# Temporal Query Entailment in the Description Logic $\mathcal{SHQ}$

Franz Baader, Stefan Borgwardt, Marcel Lippmann

*Institute of Theoretical Computer Science, Technische Universität Dresden, 01062 Dresden, Germany*

**Abstract**

Ontology-based data access (OBDA) generalizes query answering in databases towards deductive entailment since (i) the fact base is not assumed to contain complete knowledge (i.e., there is no closed world assumption), and (ii) the interpretation of the predicates occurring in the queries is constrained by axioms of an ontology. OBDA has been investigated in detail for the case where the ontology is expressed by an appropriate Description Logic (DL) and the queries are conjunctive queries. Motivated by situation awareness applications, we investigate an extension of OBDA to the temporal case. As the query language we consider an extension of the well-known propositional temporal logic LTL where conjunctive queries can occur in place of propositional variables, and as the ontology language we use the expressive DL $\mathcal{SHQ}$. For the resulting instance of temporalized OBDA, we investigate both data complexity and combined complexity of the query entailment problem. In the course of this investigation, we also establish the complexity of consistency of Boolean knowledge bases in $\mathcal{SHQ}$.

*Keywords:* Description Logic, Ontology-Based Data Access, Linear Temporal Logic, Complexity, Data Complexity

## 1. Introduction

Situation awareness tools [1, 2] try to help the user to detect certain situations within a running system. Here "system" is seen in a broad sense: it may be a computer system, air traffic observed by radar, or a patient in an intensive care unit. From an abstract point of view, the system is observed by certain "sensors" (e.g., heart rate and blood pressure monitors for a patient), and the results of sensing, possibly already preprocessed and aggregated appropriately, are stored in a fact base. Based on the information available in the fact base, the situation awareness tool is supposed to detect certain predefined situations (e.g., heart rate very high and blood pressure low), which require a reaction (e.g., fetch a doctor or give medication).

In a simple setting, one could realize such a tool by using standard database techniques: the information obtained from the sensors is stored in a relational database, and the situations to be recognized are specified by queries in an appropriate query language (e.g., conjunctive queries [3]). However, in general we cannot assume that the sensors provide us with a complete description of the current state of the system, and thus the closed world assumption (CWA) employed by database systems (where facts not occurring in the database are assumed to be false) is not appropriate (since there may be facts for which it is not known whether they are true or false). In addition, though one usually does not have a complete specification of the working of the system (e.g., a complete biological model of a human patient), one has some knowledge about how the system works. This knowledge can be used to formulate constraints on the interpretation of the predicates used in the queries, which may cause more answers to be found.

Ontology-based data access [4, 5] addresses these requirements. The fact base is viewed to be a Description Logic ABox (which is not interpreted with the CWA), and an ontology, also formulated in an appropriate DL, constrains the interpretations of unary and binary predicates, called concepts and roles in the DL community. In OBDA, one usually assumes that the ABox is obtained from external data sources (in the case of situation awareness, the raw sensor data) through appropriate mappings (which in our case realize the preprocessing and aggregation of the sensor data), but for the purpose of this paper we abstract from the mapping step, assuming that the result of the preprocessing is explicitly represented in an ABox.

As an example, assume that the ABox $\mathcal{A}$ contains the following assertions about patient Bob:

$$\begin{gathered}\mathit{systolic\_pressure}(\mathit{BOB}, \mathit{P1}),\ \mathit{High\_pressure}(P1),\\ \mathit{history}(\mathit{BOB}, H1),\quad \mathit{Hypertension}(H1),\ \mathit{Male}(\mathit{BOB}),\end{gathered}$$

which say that Bob has high blood pressure (obtained from sensor data), and is male and has a history of hypertension (obtained from the patient records). In addition, we have an ontology that says that patients with high blood pressure have hypertension and that patients that currently have hypertension and also have a history of hypertension are

*Email addresses:* baader@tcs.inf.tu-dresden.de (Franz Baader), stefborg@tcs.inf.tu-dresden.de (Stefan Borgwardt), lippmann@tcs.inf.tu-dresden.de (Marcel Lippmann)

at risk of a heart attack:

$$\exists \mathit{systolic\_pressure}.\mathit{High\_pressure} \sqsubseteq \exists \mathit{finding}.\mathit{Hypertension}$$
$$\exists \mathit{finding}.\mathit{Hypertension} \sqcap \exists \mathit{history}.\mathit{Hypertension} \sqsubseteq \exists \mathit{risk}.\mathit{Myocardial\_infarction}$$

The situation we want to recognize for a given patient $x$ is whether this patient is a male person who is at risk of a heart attack. This situation can be described by the conjunctive query

$$\exists y.\mathit{Male}(x) \land \mathit{risk}(x, y) \land \mathit{Myocardial\_infarction}(y).$$

Given the information in the ABox and the axioms in the ontology, we can derive that Bob satisfies this query, i.e., he is a *certain answer* of the query. Obviously, without the ontology this answer could not be derived.

The complexity of *query entailment* w.r.t. an ontology, i.e., the complexity of checking whether a given tuple of individuals is a certain answer of a query in an ABox w.r.t. an ontology, has been investigated in detail for cases where the ontology is expressed in an appropriate DL and the query is a conjunctive query. One can either consider the *combined complexity*, which is measured in the size of the whole input (consisting of the query, the ontology, and the ABox), or the *data complexity*, which is measured in the size of the ABox only (i.e., the query and the ontology are assumed to be of constant size). The underlying assumption is that the query and the ontology are usually relatively small, whereas the size of the data may be huge. In the database setting (where there is no ontology and CWA is used), conjunctive query entailment is NP-complete w.r.t. combined complexity and in $\mathrm{AC}^0$ w.r.t. data complexity [3, 6]. For expressive DLs, the complexity of checking certain answers is considerably higher. For instance, for the well-known DL $\mathcal{ALC}$, the query entailment problem is ExpTime-complete w.r.t. combined complexity and co-NP-complete w.r.t. data complexity [7–9]. For this reason, the more light-weight DLs of the *DL-Lite* family have been developed, for which the entailment problem is still in $\mathrm{AC}^0$ w.r.t. data complexity, and for which computing certain answers can be reduced to answering first-order queries in the database setting [10].

Unfortunately, OBDA as described until now is not sufficient to achieve situation awareness. The reason is that the situations we want to recognize may depend on states of the system at different time points. For example, assume that we want to find male patients with a history of hypertension, i.e., patients that are male and at some previous time point had hypertension.[1] In order to express this kind of temporal queries, we propose to extend the well-known propositional temporal logic LTL [11] by allowing the use of conjunctive queries in place of propositional variables. For example, male patients with a history of hypertension can then be described by the query

$$\mathit{Male}(x) \land \bigcirc^{-} \Diamond^{-} (\exists y.\mathit{finding}(x, y) \land \mathit{Hypertension}(y)),$$

where $\bigcirc^{-}$ stands for "previous" and $\Diamond^{-}$ stands for "sometime in the past." We call the queries obtained this way temporal conjunctive queries (TCQs). These queries extend the temporal description logic $\mathcal{ALC}$-LTL introduced and investigated in [12]. In $\mathcal{ALC}$-LTL, only concept and role assertions (i.e., very restricted conjunctive queries without variables and existential quantification) can be used in place of propositional variables. As in [12], we also consider rigid concepts and roles, i.e., concepts and roles whose interpretation does not change over time. For example, we may want to assume that the concept *Male* is rigid, and thus a patient that is male now also has been male in the past and will stay male in the future.

Our overall setting for recognizing situations will thus be the following. In addition to a global ontology $\mathcal{T}$ (which describes properties of the system that hold at every time point, using the expressive DL $\mathcal{SHQ}$), we have a sequence of ABoxes $\mathcal{A}_0, \mathcal{A}_1, \ldots \mathcal{A}_n$, which (incompletely) describe the states of the system at the previous time points $0, 1, \ldots, n-1$ and the current time point $n$. The situation to be recognized is expressed by a temporal conjunctive query, as introduced above, which is evaluated w.r.t. the current time point $n$.

### *1.1. Related Work*

Our work combines results on atemporal conjunctive query answering w.r.t. DL ontologies with LTL as a temporal logic component. In the following, we describe relevant work in these two fields as well as similar approaches to temporal query answering, which have mainly been developed for the light-weight languages of the *DL-Lite* family.

We build on the the results about the complexity of conjunctive query entailment of [8, 13, 14] (see Sections 2.2 and 3 for details). Additionally, for our proofs it is not sufficient to use only the results, but we must also adapt the methods developed in these papers to show these results. For example, we adapt the constructions involving forest models and equivalence relations over individual names from [14], and we use the results about spoilers in $\mathcal{SHQ}^{\cap}$ from [8].

The temporal component of our query language is LTL [11]. As such, we adapt the automata construction for LTL satisfiability from [15, 16]. Our language also generalizes $\mathcal{ALC}$-LTL [12], which allows DL axioms in place of propositional variables, and in fact several constructions in the present paper are adaptations of those for $\mathcal{ALC}$-LTL, in particular the ones used to show Lemmata 4.3 and 6.4 in [12]. The latter result about the consistency of Boolean $\mathcal{ALC}$-knowledge bases is in turn an adaptation of Theorem 2.27 from [17]. Our hardness results for combined complexity also follow easily from the results in [12].

Instead of temporalizing the query language and using a global (atemporal) ontology, one can also temporalize

[1] Whereas in the previous example we have assumed that a history of hypertension was explicitly noted in the patient records, we now want to derive this information from previously stored information about blood pressure, etc.

the ontology language. Extensions of various description logics with temporal operators in concepts and axioms have been studied (see for example [17, 18]). A comprehensive survey of temporal description logics can be found in [19]. In [20], various light-weight DLs are extended by allowing temporal operators inside concepts. In addition to complexity results for temporal extensions of *DL-Lite*, it is also shown that reasoning easily becomes undecidable already in a small temporal extension of the description logic $\mathcal{EL}$. Although the *DL-Lite* family was developed with mainly query answering in mind, the complexity results in [20] are concerned with inference problems not involving queries.

In the literature, one can find several approaches to temporal query answering in description logics. In [21], temporal query answering over temporalized RDF triples [22] using an extension of the SPARQL query language is considered.

In [23], the very expressive temporalized DL $\mathcal{DLR}_{\mathcal{US}}$ is introduced, which is an extension of $\mathcal{DLR}$ that allows for temporal operators within concepts and roles. Moreover, the query containment problem of non-recursive Datalog queries under constraints defined in $\mathcal{DLR}_{\mathcal{US}}$ is investigated. It turns out that this problem is in general undecidable, but becomes decidable in the fragment $\mathcal{DLR}_{\mathcal{US}}^{-}$, where no temporal operators are allowed within roles. The query containment problem is then in 2-ExpTime, whereas satisfiability and subsumption in $\mathcal{DLR}_{\mathcal{US}}^{-}$ are ExpSpace-complete.

Following the ideas of [20], in [24] a temporal extension of *DL-Lite* is presented, which allows the temporal operators $\diamond^{-}$ and $\diamond$ on the left-hand side of GCIs and role inclusions. In this logic, first-order rewritability of CQs w.r.t. *DL-Lite*-knowledge bases is preserved from the atemporal case. Thus, techniques from temporal relational databases can be used to answer temporal queries that can refer to specific points in time.

