{I}$ satisfies a TBox (ABox, ontology) if it satisfies every axiom therein. It is then called a *model* of this set of axioms.

Let B be a finite set of Boolean operators and $\mathcal{Q} \subseteq \{\exists, \forall\}$. We use $\mathsf{Con}_\mathcal{Q}(B)$, $\mathfrak{T}_\mathcal{Q}(B)$ and $\mathfrak{O}_\mathcal{Q}(B)$ to denote the set of all concepts, TBoxes and ontologies that use operators in B only and quantifiers from $\mathcal{Q}$ only. The following decision problems are of interest for this article.

Concept satisfiability CSAT$_\mathcal{Q}(B)$:
Given a concept $C \in \mathsf{Con}_\mathcal{Q}(B)$, is there an interpretation $\mathcal{I}$ s.t. $C^\mathcal{I} \neq \emptyset$?

TBox satisfiability TSAT$_\mathcal{Q}(B)$:
Given a TBox $\mathcal{T} \in \mathfrak{T}_\mathcal{Q}(B)$, is there an interpretation $\mathcal{I}$ s.t. $\mathcal{I} \models \mathcal{T}$?

TBox-concept satisfiability TCSAT$_\mathcal{Q}(B)$:
Given $\mathcal{T} \in \mathfrak{T}_\mathcal{Q}(B)$ and $C \in \mathsf{Con}_\mathcal{Q}(B)$, is there an $\mathcal{I}$ s.t. $\mathcal{I} \models \mathcal{T}$ and $C^\mathcal{I} \neq \emptyset$?

TBox subsumption TSUBS$_\mathcal{Q}(B)$:
Given $\mathcal{T} \in \mathfrak{T}_\mathcal{Q}(B)$ and $C, D \in \mathsf{Con}_\mathcal{Q}(B)$, is there an $\mathcal{I}$ s.t. $\mathcal{I} \models \mathcal{T}$ and $C^\mathcal{I} \subseteq D^\mathcal{I}$?

Ontology satisfiability OSAT$_\mathcal{Q}(B)$:
Given an ontology $\mathcal{O} \in \mathfrak{O}_\mathcal{Q}(B)$, is there an interpretation $\mathcal{I}$ s.t. $\mathcal{I} \models \mathcal{O}$?

Ontology-concept satisfiability OCSAT$_\mathcal{Q}(B)$:
Given $\mathcal{O} \in \mathfrak{O}_\mathcal{Q}(B)$ and $C \in \mathsf{Con}_\mathcal{Q}(B)$, is there an $\mathcal{I}$ s.t. $\mathcal{I} \models \mathcal{O}$ and $C^\mathcal{I} \neq \emptyset$?

Function symbol	Description	DL operator
0, 1	constant 0, 1	$\bot$, $\top$
and, or	binary conjunction/disjunction $\wedge$, $\vee$	$\sqcap$, $\sqcup$
neg	unary negation $\neg$	$\neg$
xor	binary exclusive *or* $\oplus$	$\boxplus$
andor	$x \wedge (y \vee z)$	
maj	majority	
equiv	binary equivalence function	

Table 1: Boolean functions with description and corresponding DL operator symbol.

We are interested in the complexity of the satisfiability problems. Subsumption will only be considered for certain reductions. By abuse of notation, we will omit set parentheses and commas when stating $\mathcal{Q}$ explicitly, as in $\text{TSAT}_{\exists\forall}(B)$. The satisfiability problems in the list above are interreducible independently of B and $\mathcal{Q}$ in the following way:

$$\text{CSAT}_{\mathcal{Q}}(B) \leq_{\mathrm{m}}^{\log} \text{TCSAT}_{\mathcal{Q}}(B) \text{ and}$$
$$\text{TSAT}_{\mathcal{Q}}(B) \leq_{\mathrm{m}}^{\log} \text{TCSAT}_{\mathcal{Q}}(B) \leq_{\mathrm{m}}^{\log} \text{OSAT}_{\mathcal{Q}}(B) \equiv_{\mathrm{m}}^{\log} \text{OCSAT}_{\mathcal{Q}}(B)$$

as a concept C is satisfiable iff the concept C is satisfiable w.r.t. the empty terminology; a terminology $\mathcal{T}$ (resp. ontology $\mathcal{O}$) is satisfiable iff a fresh atomic concept A is satisfiable w.r.t. $\mathcal{T}$ (resp. $\mathcal{O}$); C is satisfiable w.r.t. $\mathcal{T}$ (resp. $\mathcal{O}$) iff $\mathcal{T} \cup \{C(a)\}$ (resp. $\mathcal{O} \cup \{C(a)\}$) is satisfiable, for a fresh individual a.

Boolean operators. This study is complete with respect to Boolean operators, which correspond to Boolean functions. Table 1 lists all Boolean functions that we will mention, together with the associated DL operator where applicable.

An n-ary Boolean function is called a *projection* (also known as identity function) if $f(x_1, \ldots, x_n) = x_i$ for some $i = 1, \ldots n$. A set of Boolean functions is called a *clone* if it contains all projections and is closed under composition (also known as superposition). The smallest clone I_2 consists just of the projections. The lattice of all clones has been established in [33], see [13] for a more succinct but complete presentation.

Given a finite set B of functions, the smallest clone containing B is denoted by $[B]$. The set B is called a *base* of $[B]$, but $[B]$ often has other bases as well. For example, nesting of binary conjunction yields conjunctions of arbitrary

Clone	Description	Base
BF	all Boolean functions	$\{\mathsf{and}, \mathsf{neg}\}$
R_0, R_1	$\mathsf{0}$-, $\mathsf{1}$-reproducing functions	$\{\mathsf{and}, \mathsf{xor}\}$, $\{\mathsf{or}, \mathsf{equiv}\}$
M	all monotone functions	$\{\mathsf{and}, \mathsf{or}, \mathsf{0}, \mathsf{1}\}$
S_1	$\mathsf{1}$-separating functions	$\{x \text{ and } (\mathsf{neg}(y))\}$
S_{11}	$\mathsf{1}$-separating, monotone functions	$\{\mathsf{andor}, \mathsf{0}\}$
D	self-dual functions	$\{\mathsf{maj}(x, \mathsf{neg}(y), \mathsf{neg}(z))\}$
L	affine functions	$\{\mathsf{xor}, \mathsf{1}\}$
L_0	affine, $\mathsf{0}$-reproducing functions	$\{\mathsf{xor}\}$
L_3	affine, $\mathsf{0}$- and $\mathsf{1}$-reproducing functions	$\{x \text{ xor } y \text{ xor } z \text{ xor } \mathsf{1}\}$
E_0, E	conjunctions and $\mathsf{0}$ (and $\mathsf{1}$)	$\{\mathsf{and}, \mathsf{0}\}$, $\{\mathsf{and}, \mathsf{0}, \mathsf{1}\}$
V_0, V	disjunctions and $\mathsf{0}$ (and $\mathsf{1}$)	$\{\mathsf{or}, \mathsf{0}\}$, $\{\mathsf{or}, \mathsf{0}, \mathsf{1}\}$
N_2, N	negation (and $\mathsf{1}$)	$\{\mathsf{neg}\}$, $\{\mathsf{neg}, \mathsf{1}\}$
I_0, I	$\mathsf{0}$ (and $\mathsf{1}$)	$\{\mathsf{0}\}$, $\{\mathsf{0}, \mathsf{1}\}$

Table 2: List of all clones relevant for this article with their standard bases. Where a base consists of a function expressed by a composition of functions (Lines $\mathsf{S}_1, \mathsf{D}, \mathsf{L}_3$), observe that functions used in this composition are not necessarily available within the corresponding clone: e.g., $\mathsf{neg} \notin \mathsf{S}_1$ although neg is used in the expression that defines the base $\{x \text{ and } (\mathsf{neg}(y))\}$ of S_1 (however, $\mathsf{and} \in \mathsf{S}_1$).

arity. Table 2 lists all clones that we will refer to, using the function names from Table 1 and the following definitions. Let f be an n-ary Boolean function. The *dual* of f, denoted $\mathrm{dual}(f)$, is the n-ary function g with $g(x_1, \ldots, x_n) = \neg f(\neg x_1, \ldots, \neg x_n)$. Further, f is called *self-dual* if $f(x_1, \ldots, x_n) = \mathrm{dual}(f)$, *c-reproducing* if $f(c, \ldots, c) = c$ for $c \in \{\mathsf{0}, \mathsf{1}\}$, and $\mathsf{1}$-*separating* if there is an $1 \leq i \leq n$ such that $b_i = \mathsf{1}$ for each $(b_1, \ldots, b_n) \in f^{-1}(\mathsf{1})$.

From now on, we will equate sets of Boolean operators with their corresponding sets of Boolean functions and use the same notation B and $[B]$ for finite sets B of operators. Closure of a clone under composition implies that the corresponding set of operators is closed under nesting. Consequently, we will denote operator sets with the above clone names. Furthermore, we call a Boolean operator corresponding to a monotone (self-dual, $\mathsf{0}$-reproducing, $\mathsf{1}$-reproducing, $\mathsf{1}$-separating) function a monotone (self-dual, $\bot$-reproducing, $\top$-reproducing, $\top$-separating) operator.

For two sets B, B' of Boolean functions, we say B' *efficiently implements (every function in)* B iff for every function $f(x_1, \ldots, x_n) \in B$ there exists a composition $\phi(g_1, \ldots, g_k)$ of functions $g_i \in B'$ such that $f \equiv \phi$ and no x_i

occurs more than once in the body of ϕ.

Via the inclusion structure of the lattice, lower and upper complexity bounds of problems parameterized by a clone can be carried over to higher and lower clones under certain succinctness conditions as mentioned in Lemma 1. In particular, this means that, if B, B' are finite sets of Boolean functions with $B \subseteq [B']$, and if B' efficiently implements B, then any decision problem restricted to B can be efficiently reduced to the same problem restricted to B': for example, $\text{TS}\textsc{at}_{\mathcal{Q}}(B)$ is reducible to $\text{TS}\textsc{at}_{\mathcal{Q}}(B')$ for any $\mathcal{Q} \subseteq \{\exists, \forall\}$. We will therefore state our results for minimal and maximal clones only, together with those conditions.

The following lemma will help restrict the length of concepts in some of our reductions. It shows that for certain operator sets B, there are always short concepts representing the operators $\sqcap$, $\sqcup$, or $\neg$, respectively. Points (2) and (3) follow directly from the proofs in [27], Point (1) is Lemma 1.4.5 from [38].

Lemma 1. *Let B be a finite set of Boolean operators.*

1. *If $[B]$ contains conjunction (resp. disjunction) and only monotone operators, i.e., $\mathsf{V} \subseteq [B] \subseteq \mathsf{M}$ ($\mathsf{E} \subseteq [B] \subseteq \mathsf{M}$, resp.), then there exists a B-concept C such that C is equivalent to $A_1 \sqcup A_2$ ($A_1 \sqcap A_2$, resp.) and each of the atomic concepts A_1, A_2 occurs exactly once in C.*

2. *If all operators are present, i.e. $[B] = \mathsf{BF}$, then there are B-concepts C and D such that C is equivalent to $A_1 \sqcup A_2$, D is equivalent to $A_1 \sqcap A_2$, and each of the atomic concepts A_1, A_2 occurs in C and D exactly once.*

3. *If $[B]$ contains all unary operators and both constants, i.e. $\mathsf{N} \subseteq [B]$, then there is a B-concept C such that C is equivalent to $\neg A$ and the atomic concept A occurs in C only once.*

Auxiliary results. The following lemmata contain technical results that will be useful to formulate our main results. We use $\star\text{S}\textsc{at}_{\mathcal{Q}}(B)$ to speak about any of the four satisfiability problems $\text{TS}\textsc{at}_{\mathcal{Q}}(B)$, $\text{TCS}\textsc{at}_{\mathcal{Q}}(B)$, $\text{OS}\textsc{at}_{\mathcal{Q}}(B)$ and $\text{OCS}\textsc{at}_{\mathcal{Q}}(B)$ introduced above; the three problems $\text{TCS}\textsc{at}_{\mathcal{Q}}(B)$, $\text{OS}\textsc{at}_{\mathcal{Q}}(B)$ and $\text{OCS}\textsc{at}_{\mathcal{Q}}(B)$ having the power to speak about a single individual will be conflated into $\star\text{S}\textsc{at}^{\text{IND}}_{\mathcal{Q}}(B)$.

Lemma 2 ([30]). *Let B be a finite set of Boolean operators such that $[B]$ contains all unary operators (i.e., $\mathsf{N}_2 \subseteq [B]$) and $\mathcal{Q} \subseteq \{\exists, \forall\}$. Then $\star\text{S}\textsc{at}_{\mathcal{Q}}(B) \equiv^{\log}_{\text{m}} \star\text{S}\textsc{at}_{\mathcal{Q}}(B \cup \{\top, \bot\})$.*

Lemma 3 ([30]). *Let B be a finite set of Boolean operators and $\mathcal{Q} \subseteq \{\exists, \forall\}$. Then* $\mathrm{TCS}\textsc{at}_{\mathcal{Q}}(B) \leq_{\mathrm{m}}^{\log} \mathrm{TS}\textsc{at}_{\mathcal{Q}\cup\{\exists\}}(B \cup \{\top\})$.

Furthermore, we observe that, for each set B of Boolean operators with $\top, \bot \in [B]$, we can simulate the negation of an atomic concept A using a fresh atomic concept A' and role R_A whenever we have access to both quantifiers: if we add the axioms $A \equiv \exists R_A.\top$ and $A' \equiv \forall R_A.\bot$ to any given ontology $\mathcal{O}$, then each model of $\mathcal{O}$ has to interpret A' as the complement of A.

In order to generalize complexity results from $\star\mathrm{S}\textsc{at}_{\mathcal{Q}}(B_1)$ to $\star\mathrm{S}\textsc{at}_{\mathcal{Q}}(B_2)$ for *arbitrary* bases B_2 of $[B_1]$, we need the following lemma.

Lemma 4 ([30]). *Let B_1, B_2 be two sets of Boolean operators such that $[B_1] \subseteq [B_2]$, and let $\mathcal{Q} \subseteq \{\exists, \forall\}$. Then* $\star\mathrm{S}\textsc{at}_{\mathcal{Q}}(B_1) \leq_{\mathrm{m}}^{\log} \star\mathrm{S}\textsc{at}_{\mathcal{Q}}(B_2)$.

PROOF. The proof makes use of DL-circuits, the DL-variant of modal circuits defined in [24]. A DL-circuit over the basis B_1 and quantifier set $\mathcal{Q}$ is a tuple $X = (G, I, E, \alpha, \beta, \mathsf{out})$, where

- (G, E) is a finite directed acyclic graph with G being the set of gates,
- $I \subseteq G$ being the set of input gates
- $\alpha : E \to \mathbb{N}$ is an injective function which defines an ordering on the edges and thereby on the children of a gate,
- $\beta : G \to B_1 \cup \{\exists R, \forall R \mid R \in \mathsf{N_R}\} \cup \mathsf{N_C}$ is a function assigning an operator, quantified role, or atomic concept to every gate such that $\beta(g) \in \mathsf{N_C}$ iff $g \in I$, and
- $\mathsf{out} \in G$, the *output gate*,

and the following conditions are satisfied.