An approach to temporalize query answering in *DL-Lite* that is more similar to the one considered in this paper is presented in [25]. There, CQs are used as atoms in a temporal formula that does not use negation. This allows easy reuse of results about atemporal first-order rewritability in *DL-Lite*. The paper also presents an algorithm to answer such temporal queries over temporal databases, which generalizes an algorithm from [26, 27].

A similar approach is pursued in [28] to combine a generic DL query component with a linear temporal dimension. To simplify the decision procedures, both components are decoupled via an autoepistemic modal operator. This allows to use atemporal query answering procedures as a black-box inside a temporal satisfiability algorithm.

### *1.2. Our Contribution*

We investigate both the combined and the data complexity of our temporal extension of OBDA, as sketched above, in three different settings: (i) both concepts and roles may be rigid; (ii) only concepts may be rigid; and (iii) neither concepts nor roles are allowed to be rigid. It is well-known that one can simulate rigid concept names by rigid role names [12], which is why there are only three cases to consider.

The complexity results for TCQ entailment obtained in this paper are summarized in Table 1. These results hold for all description logics between $\mathcal{ALC}$ and $\mathcal{SHQ}$. In fact, we show that the hardness results already hold for $\mathcal{ALC}$ and we prove the complexity upper bounds for the more expressive DL $\mathcal{SHQ}$.

$\mathcal{SHQ}$ extends $\mathcal{ALC}$ with transitive roles, subroles, and qualified number restrictions. In the conference paper [29], which is a precursor of the present paper, we showed these results for $\mathcal{ALC}$ only. From a practical point of view, we found the extension to $\mathcal{SHQ}$ interesting since the additional means of expressiveness are important for biomedical ontologies. For instance, one usually wants the part-of role (which is, e.g., extensively used in medical ontologies to define human anatomy) to be transitive, and it is also useful to distinguish the proper-part-of role from the part-of role and to declare that the former is a subrole of the latter [30]. Number restrictions can, among other things, be used to express that certain roles are functional. In our introductory example, it makes sense to require that a patient can have only one systolic blood pressure at each point in time. More general number restrictions can be used to express anatomical facts such as that humans have exactly two kidneys. From a more theoretical point of view, we wanted to know how far one can extend $\mathcal{ALC}$ without increasing the complexity of query entailment. $\mathcal{SHQ}$ is here the limit. If we add inverse roles, which are also quite useful when defining medical ontologies, then the combined complexity increases. In fact, for $\mathcal{ALCI}$ query entailment is already 2-ExpTime-complete w.r.t. combined complexity in the atemporal case [8]. For $\mathcal{SHOQ}$ (extending SHQ by nominals) and $\mathcal{SROQ}$ (further extending $\mathcal{SHOQ}$ by complex role inclusions), the best known upper bounds are respectively 2-ExpTime and 3-ExpTime [31, 32]. Also, we restrict the query language such that transitive roles (e.g. the part-of role) and roles having transitive subroles cannot *directly* be used in queries. The reason is that otherwise query entailment is known to be co-NExpTime-hard in $\mathcal{S}$ and 2-ExpTime-hard in $\mathcal{SH}$ even in the atemporal case [33]. Note, however, that such roles can be used indirectly since concept names whose definition in the global ontology involves such a role can be used in queries.

Though our complexity results are the same for $\mathcal{ALC}$ and $\mathcal{SHQ}$, and in principle the approaches used below to prove the upper bounds for $\mathcal{SHQ}$ are similar to the ones employed in [29, 34] for $\mathcal{ALC}$, the proof details are considerably more complex for $\mathcal{SHQ}$. In particular, the proof of Theorem 4.1 uses a construction different from that of Theorem 3.2 in [34] since in the presence of number restrictions it is not so easy to simply copy elements of a model while retaining the satisfaction of the knowledge base. Furthermore, the quasimodel construction in Section 6.3 uses new notions to deal with role axioms, and systems of linear equations to simulate the semantics of number restrictions.

Table 1: The complexity of simple TCQ entailment for all DLs between $\mathcal{ALC}$ and $\mathcal{SHQ}$.

| | *data complexity* | *combined complexity* |
|---|---|---|
| without rigid names | co-NP-complete (Corollary 4.2 and Theorem 4.13) | ExpTime-complete (Theorems 4.3 and 4.13) |
| without rigid role names | co-NP-complete (Corollary 4.2 and Theorem 5.2) | co-NExpTime-complete (Theorems 4.3 and 6.3) |
| with rigid names | co-NP-hard/in ExpTime (Corollary 4.2 and Theorem 4.15) | 2-ExpTime-complete (Theorems 4.3 and 4.15) |

For the combined complexity, the results obtained in the present paper are actually identical to the ones for $\mathcal{ALC}$-LTL [12], though the upper bounds are considerably harder to show. The data complexity results in Settings (ii) and (iii) coincide with the ones for atemporal query entailment, which is co-NP-complete w.r.t. data complexity. For Setting (i), we can show that the entailment problem is in ExpTime w.r.t. data complexity (in contrast to 2-ExpTime-completeness w.r.t. combined complexity), but we do not have a matching lower bound. To show the result for combined complexity in Setting (ii), we additionally establish the complexity of the atemporal problem of consistency of Boolean knowledge bases in $\mathcal{SHQ}$ extended with a limited form of role conjunctions.

Of the other related work mentioned in the previous subsection, the ones described in [23–25, 28] are most closely related to our work. Nevertheless, they differ from our approach in several ways:

- We consider the expressive DL $\mathcal{SHQ}$ instead of light-weight DLs such as *DL-Lite* [24, 25].
- We consider a temporal query language instead of a temporal ontology language [23, 24].
- In contrast to [28], we consider also the case of rigid concept and role names. In [24, 25], rigid names are also used, but in the context of light-weight DLs.

## 2. Preliminaries

In this section, we introduce the description logics $\mathcal{ALC}$ and $\mathcal{SHQ}$, conjunctive queries, and the temporal logic LTL. These are the main ingredients for our temporal query language, which will be defined in Section 3.

### 2.1. Description Logics

Description Logics (DLs) are a family of knowledge representation formalisms (for an introduction, see [35]). While our temporal query language can be parameterized with any DL, in this paper we consider the DLs between $\mathcal{ALC}$ and $\mathcal{SHQ}$ [36]. In the proof of Theorem 6.3, we additionally use the DL $\mathcal{SHQ}^\cap$ that extends $\mathcal{SHQ}$ with role conjunctions.

**Definition 2.1 (syntax of $\mathcal{SHQ}^\cap$).** Let $N_\mathrm{C}$, $N_\mathrm{R}$, and $N_\mathrm{I}$, be sets of *concept names*, *role names*, and *individual names*, respectively. The set of $\mathcal{SHQ}^\cap$*-concepts* is the smallest set such that

- all concept names $A \in N_\mathrm{C}$ are $\mathcal{SHQ}^\cap$-concepts, and
- if $C, D$ are $\mathcal{SHQ}^\cap$-concepts, $r, r_1, \dots, r_\ell \in N_\mathrm{R}$, and $n$ is a non-negative integer, then $\neg C$ (negation), $C \sqcap D$ (conjunction), $\exists(r_1 \cap \dots \cap r_\ell).C$ (existential restriction), and $\geq n\, r.C$ (at-least restriction) are also $\mathcal{SHQ}^\cap$-concepts.

A *general concept inclusion* in $\mathcal{SHQ}^\cap$ ($\mathcal{SHQ}^\cap$*-GCI*) is of the form $C \sqsubseteq D$, where $C, D$ are $\mathcal{SHQ}^\cap$-concepts. A *role inclusion* is of the form $r \sqsubseteq s$, and a *transitivity axiom* is of the form $\mathsf{trans}(r)$, where where $r$ and $s$ are role names. An *assertion* is of the form $A(a)$ (concept assertion) or $r(a, b)$ (role assertion), where $A \in N_\mathrm{C}$, $r \in N_\mathrm{R}$, and $a, b \in N_\mathrm{I}$. An $\mathcal{SHQ}^\cap$*-axiom* is either an $\mathcal{SHQ}^\cap$-GCI, a role inclusion, a transitivity axiom, or an assertion.

An $\mathcal{SHQ}^\cap$*-TBox* is a finite set of $\mathcal{SHQ}^\cap$-GCIs, an $\mathcal{SHQ}^\cap$*-RBox* is a finite set of role inclusions and transitivity axioms, and an *ABox* is a finite set of assertions. An $\mathcal{SHQ}^\cap$*-knowledge base* $\mathcal{K} = \langle \mathcal{A}, \mathcal{T}, \mathcal{R} \rangle$ consists of an ABox $\mathcal{A}$, an $\mathcal{SHQ}^\cap$-TBox $\mathcal{T}$, and an $\mathcal{SHQ}^\cap$-RBox $\mathcal{R}$. We denote the set of individual names occurring in an $\mathcal{SHQ}^\cap$-knowledge base $\mathcal{K}$ by $\mathsf{Ind}(\mathcal{K})$.

Other constructors that are often used in $\mathcal{SHQ}^\cap$ can be defined as follows:

- $\top := A \sqcup \neg A$ (top), where $A$ is an arbitrary, but fixed, concept name;
- $\bot := \neg\top$ (bottom);
- $C \sqcup D := \neg(\neg C \sqcap \neg D)$ (disjunction);
- $\forall(r_1 \cap \dots \cap r_\ell).C := \neg(\exists(r_1 \cap \dots \cap r_\ell).\neg C)$ (value restriction); and
- $\leq n\, r.C := \neg(\geq (n+1)\, r.C)$ (at-most restriction).

As mentioned above, most of the time, we consider the description logic $\mathcal{SHQ}$ that does not allow role conjunctions in existential restrictions, i.e., requires that $\ell = 1$. We sometimes restrict the DL under consideration, e.g., to

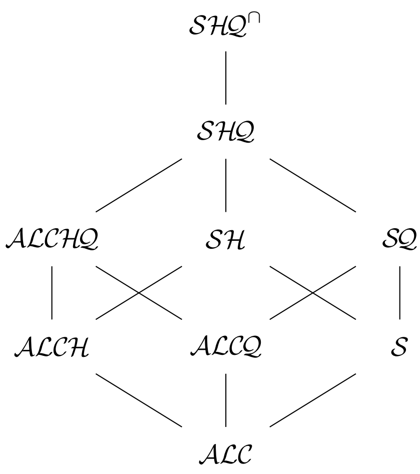


Figure 1: The relative expressivity of the DLs between $\mathcal{ALC}$ and $\mathcal{SHQ}^\cap$.

the sublogic $\mathcal{ALC}$ of $\mathcal{SHQ}^\cap$ which does not allow role conjunctions, transitivity axioms, role inclusions, or at-least restrictions, and then write, e.g., $\mathcal{ALC}$-knowledge base instead of $\mathcal{SHQ}^\cap$-knowledge base. The extension of $\mathcal{ALC}$ with transitivity axioms is usually denoted by $\mathcal{S}$. The letters $\mathcal{H}$ and $\mathcal{Q}$ respectively denote the presence of role inclusions and number restrictions. In Figure 1, all relevant DLs and their relations are depicted.

From now on, we consider an arbitrary (but fixed) DL between $\mathcal{ALC}$ and $\mathcal{SHQ}$, and therefore we often drop this prefix. Moreover, some notions, like interpretations and conjunctive queries, do not even depend on the DL under consideration.

**Definition 2.2 (semantics of $\mathcal{SHQ}^\cap$).** An interpretation is a pair $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$, consisting of a non-empty set $\Delta^\mathcal{I}$ (called *domain*) and an *interpretation function* $\cdot^\mathcal{I}$ that assigns to every $A \in N_\mathrm{C}$ a set $A^\mathcal{I} \subseteq \Delta^\mathcal{I}$, to every $r \in N_\mathrm{R}$ a binary relation $r^\mathcal{I} \subseteq \Delta^\mathcal{I} \times \Delta^\mathcal{I}$, and to every $a \in N_\mathrm{I}$ an element $a^\mathcal{I} \in \Delta^\mathcal{I}$ such that the *unique name assumption (UNA)* is satisfied, i.e., for all $a, b \in N_\mathrm{I}$ with $a \neq b$ we have $a^\mathcal{I} \neq b^\mathcal{I}$. The interpretation function is extended to concepts as follows:

- $(\neg C)^\mathcal{I} := \Delta^\mathcal{I} \setminus C^\mathcal{I}$;
- $(C \sqcap D)^\mathcal{I} := C^\mathcal{I} \cap D^\mathcal{I}$;
- $(\exists(r_1 \cap \dots \cap r_\ell).C)^\mathcal{I} := \{d \in \Delta^\mathcal{I} \mid$ there is an $e \in C^\mathcal{I}$ with $(d, e) \in r_1^\mathcal{I} \cap \dots \cap r_\ell^\mathcal{I}\}$; and
- $({\geq} n\, r.C)^\mathcal{I} := \{d \in \Delta^\mathcal{I} \mid |\{e \in C^\mathcal{I} \mid (d, e) \in r^\mathcal{I}\}| \geq n\}$.

An interpretation $\mathcal{I}$ is a *model* of an axiom $\alpha$ if

- $C^\mathcal{I} \subseteq D^\mathcal{I}$ for $\alpha = C \sqsubseteq D$;
- $r^\mathcal{I} \subseteq s^\mathcal{I}$ for $\alpha = r \sqsubseteq s$;
- $r^\mathcal{I} \circ r^\mathcal{I} \subseteq r^\mathcal{I}$, i.e., $r^\mathcal{I}$ is transitive, for $\alpha = \mathsf{trans}(r)$;
- $a^\mathcal{I} \in A^\mathcal{I}$ for $\alpha = A(a)$; and
- $(a^\mathcal{I}, b^\mathcal{I}) \in r^\mathcal{I}$ for $\alpha = r(a, b)$.

We say that $\mathcal{I}$ is a *model* of a set of axioms if it is a model of all axioms contained in it, and $\mathcal{I}$ is a *model* of a knowledge base $\mathcal{K} = \langle \mathcal{A}, \mathcal{T}, \mathcal{R} \rangle$ if it is a model of $\mathcal{A}$, $\mathcal{T}$, and $\mathcal{R}$. We write $\mathcal{I} \models \alpha$ if $\mathcal{I}$ is a model of the axiom $\alpha$, and similarly for sets of axioms and knowledge bases.

A knowledge base is *consistent* if it has a model. An axiom $\alpha$ is *entailed* by a knowledge base $\mathcal{K}$ (written $\mathcal{K} \models \alpha$) if all models of $\mathcal{K}$ are also models of $\alpha$, and similarly for sets of axioms.

Motivated by the semantics of GCIs, we often use the expression $C \equiv D$ for two concepts $C$ and $D$ to abbreviate the two GCIs $C \sqsubseteq D$ and $D \sqsubseteq C$, restricting any model to interpret $C$ and $D$ by the same set.

Recall that, contrary to the usual definition of concept assertions, we only allow concept names to occur in them, but no complex concepts. One can circumvent this by introducing abbreviations $A$ for complex concepts $C$ via $A \equiv C$. However, this restriction is useful to separate the influence of the ABox and the TBox on the complexity of reasoning problems.

If one or more components of a knowledge base $\langle \mathcal{A}, \mathcal{T}, \mathcal{R} \rangle$ are empty, we may also shorten it to, e.g., $\langle \mathcal{T}, \mathcal{R} \rangle$ or $\mathcal{R}$. Given an RBox $\mathcal{R}$, we say that a role name $r$ is *transitive* (w.r.t. $\mathcal{R}$) if $\mathcal{R} \models \mathsf{trans}(r)$, and $r$ is a *subrole* of a role name $s$ (w.r.t. $\mathcal{R}$) if $\mathcal{R} \models r \sqsubseteq s$. Furthermore, $r$ is *simple* (w.r.t. $\mathcal{R}$) if it has no transitive subrole. Entailments of the form $\mathcal{R} \models \mathsf{trans}(r)$ and $\mathcal{R} \models r \sqsubseteq s$ can be decided in polynomial time in the size of $\mathcal{R}$ [36].

Unfortunately, consistency of knowledge bases in $\mathcal{SHQ}$ is undecidable, even if all at-least restrictions are unqualified, i.e., of the form ${\geq} n\, r.\top$ [36]. One cause of undecidability is the occurrence of non-simple role names in such restrictions. To regain decidability, role names occurring in number restrictions are therefore usually required to be simple. In the following, we also make this restriction to the syntax of $\mathcal{SHQ}^\cap$. We further require that role conjunctions with at least two conjuncts contain only simple roles.

Under this assumption, the problem of deciding the consistency of $\mathcal{SHQ}$-knowledge bases is in ExpTime, even if the numbers occurring in at-least restrictions are given in binary encoding [37]. On the other hand, the problem is ExpTime-hard already in $\mathcal{ALC}$ [35].

The notion of a knowledge base can be generalized to arbitrary Boolean combinations of axioms.

**Definition 2.3 (Boolean knowledge base).** The pair $\mathcal{B} = \langle \Psi, \mathcal{R} \rangle$ is called a *Boolean knowledge base* if $\mathcal{R}$ is an RBox and $\Psi$ is a *Boolean axiom formula (w.r.t. $\mathcal{R}$)*. The set of Boolean axiom formulae (w.r.t. $\mathcal{R}$) is the smallest set such that

- every assertion is a Boolean axiom formula,

- every GCI in which number restrictions only contain simple role names (w.r.t. $\mathcal{R}$) is a Boolean axiom formula, and
- if $\Psi_1$ and $\Psi_2$ are Boolean axiom formulae, then so are $\neg\Psi_1$ (negation) and $\Psi_1 \wedge \Psi_2$ (conjunction).

The interpretation $\mathcal{I}$ is a *model* of the Boolean knowledge base $\langle\Psi, \mathcal{R}\rangle$ if $\mathcal{I} \models \mathcal{R}$ and $\mathcal{I} \models \Psi$ holds, which is also defined inductively: $\mathcal{I} \models \neg\Psi_1$ iff $\mathcal{I} \not\models \Psi_1$, and $\mathcal{I} \models \Psi_1 \wedge \Psi_2$ iff $\mathcal{I} \models \Psi_1$ and $\mathcal{I} \models \Psi_2$. A Boolean knowledge base is *consistent* if it has a model.

The reason that role inclusions and transitivity axioms are not included in the Boolean axiom formula is that the notion of simple role names does not make sense w.r.t. a Boolean combination of role axioms. Observe that every classical knowledge base $\langle\mathcal{A}, \mathcal{T}, \mathcal{R}\rangle$ is equivalent to the Boolean knowledge base $\langle\Psi, \mathcal{R}\rangle$, where $\Psi$ is the conjunction of all axioms contained in $\mathcal{A}$ and $\mathcal{T}$, and thus Boolean knowledge bases generalize classical knowledge bases. We denote by $\mathsf{Ind}(\Psi)$ the set of individuals occurring in the Boolean knowledge base $\mathcal{B} = \langle\Psi, \mathcal{R}\rangle$.

### 2.2. Conjunctive queries

In addition to consistency and entailment, there are many other inference problems relevant for DLs. One such problem is answering so-called conjunctive queries, which generalizes the entailment of assertions, e.g., deciding whether $\mathcal{K} \models r(a, b)$ holds.

**Definition 2.4 (CQ).** Let $N_\mathrm{V}$ be a set of variables. A *conjunctive query (CQ)* is of the form $\phi = \exists y_1, \ldots, y_m.\psi$, where $y_1, \ldots, y_m \in N_\mathrm{V}$ and $\psi$ is a (possibly empty) finite conjunction of *atoms* of the form

- $A(z)$ for $A \in N_\mathrm{C}$ and $z \in N_\mathrm{V} \cup N_\mathrm{I}$ *(concept atom)*; or
- $r(z_1, z_2)$ for $r \in N_\mathrm{R}$ and $z_1, z_2 \in N_\mathrm{V} \cup N_\mathrm{I}$ *(role atom)*.

The empty conjunction is denoted by $\mathsf{true}$.

A *union of conjunctive queries* (UCQ) is of the form $\phi_1 \vee \cdots \vee \phi_n$, where $\phi_1, \ldots, \phi_n$, $n \geq 1$, are CQs.

We denote the set of individual names occurring in a UCQ $\phi$ by $\mathsf{Ind}(\phi)$, the set of variables occurring in $\phi$ by $\mathsf{Var}(\phi)$, the set of free variables of $\phi$ by $\mathsf{FVar}(\phi)$, and the set of atoms occurring in $\phi$ by $\mathsf{At}(\phi)$. A UCQ $\phi$ with $\mathsf{FVar}(\phi) = \emptyset$ is called *Boolean.*

Given a UCQ $\phi$ and a knowledge base $\mathcal{K}$, we want to find all *certain answers* to $\phi$ w.r.t. $\mathcal{K}$, i.e., instantiations of the free variables in $\phi$ such that the resulting sentence is satisfied in all models of $\mathcal{K}$. We first define the semantics for Boolean UCQs, using the notion of homomorphisms [6]. This is then extended to answering arbitrary UCQs.

**Definition 2.5 (UCQ answering).** Let $\mathcal{I} = (\Delta, \cdot^\mathcal{I})$ be an interpretation and $\phi$ be a Boolean CQ. A mapping $\pi\colon \mathsf{Var}(\phi) \cup \mathsf{Ind}(\phi) \to \Delta$ is a *homomorphism* of $\phi$ into $\mathcal{I}$ if

- $\pi(a) = a^\mathcal{I}$ for all $a \in \mathsf{Ind}(\phi)$;
- $\pi(z) \in A^\mathcal{I}$ for all concept atoms $A(z) \in \mathsf{At}(\phi)$; and
- $(\pi(z_1), \pi(z_2)) \in r^\mathcal{I}$ for all role atoms $r(z_1, z_2) \in \mathsf{At}(\phi)$.

We say that $\mathcal{I}$ is a *model* of $\phi$ (written $\mathcal{I} \models \phi$) if there is such a homomorphism. Furthermore, $\mathcal{I}$ is a *model* of a Boolean UCQ $\phi_1 \vee \cdots \vee \phi_n$ if it is a model of $\phi_i$ for some $i$, $1 \leq i \leq n$.

A Boolean UCQ $\phi$ is *entailed* by a knowledge base $\mathcal{K}$ (written $\mathcal{K} \models \phi$) if every model of $\mathcal{K}$ is also a model of $\phi$. Given a (not necessarily Boolean) UCQ $\phi$, a mapping $\mathfrak{a}\colon \mathsf{FVar}(\phi) \to \mathsf{Ind}(\mathcal{K})$ is a *certain answer* to $\phi$ w.r.t. $\mathcal{K}$ if $\mathcal{K} \models \mathfrak{a}(\phi)$, where $\mathfrak{a}(\phi)$ is the Boolean UCQ obtained from $\phi$ by replacing the free variables according to $\mathfrak{a}$.