- If $g \in G$ has in-degree 0, then $\beta(g)$ is an atomic concept or one of the constants $\top, \bot$ (if they are in $[B]$).
- If $g \in G$ has in-degree 1, then $\beta(g)$ is a unary operator from B_1 or some $\exists R$ (if $\exists \in \mathcal{Q}$) or $\forall R$ (if $\forall \in \mathcal{Q}$).
- If $g \in G$ has in-degree $d > 1$, then $\beta(g)$ is a d-ary operator from B_1.

The function α is needed to define the order of arguments of non-symmetric functions. The size of a DL-circuit is the number of its gates.

Every concept expression C can straightforwardly be transformed into a DL-circuit of linear size that resembles the ordered tree induced by C. For the backward transformation, an exponential blowup may occur if the circuit is not tree-shaped.

In order to establish the reduction $\star\textsc{Sat}_{\mathcal{Q}}(B_1) \leq_{\mathrm{m}}^{\log} \star\textsc{Sat}_{\mathcal{Q}}(B_2)$, we proceed analogously to [24, Theorem 3.6] and translate, for any given instance of $\star\textsc{Sat}_{\mathcal{Q}}(B_1)$, each concept (hence each side of an axiom) into a DL-circuit X_1 over the basis B_1 and quantifier set $\mathcal{Q}$. This circuit can be easily transformed into a circuit X_2 over the basis B_2 by replacing every $\circ$-gate, with $\circ \in B_1$, with a sub-circuit over B_2. This replacement is possible because of $[B_1] \subseteq [B_2]$, and it causes only linear blowup because the size of the sub-circuits is bounded by a constant. However, since the sub-circuits may not be tree-shaped, we cannot directly transform X_2 back to a concept expression without exponential blowup. Instead, we will express it using new axioms that are constructed in the style of the formulae in [24]:

- For input gates g, we add the axiom $g \equiv x_i$.
- If g is a gate computing the Boolean operator $\circ$ and $h_1, \ldots, h_n$ are the respective predecessor gates in this circuit, we add the axiom $g \equiv \circ(h_1, \ldots, h_n)$.
- For $\exists R$-gates g, we add the axiom $g \equiv \exists R.h$.
- For $\forall R$-gates g, we add the axiom $g \equiv \forall R.h$.

For each axiom $C \sqsubseteq D$, let out_C and out_D be the output gates of the appropriate circuits. Then we need to add one new axiom $\mathsf{out}_C \sqsubseteq \mathsf{out}_D$ to express the axiom $C \sqsubseteq D$. For a concept C in the input (relevant for the problems $\mathrm{TC}\textsc{Sat}_{\mathcal{Q}}, \mathrm{OC}\textsc{Sat}_{\mathcal{Q}}$), its translation is mapped to the respective output gate out_C.

This reduction is computable in logarithmic space and its correctness can be shown in the same way as in the proof of [24, Theorem 3.6]. □

Note that this reduction does not generally hold for other logics, e.g., propositional logic or modal logic. In our case, the reduction is possible because implication and conjunction are inherently available in terminologies. They make it possible to describe the transformed circuit, which may be a more succinct encoding of an exponentially large concept expression, with linearly many axioms. This enables us to obtain the base independence statement of Lemma 4, which is somewhat exceptional and relies on the expressive power of TBoxes.

The idea for the following lemma goes back to Lewis [27].

Lemma 5 (Lewis Trick). *Let B be a set of Boolean operators and $\mathcal{Q} \subseteq \{\forall, \exists\}$. Then* $\mathrm{TS}\textsc{at}_{\mathcal{Q}}(B \cup \{\top\}) \leq_{\mathrm{m}}^{\log} \mathrm{TCS}\textsc{at}_{\mathcal{Q}}(B)$.

PROOF. Let $\mathrm{SC}(\mathcal{T})$ be the set of all (sub-)concepts occurring in $\mathcal{T}$. For every $C \in \mathrm{SC}(\mathcal{T})$, we use C_T to denote C with all occurrences of $\top$ replaced by T. Now let

$$\mathcal{T}' := \{C_T \sqsubseteq D_T \mid C \sqsubseteq D \in \mathcal{T}\} \cup \{C_T \sqsubseteq T \mid C \in \mathrm{SC}(\mathcal{T})\}\,.$$

We claim:

$$\mathcal{T} \in \mathrm{TS}\textsc{at}_{\mathcal{Q}}(B) \iff (\mathcal{T}', T) \in \mathrm{TCS}\textsc{at}_{\mathcal{Q}}(B)$$

For the direction "$\Rightarrow$" observe that for any interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ with $\mathcal{I} \models \mathcal{T}$, we can set $T^{\mathcal{I}} := \Delta^{\mathcal{I}}$ and then have $\mathcal{I} \models \mathcal{T}'$ and obviously $T^{\mathcal{I}} \neq \emptyset$.

Now consider the opposite direction "$\Leftarrow$". Let $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ be an interpretation with $\mathcal{I} \models \mathcal{T}'$ and $T^{\mathcal{I}} \neq \emptyset$. We construct $\mathcal{J}$ from $\mathcal{I}$ via restriction to $T^{\mathcal{I}}$, i.e., $\Delta^{\mathcal{J}} := T^{\mathcal{I}}$, $A^{\mathcal{J}} := A^{\mathcal{I}} \cap T^{\mathcal{I}}$ for atomic concepts A, and $R^{\mathcal{J}} := R^{\mathcal{I}} \cap (T^{\mathcal{I}} \times T^{\mathcal{I}})$ for roles R. We claim the following:

Claim. For every individual $x \in T^{\mathcal{I}}$ and every (sub-)concept C occurring in $\mathcal{T}$, it holds that $x \in C_T^{\mathcal{I}}$ if and only if $x \in C^{\mathcal{J}}$.

The claim implies $\mathcal{J} \models \mathcal{T}$. Indeed, consider an axiom $D \sqsubseteq E \in \mathcal{T}$ and any $x \in D^{\mathcal{J}}$. We clearly have $x \in \Delta^{\mathcal{J}} = T^{\mathcal{I}}$ and so, by the claim, $x \in D_T^{\mathcal{I}}$. Then $\mathcal{I} \models \mathcal{T}'$ implies $x \in E_T^{\mathcal{I}}$, whence, by the claim, $x \in E^{\mathcal{J}}$.

Proof of Claim. We proceed by induction on the structure of C. The base case includes atomic C as well as $\top$ and $\bot$, and follows from the construction of $\mathcal{J}$.

For the induction step, we consider the following cases.

- In case $C = \circ_f(C^1, \ldots, C^n)$, where $\circ_f$ is an arbitrary n-ary boolean operator corresponding to an n-ary Boolean function f, and the C^i are smaller subconcepts of C, the following holds.

$$\begin{array}{llll} x \in C_T^{\mathcal{I}} & \text{iff} & f(\|x \in (C_T^1)^{\mathcal{I}}\|, \ldots, \|x \in (C_T^n)^{\mathcal{I}}\|) = 1 & \text{def. satisfaction} \\ & \text{iff} & f(\|x \in (C^1)^{\mathcal{J}}\|, \ldots, \|x \in (C^n)^{\mathcal{J}}\|) = 1 & \text{induction hyp.} \\ & \text{iff} & x \in C^{\mathcal{J}} & \text{def. satisfaction} \end{array}$$

- In case $C = \exists R.D$, the following holds.

$$\begin{array}{lll} x \in C_T^{\mathcal{I}} & \text{iff} & \text{for some } y \in \Delta^{\mathcal{I}} : (x,y) \in R^{\mathcal{I}} \text{ and } y \in D_T^{\mathcal{I}} \\ & \text{iff} & \text{for some } y \in T^{\mathcal{I}} : (x,y) \in R^{\mathcal{I}} \text{ and } y \in D_T^{\mathcal{I}} \\ & \text{iff} & \text{for some } y \in T^{\mathcal{I}} : (x,y) \in R^{\mathcal{J}} \text{ and } y \in D^{\mathcal{J}} \\ & \text{iff} & x \in C^{\mathcal{J}} \end{array}$$

  The first equivalence is due to the definition of satisfaction. The second's "$\Rightarrow$" direction is due to the additional axiom $D_T \sqsubseteq T$ in $\mathcal{T}'$, while the "$\Leftarrow$" direction is obvious. The third equivalence is again due to the definition of satisfaction and the construction $\Delta^{\mathcal{J}} = T^{\mathcal{I}}$.

- Case $C = \forall R.D$ follows from the previous case.

□

Lemma 6 (Contraposition). *Let B be a finite set of Boolean operators and $\mathcal{Q} \subseteq \{\exists, \forall\}$. Then*

1. $\text{TSAT}_{\mathcal{Q}}(B) \leq_{\text{m}}^{\log} \text{TSAT}_{\text{dual}(\mathcal{Q})}(\text{dual}(B))$, *and*
2. $\text{TCSAT}_{\mathcal{Q}}(B) \leq_{\text{m}}^{\log} \text{TCSAT}_{\text{dual}(\mathcal{Q})}(\text{dual}(B) \cup \{\bot, \sqcap\})$,

where $\text{dual}(B) := \{\text{dual}(f) \mid f \in B\}$ *and* $\text{dual}(\mathcal{Q}) := \{\text{dual}(Q) \mid Q \in \mathcal{Q}\}$ *for* $\text{dual}(\exists) := \forall$ *and* $\text{dual}(\forall) := \exists$.

PROOF. 1. Let $\mathcal{T}$ be a terminology with operators from B and $\mathcal{Q}$. For every concept C in $\mathcal{T}$, let $C^{\neg}$ be the concept obtained by transforming $\neg C$ into negation normal form (all negations are moved inside until they occur only in front of atomic concepts). Denote these changes by $\mathcal{T}^{\text{con}} := \{D^{\neg} \sqsubseteq C^{\neg} \mid (C \sqsubseteq D) \in \mathcal{T}\}$. This transformation has replaced all Boolean operators and quantifiers with their duals.

Now we will distinguish two cases. At first assume $\neg \notin [B]$. Hence, after negating and transforming into negation normal form, every atomic concept appears negated in $\mathcal{T}^{\text{con}}$. Then replace every negated atomic concept $\neg A$ with a fresh atomic concept A'. Denote these substitutions by $\mathcal{T}^{\text{con}\prime}$. Now $\mathcal{T}^{\text{con}\prime}$ is equisatisfiable with $\mathcal{T}$: it holds that $\mathcal{T} \in \text{TSAT}_{\mathcal{Q}}(B)$ iff $\mathcal{T}^{\text{con}\prime} \in \text{TSAT}_{\text{dual}(\mathcal{Q})}(\text{dual}(B))$, based on the observation that $\mathcal{I} \models C \sqsubseteq D$ iff $\mathcal{I} \models \neg D \sqsubseteq \neg C$, for every interpretation $\mathcal{I}$, together with the fact that $\mathcal{T}$ does not use negation.

Now consider the case $\neg \in [B]$. Then we do not need to substitute the negated atomic concepts in $\mathcal{T}^{\text{con}}$ as negation is self-dual and therefore included in $\text{dual}(B)$. Hence, $\mathcal{T}^{\text{con}}$ is already in $\mathfrak{T}_{\text{dual}(\mathcal{Q})}(\text{dual}(B))$ and equisatisfiable with $\mathcal{T}$.

2. Here we need the operators $\bot$ and $\sqcap$ to ensure that the input concept C is disjoint with C'. Now observe that $(C, \mathcal{T}) \in \text{TCSAT}_{\mathcal{Q}}(B)$ iff $(C, \mathcal{T}^{\text{con}\prime} \cup \{C \sqcap C' \sqsubseteq \bot\}) \in \text{TCSAT}_{\text{dual}(\mathcal{Q})}(\text{dual}(B))$, where $\mathcal{T}^{\text{con}\prime}$ is as in (1.).

□

Known complexity results for CSAT*.* The following classification of concept satisfiability has been obtained in [24].

Theorem 1 ([24]). *Let B be a finite set of Boolean operators.*

1. *If $[B]$ contains all operators that are monotone and* 1*-separating (i.e., $\mathsf{S}_{11} \subseteq [B]$), then* CSAT$(B)$ *is* PSPACE*-complete.*

2. *If $[B]$ consists of all conjunctions, plus the constant $\bot$ or both constants (i.e., $[B] \in \{\mathsf{E}, \mathsf{E}_0\}$), then* CSAT$(B)$ *is* coNP*-complete.*

3. *If $[B]$ contains only* 1*-reproducing operators (i.e., $[B] \subseteq \mathsf{R}_1$), then* CSAT$(B)$ *is trivial.*

4. *Otherwise* CSAT$(B) \in$ P*.*

We briefly put these previous results in context with existing results on sub-Boolean DLs, and we will later do the same for our own results. Part (1) of Theorem 1 is in contrast with the coNP-completeness of $\mathcal{ALU}$ satisfiability [37] because the operators in $\mathcal{ALU}$ can express the canonical base of S_{11}. The difference is caused by the fact that $\mathcal{ALU}$ allows only *unqualified* existential restrictions. Part (2) generalises the coNP-completeness of $\mathcal{ALE}$ satisfiability, where hardness is proven in [20] without using atomic negation. It is in contrast with the tractability of $\mathcal{AL}$ satisfiability [21], again because of the unqualified restrictions. Part (3) generalises the known fact that every $\mathcal{EL}$, $\mathcal{FL}_0$, and $\mathcal{FL}^-$ concept is satisfiable, which is immediate from the observation that the base $\{\vee, \equiv\}$ of R_1 can express conjunction (as $A_1 \sqcap A_2$ is equivalent to $(A_1 \sqcup A_2) \equiv (A_1 \equiv A_2)$), confirmed by the fact that $\wedge$ is $\mathtt{1}$-reproducing.

3. Complexity Results for TSat, TCSat, OSat, OCSat

In this section we will completely classify the above mentioned satisfiability problems for their complexity with respect to sub-Boolean fragments. The first five subsections contain all technical lemmas necessary to prove upper and lower complexity bounds, divided into the number of quantifiers and the kind of satisfiability problem. Each of these subsections will begin with an overview of the results to be proven. The last subsection puts all results together into five main theorems, and comments on their relation with existing results for fragments of $\mathcal{ALC}$.

3.1. Both quantifiers

The results for fragments that contain both quantifiers $\forall, \exists$ are summarized in Theorem 2.