For a UCQ $\phi$ and a knowledge base $\mathcal{K}$, one can compute all certain answers by enumerating all candidate mappings $\mathfrak{a}\colon \mathsf{FVar}(\phi) \to \mathsf{Ind}(\mathcal{K})$ and then solving the entailment problem $\mathcal{K} \models \mathfrak{a}(\phi)$ for each $\mathfrak{a}$. Since there are $|\mathsf{Ind}(\mathcal{K})|^{|\mathsf{FVar}(\phi)|}$ such mappings, we have to solve exponentially many such entailment problems.

To analyze the complexity of deciding $\mathcal{K} \models \mathfrak{a}(\phi)$, it obviously suffices to consider Boolean UCQs only. Usually, two kinds of complexity measures are considered: combined complexity and data complexity. For the *combined complexity*, all parts of the input, i.e., the UCQ $\phi$ and the knowledge base $\mathcal{K}$, are taken into account. For the *data complexity*, the UCQ, the TBox, and the RBox are assumed to be constant, and the complexity is measured only w.r.t. the data, i.e., the ABox. For this analysis, we assume in the following that the query does not introduce new names, i.e., it contains only concept and role names that also occur in the TBox or the RBox. This is without loss of generality since we can always introduce trivial axioms like $A \sqsubseteq A$ or $r \sqsubseteq r$ into the TBox and RBox without affecting data complexity or combined complexity.

Regarding data complexity, the entailment problem for *concept assertions* in $\mathcal{ALC}$ is already co-NP-hard [38], and a matching upper bound has been established for UCQ entailment in $\mathcal{SHQ}$ [14].

The entailment problem for concept assertions in $\mathcal{ALC}$ is ExpTime-hard w.r.t. combined complexity [35], and a matching upper bound is known for entailment of UCQs in $\mathcal{ALCHQ}$ [8]. In $\mathcal{S}$, the problem is already co-NExpTime-hard, while it becomes 2-ExpTime-hard in $\mathcal{SH}$ [33]. In this paper, we focus on a variant of the UCQ entailment problem that is ExpTime-complete even for $\mathcal{SHQ}$, namely, we restrict to *simple* queries, which are only allowed to use simple role names. Note that this is only a restriction in extensions of $\mathcal{S}$.

### 2.3. Linear Temporal Logic

We now come to the temporal component of our query language, which is based on propositional linear temporal logic (LTL) [11].

**Definition 2.6 (LTL).** Let $\{p_1, \dots, p_m\}$ be a finite set of *propositional variables*. The set of *LTL-formulae* is the smallest set such that

- $p_1, \dots, p_m$ are LTL-formulae, and
- if $\phi_1$ and $\phi_2$ are LTL-formulae, then so are $\neg\phi_1$ (negation), $\phi_1 \wedge \phi_2$ (conjunction), $\bigcirc\phi_1$ (next), $\bigcirc^-\phi_1$ (previous), $\phi_1 \,\mathsf{U}\, \phi_2$ (until), and $\phi_1 \,\mathsf{S}\, \phi_2$ (since).

An *LTL-structure* is an infinite sequence $\mathfrak{J} = (w_i)_{i\geq 0}$ of *worlds* $w_i \subseteq \{p_1, \dots, p_m\}$. The LTL-structure $\mathfrak{J}$ is a *model* of an LTL-formula $\phi$ at time point $i \geq 0$ iff $\mathfrak{J}, i \models \phi$ holds, which is defined inductively as follows:

| | | |
|---|---|---|
| $\mathfrak{J}, i \models p_j$ | iff | $p_j \in w_i$ |
| $\mathfrak{J}, i \models \neg\phi_1$ | iff | $\mathfrak{J}, i \not\models \phi_1$ |
| $\mathfrak{J}, i \models \phi_1 \wedge \phi_2$ | iff | $\mathfrak{J}, i \models \phi_1$ and $\mathfrak{J}, i \models \phi_2$ |
| $\mathfrak{J}, i \models \bigcirc\phi_1$ | iff | $\mathfrak{J}, i+1 \models \phi_1$ |
| $\mathfrak{J}, i \models \bigcirc^-\phi_1$ | iff | $i > 0$ and $\mathfrak{J}, i-1 \models \phi_1$ |
| $\mathfrak{J}, i \models \phi_1 \,\mathsf{U}\, \phi_2$ | iff | there is $k \geq i$ with $\mathfrak{J}, k \models \phi_2$ and $\mathfrak{J}, j \models \phi_1$ for all $j$, $i \leq j < k$ |
| $\mathfrak{J}, i \models \phi_1 \,\mathsf{S}\, \phi_2$ | iff | there is $k$, $0 \leq k \leq i$ with $\mathfrak{J}, k \models \phi_2$ and $\mathfrak{J}, j \models \phi_1$ for all $j$, $k < j \leq i$ |

An LTL-formula $\phi$ is *satisfiable* if it has a model at time point 0.

Note that what we introduced above would usually be called Past-LTL, as LTL is normally defined using only the operators $\bigcirc$ and $\mathsf{U}$ [11].

Our temporal query language is based on the temporal DL $\mathcal{ALC}$-LTL, which extends LTL by allowing GCIs and assertions in place of propositional variables [12]. The semantics of this logic is determined by infinite sequences of interpretations, which will be defined more formally in the next section. It is possible to designate certain concept and role names as *rigid*, which means that their interpretation is not allowed to change over time. Satisfiability of $\mathcal{ALC}$-LTL-formulae is ExpTime-complete without rigid names, NExpTime-complete if only concept names are allowed to be rigid, and 2-ExpTime-complete in general [12].

## 3. Temporal Conjunctive Queries

We now combine the notions of (simple) conjunctive queries in $\mathcal{SHQ}$ and $\mathcal{ALC}$-LTL-formulae into a new formalism, called *temporal conjunctive queries*.

In the following, we assume (as in [12]) that a subset of the concept and role names is designated as being rigid. Let $N_{\mathrm{RC}} \subseteq N_{\mathrm{C}}$ denote the *rigid concept names*, and $N_{\mathrm{RR}} \subseteq N_{\mathrm{R}}$ the *rigid role names*. The names in $N_{\mathrm{C}} \setminus N_{\mathrm{RC}}$ and $N_{\mathrm{R}} \setminus N_{\mathrm{RR}}$ are called *flexible*. Individual names are also rigid, i.e., an individual always keeps its name.

We first extend the notion of knowledge bases and models into the temporal setting. The idea is that there is a global TBox and a global RBox that define the terminology, and several ABoxes that contain information about the state of the world at the time points we have observed so far.

**Definition 3.1 (TKB).** A *temporal knowledge base* (TKB) $\mathcal{K} = \langle (\mathcal{A}_i)_{0\leq i\leq n}, \mathcal{T}, \mathcal{R} \rangle$ consists of a finite sequence of ABoxes $\mathcal{A}_i$, a TBox $\mathcal{T}$, and an RBox $\mathcal{R}$.

Let $\mathfrak{I} = (\mathcal{I}_i)_{i\geq 0}$ be an infinite sequence of interpretations $\mathcal{I}_i = (\Delta, \cdot^{\mathcal{I}_i})$ over a fixed domain $\Delta$ (*constant domain assumption*). Then $\mathfrak{I}$ is a *model* of $\mathcal{K}$ (written $\mathfrak{I} \models \mathcal{K}$) if

- $\mathcal{I}_i \models \mathcal{A}_i$ for all $i, 0 \leq i \leq n$,
- $\mathcal{I}_i \models \mathcal{T}$ and $\mathcal{I}_i \models \mathcal{R}$ for all $i \geq 0$, and
- $\mathfrak{I}$ respects rigid names, i.e., we have $x^{\mathcal{I}_i} = x^{\mathcal{I}_j}$ for all $x \in N_{\mathrm{I}} \cup N_{\mathrm{RC}} \cup N_{\mathrm{RR}}$ and all time points $i, j \geq 0$.

As for atemporal knowledge bases, we denote by $\mathsf{Ind}(\mathcal{K})$ the set of all individual names occurring in a TKB $\mathcal{K}$.

**Definition 3.2 (TCQ).** The set of *simple temporal conjunctive queries (TCQs)* is the smallest set such that

- every simple CQ is a simple TCQ, and
- if $\phi_1$ and $\phi_2$ are simple TCQs, then so are $\neg\phi_1$ (negation), $\phi_1 \wedge \phi_2$ (conjunction), $\bigcirc\phi_1$ (next), $\bigcirc^-\phi_1$ (previous), $\phi_1 \,\mathsf{U}\, \phi_2$ (until), and $\phi_1 \,\mathsf{S}\, \phi_2$ (since).

In the following, we usually drop the qualifier *simple*. As for conjunctive queries, the sets $\mathsf{Ind}(\phi)$ and $\mathsf{FVar}(\phi)$ contain all individuals and free variables, respectively, of a TCQ $\phi$, and a *Boolean TCQ* is a TCQ without free variables.

As usual in temporal logics, one can define the following abbreviations:

- $\phi_1 \vee \phi_2 := \neg(\neg\phi_1 \wedge \neg\phi_2)$ (disjunction);
- $\Diamond\phi := \mathsf{true}\,\mathsf{U}\,\phi$ (eventually);
- $\Box\phi := \neg\Diamond\neg\phi$ (always);
- $\Diamond^-\phi := \mathsf{true}\,\mathsf{S}\,\phi$ (once); and
- $\Box^-\phi := \neg\Diamond^-\neg\phi$ (historically).

As before, we first define the semantics for Boolean queries, which is a straightforward extension of the semantics of CQs and LTL-formulae. The main difference is that the point of reference is not the first time point 0, as in LTL, but rather the last time point $n$ of a given temporal knowledge base. This can be seen as the current time point, at which we have information (e.g., sensor data) about the past, but not yet about the future. The notion of certain answers can then be defined exactly as in the atemporal case.

**Definition 3.3 (semantics of TCQs).** An infinite sequence of interpretations $\mathfrak{I} = (\mathcal{I}_i)_{i\geq 0}$ is a *model* of a Boolean TCQ $\phi$ at time point $i \geq 0$ iff $\mathfrak{I}, i \models \phi$ holds, which is defined inductively as follows (cf. Definition 2.6):

| | | |
|---|---|---|
| $\mathfrak{I}, i \models \exists y_1, \ldots, y_m.\psi$ | iff | $\mathcal{I}_i \models \exists y_1, \ldots, y_m.\psi$ |
| $\mathfrak{I}, i \models \neg\phi_1$ | iff | $\mathfrak{I}, i \not\models \phi_1$ |
| $\mathfrak{I}, i \models \phi_1 \land \phi_2$ | iff | $\mathfrak{I}, i \models \phi_1$ and $\mathfrak{I}, i \models \phi_2$ |
| $\mathfrak{I}, i \models \bigcirc\phi_1$ | iff | $\mathfrak{I}, i+1 \models \phi_1$ |
| $\mathfrak{I}, i \models \bigcirc^-\phi_1$ | iff | $i > 0$ and $\mathfrak{I}, i-1 \models \phi_1$ |
| $\mathfrak{I}, i \models \phi_1 \mathsf{U} \phi_2$ | iff | there is $k \geq i$ with $\mathfrak{I}, k \models \phi_2$ and $\mathfrak{I}, j \models \phi_1$ for all $j$, $i \leq j < k$ |
| $\mathfrak{I}, i \models \phi_1 \mathsf{S} \phi_2$ | iff | there is $k$, $0 \leq k \leq i$ with $\mathfrak{I}, k \models \phi_2$ and $\mathfrak{I}, j \models \phi_1$ for all $j$, $k < j \leq i$ |

Given a TKB $\mathcal{K} = \langle (\mathcal{A}_i)_{0 \leq i \leq n}, \mathcal{T}, \mathcal{R} \rangle$, we say that $\mathfrak{I}$ is a *model* of $\phi$ w.r.t. $\mathcal{K}$ if $\mathfrak{I} \models \mathcal{K}$ and $\mathfrak{I}, n \models \phi$. We call $\phi$ *satisfiable* w.r.t. $\mathcal{K}$ if it has a model w.r.t. $\mathcal{K}$, and it is *entailed* by $\mathcal{K}$ (written $\mathcal{K} \models \phi$) if every model $\mathfrak{I}$ of $\mathcal{K}$ satisfies $\mathfrak{I}, n \models \phi$.