Theorem 2. *Let B be a finite set of Boolean operators.*

1. *If $[B]$ contains all identity operators and both constants, or all unary operators (i.e., $\mathsf{I} \subseteq [B]$ or $\mathsf{N}_2 \subseteq [B]$), then $\text{TSat}_{\exists\forall}(B)$ is ExpTime-complete.*

2. *If $[B]$ contains all identity operators and the constant $\bot$, or all unary operators (i.e., $\mathsf{I}_0 \subseteq [B]$ or $\mathsf{N}_2 \subseteq [B]$), then $\star\text{Sat}^{\text{IND}}_{\exists\forall}(B)$ is ExpTime-complete.*

3. *If $[B]$ contains only $\mathtt{0}$-reproducing operators (i.e., $[B] \subseteq \mathsf{R}_0$), then $\text{TSat}_{\exists\forall}(B)$ is trivial.*

4. *If $[B]$ contains only $\mathtt{1}$-reproducing operators (i.e., $[B] \subseteq \mathsf{R}_1$), then $\star\text{Sat}_{\exists\forall}(B)$ is trivial.*

PROOF. Parts 1.–4. are formulated as Lemmas 8 to 12. (Due to the interreducibilities stated in Section 2, it suffices to show lower bounds for TSAT and upper bounds for OCSAT. Moreover Lemma 4 enables us to restrict the proofs to the standard basis of each clone for stating general results.) □

Part (2) for I_0 generalizes the EXPTIME-hardness of subsumption for $\mathcal{FL}_0$ and $\mathcal{AL}$ with respect to GCIs [6; 19; 23; 25]. The contrast to the tractability of subsumption with respect to GCIs in $\mathcal{EL}$, which uses only existential quantifiers, undermines the observation that, for negation-free fragments, the choice of the quantifier affects tractability and not the choice between conjunction and disjunction. DL-Lite and $\mathcal{ALU}$ cannot be put into this context because they use unqualified restrictions.

Parts (1) and (2) show that satisfiability with respect to theories is already intractable for even smaller sets of Boolean operators. One reason is that sets of axioms already contain limited forms of implication and conjunction. This also causes the results of this analysis to differ from similar analyses for sub-Boolean modal logics in that hardness already holds for bases of clones that are comparatively low in Post's lattice.

Part (3) reflects the fact that TSAT is less expressive than the other three decision problems: it cannot speak about one single individual.

Lemma 7 ([22; 34; 39]). $\mathrm{OCSAT}_{\exists\forall}(\mathsf{BF}) \in$ EXPTIME.

Lemma 8 ([30]). *Let B be a finite set of Boolean operators such that $[B]$ contains only* $\mathtt{1}$*-reproducing operators (i.e., $[B] \subseteq \mathsf{R}_1$). Then $\mathrm{OCSAT}_{\exists\forall}(B)$ is trivial.*

Lemma 9 ([30]). *Let B be a finite set of Boolean operators such that $[B]$ contains only* $\mathtt{0}$*-reproducing operators (i.e., $[B] \subseteq \mathsf{R}_0$). Then $\mathrm{TSAT}_{\exists\forall}(B)$ is trivial.*

Lemma 10. *Let B be a finite set of Boolean operators.*

1. *If $[B]$ contains all conjunctions and the constant $\bot$ (i.e., $\mathsf{E}_0 \subseteq [B]$), then $\star\mathrm{SAT}^{\mathrm{IND}}_{\exists\forall}(B)$ is* EXPTIME*-complete.*

2. *If $[B]$ contains all disjunctions and the constant $\bot$ (i.e., $\mathsf{V}_0 \subseteq [B]$), then $\star\mathrm{SAT}^{\mathrm{IND}}_{\exists\forall}(B)$ is* EXPTIME*-complete.*

3. If $[B]$ contains only self-dual operators (i.e., $[B] \subseteq \mathsf{D}$), then $\mathrm{TS}\textsc{at}_{\exists\forall}(B)$ *is* ExpTime-*complete.*

PROOF. Membership in ExpTime follows from Lemma 7 in combination with Lemma 4. For ExpTime-hardness, we proceed as follows.

1. We reduce from the positive entailment problem for Tarskian set constraints for $L(\langle\,\rangle, [\,], \cap)$ in [23] which, adjusted to our notation, is defined as follows. Given two sets Σ, Φ of pairs of concept expressions $C_1 \sqsubseteq C_2$ over the operators $\exists, \forall, \sqcap$, does every interpretation satisfying all expressions in Σ also satisfy all expressions in Φ? The proof in [23] showed ExpTime-hardness of this problem already for the case $|\Phi| = 1$. This implies ExpTime-completeness of $\mathrm{TS}\textsc{ubs}_{\exists\forall}(\{\sqcap\})$.

   We first reduce the *complement* of this problem to $\mathrm{TS}\textsc{at}_{\exists\forall}(\{\sqcap, \top, \bot\})$ by adding two axioms to $\mathcal{T}$ which express that there is an instance of C which is not an instance of D, using a fresh atomic concept D' and role R: $\mathcal{T}' := \mathcal{T} \cup \{\top \sqsubseteq \exists R.(C \sqcap D'),\ D' \sqcap D \sqsubseteq \bot\}$. The reduction is established by observing that $\mathcal{T} \not\models C \sqsubseteq D$ if and only if $\mathcal{T}'$ is satisfiable. If $\mathcal{T} \not\models C \sqsubseteq D$, then there is a model $\mathcal{I}$ of $\mathcal{T}$ with an instance x of C which is not an instance of D. If we additionally interpret D' as the complement of $D^{\mathcal{I}}$ and R as $\Delta^{\mathcal{I}} \times \{x\}$, we obtain a model of the two additional axioms of $\mathcal{T}'$. Conversely, a model of $\mathcal{T}'$ is a model of $\mathcal{T}$ that does not satisfy $C \sqsubseteq D$.

   Now for $\mathrm{TCS}\textsc{at}_{\exists\forall}(\{\bot, \sqcap\})$, we introduce an additional fresh concept name T that replaces the introduced occurrence of $\top$, and consider $\mathcal{T}'' := \mathcal{T} \cup \{T \sqsubseteq \exists R.(C \sqcap D'),\ D' \sqcap D \sqsubseteq \bot\}$. Then, $\mathcal{T} \not\models C \sqsubseteq D$ iff $(\mathcal{T}'', T) \in \mathrm{TCS}\textsc{at}_{\exists\forall}(\{\bot, \sqcap\})$, with the same reasoning as above.

2. For $\mathrm{TCS}\textsc{at}_{\exists\forall}(\{\bot, \sqcup\})$, we modify the above definition of $\mathcal{T}''$ to dispose of the two introduced conjunctions, using an additional fresh atomic concept E and role S:

   $$\mathcal{T}'' := \mathcal{T} \cup \{E \sqsubseteq C,\ E \sqsubseteq D',\ T \sqsubseteq \exists R.E,\ D \equiv \exists S.T,\ D' \equiv \forall S.\bot\}$$

   The last two axioms imply the disjointness of D and D'. Then, $\mathcal{T} \not\models C \sqsubseteq D$ iff $(\mathcal{T}'', T) \in \mathrm{TCS}\textsc{at}_{\exists\forall}(\{\bot, \sqcup\})$. If $\mathcal{T} \not\models C \sqsubseteq D$, then there is a model $\mathcal{I}$ of $\mathcal{T}$ with an instance x of C which is not an instance of D. If we extend $\mathcal{I}$ to $\mathcal{I}'$ via $(D')^{\mathcal{I}'} := \Delta^{\mathcal{I}} \setminus D^{\mathcal{I}}$, $E^{\mathcal{I}'} := C^{\mathcal{I}} \cap (D')^{\mathcal{I}}$, $T^{\mathcal{I}'} := \Delta^{\mathcal{I}}$,

$R^{\mathcal{I}'} := \Delta^{\mathcal{I}} \times \{x\}$, and $S^{\mathcal{I}'} := D^{\mathcal{I}'} \times \{x\}$, we obtain a model of the two additional axioms of $\mathcal{T}''$ where x is an instance of T. Conversely, any model $\mathcal{I}$ of $\mathcal{T}''$ with an instance x of T, must have an instance y of E (third axiom), which is also an instance of C (first axiom) and D' (second axiom), hence not an instance of D (last two axioms). Therefore, $\mathcal{I}$ is a model of $\mathcal{T}$ that does not satisfy $C \sqsubseteq D$.

3. The remaining case for the self-dual operators follows from Lemmas 1 and 2, as all self-dual operators in combination with the constants $\top, \bot$ (to which we have access as $\neg$ is self-dual) can express any arbitrary Boolean operator.

□

Lemma 11. *Let B, B' be finite sets of Boolean operators such that $[B]$ contains all identity operators and the constant $\bot$ (i.e., $\mathsf{I}_0 \subseteq [B]$) and $[B']$ contains all identity operators and both constants (i.e., $\mathsf{I} \subseteq [B']$). Then $\star\text{SAT}^{\text{IND}}_{\exists\forall}(B)$ and $\text{TSAT}_{\exists\forall}(B')$ are* EXPTIME*-complete.*

PROOF. For the upper bound apply Lemma 7 and Lemma 4. For hardness, we reduce from $\text{TSAT}_{\exists\forall}(\{\sqcap, \bot, \top\})$ to $\text{TSAT}_{\exists\forall}(\{\bot, \top\})$—the former is shown to be EXPTIME-complete in the proof of Lemma 10. The main idea is an extension of the normalization rules in [15] where also the following normalization rules have been stated and proven to be correct:

$$
\begin{array}{lllll}
(\textbf{NF1}) & \hat{C} \sqcap D \sqsubseteq E & \rightsquigarrow & \{A \equiv \hat{C}, A \sqcap D \sqsubseteq E\} \\
(\textbf{NF2}) & C \sqsubseteq D \sqcap \hat{E} & \rightsquigarrow & \{C \sqsubseteq D \sqcap A, A \equiv \hat{E}\} \\
(\textbf{NF3}) & \exists R.\hat{C} \sqsubseteq D & \rightsquigarrow & \{A \equiv \hat{C}, \exists R.A \sqsubseteq D\} \\
(\textbf{NF4}) & C \sqsubseteq \exists R.\hat{D} & \rightsquigarrow & \{C \sqsubseteq \exists R.A, A \equiv \hat{D}\} \\
(\textbf{NF5}) & C \sqsubseteq D \sqcap E & \rightsquigarrow & \{C \sqsubseteq D, C \sqsubseteq E\}
\end{array}
$$

where R is a role, C, D, E denote arbitrary concepts, $\hat{C}, \hat{D}$ denote concepts that are not atomic, and A is a fresh atomic concept.

Now we want to extend these rules for conjunctions on the left side of GCIs and for $\forall$-quantification:

$$
\begin{array}{llll}
(\textbf{NF6}) & \forall R.\hat{C} \sqsubseteq D & \rightsquigarrow & \{A \equiv \hat{C}, \forall R.A \sqsubseteq D\} \\
(\textbf{NF7}) & C \sqsubseteq \forall R.\hat{D} & \rightsquigarrow & \{A \equiv \hat{D}, C \sqsubseteq \forall R.A\} \\
(\textbf{NF8}) & C \sqcap D \sqsubseteq E & \rightsquigarrow & \{C \sqsubseteq \exists R_C.\top, D \sqsubseteq \forall R_C.C', \exists R_C.C' \sqsubseteq E\}
\end{array}
$$

where R is a role, C, D, E denote arbitrary concepts, $\hat{C}, \hat{D}$ denote concepts that are not atomic, R_C is a fresh role, and C' is a fresh atomic concept. For (**NF8**) we will prove correctness.

Assume $C \sqcap D \sqsubseteq E$ holds in an interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$. Thus $w \in E^{\mathcal{I}}$ for each $w \in C^{\mathcal{I}} \cap D^{\mathcal{I}}$. In the following we will construct a modified interpretation $\mathcal{I}'$ from $\mathcal{I}$ that satisfies the axioms constructed by (**NF8**). As $w \in C^{\mathcal{I}'}$, we add one R_C-edge to the same individual w, and due to $D \sqsubseteq \forall R_C.C'$ we must add w to $(C')^{\mathcal{I}'}$. Finally the last GCI is satisfied as we have $w \in E^{\mathcal{I}'}$.

For the opposite direction assume $C \sqcap D \sqsubseteq E$ cannot be satisfied, i.e., in every interpretation there is an individual which is an instance of C and D but not of E. Hence we take an arbitrary interpretation $\mathcal{I}$ such that it satisfies the first two axioms $C \sqsubseteq \exists R_C.\top$ and $D \sqsubseteq \forall R_C.C'$. Due to our assumption every individual w is an instance of C and D, and hence we have an R_C-edge to an individual where C' must hold. Therefore w is an instance of the left side of the third axiom but not of its right side E. Hence this axiom is not satisfied and we have the desired contradiction.

As this normalization procedure runs in polynomial time and eliminates every conjunction of concepts, we have a reduction from $\text{TCSAT}_{\exists\forall}(\{\sqcap, \bot\})$ to $\text{TCSAT}_{\exists\forall}(\{\bot\})$, and also from $\text{TSAT}_{\exists\forall}(\{\sqcap, \bot, \top\})$ to $\text{TSAT}_{\exists\forall}(\{\top, \bot\})$. Hence the Lemma applies. □

Lemma 12. *Let B be a finite set of Boolean operators such that $[B]$ contains all unary operators (i.e., $\mathsf{N}_2 \subseteq [B]$). Then $\star\text{SAT}_{\exists\forall}(B)$ is* EXPTIME-*complete.*

PROOF. The upper bound follows from Lemma 7 and Lemma 4. For the lower bound use Lemma 2 to simulate $\top$ and $\bot$ with fresh atomic concepts. Then the argumentation follows similarly to Lemmas 10 and 11. □

3.2. No quantifiers, TSAT

In this subsection we investigate the complexity of the problems $\text{TSAT}_{\emptyset}$. The quantifier-free case is nontrivial: for example, $\text{TSAT}_{\emptyset}(B)$ does not reduce to propositional satisfiability for B because restricted use of implication and conjunction are implicit in sets of axioms.

Theorem 3. *Let B be a finite set of Boolean operators.*

1. *If $[B]$ contains all operators that are affine and self-dual, or all monotone operators plus both constants (i.e., $\mathsf{L}_3 \subseteq [B]$ or $\mathsf{M} \subseteq [B]$), then $\text{TSAT}_{\emptyset}(B)$ is* NP-*complete.*

2. *If $[B]$ equals the set of all conjunctions (resp., disjunctions) plus both constants (i.e., $[B] = \mathsf{E}$ or $[B] = \mathsf{V}$), then* $\mathrm{TS}\textsc{at}_\emptyset(B)$ *is* P*-complete.*

3. *If $[B]$ equals the set of all identities plus both constants, or all unary functions with or without constants (i.e., $[B] \in \{\mathsf{I}, \mathsf{N}_2, \mathsf{N}\}$), then* $\mathrm{TS}\textsc{at}_\emptyset(B)$ *is* NL*-complete.*

4. *Otherwise (if $[B]$ contains only* `1`*- or only* `0`*-reproducing operators, i.e. $[B] \subseteq \mathsf{R}_1$ or $[B] \subseteq \mathsf{R}_0$), then* $\mathrm{TS}\textsc{at}_\emptyset(B)$ *is trivial.*

PROOF. NP-completeness for (1) is composed of on the one hand the upper bound which results from $\mathrm{OCS}\textsc{at}_\exists(\{\sqcap, \neg, \top, \bot\})$ which is proven to be in NP in Lemma 25 and on the other hand the lower bounds which are proven in Lemmas 13 and 14. Both lower bounds of (2) will be proven through Lemmas 15 and 16. The upper bound is due to $\mathrm{OCS}\textsc{at}_\exists(\{\sqcap, \top, \bot\})$ which is shown to be in P in Lemma 30. The membership in (3) results from Lemmas 18 and 26. Hardness for I is proven in Lemma 17. Together with Lemma 2, we obtain hardness for N_2. Item (4) follows through Lemmas 8 and 9. □

Lemma 13. *Let B be a finite set of Boolean operators such that $[B]$ contains all monotone operators (i.e., $\mathsf{M} \subseteq [B]$). Then* $\mathrm{TS}\textsc{at}_\emptyset(B)$ *is* NP*-hard.*

PROOF. We reduce from the complements of the implication problems for the self-dual and monotone fragments of propositional logic. Those problems are shown to be coNP-complete in [12]. NP-hardness of $\mathrm{TS}\textsc{at}_\emptyset(\mathsf{M})$ follows from the fact that, for propositional formulae φ, ψ with monotone operators only, it holds that

$$\varphi \not\models \psi \iff \{C_\psi \sqsubseteq \bot, \top \sqsubseteq C_\varphi\} \in \mathrm{TS}\textsc{at}_\emptyset(\mathsf{M}),$$

where C_φ and C_ψ are concepts corresponding to φ, ψ in the usual way. □

Lemma 14. *Let B be a finite set of Boolean operators such that $[B]$ contains all operators that are self-dual and affine (i.e., $\mathsf{L}_3 \subseteq [B]$). Then* $\mathrm{TS}\textsc{at}_\emptyset(B)$ *is* NP*-hard.*

PROOF. Here we will provide a reduction from the NP-complete problem 1-IN-3-SAT which is defined as follows: given a formula $\varphi = \bigwedge_{i=1}^{n} \bigvee_{j=1}^{3} \ell_{ij}$, where ℓ_{ij} are literals (propositional variables or their negations), we ask for the existence of a satisfying assignment which makes exactly one literal per clause true [35]. In the following we are allowed to use the binary exclusive-or by $x \boxplus y \boxplus \top \boxplus \top \equiv x \boxplus y$ because the following holds: the connective "negation" is available as $x \boxplus x \boxplus z \boxplus \top \equiv \neg z$, and therefore we have access to both constants $\top$ and $\bot$ due to Lemma 2.

The main idea of the reduction is to use for each clause $(\ell_{i1} \vee \ell_{i2} \vee \ell_{i3}) \in \varphi$ an axiom $\top \sqsubseteq \tilde{\ell}_{i1} \boxplus \tilde{\ell}_{i2} \boxplus \tilde{\ell}_{i3}$, where $\tilde{\ell} := A_x$ for a positive literal $\ell = x$ and $\tilde{\ell} := \neg A_x$ for a negative literal $\ell = \neg x$, to enforce that only one literal is satisfied. Unfortunately for this axiom it is possible to have all literals satisfied which is not a valid 1-IN-3-SAT-assignment. But in the following we show how the addition of some other axioms helps circumventing this problem.

Let φ be defined as above, then the reduction is defined as $\varphi \mapsto \mathcal{T}$, where

$$
\begin{aligned}
\mathcal{T} := & \left\{ \top \sqsubseteq (\tilde{\ell}_{i1} \boxplus \tilde{\ell}_{i2}) \boxplus \tilde{\ell}_{i3} \boxplus s^i \boxplus \top \;\middle|\; 1 \leq i \leq n \right\} \cup && (1)\\
& \cup \left\{ \top \sqsubseteq (\tilde{\ell}_{i1} \boxplus \tilde{\ell}_{i2}) \boxplus \tilde{\ell}_{i3} \;\middle|\; 1 \leq i \leq n \right\} \cup && (2)\\
& \cup \left\{ s_1^i \sqsubseteq \tilde{\ell}_{i1} \boxplus \tilde{\ell}_{i2} \;\middle|\; 1 \leq i \leq n \right\} \cup \\
& \cup \left\{ s_2^i \sqsubseteq \tilde{\ell}_{i1} \boxplus \tilde{\ell}_{i3} \;\middle|\; 1 \leq i \leq n \right\} \cup \\
& \cup \left\{ s_3^i \sqsubseteq \tilde{\ell}_{i2} \boxplus \tilde{\ell}_{i3} \;\middle|\; 1 \leq i \leq n \right\} \cup \\
& \cup \left\{ s^i \sqsubseteq (s_1^i \boxplus s_2^i) \boxplus s_3^i \;\middle|\; 1 \leq i \leq n \right\}.
\end{aligned}
$$

Now we claim that $\varphi \in$ 1-IN-3-SAT iff $\mathcal{T} \in \mathrm{TSAT}_\emptyset(\mathsf{L}_0)$.

For the correctness consider an arbitrary clause $\ell_{i1} \vee \ell_{i2} \vee \ell_{i3}$ from φ with ℓ_{ij} its literals.

The following table shows each possible assignment for the ℓ_{ij} and suitable assignments for the s_k^i and s^i, and the validity of the axioms (1) and (2). A bold truth value of an s_k^i or s^i in the table denotes that this assignment is enforced whereas a blank cell denotes arbitrary choices. If at least one of (1) and (2) are contradicted then there exists no model for $\mathcal{T}$.

ℓ_{i1}	ℓ_{i2}	ℓ_{i3}	s_1^i	s_2^i	s_3^i	s^i	(1)	(2)
0	0	0						↯
0	0	1	**0**	1	0	**1**	✓	✓
0	1	0	1	**0**	0	**1**	✓	✓
0	1	1						↯
1	0	0	1	0	**0**	**1**	✓	✓
1	0	1						↯
1	1	0						↯
1	1	1	**0**	**0**	**0**	**0**	↯	✓

At first we start with an interpretation and some individual w. Through the mapping of the literals ℓ_{ij} to w we immediately observe whether axiom (2) is contradicted or not. If (2) is not contradicted then we have to consider the remaining axioms having s_k^i on the left side in order to find an extension of this interpretation which assigns the s_k^i and s^i in a way that (2) is not violated whenever we have an interpretation which corresponds to a valid 1-in-3-Sat assignment. For the opposite direction we have to show that there exists no possible extension that satisfies axiom (2) whenever we have an assignment which is not a valid for 1-in-3-Sat.

The table now proves that, if there exists a valid 1-in-3-Sat assignment, we can always construct an interpretation which satisfies all axioms (the checkmark cases), and for every non 1-in-3-Sat assignment it is not possible to construct an interpretation which satisfies every axiom (the remaining cases where either in column (1) or (2) there is a lightning symbol stating that the axiom is violated). □

Lemma 15. *Let B be a finite set of Boolean operators such that $[B]$ equals the set of all conjunctions plus constants (i.e., $[B] = \mathsf{E}$). Then* $\text{TSat}_\emptyset(B)$ *is* P*-hard.*

proof. We reduce from HornSat, a problem that is known to be P-complete [32, p. 176], and which is defined as follows. A Boolean formula is in Horn-CNF if it is a conjunction of Horn clauses. A Horn clause is a disjunction of literals, of which at most one is positive. Hence, every Horn clause is either a positive unit clause p, or has only negative literals $\neg p_1 \vee \cdots \vee \neg p_n$, or has some negative and one positive literal $\neg p_1 \vee \cdots \vee \neg p_n \vee p_{n+1}$, for $n \geqslant 1$. The latter case can be equivalently rewritten as $p_1 \wedge \cdots \wedge p_n \rightarrow p_{n+1}$. HornSat is the problem of deciding, given a Boolean formula φ in HORN-CNF, whether φ is satisfiable.

The reduction is as follows. Construct the following TBox, using concept names P_i to represent the propositional variables p_i.

$$
\begin{aligned}
\mathcal{T}_\varphi := & \{\top \sqsubseteq P \mid p \text{ is a positive unit clause of } \varphi\} \\
& \cup \{P_1 \sqcap \cdots \sqcap P_n \sqsubseteq \bot\} \mid (\neg p_1 \vee \cdots \vee \neg p_n) \in \varphi\} \\
& \cup \{P_1 \sqcap \cdots \sqcap P_n \sqsubseteq P_{n+1}\} \mid (\neg p_1 \vee \cdots \vee \neg p_n \vee p_{n+1}) \in \varphi\}
\end{aligned}
$$

It is straightforward to prove that φ is satisfiable if and only if $\mathcal{T}_\varphi$ is. □

Lemma 16. *Let B be a finite set of Boolean operators such that $[B]$ equals the set of all disjunctions plus constants (i.e., $[B] = \mathsf{V}$). Then $\text{TSAT}_\emptyset(B)$ is* P*-hard.*

PROOF. This is a consequence of Lemma 15 and the reduction $\text{TSAT}_\emptyset(\mathsf{E}) \leq_{\text{m}}^{\log} \text{TSAT}_{\text{dual}(\emptyset)}(\text{dual}(\mathsf{E})) = \text{TSAT}_\emptyset(\mathsf{V})$ from Lemma 6. □

Lemma 17. *Let B be a finite set of Boolean operators such that $[B]$ equals the set of all identities plus constants (i.e., $[B] = \mathsf{I}$). Then $\text{TSAT}_\emptyset(B)$ is* NL*-hard.*

PROOF. The underlying intuition is that any TBox $\mathcal{T}$ over the base B is a set of inclusions between atomic concepts that is isomorphic to a graph, and a deduction chain is isomorphic to a path in that graph. It is therefore straightforward to obtain NL-hardness via reduction from the complement of the graph accessibility problem GAP. Consider a given directed graph $G = (V, E)$ and two nodes $s, t \in V$ as the recent instance for GAP asking for a path from s to t in G. We introduce a concept name A_v per node $v \in V$ and define $\mathcal{T} := \{A_u \sqsubseteq A_v \mid (u, v) \in E\} \cup \{\top \sqsubseteq A_s, A_t \sqsubseteq \bot\}$.

To prove that $(G, s, t) \notin \text{GAP} \iff \mathcal{T} \in \text{TSAT}_\emptyset(B)$, observe that $\mathcal{T} \models A_v \sqsubseteq A_w$ iff G has a path from v to w, for any nodes v, w. The last two axioms of $\mathcal{T}$ then ensure that $\mathcal{T}$ contains a contradiction iff G has a path from s to t. □

Lemma 18. *Let B be a finite set of Boolean operators such that $[B]$ equals the set of all identities plus constants (i.e., $[B] = \mathsf{I}$). Then $\text{TSAT}_\emptyset(B)$ is in* NL*.*

PROOF. With the same intuition as in the proof of Lemma 17, we reduce to the complement of GAP. Let $\mathcal{T}$ be a TBox, where the set of concept names occurring in $\mathcal{T}$ plus $\top, \bot$ is $A_0, \ldots, A_n$ with $A_0 = \bot$ and $A_1 = \top$. Then $\mathcal{T}$ is mapped to $G = (V, E)$ with $V := \{v_0, \ldots, v_n\}$ and $E := \{(v_i, v_j) \mid A_i \sqsubseteq A_j\}$. Now it holds that $\mathcal{T} \in \text{TSAT}_\emptyset(B)$ iff $(G, v_1, v_0) \notin \text{GAP}$. As a corner case, if $\top$ and $\bot$ do not both occur in $\mathcal{T}$, $\mathcal{T}$ is trivially satisfiable and there is no path in G from v_1 to v_0. In general, $\mathcal{T} \models A_i \sqsubseteq A_j$ iff there is a path from v_i to v_j in G, for any $0 \leq i, j \leq n$. Hence, $\mathcal{T}$ is satisfiable, which is equivalent to $\mathcal{T} \not\models \top \sqsubseteq \bot$, iff there is no path from v_1 to v_0 in G. □

3.3. One quantifier, TSAT

In this subsection we investigate the complexity of the problems $\text{TSAT}_\exists$ and $\text{TSAT}_\forall$.

Theorem 4. *Let B be a finite set of Boolean operators and $Q \in \{\forall, \exists\}$.*

1. *If $[B]$ contains all monotone or all unary operators (i.e., $\mathsf{M} \subseteq [B]$ or $\mathsf{N}_2 \subseteq [B]$), then $\text{TSAT}_Q(B)$ is* EXPTIME*-complete.*
2. *If $[B]$ equals the set of all identities and constants (i.e., $[B] = \mathsf{I}$), then $\text{TSAT}_Q(B)$ is* P*-complete.*
3. *If $[B]$ equals the set of all conjunctions and constants (i.e., $[B] = \mathsf{E}$), then $\text{TSAT}_\exists(B)$ is* P*-complete, and $\text{TSAT}_\forall(B)$ is* EXPTIME*-complete.*
4. *If $[B]$ equals the set of all disjunctions and constants (i.e., $[B] = \mathsf{V}$), then $\text{TSAT}_\forall(B)$ is* P*-complete, and $\text{TSAT}_\exists(B)$ is* EXPTIME*-complete.*
5. *Otherwise (if $[B]$ contains only* `1`*- or only* `0`*-reproducing operators, i.e., $[B] \subseteq \mathsf{R}_1$ or $[B] \subseteq \mathsf{R}_0$), then $\text{TSAT}_Q(B)$ is trivial.*

PROOF. EXPTIME-hardness for the monotone case in (1) follows from Lemmas 19 and 20. For N_2, see Lemma 22. EXPTIME-membership results from Lemmas 7 and 4, and applies to Items 3 and 4, too.

P-hardness in (2) for $\text{TSAT}_\exists(\mathsf{I})$ is shown in Lemma 23, and the case $\forall$ is due to Lemma 6. This applies to Items 3 and 4, too. P-membership follows from Items 3 and 4.

In (3), P-membership of $\text{TSAT}_\exists(\mathsf{E})$ follows from Lemma 30 (P-membership of $\text{OCSAT}_\exists(\sqcap, \top, \bot)$). For EXPTIME-hardness of $\text{TSAT}_\forall(\mathsf{E})$, see Lemma 20.

In (4), P-membership of $\text{TSAT}_\forall(\mathsf{V})$ follows from Lemma 31 (P-membership of $\text{OCSAT}_\forall(\sqcup, \top, \bot)$). For EXPTIME-hardness of $\text{TSAT}_\exists(\mathsf{V})$, see Lemma 21.

(5) follows from Lemmas 8 and 9. □

Item 5 generalizes the fact that every $\mathcal{EL}$- and $\mathcal{FL}_0$-TBox is satisfiable, and the whole theorem shows that separating either conjunction and disjunction, or the constants is the only way to achieve tractability for TSAT.