Given a (not necessarily Boolean) TCQ $\phi$, a mapping $\mathfrak{a} \colon \mathsf{FVar}(\phi) \to \mathsf{Ind}(\mathcal{K})$ is a *certain answer* to $\phi$ w.r.t. $\mathcal{K}$ if $\mathcal{K} \models \mathfrak{a}(\phi)$, where $\mathfrak{a}(\phi)$ is the Boolean TCQ obtained from $\phi$ by replacing the free variables according to $\mathfrak{a}$.

As in the atemporal case, one can compute all certain answers by enumerating the (exponentially many) mappings $\mathfrak{a} \colon \mathsf{FVar}(\phi) \to \mathsf{Ind}(\mathcal{K})$ and then solving the entailment problem $\mathcal{K} \models \mathfrak{a}(\phi)$ for each $\mathfrak{a}$. Therefore, it is enough to consider the entailment problem. We instead analyze the complexity of deciding *non-entailment* $\mathcal{K} \not\models \phi$. This problem has the same complexity as the satisfiability problem of $\phi$ w.r.t. $\mathcal{K}$. In fact, $\mathcal{K} \not\models \phi$ iff $\neg\phi$ has a model w.r.t. $\mathcal{K}$, and conversely $\phi$ has a model w.r.t. $\mathcal{K}$ iff $\mathcal{K} \not\models \neg\phi$.

Note that, for the data complexity, we have to measure the complexity in the size of the sequence of ABoxes in the temporal knowledge base, instead of just a single ABox. As for the data complexity of the UCQ entailment problem, we assume that the ABoxes occurring in a temporal knowledge base and the query contain only concept and role names that also occur in the global TBox or the global RBox.

Obviously, TCQ entailment includes as a special case the entailment of CQs by atemporal knowledge bases, which can be seen as temporal knowledge bases with a sequence of ABoxes of length 1, i.e., having $n = 0$. Although models of such knowledge bases are formally infinite sequences of interpretations, all but the first interpretation are irrelevant for CQs.

On the temporal side, the TCQ satisfiability problem generalizes the satisfiability problem for $\mathcal{ALC}$-LTL-formulae since assertions are Boolean CQs. Although $\mathcal{ALC}$-LTL-formulae may additionally contain GCIs, they can equivalently be expressed by negated CQs (see the proof of Theorem 4.3 for details). On the other hand, TCQs are more expressive than $\mathcal{ALC}$-LTL-formulae since CQs like $\exists y.r(y, y)$, which says that there is a loop in the model without naming the individual which has the loop, can clearly not be expressed in $\mathcal{ALC}$.

## 4. Complexity of TCQ Entailment

We now analyze the complexity of TCQ entailment in DLs between $\mathcal{ALC}$ and $\mathcal{SHQ}$. We emphasize again that our queries only use simple role names. Without this restriction, UCQ entailment is already 2-ExpTime-hard in $\mathcal{SH}$ [33]. It is not clear whether our methods would allow us to show tight upper bounds, as they presently rely on the fact that UCQ entailment is in ExpTime (see Theorem 4.1). This allows us to show the same complexity results for simple TCQ entailment for all logics between $\mathcal{ALC}$ and $\mathcal{SHQ}$, i.e., we show the lower bounds for $\mathcal{ALC}$ and the upper bounds for $\mathcal{SHQ}$.

The restriction that all interpretations satisfy the UNA simplifies some of the proofs, but does not affect the results in this paper. More precisely, the complexity lower bounds follow from hardness results in [12, 38], the proofs of which are independent of the unique name assumption. For the upper bounds, observe that, to find a model that does not necessarily satisfy the UNA, one can guess in nondeterministic polynomial time an equivalence relation on the individual names that collects those names that will be interpreted as the same domain element, replace all names by a fixed representative of their equivalence class, and then ask for a model satisfying the UNA. For details on this construction, see [14] or the proof of Theorem 4.1 below, where we need to enforce the UNA on newly introduced individual names. This additional guessing step does not affect our complexity results.

We first take a look at the atemporal special case of the satisfiability problem for conjunctions $\phi$ of *CQ-literals*, which are either Boolean CQs or negated Boolean CQs. Since such a Boolean TCQ $\phi$ contains no temporal operators, for the satisfiability problem it suffices to consider a single interpretation instead of an infinite sequence $\mathfrak{I} = (\mathcal{I}_i)_{i \geq 0}$ of interpretations. Extending the notation for UCQs, we often write $\mathcal{I}_i \models \phi$ instead of $\mathfrak{I}, i \models \phi$ in this case. Furthermore, it is sufficient to consider TKBs with only one ABox, which can be viewed as classical knowledge bases. The following result will prove useful also for analyzing entailment of arbitrary TCQs.

**Theorem 4.1.** *Deciding satisfiability of a conjunction of CQ-literals w.r.t. a knowledge base is*

- ExpTime-*complete w.r.t. combined complexity and*
- NP-*complete w.r.t. data complexity.*

Proof. Deciding CQ entailment in $\mathcal{ALC}$ is ExpTime-hard w.r.t. combined complexity and co-NP-hard w.r.t. data complexity [9, 35, 38]. This problem is a special case of the complement of our problem.

Let now $\mathcal{K} = \langle \mathcal{A}, \mathcal{T}, \mathcal{R} \rangle$ be an $\mathcal{SHQ}$-knowledge base and $\phi$ be a conjunction of CQ-literals. To check whether there is an interpretation $\mathcal{I}$ with $\mathcal{I} \models \mathcal{K}$ and $\mathcal{I} \models \phi$, we reduce this problem to a query non-entailment problem of known complexity. Let

$$\phi = \chi_1 \wedge \ldots \wedge \chi_\ell \wedge \neg\rho_1 \wedge \ldots \wedge \neg\rho_m$$

for Boolean CQs $\chi_1, \ldots, \chi_\ell$, $\rho_1, \ldots, \rho_m$. First, we instantiate the non-negated CQs $\chi_1, \ldots, \chi_\ell$ by omitting the existential quantifiers and replacing the variables by fresh individual names. The set $\mathcal{A}'$ of all resulting atoms can thus be viewed as an additional ABox that restricts the interpretation $\mathcal{I}$.

However, we also have to ensure that the UNA is respected for the newly introduced individual names. To do this, we employ a trick from [14], which consists in guessing an equivalence relation $\approx$ on $\mathsf{Ind}(\mathcal{A} \cup \mathcal{A}')$ that specifies which individual names are allowed to be mapped to the same domain element, with the additional restriction that each equivalence class can contain at most one element from $\mathsf{Ind}(\mathcal{A})$. For such a relation $\approx$, we fix a representative for each equivalence class such that every class that contains an $a \in \mathsf{Ind}(\mathcal{A})$ has $a$ as its representative. We denote by $\mathcal{A}_\approx$ the ABox resulting from $\mathcal{A}'$ by replacing each new individual name by the representative of its equivalence class. Note that there are exponentially many such equivalence relations, each of which is of size polynomial in the size of $\phi$.

We now show that the existence of an interpretation $\mathcal{I}$ with $\mathcal{I} \models \mathcal{K}$ for $\mathcal{K} = \langle \mathcal{A}, \mathcal{T}, \mathcal{R} \rangle$ and $\mathcal{I} \models \phi$ is equivalent to the existence of an equivalence relation $\approx$ as above and an interpretation $\mathcal{I}'$ with $\mathcal{I}' \models \langle \mathcal{A} \cup \mathcal{A}_\approx, \mathcal{T}, \mathcal{R} \rangle$ and $\mathcal{I}' \models \neg\rho_1 \wedge \ldots \wedge \neg\rho_m$.

For the "if" direction, assume that $\approx$ is an equivalence relation on the individual names and $\mathcal{I}'$ is a model of $\mathcal{A}$, $\mathcal{T}$, $\mathcal{R}$, $\mathcal{A}_\approx$, and $\neg\rho_1 \wedge \ldots \wedge \neg\rho_m$. By mapping each variable occurring in $\chi_1 \wedge \ldots \wedge \chi_\ell$ to the interpretation of the representative of the equivalence class of the corresponding fresh individual name, we obtain homomorphisms from $\chi_i$ into $\mathcal{I}'$, for each $i$, $1 \leq i \leq n$. This shows that $\mathcal{I}'$ is also a model of $\phi$.

For the "only if" direction, assume that $\mathcal{I} \models \mathcal{K}$ and $\mathcal{I} \models \phi$. Thus, there are homomorphisms from each $\chi_i$, $1 \leq i \leq n$, into $\mathcal{I}$. We define any pair of individual names in $\mathcal{A} \cup \mathcal{A}'$ equivalent w.r.t. $\approx$ iff they are mapped to the same domain element by their respective homomorphisms or $\mathcal{I}$. The extension of $\mathcal{I}$ that maps each representative of its equivalence class to exactly this domain element is obviously a model of $\mathcal{A}_\approx$. It still satisfies $\mathcal{A}$, $\mathcal{T}$, $\mathcal{R}$, and $\neg\rho_1 \wedge \ldots \wedge \neg\rho_m$ since they do not contain the new individual names, and thus it is of the required form.

The above problem is thus equivalent to finding an equivalence relation $\approx$ and an interpretation $\mathcal{I}$ with $\mathcal{I} \models \langle \mathcal{A} \cup \mathcal{A}_\approx, \mathcal{T}, \mathcal{R} \rangle$ and $\mathcal{I} \not\models \rho$, where $\rho := \rho_1 \vee \cdots \vee \rho_m$ is the Boolean UCQ that results from negating the conjunction of all negated CQs in $\phi$. This is the same as asking whether $\langle \mathcal{A} \cup \mathcal{A}_\approx, \mathcal{T}, \mathcal{R} \rangle$ does not entail $\rho$.

For the combined complexity, we can enumerate all equivalence relations $\approx$ in exponential time, and check the above non-entailment for the polynomial-size $\mathcal{SHQ}$-knowledge base and UCQ resulting from each $\approx$, which can be done in ExpTime [8]. For the data complexity, we can guess $\approx$ in nondeterministic polynomial time, and check the non-entailment in NP [13]. □

In the remainder of this paper, we will present several constructions, most of which use the above theorem, to derive the complexity results shown in Table 1 for TCQ entailment in all DLs between $\mathcal{ALC}$ and $\mathcal{SHQ}$. The results depend on which symbols are allowed to be rigid.

*4.1. Lower Bounds for the Entailment Problem*

For the data complexity, we obtain the lower bounds from Theorem 4.1.

**Corollary 4.2.** *TCQ entailment is* co-NP*-hard w.r.t. data complexity.*

Proof. Theorem 4.1 states that for conjunctions of CQ-literals $\phi$ and atemporal knowledge bases $\mathcal{K}$, deciding whether $\phi$ has a model w.r.t. $\mathcal{K}$ is NP-complete w.r.t. data complexity. Since $\phi$ is a special TCQ and rigid names are irrelevant in the atemporal case, we obtain co-NP-hardness w.r.t. data complexity for the entailment problem for all the cases in Table 1. □

For the combined complexity, we get the lower bounds by a simple reduction of the satisfiability problem for $\mathcal{ALC}$-LTL [12].

**Theorem 4.3.** *TCQ entailment w.r.t. combined complexity is*

- ExpTime*-hard if $N_{RC} = N_{RR} = \emptyset$;*
- co-NExpTime*-hard if $N_{RC} \neq \emptyset$ and $N_{RR} = \emptyset$; and*
- 2-ExpTime*-hard if $N_{RR} \neq \emptyset$.*

Proof. We reduce the satisfiability problem of $\mathcal{ALC}$-LTL to the TCQ *non-entailment* problem.

Let $\psi$ be the Boolean TCQ and $\mathcal{T}$ be the TBox obtained from an $\mathcal{ALC}$-LTL-formula $\phi$ as follows. We replace each GCI $C \sqsubseteq D$ in $\phi$ by $\neg(\exists x.A(x))$ and add $A \equiv C \sqcap \neg D$ to $\mathcal{T}$, where $A$ is a fresh concept name. Similarly, we replace every complex concept assertion $E(a)$ in $\phi$ by $B(a)$ and add $B \equiv E$ to $\mathcal{T}$. Then $\phi$ is satisfiable iff $\langle \emptyset, \mathcal{T}, \emptyset \rangle \not\models \neg\psi$.

Since satisfiability of $\mathcal{ALC}$-LTL-formulae is ExpTime-complete without rigid names, NExpTime-complete with rigid concept names, and 2-ExpTime-complete with rigid concept and role names [12], this shows the claimed lower bounds. □

In the following sections, we present the ideas for the upper bounds w.r.t. combined complexity and data complexity. For the former, we can match all lower bounds we have from Theorem 4.3. For the latter, unfortunately we cannot match the lower bound of co-NP in the case where we have rigid role names. While our constructions need to deal with CQs and the additional expressivity of $\mathcal{SHQ}$ in an appropriate way, the basic ideas are similar to those presented for $\mathcal{ALC}$-LTL in [12].

*4.2. Upper Bounds for the Entailment Problem*

We divide the *satisfiability* problem of a Boolean TCQ $\phi$ w.r.t. a TKB $\mathcal{K} = \langle(\mathcal{A}_i)_{0\leq i\leq n}, \mathcal{T}, \mathcal{R}\rangle$ into two separate satisfiability problems, similar to what was done for $\mathcal{ALC}$-LTL in Lemma 4.3 of [12]. The *t-satisfiability* expresses that the *temporal* structure of $\phi$ is consistent, while the *r-satisfiability* determines whether it is possible to satisfy the *rigidity* constraints for the names in $N_{\mathrm{RC}}$ and $N_{\mathrm{RR}}$.

We consider the *propositional abstraction* $\widehat{\phi}$ of $\phi$, which is the propositional LTL-formula built from $\phi$ by replacing each CQ by a unique propositional variable. We assume that $\alpha_1, \ldots, \alpha_m$ are the CQs occurring in $\phi$, $p_1, \ldots, p_m$ are the propositional variables of $\widehat{\phi}$, and that each $\alpha_i$ is replaced by $p_i$ for all $i$, $1 \leq i \leq m$. This LTL-formula allows us to analyze the temporal structure of $\phi$ separately from the DL query component.

We now consider a set $\mathcal{S} \subseteq 2^{\{p_1,\ldots,p_m\}}$, which intuitively specifies the worlds that are allowed to occur in an LTL-structure satisfying $\widehat{\phi}$. To express this restriction, we define the propositional LTL-formula

$$\widehat{\phi}_{\mathcal{S}} := \widehat{\phi} \wedge \Box^{-}\Box\left(\bigvee_{X\in\mathcal{S}}\left(\bigwedge_{p\in X} p \wedge \bigwedge_{p\notin X} \neg p\right)\right).$$

Note that a formula $\Box^{-}\Box\psi$ is satisfied iff $\psi$ holds at all time points. An immediate connection between $\phi$ and $\widehat{\phi}_{\mathcal{S}}$ is formalized in the next lemma.

**Lemma 4.4.** *If $\phi$ has a model w.r.t. $\mathcal{K}$, then there is a set $\mathcal{S} \subseteq 2^{\{p_1,\ldots,p_m\}}$ and a propositional LTL-structure that is a model of $\widehat{\phi}_{\mathcal{S}}$ at time point $n$.*

Proof. Let $\mathfrak{I} = (\mathcal{I}_i)_{i\geq 0}$ be a sequence of interpretations that respects rigid names, is a model of $\mathcal{K}$, and satisfies $\mathfrak{I}, n \models \phi$. For each interpretation $\mathcal{I}_i$ of $\mathfrak{I}$, we set

$$X_i := \{p_j \mid 1 \leq j \leq m \text{ and } \mathcal{I}_i \text{ satisfies } \alpha_j\},$$

and then consider the set $\mathcal{S} := \{X_i \mid i \geq 0\}$ induced by $\mathfrak{I}$. The *propositional abstraction* $\widehat{\mathfrak{I}} = (w_i)_{i\geq 0}$ of $\mathfrak{I}$ is now defined by $w_i := X_i$ for all $i \geq 0$. It is easy to check that the fact that $\mathfrak{I}$ satisfies $\phi$ at time point $n$ implies that $\widehat{\mathfrak{I}}$ is a model of $\widehat{\phi}_{\mathcal{S}}$ at time point $n$. □

However, guessing a set $\mathcal{S}$ and then testing whether the induced LTL-formula $\widehat{\phi}_{\mathcal{S}}$ is has a model at time point $n$ is not sufficient for checking whether $\phi$ has a model w.r.t. $\mathcal{K}$. We must also check whether $\mathcal{S}$ can indeed be induced by some sequence of interpretations that is a model of $\mathcal{K}$. In the following, let $\mathcal{S} = \{X_1, \ldots, X_k\} \subseteq 2^{\{p_1,\ldots,p_m\}}$, and $\iota\colon \{0, \ldots, n\} \to \{1, \ldots, k\}$ be a mapping that specifies a set $X_{\iota(i)}$ for each of the ABoxes $\mathcal{A}_i$, $0 \leq i \leq n$.

**Definition 4.5 (r-satisfiability).** We call $\mathcal{S}$ *r-satisfiable* w.r.t. $\iota$ and $\mathcal{K}$ if there exist interpretations $\mathcal{J}_1, \ldots, \mathcal{J}_k$, $\mathcal{I}_0, \ldots, \mathcal{I}_n$ such that

- they share the same domain and respect rigid names;[2]
- they are models of $\mathcal{T}$ and $\mathcal{R}$;
- each $\mathcal{J}_i$, $1 \leq i \leq k$, is a model of
$$\chi_i := \bigwedge_{p_j\in X_i} \alpha_j \wedge \bigwedge_{p_j\notin X_i} \neg\alpha_j; \text{ and}$$
- each $\mathcal{I}_i$, $0 \leq i \leq n$, is a model of $\mathcal{A}_i$ and $\chi_{\iota(i)}$.

The intuition underlying this definition is the following. The existence of the interpretation $\mathcal{J}_i$, $1 \leq i \leq k$, ensures that the conjunction $\chi_i$ of the CQ-literals specified by $X_i$ is consistent. In fact, a set $\mathcal{S}$ containing a set $X_i$ for which this does not hold cannot be induced by a sequence of interpretations. The interpretations $\mathcal{I}_i$, $0 \leq i \leq n$, constitute the first $n+1$ interpretations in such a sequence. In addition to inducing a set $X_{\iota(i)} \in \mathcal{S}$ and thus satisfying the corresponding conjunction $\chi_{\iota(i)}$, the interpretation $\mathcal{I}_i$ must also satisfy the ABox $\mathcal{A}_i$. The first and the second condition ensure that a sequence of interpretations built from $\mathcal{J}_1, \ldots, \mathcal{J}_k, \mathcal{I}_0, \ldots, \mathcal{I}_n$ respects rigid names and satisfies the global TBox $\mathcal{T}$ and the global RBox $\mathcal{R}$. Note that we can use Theorem 4.1 to check whether interpretations satisfying the last three conditions of Definition 4.5 exist. As we will see below, the difficulty lies in ensuring that they also satisfy the first condition.

Satisfaction of the temporal structure of $\phi$ by a sequence of interpretations built this way is ensured by testing $\widehat{\phi}_{\mathcal{S}}$ for satisfiability w.r.t. a side condition that ensures that the first $n$ worlds are those chosen by $\iota$.

**Definition 4.6 (t-satisfiability).** The LTL-formula $\widehat{\phi}$ is *t-satisfiable* w.r.t. $\mathcal{S}$ and $\iota$ if there exists an LTL-structure $\mathfrak{I} = (w_i)_{i\geq 0}$ such that

- $\mathfrak{I}, n \models \widehat{\phi}_{\mathcal{S}}$ and
- $w_i = X_{\iota(i)}$ for all $i$, $0 \leq i \leq n$.

We can now combine these two satisfiability tests to decide satisfiability of a TCQ w.r.t. a TKB.

**Lemma 4.7.** *The TCQ $\phi$ is satisfiable w.r.t. the TKB $\mathcal{K}$ iff there is a set $\mathcal{S} = \{X_1, \ldots, X_k\} \subseteq 2^{\{p_1,\ldots,p_m\}}$ and a mapping $\iota\colon \{0, \ldots, n\} \to \{1, \ldots, k\}$ such that*

- *$\mathcal{S}$ is r-satisfiable w.r.t. $\iota$ and $\mathcal{K}$, and*
- *$\widehat{\phi}$ is t-satisfiable w.r.t. $\mathcal{S}$ and $\iota$.*

Proof. For the "only if" direction, assume that there is a sequence of interpretations $\mathfrak{I} = (\mathcal{I}_i)_{i\geq 0}$ with $\mathfrak{I} \models \mathcal{K}$ and $\mathfrak{I}, n \models \phi$. Recall that we have already seen in Lemma 4.4 that $\mathfrak{I}$ induces a set $\mathcal{S} \subseteq 2^{\{p_1,\ldots,p_m\}}$ such that $\widehat{\phi}_{\mathcal{S}}$ is satisfiable at time point $n$. Let $\mathcal{S} = \{X_1, \ldots, X_k\}$. For each

[2] This is defined analogously to the case of sequences of interpretations (Definition 3.1).