Lemma 19. *Let B be a finite set of Boolean operators such that $[B]$ equals the set of all monotone operators plus constants (i.e., $[B] = \mathsf{M}$). Then* $\text{TSAT}_\exists(B)$ *is* EXPTIME*-hard.*

PROOF. We reduce from the complement of $\text{TSUBS}_\exists(\{\top, \sqcap, \sqcup\})$. That problem coincides with $\mathcal{ELU}$ TBox subsumption, which is EXPTIME-complete [6, Thm. 7]. It holds that $\mathcal{T} \not\models C \sqsubseteq D$ if and only if $\mathcal{T} \cup \{\top \sqsubseteq \exists R.(C \sqcap D'),\ D \sqcap D' \sqsubseteq \bot\}$ is satisfiable, with the same justification as in the first part of the proof of Lemma 10, Item 1. □

Lemma 20. *Let B be a finite set of Boolean operators such that $[B]$ equals the set of all conjunctions plus constants (i.e., $[B] = \mathsf{E}$). Then* $\text{TSAT}_\forall(B)$ *is* EXPTIME*-hard.*

PROOF. We reduce from the complement of $\text{TSUBS}_\forall(\{\sqcap\})$. That problem coincides with $\mathcal{FL}_0$ TBox subsumption, which is EXPTIME-complete [6; 25]. We transform a given instance $(\mathcal{T}, C, D)$ into

$$\mathcal{T}' := \mathcal{T} \cup \{\forall R.\bot \sqsubseteq \bot,\ \top \sqsubseteq \forall R.C \sqcap \forall R.D',\ D' \sqcap D \sqsubseteq \bot\},$$

where D is a fresh concept name and R a fresh role. It is straightforward to show that $\mathcal{T} \not\models C \sqsubseteq D$ if and only if $\mathcal{T}'$ is satisfiable:

For "$\Rightarrow$", assume that $\mathcal{T} \not\models C \sqsubseteq D$ and consider a model $\mathcal{I}$ of $\mathcal{T}$ that does not satisfy $C \sqsubseteq D$. Take an instance $x \in C^{\mathcal{I}} \setminus D^{\mathcal{I}}$ and construct the interpretation $\mathcal{J}$ from $\mathcal{I}$ by setting $\Delta^{\mathcal{J}} = \Delta^{\mathcal{I}}$, $R^{\mathcal{J}} = \Delta^{\mathcal{J}} \times \{x\}$, $(D')^{\mathcal{J}} = \Delta^{\mathcal{J}} \setminus D^{\mathcal{J}}$, and $X^{\mathcal{J}} = X^{\mathcal{I}}$ for all other concept names and roles. Clearly, $\mathcal{J}$ satisfies $\mathcal{T}'$.

For "$\Leftarrow$", observe that every model of $\mathcal{T}'$ is also one of $\mathcal{T}$, and the three new axioms imply that $C \sqcap \neg D$ has an instance: the first says that every point has an outgoing R-edge; the second says that each such edge must lead to an instance of $C \sqcap D'$, and the third says that D' is subsumed by $\neg D$. □

Lemma 21. *Let B be a finite set of Boolean operators such that $[B]$ equals the set of all disjunctions plus constants (i.e., $[B] = \mathsf{V}$). Then* $\text{TSAT}_\exists(B)$ *is* EXPTIME*-hard.*

PROOF. This follows from Lemma 20 together with the contraposition argument of Lemma 6. □

Lemma 22. *Let B be a finite set of Boolean operators such that $[B]$ equals the set of all unary operators (i.e., $[B] = \mathsf{N}_2$) and $Q \in \{\forall, \exists\}$. Then $\text{TSAT}_Q(B)$ is* EXPTIME*-hard.*

PROOF. We reduce from $\text{TSAT}_{\exists\forall}(\mathsf{I})$, whose EXPTIME-completeness is proven in Lemma 11. The reduction from $\text{TSAT}_{\exists\forall}(\mathsf{I})$ to $\text{TSAT}_{\exists}(\mathsf{N})$ and $\text{TSAT}_{\forall}(\mathsf{N})$ is obvious because one of the two quantifiers can be expressed using $\neg$ and the other quantifier. Furthermore, we can reduce $\text{TSAT}_Q(\mathsf{N})$ to $\text{TSAT}_Q(\mathsf{N}_2)$ by simulating the constants using new concept names and negation, as follows from Lemma 2. □

Lemma 23. *Let B be a finite set of Boolean operators such that $[B]$ equals the set of all identities plus constants (i.e., $[B] = \mathsf{I}$). Then $\text{TSAT}_{\exists}(B)$ is* P*-hard.*

PROOF. Cook [18] proved that the complexity class P can be represented by nondeterministic Turing machines running in logarithmic space using a stack.[2] We will reduce the word problem for this machine model to $\text{TSUBS}_{\exists}(\emptyset)$, which in turn can be reduced to the complement of $\text{TSAT}_{\exists}(\mathsf{I})$: $(\mathcal{T}, C, D) \in \text{TSUBS}_{\exists}(\emptyset)$ iff $(\mathcal{T} \cup \{\top \sqsubseteq C, D \sqsubseteq \bot\}) \notin \text{TSAT}_{\exists}(\mathsf{I})$. This will provide P-hardness of $\text{TSAT}_{\exists}(\mathsf{I})$.

Let M be a nondeterministic Turing machine, which has access to a read-only input tape, a read-write work tape and a stack. Let M be the 6-tuple $(\Sigma, \Psi, \Gamma, Q, f, q_0)$, where

- Σ is the input alphabet;
- Ψ is the work alphabet containing the empty-cell symbol #;
- Γ is the stack alphabet containing the bottom-of-stack symbol $\Box$;
- Q is the set of states;

[2] If the machine model is further restricted to polynomial runtime, then it characterises the complexity class LOGCFL, which consists of all problems reducible in logarithmic space to deciding membership in a context-free language.

- f is the state transition function which maps $Q \times \Sigma \times \Psi \times \Gamma$ to the power set of $Q \times \Psi \times \{-,+\}^2 \times (\Gamma \setminus \{\Box\})^\star$ and describes a transition where the machine is in a state, reads an input symbol, reads a work symbol and takes a symbol from the stack, and goes into another state, writes a symbol to the work tape, makes a step on each tape (left or right) and possibly adds a sequence of symbols to the stack;

- $q_0 \in Q$ is the initial state.

We assume that each computation of M starts in q_0 with the heads at the left-most position of each tape and with exactly the symbol $\Box$ on the stack. W.l.o.g. the machine accepts whenever the stack is empty, regardless of its current state.

We now fix a machine M and an input word $\vec{w} = w_1 \dots w_n$ of length n and consider the configurations that can occur during any computation of $M(\vec{w})$. Since M is logarithmically space-bounded, the size of these configurations does not exceed some value $\ell \in O(\log n)$ that only depends on n and not on the w_i.

A *shallow configuration* of $M(\vec{w})$ is a sequence $(p\delta_1 \dots \delta_{k-1} q \delta_k \dots \delta_\ell)$, where

- $p \in \{1, \dots, n\}$ is the current position on the input tape, represented in binary;

- $\delta_1, \dots, \delta_\ell$ is the current content of the work tape;

- k is the current position on the work tape;

- q is the current state of M.

The initial shallow configuration $(0q_0\# \dots \#)$ is denoted by S_0. Let $\mathcal{SC}$ be the set of all possible shallow configurations that can occur during any computation of $M(\vec{w})$. The cardinality of this set is bounded by a polynomial in n because the number of work-tape cells used is logarithmic in n and the binary counter for the position on the input tape is logarithmic in n.

A *deep configuration* of $M(\vec{w})$ is a sequence

$$(R_1 \dots R_m p \delta_1 \dots \delta_{k-1} q \delta_k \dots \delta_\ell),$$

where the R_i are the symbols currently on the stack and the remaining components are as above. Let $\mathcal{DC}$ be the set of all possible deep configurations that can occur during any computation of $M(\vec{w})$. The cardinality of this set can be exponential as soon as Γ has more than two elements besides $\square$. This is not a problem for our reduction, which will only touch shallow configurations.

We now construct an instance of $\text{TSubs}_\exists(\emptyset)$ from M and $\vec{w}$. We use each shallow configuration $X \in \mathcal{SC}$ as a concept name and each stack symbol as a role name. The TBox $\mathcal{T}$ describes all possible computations of $M(\vec{w})$ by containing an axiom for every two deep configurations that the machine can take on before and after some computation step. A deep configuration D is represented by the concept corresponding to D's shallow part, preceded by the sequence of existentially quantified stack symbols corresponding to the stack content in D. The TBox $\mathcal{T}$ is constructed from a set of axioms per entry in f. (We will omit the subscript from now on.) For the instruction

$$(q, \sigma, \delta, R) \mapsto (q', \delta', -, -, R_1 \dots R_k)$$

of f, we add the axioms

$$\begin{aligned} \exists R.(\text{bin}(p)\delta_0 \dots \delta_{i-1} q \delta \delta_{i+1} \dots \delta_\ell) \; &\sqsubseteq \\ \exists R_1 \dots \exists R_k.(\text{bin}(p \dot{-} 1)\delta_0 \dots \delta_{i-2} q' \delta_{i-1} \delta' \delta_{i+1} \dots \delta_\ell) & \end{aligned} \tag{3}$$

for every p with $w_p = \sigma$, every $i = 1, \dots, \ell$, and all $\delta_0, \dots, \delta_{i-1}, \delta_{i+1}, \dots, \delta_\ell$. The expression $p \dot{-} 1$ stands for $p - 1$ if $p \geqslant 2$ and for 1 otherwise, reflecting the assumption that the machine does not move on the input tape on a "go left" instruction if it is already on the left-most input symbol. This behaviour can always be assumed w.l.o.g. In case $k = 0$, the quantifier prefix on the right-hand side is empty. For instructions of f requiring "+" steps on any of the tapes, the construction is analogous. The number of axioms generated by each instruction is bounded by the number of shallow configurations; therefore the overall number of axioms is bounded by a polynomial in $n \cdot |f|$.

Furthermore, we use a fresh concept name A and add an axiom $X \sqsubseteq A$ for each shallow configuration X. Also we add a single axiom $S \sqsubseteq \exists \square . S_0$ to $\mathcal{T}$. The instance of $\text{TSubs}_\exists(\emptyset)$ is constructed as $(\mathcal{T}, S, A)$. $\mathcal{T}$ can be constructed in logarithmic space. The number of axioms in $\mathcal{T}$ is bounded by $n \cdot |f| + 2$, which depends only on n and not on the contents of the input word. It remains to prove the following claim.

Claim. $M(\vec{w})$ has an accepting computation if and only if $S \sqsubseteq_\mathcal{T} A$.

Proof of Claim. For the "⇒" direction, we observe that, for each step in the accepting computation, the (arbitrary) concept associated with the predecessor configuration is subsumed by the concept associated with the successor configuration. More precisely, if $M(\vec{w})$ makes a step

$$(q, \sigma, \delta, R) \mapsto (q', \delta', -, -, R_1 \ldots R_k),$$

then its deep configuration before that step has to be

$$S_1 \ldots S_j R p \delta_0 \ldots \delta_{i-1} q \delta \delta_{i+1} \ldots \delta_\ell,$$

for some $S_1, \ldots, S_j \in \Gamma$, $\delta_0, \ldots, \delta_{i-1}, \delta_{i+1}, \ldots, \delta_\ell \in \Psi$ and $p \in \mathbb{N}$, and the deep configuration after that step is

$$S_1 \ldots S_j R_1 \ldots R_k (p \dot{-} 1) \delta_0 \ldots \delta_{i-2} q' \delta_{i-1} \delta' \delta_{i+1} \ldots \delta_\ell.$$

The set of axioms in (3) ensures that there is an axiom that implies

$$\begin{gathered} \exists S_1 \ldots \exists S_j . \exists R . (\mathrm{bin}(p) \delta_0 \ldots \delta_{i-1} q \delta \delta_{i+1} \ldots \delta_\ell) \ \sqsubseteq_{\mathcal{T}} \\ \exists S_1 \ldots \exists S_j . \exists R_1 \ldots R_k . (\mathrm{bin}(p \dot{-} 1) \delta_0 \ldots \delta_{i-2} q' \delta_{i-1} \delta' \delta_{i+1} \ldots \delta_\ell). \end{gathered}$$

Since some computation of $M(\vec{w})$ reaches a configuration with an empty stack, we can conclude that some atomic concept corresponding to a shallow configuration $\mathcal{S}$, and therefore also A, subsumes $\exists \square . S_0$ which subsumes S (per definition).

For the "$\Leftarrow$" direction, we assume that $M(\vec{w})$ has no accepting computation. This means that, during every computation of $M(\vec{w})$, the stack does never become empty. From the set of all computations of $M(\vec{w})$, we will show that there exists an interpretation $\mathcal{I}$ that satisfies $\mathcal{T}$, but not $S \sqsubseteq A$; hereby we can conclude $(\mathcal{T}, S, A) \notin \text{TS}\textsc{ubs}_\exists(\emptyset)$.

Observe that any atomic concept besides S and A in $\mathcal{T}$ correspond to a specific shallow configuration of $M(\vec{w})$. Let $T = (V, E)$ denote the computation tree of $M(\vec{w})$. Thus every node $v \in V$ represents a deep configuration of $M(\vec{w})$ which will be denoted via C_v. Then for two nodes $u, v \in V$ with $(u, v) \in E$ it holds that $C_u \vdash_M C_v$. In the following we will describe how to construct an interpretation $\mathcal{I}$ from T which has a witness for $S^\mathcal{I} \not\subseteq A^\mathcal{I}$. To simplify notion, we will use μ to denote a shallow configuration $\mu \in \mathcal{SC}$ as well as the respecting concept in $\mathcal{T}$.

The root of T is the initial configuration $\square 0 q_0 \overbrace{\# \ldots \#}^{\ell}$. Now we will define $\mathcal{I}(S) := \bigcup_{i \geq 0} \mathcal{I}_i(S)$ starting with $\Delta^{\mathcal{I}_0(S)} := \{x\}$ and

- $S^{\mathcal{I}_0(S)} := \{x\}$, and
- $y \in (S_0)^{\mathcal{I}_0(S)}$ with $(x, y) \in \Box^{\mathcal{I}_0(S)}$ (i.e., $(\exists\Box.S_0)^{\mathcal{I}_0(S)} = \{x\}$)

inductively as follows.
(**1**) For every node $v \in V$ with deep configuration $C_v = S_1 \dots S_j R\mu$, where μ is a sequence of some bin(p) followed by a string in $\Psi^h \cdot Q \cdot \Psi^k$ with $h+k = \ell-1$, proceed as follows. Let $x_1, \dots, x_j, x_r, x_\mu \in \Delta^{\mathcal{I}_i(S)}$ be individuals such that $(x_1, x_2) \in (S_1)^{\mathcal{I}_i(S)}$, $(x_2, x_3) \in (S_2)^{\mathcal{I}_i(S)}$, $\dots$, $(x_j, x_r) \in (S_j)^{\mathcal{I}_i(S)}$, $(x_r, x_\mu) \in R^{\mathcal{I}_i(S)}$ and $x_\mu \in \mu^{\mathcal{I}_i(S)}$. For every $u \in V$ with $(v, u) \in E$ that corresponds to a successor configuration $C_u = S_1 \dots S_j R_1 \dots R_k \lambda$ of C_v, i.e., $C_v \vdash_M C_u$:

- add x_r to $\lambda^{\mathcal{I}_{i+1}(S)}$ for $k = 0$, and otherwise
- if there are no $y_1, \dots, y_k \in \Delta^{\mathcal{I}_i(S)}$ with $(x_r, y_1) \in (R_1)^{\mathcal{I}_i(S)}$, $(y_1, y_2) \in (R_2)^{\mathcal{I}_i(S)}, \dots, (y_{k-1}, y_k) \in (R_k)^{\mathcal{I}_i(S)}$ and $y_k \in \lambda^{\mathcal{I}_i(S)}$, then introduce new individuals $y_1, \dots, y_k$ to $\Delta^{\mathcal{I}_{i+1}(S)}$ and add (x_μ, y_1) to $(R_1)^{\mathcal{I}_{i+1}(S)}$, (y_1, y_2) to $(R_2)^{\mathcal{I}_{i+1}(S)}$, $\dots$, (y_{k-1}, y_k) to $(R_k)^{\mathcal{I}_{i+1}(S)}$ and include y_k into $\lambda^{\mathcal{I}_{i+1}(S)}$.