$i \geq 0$, there is an index $\nu_i \in \{1, \dots, k\}$ such that $\mathcal{I}_i$ induces the set $X_{\nu_i}$, i.e.,

$$X_{\nu_i} = \{p_j \mid 1 \leq j \leq m \text{ and } \mathcal{I}_i \text{ satisfies } \alpha_j\},$$

and, conversely, for each $\nu \in \{1, \dots, k\}$, there is an index $i \geq 0$ such that $\nu = \nu_i$. We define the mapping $\iota$ as follows: $\iota(i) = \nu_i$ for all $i$, $0 \leq i \leq n$. Let $\widehat{\mathfrak{I}} = (w_i)_{i \geq 0}$ be the propositional abstraction of $\mathfrak{I}$. As argued in Lemma 4.4, $\widehat{\mathfrak{I}}$ is a model of $\widehat{\phi}_{\mathcal{S}}$ at time point $n$. By definition of $\iota$, $X_{\nu_i}$ and $\widehat{\mathfrak{I}}$, we also have $w_i = X_{\iota(i)}$ for all $i$, $0 \leq i \leq n$.

For $i$, $1 \leq i \leq k$, the interpretation $\mathcal{J}_i$ is obtained as follows. Let $\ell_1, \dots, \ell_k$ be such that $\nu_{\ell_1} = 1, \dots, \nu_{\ell_k} = k$. Now, if we set $\mathcal{J}_i := \mathcal{I}_{\ell_i}$, then it is clear that $\mathcal{J}_i$ is a model of $\chi_i$. It is now easy to see that the interpretations $\mathcal{J}_1, \dots, \mathcal{J}_k, \mathcal{I}_0, \dots, \mathcal{I}_n$ satisfy the conditions for r-satisfiability of $\mathcal{S}$ w.r.t. $\iota$ and $\mathcal{K}$.

To show the "if" direction, assume that there is a set $\mathcal{S} = \{X_1, \dots, X_k\}$, a mapping $\iota\colon \{0, \dots, n\} \to \{1, \dots, k\}$, an LTL-structure $\mathfrak{J} = (w_i)_{i \geq 0}$ such that $\mathfrak{J}$ is a model of $\widehat{\phi}_{\mathcal{S}}$ at time point $n$ and $w_i = X_{\iota(i)}$ for all $i$, $0 \leq i \leq n$, and models $\mathcal{J}_1, \dots, \mathcal{J}_k, \mathcal{I}_0, \dots, \mathcal{I}_n$ of $\mathcal{T}$ and $\mathcal{R}$ with the properties of Definition 4.5.

By the definition of $\widehat{\phi}_{\mathcal{S}}$, for every world $w_i$, there is exactly one index $\nu_i \in \{1, \dots, k\}$ such that $w_i$ satisfies

$$\bigwedge_{p \in X_{\nu_i}} p \wedge \bigwedge_{p \notin X_{\nu_i}} \neg p.$$

Since $w_i$, $0 \leq i \leq n$, satisfies exactly the propositional variables of $X_{\iota(i)}$, we have $\iota(i) = \nu_i$. We can now define a sequence of interpretations $\mathfrak{I} := (\mathcal{I}_i)_{i \geq 0}$ respecting rigid names as follows: we set $\mathcal{I}_i := \mathcal{J}_{\nu_i}$ for $i > n$. By Definition 4.5, each $\mathcal{I}_i$ satisfies exactly the CQs specified by the propositional variables in $X_{\nu_i}$. Since $\mathfrak{J}, n \models \widehat{\phi}_{\mathcal{S}}$, this means that $\mathfrak{I}, n \models \phi$. It also follows directly from Definition 4.5 that $\mathfrak{I} \models \mathcal{K}$. Hence, we have that $\phi$ has model w.r.t. $\mathcal{K}$. $\square$

Since the overall complexity of the satisfiability problem depends on which symbols are allowed to be rigid, we obtain the set $\mathcal{S}$ and the function $\iota$ either by enumeration, guessing, or direct construction (see, e.g., Theorems 4.13 and 4.15). Given $\mathcal{S}$ and $\iota$, it remains to check the two conditions of Lemma 4.7. For the r-satisfiability test, we need to use different constructions depending on which symbols are allowed to be rigid. Using these constructions, we obtain the complexity results for the entailment problem shown in Table 1. The details can be found in later sections. First, we focus on the second condition of Lemma 4.7.

*4.2.1. An Automaton for LTL-Satisfiability*

We construct a generalized Büchi automaton, similar to the standard construction for satisfiability of LTL-formulae [15, 16], such that emptiness of this automaton is equivalent to t-satisfiability of $\widehat{\phi}$ w.r.t. $\mathcal{S}$ and $\iota$.

**Definition 4.8 (generalized Büchi automaton).** A *generalized Büchi automaton* $\mathcal{G} = (Q, \Sigma, \Delta, Q_0, \mathcal{F})$ consists of a finite set of states $Q$, a finite input alphabet $\Sigma$, a transition relation $\Delta \subseteq Q \times \Sigma \times Q$, a set $Q_0 \subseteq Q$ of initial states, and a set of sets of accepting states $\mathcal{F} \subseteq 2^Q$.

Given an infinite word $w = \sigma_0\sigma_1\sigma_2 \ldots \in \Sigma^\omega$, a *run* of $\mathcal{G}$ on $w$ is an infinite word $q_0q_1q_2 \ldots \in Q^\omega$ such that $q_0 \in Q_0$ and $(q_i, \sigma_i, q_{i+1}) \in \Delta$ for all $i \geq 0$. This run is *accepting* if, for every $F \in \mathcal{F}$, there are infinitely many $i \geq 0$ such that $q_i \in F$. The *language accepted by* $\mathcal{G}$ is defined as

$$L_\omega(\mathcal{G}) := \{w \in \Sigma^\omega \mid \text{there is an accepting run of } \mathcal{G} \text{ on } w\}.$$

The *emptiness problem* for generalized Büchi automata is the problem of deciding, given a generalized Büchi automaton $\mathcal{G}$, whether $L_\omega(\mathcal{G}) = \emptyset$ or not.

We use *generalized* Büchi automata rather than normal ones (where $|\mathcal{F}| = 1$) since this allows for a simpler construction below. It is well-known that a generalized Büchi automaton can be transformed into an equivalent normal one in polynomial time [39, 40]. Together with the fact that the emptiness problem for normal Büchi automata can be solved in polynomial time [16], this yields a polynomial time bound for the complexity of the emptiness problem for generalized Büchi automata.

To define our automaton, we need the notion of a *type for* $\widehat{\phi}$.

**Definition 4.9 (type).** A *sub-literal* of $\widehat{\phi}$ is a sub-formula of $\widehat{\phi}$ or its negation. A set $T$ of sub-literals of $\widehat{\phi}$ is a type for $\widehat{\phi}$ iff the following properties are satisfied:

1. for every sub-formula $\psi$ of $\widehat{\phi}$, we have $\psi \in T$ iff $\neg\psi \notin T$;
2. for every sub-formula $\psi_1 \wedge \psi_2$ of $\widehat{\phi}$, we have $\psi_1 \wedge \psi_2 \in T$ iff $\{\psi_1, \psi_2\} \subseteq T$;

We denote the set of all types for $\widehat{\phi}$ by $\mathfrak{T}$. We further define the set $\mathfrak{T}|_{\mathcal{S}} \subseteq \mathfrak{T}$ that contains all types $T$ for $\widehat{\phi}$ for which $T \cap \{p_1, \dots, p_m\} \in \mathcal{S}$.

The reason that we use the types for $\widehat{\phi}$ and not for $\widehat{\phi}_{\mathcal{S}}$ is that the latter formula is exponentially larger than the former. To avoid this exponential blowup in the automaton, we check the additional condition of $\widehat{\phi}_{\mathcal{S}}$, namely that each world of a model must occur in the set $\mathcal{S}$, by restricting the first component of the state set of the automaton to $\mathfrak{T}|_{\mathcal{S}}$.

Another difference to the standard construction for LTL is the additional condition that $w_i = X_{\iota(i)}$ should hold for all $i$, $0 \leq i \leq n$. We check this by attaching a counter from $\{0, \dots, n+1\}$ to the states of the automaton. Transitions where the counter is $i < n + 1$ check if the current world corresponds to $X_{\iota(i)}$ and increase the counter by 1. At $i = n$, we ensure that $\widehat{\phi}$ is satisfied.

**Definition 4.10 (automaton for t-satisfiability).** The generalized Büchi automaton $\mathcal{G} = (Q, \Sigma, \Delta, Q_0, \mathcal{F})$ is defined as follows:

- $Q := \mathfrak{T}|_{\mathcal{S}} \times \{0, \dots, n+1\}$;

- $\Sigma := 2^{\{p_1,\dots,p_m\}}$;
- $((T,k),\sigma,(T',k')) \in \Delta$ iff
  - $\sigma = T \cap \{p_1,\dots,p_m\}$;
  - $\bigcirc\psi \in T$ iff $\psi \in T'$;
  - $\bigcirc^-\psi \in T'$ iff $\psi \in T$;
  - $\psi_1 \mathsf{U} \psi_2 \in T$ iff (i) $\psi_2 \in T$ or (ii) $\psi_1 \in T$ and $\psi_1 \mathsf{U} \psi_2 \in T'$;
  - $\psi_1 \mathsf{S} \psi_2 \in T'$ iff (i) $\psi_2 \in T'$ or (ii) $\psi_1 \in T'$ and $\psi_1 \mathsf{S} \psi_2 \in T$;
  - $k < n+1$ implies $\sigma = X_{\iota(k)}$;
  - $k = n$ implies $\widehat{\phi} \in T$; and
  - $k' = \begin{cases} k+1 & \text{if } k < n+1, \text{ and} \\ k & \text{otherwise;} \end{cases}$
- $Q_0 := \{(T,0) \mid \psi_1 \mathsf{S} \psi_2 \in T \Rightarrow \psi_2 \in T \text{ and } \bigcirc^-\psi \notin T\}$; and
- $\mathcal{F}$ contains, for each sub-formula of $\widehat{\phi}$ of the form $\psi_1 \mathsf{U} \psi_2$, the set $F_{\psi_1 \mathsf{U} \psi_2} \times \{n+1\}$, where

$$F_{\psi_1 \mathsf{U} \psi_2} := \{T \in \mathfrak{T}|_{\mathcal{S}} \mid \psi_1 \mathsf{U} \psi_2 \in T \Rightarrow \psi_2 \in T\}.$$

This automaton accepts exactly those sequences of worlds that satisfy the conditions for t-satisfiability of $\widehat{\phi}$ w.r.t. $\mathcal{S}$ and $\iota$. The proof is a straightforward extension of the original proof for LTL-satisfiability [15, 16], and can be found in the appendix.

**Lemma 4.11.** *For every infinite word $w = w_0w_1 \ldots \in \Sigma^\omega$, we have $w \in L_\omega(\mathcal{G})$ iff the LTL-structure $\mathfrak{J} := (w_i)_{i\geq 0}$ satisfies $\mathfrak{J}, n \models \widehat{\phi}_{\mathcal{S}}$ and $w_i = X_{\iota(i)}$ for all $i$, $0 \leq i \leq n$.*

This implies that $L_\omega(\mathcal{G}) \neq \emptyset$ iff $\widehat{\phi}$ is t-satisfiable w.r.t. $\mathcal{S}$ and $\iota$. We can thus decide the latter problem by testing $\mathcal{G}$ for emptiness, which yields the following complexity results.