(**2**) For every individual $x \in \Delta^{\mathcal{I}_i(S)}$ and deep configuration χ that is also a shallow configuration with $x \in \chi^{\mathcal{I}_i(S)}$ include x into $B^{\mathcal{I}_{i+1}(S)}$.

In the following we will show that $\mathcal{I}(S)$ is indeed a valid interpretation for $\mathcal{T}$ but $S \not\sqsubseteq_{\mathcal{T}} A$. As there is no axiom in $\mathcal{T}$ with S on the right side it holds that $|S^{\mathcal{I}(S)}| = 1$. Assume there is some GCI $G = A_G \sqsubseteq A'_G \in \mathcal{T}$ which is violated in $\mathcal{I}(S)$, i.e., we have some individual $x' \in \Delta^{\mathcal{I}(S)}$ such that $x' \in (A_G)^{\mathcal{I}(S)}$ but $x' \notin (A'_G)^{\mathcal{I}(S)}$. As in $\mathcal{T}$ there are two different kinds of axioms we have to distinguish these cases (because the axiom with S on the left side cannot be such a violated axiom):

1. If $G = \alpha \sqsubseteq \beta \in \mathcal{T}$ for α and β being atomic (this is the case for axioms with concepts representing shallow configurations on the left side and A on the right side), then $x' \in \alpha^{\mathcal{I}(S)}$ but $x' \notin \alpha^{\mathcal{I}(S)}$. Now consider the least index n such that $x' \in \alpha^{\mathcal{I}_n(S)}$. As α represents clearly a shallow configuration and $\beta = A$ then x' is added to $\beta^{\mathcal{I}_{n+1}(S)} \subseteq \beta^{\mathcal{I}(S)}$ by (**2**), which contradicts the assumption.

2. If $G = \exists R.\mu \sqsubseteq \exists R_1. \dots .\exists R_k.\lambda \in \mathcal{T}$ wherefore exist some entry in f from M such that $(S_1 \dots S_j R\mu) \vdash_M (S_1 \dots S_j R_1 \dots R_k \lambda)$ for some stack symbols $S_1, \dots, S_j$, then $x' \in (\exists R.\mu)^{\mathcal{I}(S)}$ but $x' \notin (\exists R_1. \dots .\exists R_k.\lambda)^{\mathcal{I}(S)}$.

Now let n denote the least index such that y is added to $(\mu)^{\mathcal{I}_n(S)}$ and there must be some $m < n$ such that (x', y) is added to $R^{\mathcal{I}_m(S)}$. Then in step (**1**) there are $y_1, \ldots, y_k$ added to $\Delta^{\mathcal{I}_{n+1}(S)}$, the corresponding R_i-edges are added to their respective $(R_i)^{\mathcal{I}_{n+1}(S)}$-set and y_k is added to $\lambda^{\mathcal{I}_{n+1}(S)}$ obeying $x \in (\exists R_1 . \ldots \exists R_k . \lambda)^{\mathcal{I}_{n+1}(S)} \subseteq (\exists R_1 . \ldots \exists R_k . \lambda)^{\mathcal{I}(S)}$. This contradicts our assumption again.

Consequently $\mathcal{I}(S)$ is a model of $\mathcal{T}$. Now assume that $S^{\mathcal{I}(S)} \subseteq A^{\mathcal{I}(S)}$. Thus for the starting point x which is added to $S^{\mathcal{I}(S)}$ at the initial construction step of $\mathcal{I}(S)$, it holds in particular that $x \in A^{\mathcal{I}(S)}$. As x is added to $A^{\mathcal{I}(S)}$ if and only if x is added to $\mu^{\mathcal{I}(S)}$ for some shallow configuration μ, we can conclude that an accepting configuration must be reachable in T which contradicts our assumption (of the absence of such a computation sequence). Thus an inductive argument proves that $\mu \in x^{\mathcal{I}_n(S)}$ for $\{x\} = S^{\mathcal{I}(S)}$ implies that M reaches an accepting configuration on $\vec{w}$ in T.

Claim. Let $C = (R_1 \ldots R_k \mu)$ be a configuration. It holds for all $n \in \mathbb{N}$ that if $x \in (\exists R_1 . \ldots \exists R_k . \mu)^{\mathcal{I}_n(S)}$ and $\{x\} = S^{\mathcal{I}(S)}$ then M reaches C in the computation on $\vec{w}$ in its computation tree T.

Induction basis. Let $n = 1$ and $C = (R_1 \ldots R_k . \mu)$ for $\mu \in \mathcal{SC}$ be some configuration with $x \in (\exists R_1 . \ldots \exists R_k . \mu)^{\mathcal{I}_1(S)}$ and $\{x\} = S^{\mathcal{I}(S)}$. Thus the individual x is added to $(\exists R_1 . \ldots \exists R_k . \mu)^{\mathcal{I}_1(S)}$ because we have some axiom such that $\exists \Box . (\mathrm{bin}(0) \# \ldots \#) \sqsubseteq \exists R_1 . \ldots \exists R_k . \mu \in \mathcal{T}$ as we only have one step in this case. Hence C can be reached from the initial configuration $\Box 0 q_0 \# \ldots \#$ in one step via the transition that corresponds to the before mentioned axiom, i.e., $\Box 0 q_0 \# \ldots \# \vdash_M R_1 . \ldots R_k \mu$.

Induction step. Let $n > 1$ and assume the claim holds for all $m < n$. Now we have some configuration $C = (S_1 \ldots S_j R_1 \ldots R_k \mu)$ for $\mu \in \mathcal{SC}$ with $x \in (\exists S_1 . \ldots \exists S_j . \exists R_1 . \ldots \exists R_k . \mu)^{\mathcal{I}_n(S)}$ and $\{x\} = S^{\mathcal{I}(S)}$. By induction hypothesis we have some other configuration $C' = (S_1 \ldots S_j R \lambda)$ with $\lambda \in \mathcal{SC}$ from which C occurs in one step, i.e., $C' \vdash_M C$, and C is reachable on the computation of $M(\vec{w})$ and $x \in (\exists S_1 . \ldots \exists S_j . \exists R . \lambda)^{\mathcal{I}_{n-1}(S)}$. Thus we have also some axiom that adds x to $(\exists S_1 . \ldots \exists S_j . \exists R_1 . \ldots \exists R_k . \mu)^{\mathcal{I}_n(S)}$ in (**1**). This axiom is of the form $\exists R . \lambda \sqsubseteq \exists R_1 . \ldots \exists R_k . \mu \in \mathcal{T}$. As M reaches C' by induction hypothesis and C can be reached via one step from C' and x is an instance of $\exists S_1 . \ldots \exists S_j . \exists R_1 . \ldots \exists R_k . \mu$, M can also reach C within the computation on $\vec{w}$.

Hence this contradicts our assumption that M does not accept $\vec{w}$ and completes our proof. □

3.4. No quantifiers, TCSat-, OSat-, OCSat-*Results.*

In this subsection we investigate the complexity of the problems $\text{TCSAT}_\emptyset$, $\text{OSAT}_\emptyset$, and $\text{OCSAT}_\emptyset$.

Theorem 5. *Let B be a finite set of Boolean operators.*

1. *If $[B]$ contains all operators that are monotone and* 1*-separating, or all operators that are self-dual and affine, or all operators that are affine and* 0*-reproducing (i.e., $\mathsf{S}_{11} \subseteq [B]$ or $\mathsf{L}_3 \subseteq [B]$ or $\mathsf{L}_0 \subseteq [B]$), then $\star\text{SAT}^{\text{IND}}_\emptyset(B)$ is* NP*-complete.*

2. *If $[B]$ consists of either all conjunctions or all disjunctions, plus the constant $\bot$ or both constants (i.e., $[B] \in \{\mathsf{E}_0, \mathsf{E}, \mathsf{V}_0, \mathsf{V}\}$), then $\star\text{SAT}^{\text{IND}}_\emptyset(B)$ is* P*-complete.*

3. *If $[B]$ contains only identities, unary operators, and constants (i.e., $[B] \in \{\mathsf{I}_0, \mathsf{I}, \mathsf{N}_2, \mathsf{N}\}$), then $\star\text{SAT}^{\text{IND}}_\emptyset(B)$ is* NL*-complete.*

4. *Otherwise (if $[B]$ contains only* 1*-reproducing operators, i.e., $[B] \subseteq \mathsf{R}_1$), then $\star\text{SAT}^{\text{IND}}_\emptyset(B)$ is trivial.*

PROOF. NP-hardness for (1) follows from the respective $\text{TSAT}_\emptyset(B)$ results in Lemmas 13 and 14 in combination with Lemma 5 for the lower bound. The membership in NP is shown in Lemma 25.

The lower bounds for (2) result from $\text{TSAT}_\emptyset(\{\sqcap, \top, \bot\})$ and from $\text{TSAT}_\emptyset(\{\sqcup, \top, \bot\})$ shown in Lemmas 15 and 16 in combination with Lemma 5 while the upper bound applies due to $\text{OCSAT}_\exists(\{\sqcap, \top, \bot\})$ which is proven to be in P in Lemma 30.

The lower bound of (3) is proven in Lemma 27. The upper bound follows from Lemma 26.

(4) is due to Lemma 8. □

Lemma 24. *Let $((\mathcal{T}, \mathcal{A}), C)$ be an instance of $\text{OCSAT}_\emptyset(B)$, for an arbitrary finite set B of Boolean operators, and let $a_1, \ldots, a_n$ be the individuals in $\mathcal{A}$. Denote by $\mathcal{A}_i$ the restriction of $\mathcal{A}$ to all axioms about a_i. Then C is satisfiable with respect to $(\mathcal{T}, \mathcal{A})$ if and only if $(\mathcal{T}, \{C(a_0)\})$ and all $(\mathcal{T}, \mathcal{A}_i)$ are satisfiable, where a_0 is a fresh individual.*

PROOF. In the absence of quantifiers, $\mathcal{T}$ only makes propositional statements. The "$\Rightarrow$" direction is trivial and, for "$\Leftarrow$", we can assume w.l.o.g. that $(\mathcal{T}, \mathcal{A}_i)$ has a model with a singleton domain. The disjoint union of all these models is a model for $\big((\mathcal{T}, \mathcal{A}), C\big)$. $\square$

Lemma 25. *Let B be an arbitrary finite set of Boolean operators (i.e., $[B] \subseteq \mathsf{BF}$). Then* $\mathrm{OCS}\small{\mathrm{AT}}_{\emptyset}(B)$ *is in* NP.

PROOF. Due to Lemma 24, it suffices to check satisfiability of (a linear number of) $(\mathcal{T}, \mathcal{A})$ where $\mathcal{A}$ contains only one individual a. This problem can be further reduced to SAT, the satisfiability problem for propositional formulae. Due to Lemma 4, we can assume that $B = \{\sqcap, \neg\}$. We can now straightforwardly translate every axiom in $(\mathcal{T}, \mathcal{A})$ into a propositional formula by omitting a, replacing each atomic concept with a fresh atomic proposition p_A, and replacing $\bot, \top, \neg, \sqcap, \sqsubseteq$ with $\mathtt{0}, \mathtt{1}, \neg, \wedge, \rightarrow$, respectively. The conjunction of all these translations is equisatisfiable with $(\mathcal{T}, \mathcal{A})$ due to the singleton-domain observation made in the proof of Lemma 24. $\square$

Lemma 26. *Let B be a finite set of Boolean operators such that $[B]$ equals the set of all unary operators plus both constants (i.e., $[B] = \mathsf{N}$). Then* $\mathrm{OCSAT}_{\emptyset}(B)$ *is in* NL.

PROOF. Due to Lemma 24, it suffices to check satisfiability of (a linear number of) $(\mathcal{T}, \mathcal{A})$ where $\mathcal{A}$ contains only one individual a. This problem can be further reduced to 2SAT, the satisfiability problem for propositional formulae in 2CNF, i.e., conjunctions of clauses with two literals (a literal is an atomic proposition or its negation). Every 2CNF-clause can equivalently be rewritten into implication normal form (INF), i.e., $(\ell_1 \vee \ell_2)$ is equivalent to $({\sim}\ell_1 \rightarrow \ell_2)$, where ${\sim}p := \neg p$ and ${\sim}(\neg p) := p$, for any atomic proposition p. It is now easy to observe that every axiom in $(\mathcal{T}, \mathcal{A})$ directly corresponds to a 2CNF clause in INF: for TBox axioms, remove double negation and replace $\sqsubseteq$ with $\rightarrow$; for ABox axioms, additionally remove the reference to a and precede the remaining literal with "$\mathtt{1} \rightarrow$". The conjunction of all these clauses is equisatisfiable with $(\mathcal{T}, \mathcal{A})$ due to the singleton-domain observation made in the proof of Lemma 24. $\square$

Lemma 27. *Let B be a finite set of Boolean operators such that $[B]$ equals the set of all identities plus the constant $\bot$ (i.e., $[B] = \mathsf{I}_0$). Then* $\mathrm{TCSAT}_{\emptyset}(B)$ *is* NL*-hard.*

PROOF. In Lemma 17 $\text{TSAT}_\emptyset(\mathsf{I})$ was shown to be NL-hard. This result in combination with Lemma 5 proves the claim of this lemma by stating $\text{TSAT}_\emptyset(\mathsf{I}_0 \cup \{\top\}) \leq_{\text{m}}^{\log} \text{TCSAT}_\emptyset(\mathsf{I}_0)$ as $[\mathsf{I}_0 \cup \{\top\}] = \mathsf{I}$. □

3.5. One quantifier, TCSAT-, OSAT-, OCSAT-*Results.*

In this subsection we investigate the complexity of the problems $\text{TCSAT}_\exists$, $\text{TCSAT}_\forall$, $\text{OSAT}_\exists$, $\text{OSAT}_\forall$, $\text{OCSAT}_\exists$, and $\text{OCSAT}_\forall$.

Theorem 6. *Let B be a finite set of Boolean operators, and $Q \in \{\forall, \exists\}$.*

1. *If $[B]$ contains all operators that are monotone and* 1*-separating, or all unary operators, or all operators that are affine and* 0*-reproducing (i.e., $\mathsf{S}_{11} \subseteq [B]$ or $\mathsf{N}_2 \subseteq [B]$ or $\mathsf{L}_0 \subseteq [B]$), then $\star\text{SAT}_Q^{\text{IND}}(B)$ is* EXPTIME-*complete.*

2. *If $[B]$ contains the constant $\bot$ or both constants (i.e., $B \in \{\mathsf{I}_0, \mathsf{I}\}$), then $\star\text{SAT}_Q^{\text{IND}}(B)$ is* P*-complete.*

3. *If $[B]$ consists of all conjunctions plus the constant $\bot$ or both constants (i.e., $[B] \in \{\mathsf{E}_0, \mathsf{E}\}$), then $\star\text{SAT}_\forall^{\text{IND}}(B)$ is* EXPTIME-*complete, and $\star\text{SAT}_\exists^{\text{IND}}(B)$ is* P*-complete.*

4. *If $[B]$ consists of all disjunctions plus the constant $\bot$ or both constants (i.e., $[B] \in \{\mathsf{V}_0, \mathsf{V}\}$), then $\star\text{SAT}_\exists^{\text{IND}}(B)$ is* EXPTIME-*complete, and $\star\text{SAT}_\forall^{\text{IND}}(B)$ is* P*-complete.*

5. *If $[B]$ contains only* 1*-reproducing operators (that is, $[B] \subseteq \mathsf{R}_1$), then $\star\text{SAT}_Q^{\text{IND}}(B)$ is trivial.*

PROOF. In (1), EXPTIME-hardness of $\star\text{SAT}_Q^{\text{IND}}(\mathsf{S}_{11})$ follows from Lemma 19 (EXPTIME-completeness of $\text{TSAT}_Q(\mathsf{M})$) together with the Lewis Trick from Lemma 5. EXPTIME-hardness of $\star\text{SAT}_Q^{\text{IND}}(\mathsf{N}_2)$ is due to Lemma 22. EXPTIME-hardness of $\star\text{SAT}_Q^{\text{IND}}(\mathsf{L}_0)$ follows from Lemma 22, which implies EXPTIME-completeness of $\text{TSAT}_Q(\mathsf{L})$, together with the Lewis Trick from Lemma 5. EXPTIME-membership follows from Lemmas 7 and 4, and applies to Items 3 and 4, too.

In (2), P-hardness follows from Theorem 4 (2) and applies to Items 3 and 4, too. P-membership follows from Items 3 and 4.

In (3), P-membership of $\star\text{SAT}_\exists^{\text{IND}}(\mathsf{E})$ follows from Lemma 30. EXPTIME-hardness of $\star\text{SAT}_\forall^{\text{IND}}(\mathsf{E}_0)$ is proven in Lemma 28.

In (4), P-membership of $\star\textsc{Sat}_\forall^{\text{IND}}(\mathsf{V})$ follows from Lemma 31. ExpTime-hardness of $\star\textsc{Sat}_\exists^{\text{IND}}(\mathsf{V}_0)$ is proven in Lemma 29.

(5) is due to Lemma 8. □

Theorem 6 shows one reason why the logics in the $\mathcal{EL}$ family have been much more successful as "small" logics with efficient reasoning methods than the $\mathcal{FL}$ family: the combination of the $\forall$ with conjunction is intractable, while $\exists$ and conjunction are still in polynomial time. Again, separating either conjunction and disjunction, or the constants is crucial for tractability.

Lemma 28. *Let B be a finite set of Boolean operators such that $[B]$ equals the set of all conjunctions plus the constant $\bot$ (i.e., $[B] = \mathsf{E}_0$). Then* $\mathrm{TCS}\textsc{at}_\forall(B)$ *is* ExpTime*-hard.*

proof. As in the proof of Lemma 20, we reduce from the complement of $\mathrm{TS}\textsc{ubs}_\forall(\{\sqcap\})$, which is $\mathcal{FL}_0$ TBox subsumption. The following reduction, given an instance $(\mathcal{T}, C, D)$ of $\mathrm{TS}\textsc{ubs}_\forall(\{\sqcap\})$, is obvious: $\mathcal{T} \not\models C \sqsubseteq D$ iff $C \sqcap D'$ is satisfiable w.r.t. $\mathcal{T} \cup \{D \sqcap D' \sqsubseteq \bot\}$. □

Lemma 29. *Let B be a finite set of Boolean operators such that $[B]$ equals the set of all disjunctions plus the constant $\bot$ (i.e., $[B] = \mathsf{V}_0$). Then* $\mathrm{TCS}\textsc{at}_\exists(B)$ *is* ExpTime*-hard.*

proof. As in the proof of Lemma 20, we reduce from the complement of $\mathrm{TS}\textsc{ubs}_\forall(\{\sqcap\})$. Let $(\mathcal{T}, C, D)$ be an instance of $\mathrm{TS}\textsc{ubs}_\forall(\{\sqcap\})$. As in the proof of Lemma 6, we take an arbitrary concept E in $\mathcal{T}$, including C, D, and transform it into a concept $E^\neg$ by converting $\neg E$ into negation normal form and replacing every negated atomic concept $\neg A$ with a fresh atomic concept A'. This transformation replaces all Boolean operators and quantifiers with their duals. Let $\mathcal{T}^{\text{con}} := \{F^\neg \sqsubseteq E^\neg \mid (E \sqsubseteq F) \in \mathcal{T}\}$, and $\mathcal{T}^{\text{con}\prime}$ be the substitutions of the negated concepts. We can argue as in the proof of Lemma 6 that $\mathcal{T} \models C \sqsubseteq D$ if and only if $\mathcal{T}^{\text{con}\prime} \models D^\neg \sqsubseteq C^\neg$.

We now consider the TBox

$$\mathcal{T}' := \mathcal{T}^{\text{con}\prime} \cup \{\exists R.C^\neg \sqsubseteq \bot, E \sqsubseteq \exists R.D^\neg\},$$

for a fresh concept name E and role R. It remains to prove the following claim.

Claim. $\mathcal{T}^{\text{con}\prime} \not\models D^\neg \sqsubseteq C^\neg$ if and only if E is satisfiable w.r.t. $\mathcal{T}'$.

Proof of Claim. For the "⇒" direction, assume that $\mathcal{T}^{\text{con}\prime} \not\models D^{\neg} \sqsubseteq C^{\neg}$. Then there is a model $\mathcal{I}$ of $\mathcal{T}^{\text{con}\prime}$ with an instance x of $D^{\neg}$ that is not an instance of $C^{\neg}$. Consider the interpretation $\mathcal{J}$ with $\Delta^{\mathcal{J}} = \Delta^{\mathcal{I}}$ $R^{\mathcal{J}} = \{(x,x)\}$, $E^{\mathcal{J}} = \{x\}$, and $X^{\mathcal{J}} = X^{\mathcal{I}}$ for all other symbols. Clearly, $\mathcal{J}$ satisfies the additional axioms in $\mathcal{T}$ and interprets E as nonempty; hence E is satisfiable w.r.t. $\mathcal{T}'$.

For the "⇐" direction, assume that E is satisfiable w.r.t. $\mathcal{T}'$. Then there is a model $\mathcal{I}$ of $\mathcal{T}'$ with $E^{\mathcal{I}} \neq \emptyset$. Obviously, $\mathcal{T}$ is also a model of $\mathcal{T}^{\text{con}\prime}$. The nonemptiness of $E^{\mathcal{I}}$ together with the two additional axioms implies that there is an instance of $(D^{\neg})^{\mathcal{I}} \setminus (C^{\neg})^{\mathcal{I}}$; hence $\mathcal{T}^{\text{con}\prime} \not\models D^{\neg} \sqsubseteq C^{\neg}$. □

Lemma 30. *Let B be a finite set of Boolean operators such that $[B]$ equals the set of all conjunctions plus both constants (i.e., $[B] = \mathsf{E}$). Then* $\text{OCS}\textsc{at}_{\exists}(B)$ *is in* P.

PROOF. To provide an algorithm running in polynomial time, we will reduce the given problem to the complement of $\mathcal{EL}^{++}$ TBox subsumption, which is P-complete [8]. The logic $\mathcal{EL}^{++}$ is $\mathcal{ALC}$ restricted to the operators $\exists, \sqcap, \top, \bot$ and extended with nominals and other features that lead beyond $\mathcal{ALC}$. Nominals are singleton concepts $\{a\}$ with $a \in \mathsf{N_I}$, having the semantics $\{a\}^{\mathcal{I}} = \{a^{\mathcal{I}}\}$; the other features are not used in our reduction.

The reduction works as follows:

$$((\mathcal{T},\mathcal{A}),C) \in \text{OCS}\textsc{at}_{\exists}(B) \iff \mathcal{T} \cup \{\top \sqsubseteq \mathcal{C}_{\mathcal{A}}\} \not\models C \sqsubseteq \bot,$$

where $\mathcal{T}$ is a TBox, $\mathcal{A}$ is an ABox, and

$$\mathcal{C}_{\mathcal{A}} := \bigsqcap_{C(a)\in\mathcal{A}} \exists U.(\{a\} \sqcap C) \sqcap \bigsqcap_{R(a,b)\in\mathcal{A}} \exists U.(\{a\} \sqcap \exists R.\{b\})$$

is the concept constructed as in [7] from $\mathcal{A}$, using a fresh role name U. □

Lemma 31. *Let B be a finite set of Boolean operators such that $[B]$ equals the set of all disjunctions plus both constants (i.e., $[B] = \mathsf{V}$). Then* $\text{OCS}\textsc{at}_{\forall}(B)$ *is in* P.

PROOF. Here we use the result from Lemma 30 and reduce to the dual problem $\text{OCS}\textsc{at}_{\exists}(\mathsf{E})$. Consider an ontology $(\mathcal{T},\mathcal{A})$ where $\mathcal{T}$ is a TBox and $\mathcal{A}$ an ABox, and a concept C as the given instance of $\text{OCS}\textsc{at}_{\forall}(B)$. W.l.o.g. assume C to be atomic. Now first construct the new terminology $\mathcal{T}'$ similarly to Lemma 6(2). Then add for each $A \in \mathsf{N_C}$ and hence each A' the GCIs $A \sqcap A' \sqsubseteq \bot$ to ensure they are disjoint. Denote this change by the terminology $\mathcal{T}''$. Then it holds $((\mathcal{T},\mathcal{A}),C) \in \text{OCS}\textsc{at}_{\forall}(B) \iff ((\mathcal{T}'',\mathcal{A}),C') \in \text{OCS}\textsc{at}_{\exists}(\mathsf{E})$. □

$\mathrm{TSat}_{\mathcal{Q}}(B)$	I	V	E	N/N_2	M	L_3 to BF	else
$\mathcal{Q} = \emptyset$	NL[17,18]	P[15,16,30]		NL[17,26]	NP[13,14,25]		triv.[8,9]
$\mathcal{Q} = \{\exists\}$	P[23,30]	Exp[7,21]	P[23,30]	Exp[7,19,22]			triv.[8,9]
$\mathcal{Q} = \{\forall\}$	P[4,31]		Exp[7,20,22]				triv.[8,9]
$\mathcal{Q} = \{\exists, \forall\}$	Exp[2]						triv.[8,9]

$\star\mathrm{Sat}^{\mathrm{IND}}_{\mathcal{Q}}(B)$	I/I_0	V/V_0	E/E_0	N/N_2	S_{11} to M	L_3/L_0 to BF	else
$\mathcal{Q} = \emptyset$	NL[26,27]	P[15,16,30]		NL[26,27]	NP[13,14,25]		triv.[8]
$\mathcal{Q} = \{\exists\}$	P[23,30]	Exp[7,29]	P[23,30]	Exp[7,19,22]			triv.[8]
$\mathcal{Q} = \{\forall\}$	P[4,31]		Exp[7,22,28]				triv.[8]
$\mathcal{Q} = \{\exists, \forall\}$	Exp[2]						triv.[8]

Table 3: Complexity overview for all Boolean function and quantifier fragments. All results are completeness results for the given complexity class. The superscripts point to the respective theorem or lemma. Exp is an abbreviation of ExpTime.

4. Conclusion

Table 3 gives an overview of our results. Figures 1 and 2 show how the results arrange in Post's lattice.

With Theorems 2 to 6, we have completely classified the satisfiability problems connected to arbitrary terminologies and concepts for $\mathcal{ALC}$ fragments obtained by arbitrary sets of Boolean operators and quantifiers. In particular, we improved and finished the study of [30]. In more detail, we achieved a dichotomy for all problems using both quantifiers (ExpTime-complete vs. trivial fragments), a trichotomy when only one quantifier is allowed (trivial, ExpTime-, and P-complete fragments), and a quatrochotomy in the absence of quantifiers, ranging over trivial, NL-complete, P-complete, and NP-complete fragments. Figures 1 and 2 show how our results arrange in Post's lattice.

Furthermore the connection to established fragments of $\mathcal{ALC}$, e.g., $\mathcal{FL}$ and $\mathcal{EL}$ now enriches the complexity landscape by a generalization of the known (in-)tractability results. Our complexity classification improves the overall understanding of where the tractability border lies: the most important lesson learnt is that (a) the separation of both quantifiers *and* conjunction and disjunction, or (b) the sole use of negation or the constants, is the only way to achieve tractability in our setting. More precisely, the maximal $\mathcal{ALC}$ fragments for which satisfiability with respect to theories is tractable but non-trivial are determined by the operator sets $\{\top, \bot, \sqcap, \exists\}$, $\{\top, \bot, \sqcup, \forall\}$, and $\{\top, \bot, \neg\}$. These sets represent the logic $\mathcal{EL}$ extended by $\bot$, its dual, and a very restricted Boolean DL. We have already explained in the Introduction that we consider this insight as a systematic underpinning of the folklore knowledge that the $\mathcal{EL}$ and DL-Lite families are the only reasonably useful tractable $\mathcal{ALC}$-fragments. For subsumption, which is not interreducible with satisfiability under restricted Boolean operators, the tractable cases are essentially the same [28].

If we compare the results of our study with similar analyses using *Post's lattice* of propositional logic [27], Linear Temporal Logic [11], modal logic [24] and hybrid logic [29], our study shows intractable fragments considerably closer to the bottom of the lattice—down to the I-clones, which contain only projections and constants. This contrast is particularly striking in the closely related cases of modal logic and the more expressive hybrid logic, where [24; 29] established that intractability "requires" at least conjunction. However, this comparison is slightly distorted because the satisfiability problem considered in the cited studies corresponds to the concept satisfiability problem for description logics. The other four satisfiability problems in this study, denoted by $\star$SAT, are those with the low tractability border. This is not surprising; it confirms the expressive power implicit in terminologies and assertional boxes: restricted to only the Boolean function *false* besides both quantifiers we are still able to encode EXPTIME-hard problems into the decision problems that have a TBox and a concept as input. Thus perhaps the strongest source of intractability can be found in the fact that theories with general concept inclusions already express limited implication and conjunction, and not in the set of allowed Boolean functions alone.

For future work, it would be interesting to see whether the picture changes if the use of general concept inclusions is restricted, for example, to acyclic terminologies—theories where axioms are cycle-free definitions $A \equiv C$ with A being atomic. Theories so restricted are sufficient for establishing taxonomies. Concept satisfiability for $\mathcal{ALC}$ w.r.t. acyclic terminologies is still PSPACE-complete [10; 16]. Is the tractability border the same under this restriction? One could also look at fragments with unqualified quantifiers, e.g., $\mathcal{ALU}$ or the DL-Lite family, which are not covered by the current analysis. Furthermore, since the standard reasoning tasks are not always interreducible under restricted Boolean operators, a similar classification for other decision problems such as concept subsumption is pending.

Acknowledgements

We thank Peter Lohmann, Carsten Lutz, and the anonymous referees for very helpful comments and suggestions.

References

[1] A. Artale, D. Calvanese, R. Kontchakov, and M. Zakharyaschev. DL-Lite in the light of first-order logic. In *Proc. AAAI*, pages 361–366, 2007.

[2] A. Artale, D. Calvanese, R. Kontchakov, and M. Zakharyaschev. Adding weight to DL-Lite. In *Proc. DL*, CEUR-WS, 2009.

[3] A. Artale, D. Calvanese, R. Kontchakov, and M. Zakharyaschev. The DL-Lite family and relations. *JAIR*, 36:1–69, 2009.

[4] F. Baader. Using automata theory for characterizing the semantics of terminological cycles. *Ann. Math. Artif. Intell.*, 18(2-4):175–219, 1996.

[5] F. Baader. Terminological cycles in a description logic with existential restrictions. In *Proc. IJCAI*, pages 325–330, 2003.

[6] F. Baader, S. Brandt, and C. Lutz. Pushing the $\mathcal{EL}$ envelope. In *Proc. IJCAI*, pages 364–369, 2005.

[7] F. Baader, S. Brandt, and C. Lutz. Pushing the $\mathcal{EL}$ envelope. In *LTCS–Report*, volume 05-01, 2005.

[8] F. Baader, S. Brandt, and C. Lutz. Pushing the $\mathcal{EL}$ envelope further. In *Proc. OWLED DC*, 2008.

[9] F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi, and P. F. Patel-Schneider. *The Description Logic Handbook: Theory, Implementation, and Applications.* Cambridge University Press, 2003.

[10] F. Baader and B. Hollunder. A terminological knowledge representation system with complete inference algorithms. In *Proc. PDK*, volume 567 of *LNCS*, pages 67–86. Springer, 1991.

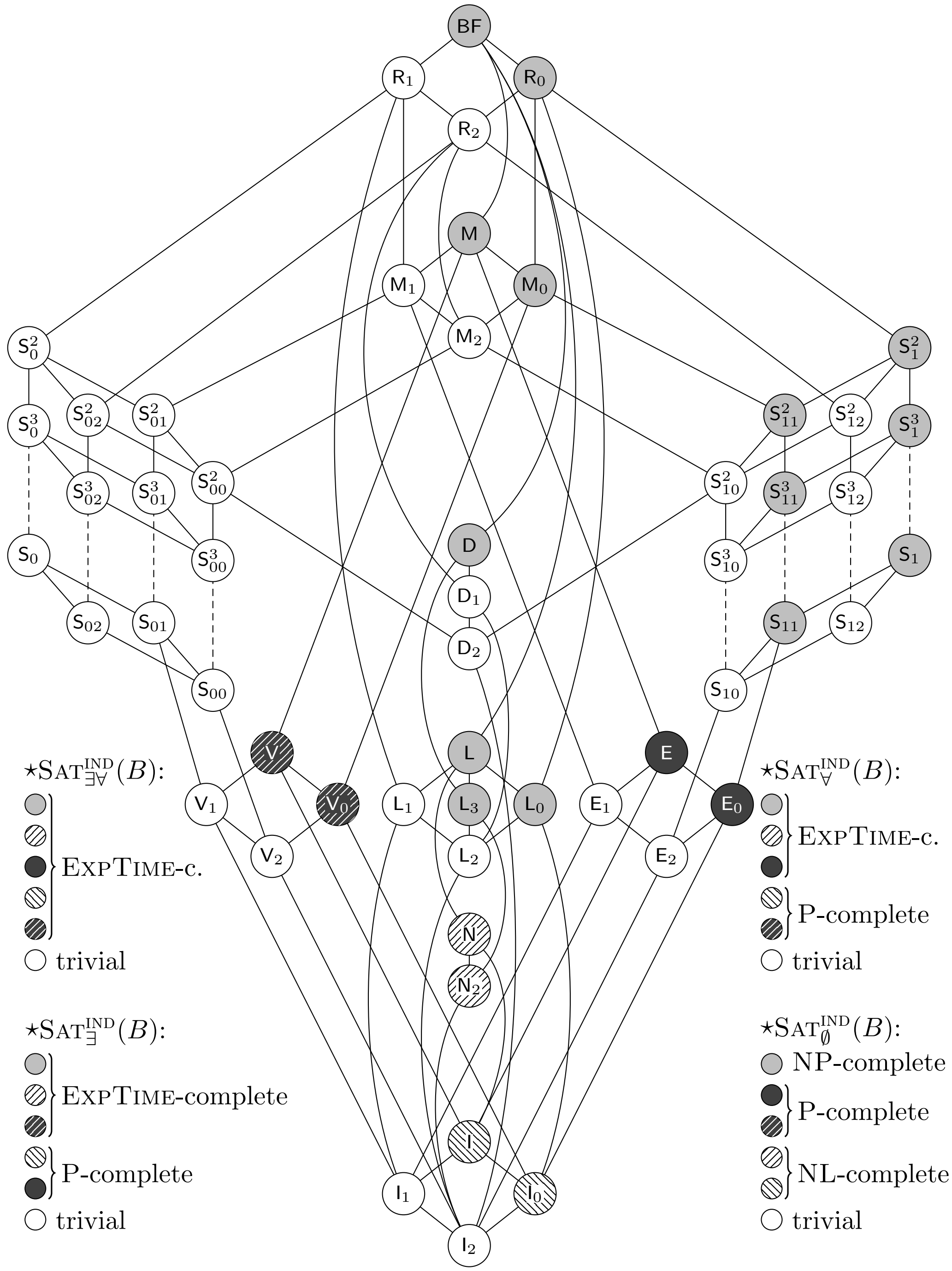

Figure 1: Complexity of $\star\mathrm{SAT}^{\mathrm{IND}}_{\mathcal{Q}}(B)$.

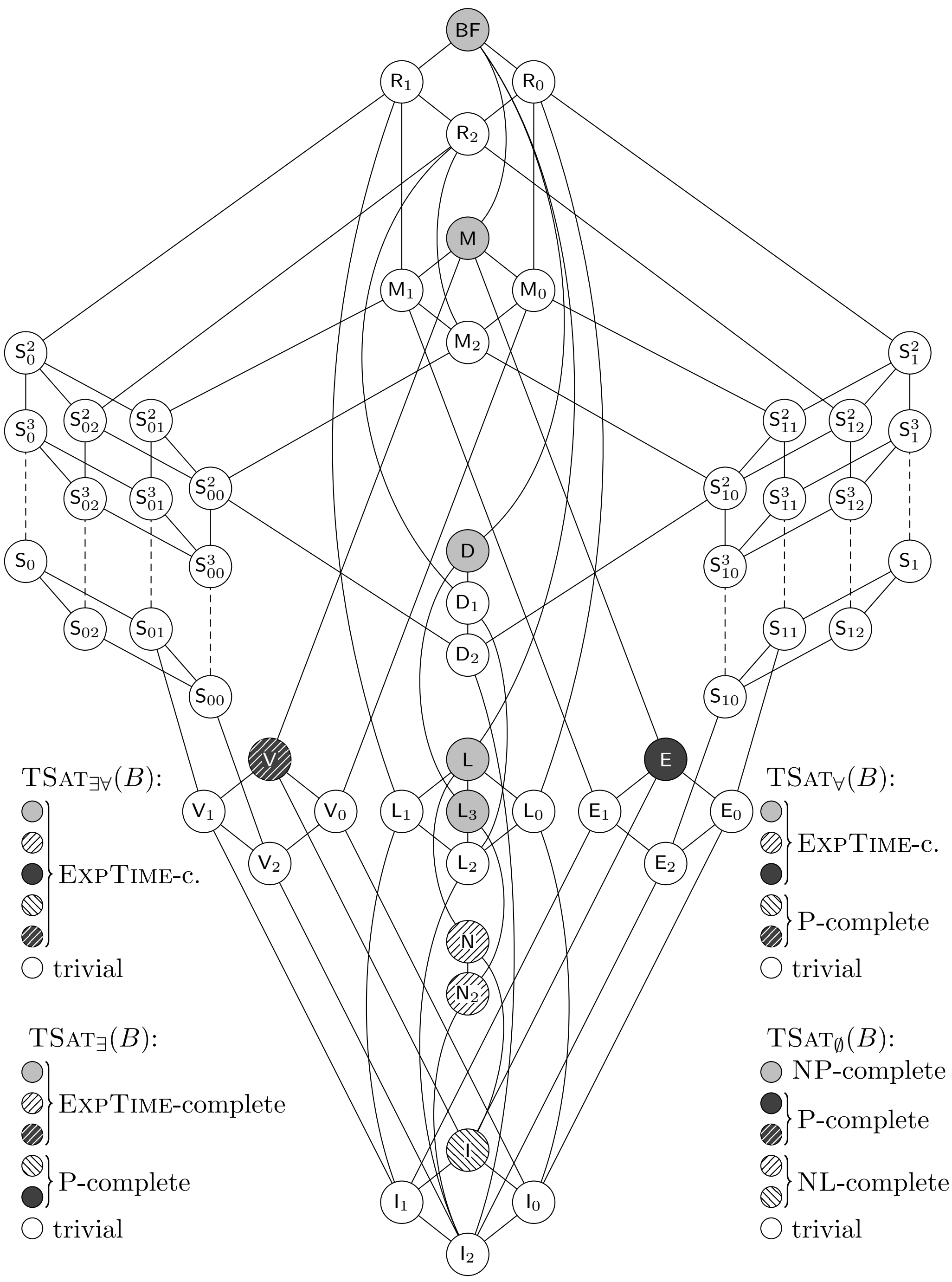

Figure 2: Complexity of $\mathrm{TSAT}_{\mathcal{Q}}(B)$.

[11] M. Bauland, T. Schneider, H. Schnoor, I. Schnoor, and H. Vollmer. The complexity of generalized satisfiability for Linear Temporal Logic. *LMCS*, 5(1), 2009.

[12] O. Beyersdorff, A. Meier, M. Thomas, and H. Vollmer. The Complexity of Propositional Implication. *IPL*, 109(18):1071–1077, 2009.

[13] E. Böhler, N. Creignou, S. Reith, and H. Vollmer. Playing with Boolean blocks, part I: Post's lattice with applications to complexity theory. *ACM-SIGACT Newsletter*, 34(4):38–52, 2003.

[14] S. Brandt. Polynomial time reasoning in a description logic with existential restrictions, GCI axioms, and—what else? In *Proc. ECAI*, pages 298–302, 2004.

[15] S. Brandt. Reasoning in $\mathcal{ELH}$ w.r.t. general concept inclusion axioms. LTCS-Report LTCS-04-03, Dresden University of Technology, Germany, 2004.

[16] D. Calvanese. Reasoning with inclusion axioms in description logics: Algorithms and complexity. In *Proc. ECAI*, pages 303–307. Wiley, 1996.

[17] D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, and R. Rosati. DL-Lite: Tractable description logics for ontologies. In *Proc. AAAI*, pages 602–607, 2005.

[18] S. Cook. Characterizations of pushdown machines in terms of time-bounded computers. *Journal of the ACM*, 18(1):4–18, January 1971.

[19] F. M. Donini. Complexity of reasoning. In *Description Logic Handbook* [9], pages 96–136.

[20] F. M. Donini, M. Lenzerini, D. Nardi, B. Hollunder, W. Nutt, and A. Marchetti-Spaccamela. The complexity of existential quantification in concept languages. *AI*, 53(2-3):309–327, 1992.

[21] F. M. Donini, M. Lenzerini, D. Nardi, and W. Nutt. The complexity of concept languages. *Inf. Comput.*, 134(1):1–58, 1997.

[22] F. M. Donini and F. Massacci. EXPTIME tableaux for $\mathcal{ALC}$. *AI*, 124(1):87–138, 2000.

[23] R. Givan, D. McAllester, C. Wittny, and D. Kozen. Tarskian set constraints. *Information and Computation*, 174:105–131, 2002.

[24] E. Hemaspaandra, H. Schnoor, and I. Schnoor. Generalized modal satisfiability. *Journal of Computer and System Sciences*, 76(7):561–578, 2010.

[25] M. Hofmann. Proof-theoretic approach to description-logic. In *Proc. LICS*, pages 229–237, 2005.

[26] Y. Kazakov and H. de Nivelle. Subsumption of concepts in $\mathcal{FL}_0$ for (cyclic) terminologies with respect to descriptive semantics is PSPACE-complete. In *Proc. DL*, CEUR-WS, 2003.

[27] H. Lewis. Satisfiability problems for propositional calculi. *Math. Sys. Theory*, 13:45–53, 1979.

[28] A. Meier. *On the Complexity of Modal Logic Variants and their Fragments.* PhD thesis, Leibniz University of Hannover, 2011.

[29] A. Meier, M. Mundhenk, T. Schneider, M. Thomas, V. Weber, and F. Weiss. The complexity of satisfiability for fragments of hybrid logic—Part I. *Journal of Applied Logic*, 8(4):409–421, 2010.

[30] A. Meier and T. Schneider. The complexity of satisfiability for sub-Boolean fragments of $\mathcal{ALC}$. In *Proc. of DL-2010.* CEUR-WS, 2010.

[31] B. Nebel. Terminological reasoning is inherently intractable. *AI*, 43(2):235–249, 1990.

[32] C. H. Papadimitriou. *Computational Complexity.* Addison-Wesley, 1994.

[33] E. Post. The two-valued iterative systems of mathematical logic. *Ann. Math. Studies*, 5:1–122, 1941.

[34] V. R. Pratt. A practical decision method for propositional dynamic logic: Preliminary report. In *STOC*, pages 326–337. ACM, 1978.

[35] T. J. Schaefer. The complexity of satisfiability problems. In *Proc. STOC*, pages 216–226. ACM Press, 1978.

[36] K. Schild. Terminological cycles and the propositional μ-calculus. In *Proc. KR*, pages 509–520, 1994.

[37] M. Schmidt-Schauß and G. Smolka. Attributive concept descriptions with complements. *AI*, 48(1):1–26, 1991.

[38] H. Schnoor. *Algebraic Techniques for Satisfiability Problems*. PhD thesis, Leibniz University of Hannover, 2007.

[39] M. Y. Vardi and P. Wolper. Automata-theoretic techniques for modal logics of programs. *JCSS*, 32(2):183–221, 1986.