**Lemma 4.12.** *Deciding t-satisfiability of $\widehat{\phi}$ w.r.t. $\mathcal{S}$ and $\iota$ can be done*

- *in* ExpTime *w.r.t. combined complexity and*
- *in* P *w.r.t. data complexity.*

Proof. For combined complexity, there are exponentially many types for $\widehat{\phi}$ and exponentially many input symbols in $2^{\{p_1,\dots,p_m\}}$. The set $\mathcal{F}$ contains linearly many sets of size at most exponential, while the size of $Q_0$ and $\Delta$ is bounded polynomially in the size of $Q$ (which is exponential). Since all conditions that need to be checked to construct the components of $\mathcal{G}$ can be checked in exponential time, and the size of $\mathcal{G}$ is exponential in the size of $\mathcal{K}$ and $\phi$, the emptiness test can be done in ExpTime.

For data complexity, the size of $\mathcal{G}$ is polynomial in $n$ because of the following reasons: the size of $\mathfrak{T}|_{\mathcal{S}}$ is constant since the size of $\mathcal{S}$ depends only on the size of $\phi$, which is constant. Thus, the size of $Q$ is linear in $n$. The size of $\Sigma$ is constant. Obviously, then the size of $\Delta$ is polynomial in $n$. The size of $Q_0$ is linear in $n$, because $Q_0 \subseteq Q$. The size of $\mathcal{F}$ is logarithmic in $n$, because each set $F_{\psi_1 \mathsf{U} \psi_2}$ is of constant size, and the number of such sets does not depend on $n$. Obviously, $\mathcal{G}$ can also be constructed in time polynomial in $n$. The data complexity of the emptiness test is thus in P. □

However, the complexity of the entailment problem also depends on the complexity of the r-satisfiability test for $\mathcal{S}$. In the following sections, we will establish some results as to this complexity in the cases without rigid names, and with rigid concept and role names. The most interesting (and most complex) case without rigid role names, but with rigid concept names, is considered in Section 5 for data complexity and in Section 6 for combined complexity.

#### *4.2.2. The Case without Rigid Names*

Assume that a set $\mathcal{S} = \{X_1,\dots,X_k\} \subseteq 2^{\{p_1,\dots,p_m\}}$ and a mapping $\iota\colon \{0,\dots,n\} \to \{1,\dots,k\}$ are given. To check r-satisfiability of $\mathcal{S}$ w.r.t. $\iota$ and $\mathcal{K}$ without rigid names, it clearly suffices to check the satisfiability of the following conjunctions of CQ-literals w.r.t. the TBox $\mathcal{T}$ and the RBox $\mathcal{R}$ individually:

- for each $i$, $1 \leq i \leq k$, the conjunction $\chi_i$; and
- for each $i$, $0 \leq i \leq n$, the conjunction $\chi_{\iota(i)} \wedge \bigwedge_{\alpha\in\mathcal{A}_i} \alpha$.[3]

Each of these conjunctions of CQ-literals is of polynomial size in the size of $\mathcal{K}$ and $\phi$. We can now use Theorem 4.1 to establish the complexity of the entailment problem without rigid names.

**Theorem 4.13.** *If $N_{RC} = N_{RR} = \emptyset$, TCQ entailment is*

- *in* ExpTime *w.r.t. combined complexity and*
- *in* co-NP *w.r.t. data complexity.*

Proof. For combined complexity, note that we do not need to guess the set $\mathcal{S}$. Since the r-satisfiability condition imposes no dependency between the sets $X \in \mathcal{S}$, it suffices to define $\mathcal{S}$ as the set of *all* sets $X_i$ that pass the satisfiability test of the corresponding conjunction $\chi_i$ w.r.t. $\langle\emptyset,\mathcal{T},\mathcal{R}\rangle$. Since there are exponentially many such sets, but each of them is of polynomial size, by Theorem 4.1 we only have to do exponentially many ExpTime-tests to construct $\mathcal{S}$. We can further enumerate all possible mappings $\iota$ in exponential time and check for each $\iota$ the satisfiability of the conjunctions $\chi_{\iota(i)} \wedge \bigwedge_{\alpha\in\mathcal{A}_i} \alpha$ again in ExpTime. For each $\iota$ that passes these tests, we can check t-satisfiability of $\widehat{\phi}$ w.r.t. $\mathcal{S}$ and $\iota$ in ExpTime by Lemma 4.12. Lemma 4.7 now

[3] We can assume that all of these models have the same domain since their domains can be assumed to be countably infinite by the Löwenheim-Skolem theorem, and that all individual names are interpreted by the same domain elements in all models.

yields a total complexity of ExpTime for the satisfiability problem, and therefore also for the entailment problem.

For data complexity, note that since $\mathcal{S}$ is of constant size w.r.t. the ABoxes and $\iota$ is linear in $n$, guessing $\mathcal{S}$ and $\iota$ can be done in NP. Since the t-satisfiability test can be done in P (Lemma 4.12) and the satisfiability tests for r-satisfiability of $\mathcal{S}$ can be done in NP (Theorem 4.1), by Lemma 4.7 the satisfiability problem is also in NP. □

*4.2.3. The Case with Rigid Role Names*

If the sets $N_{\mathrm{RC}}$ and $N_{\mathrm{RR}}$ are allowed to be non-empty, the satisfiability tests for the r-satisfiability of $\mathcal{S}$ are not independent any longer. To make sure that the models respect the rigid symbols, we use a renaming technique similar to the one used in [12] that works by introducing enough copies of the flexible symbols.

For every $i$, $1 \leq i \leq k+n+1$, and every *flexible* concept name $A$ (every *flexible* role name $r$) occurring in $\mathcal{T}$ or $\mathcal{R}$, we introduce a copy $A^{(i)}$ ($r^{(i)}$). We call $A^{(i)}$ ($r^{(i)}$) the $i$-th copy of $A$ ($r$). The conjunctive query $\alpha^{(i)}$ (the GCI/transitivity axiom/role inclusion $\beta^{(i)}$) is obtained from a CQ $\alpha$ (a GCI/transitivity axiom/role inclusion $\beta$) by replacing every occurrence of a flexible name by its $i$-th copy. Similarly, for $1 \leq \ell \leq k$, the conjunction of CQ-literals $\chi_\ell^{(i)}$ is obtained from $\chi_\ell$ (see Definition 4.5) by replacing each CQ $\alpha_j$ by $\alpha_j^{(i)}$. Finally, we define

$$\chi_{\mathcal{S},\iota} := \bigwedge_{1\leq i\leq k} \chi_i^{(i)} \wedge \bigwedge_{0\leq i\leq n} \left( \chi_{\iota(i)}^{(k+i+1)} \wedge \bigwedge_{\alpha\in\mathcal{A}_i} \alpha^{(k+i+1)} \right),$$
$$\mathcal{T}_{\mathcal{S},\iota} := \{\beta^{(i)} \mid \beta \in \mathcal{T} \text{ and } 1 \leq i \leq k+n+1\},$$
$$\mathcal{R}_{\mathcal{S},\iota} := \{\gamma^{(i)} \mid \gamma \in \mathcal{R} \text{ and } 1 \leq i \leq k+n+1\}.$$

Note that here it is essential that the ABoxes do not contain complex concepts, otherwise they could not be interpreted as conjunctions of CQ-literals.

**Lemma 4.14.** *The set $\mathcal{S}$ is r-satisfiable w.r.t. $\iota$ and $\mathcal{K}$ iff $\chi_{\mathcal{S},\iota}$ is satisfiable w.r.t. $\langle \mathcal{T}_{\mathcal{S},\iota}, \mathcal{R}_{\mathcal{S},\iota}\rangle$.*

The proof of this lemma can be found in the appendix. Unfortunately, the data complexity of this approach does not allow us to match the lower bound of co-NP for the entailment problem we have from Corollary 4.2. However, for the combined complexity we obtain containment in 2-ExpTime.

**Theorem 4.15.** *If $N_{RR} \neq \emptyset$, TCQ entailment is*

- *in* 2-ExpTime *w.r.t. combined complexity and*
- *in* ExpTime *w.r.t. data complexity.*

Proof. To check a TCQ $\phi$ for satisfiability w.r.t. a TKB $\mathcal{K}$, we first enumerate all possible sets $\mathcal{S}$ and mappings $\iota$, which can be done in 2-ExpTime w.r.t. combined complexity and in ExpTime w.r.t. data complexity since $\mathcal{S}$ is constant in this case. For each of these double-exponentially many pairs $(\mathcal{S},\iota)$, we then check t-satisfiability of $\widehat{\phi}_{\mathcal{S}}$ w.r.t. $\mathcal{S}$ and $\iota$ in exponential time (see Lemma 4.12) and test $\mathcal{S}$ for r-satisfiability w.r.t. $\iota$ and $\mathcal{K}$. By Lemma 4.7, $\phi$ has a model w.r.t. $\mathcal{K}$ iff at least one pair passes both tests.

For the combined complexity of the r-satisfiability test, observe that the conjunction of CQ-literals $\chi_{\mathcal{S},\iota}$ is of exponential size in the size of $\phi$ and $\mathcal{K}$. By Theorem 4.1, the overall combined complexity of the r-satisfiability test is thus in 2-ExpTime.

For the data complexity of the r-satisfiability test, we know that $\chi_{\mathcal{S},\iota}$ is of linear size in the size of the input ABoxes. Unfortunately, by copying each of the types $\chi_{\iota(i)}$ assigned to the ABoxes, we have introduced linearly many *negated* CQs, which is why Theorem 4.1 only yields an ExpTime upper bound for the data complexity. Note that linearly many non-negated CQs in $\chi_{\mathcal{S},\iota}$ are not problematic, as they can be instantiated and viewed as part of the ABox, as detailed in the proof of Theorem 4.1. □

However, we can match the lower bound of co-NP for the data complexity in the following special cases.

**Lemma 4.16.** *If $N_{RR} \neq \emptyset$, TCQ entailment is in* co-NP *w.r.t. data complexity if any of the following conditions apply:*

1. *The number $n$ of the input ABoxes is bounded by a constant.*
2. *The set of individual names allowed to occur in the ABoxes is fixed.*

Proof. As in the proof of Theorem 4.13, we can guess the set $\mathcal{S}$ and the mapping $\iota$ in NP and do the LTL-satisfiability test in P. Thus, it suffices to show that in the above-mentioned special cases r-satisfiability of $\mathcal{S}$ can be tested in NP.

1. If $n$ is bounded by a constant, then the number of negated CQs in $\chi_{\mathcal{S},\iota}$ is constant, and thus Theorem 4.1 yields the desired NP upper bound.
2. If the set of individual names is fixed, then the number of possible assertions involving concept names occurring in the TBox is constant. Note that the concept names occurring only in the ABoxes do not affect the entailment of the TCQ, as they can only occur in *positive* assertions, and can thus always be satisfied by appropriately interpreting the new names.

   This allows us to restrict the formula $\chi_{\mathcal{S},\iota}$ to contain at most one copy of $\chi_{\iota(i)}$ for each distinct combination of $\chi_{\iota(i)}$ and $\mathcal{A}_i$ (ignoring assertions about names that do not occur in the TBox). Clearly, consistency of each combination of an ABox with a type needs to be checked only once. Since there are now only constantly many such combinations, the modified TCQ $\chi'_{\mathcal{S},\iota}$ again contains only constantly many negated CQs. As in the previous case, Theorem 4.1 yields the result. □

## 5. Data Complexity for the Case of Rigid Concept Names

We will now show that the data complexity of TCQ entailment in the case where $N_{\mathrm{RC}} \neq \emptyset$ and $N_{\mathrm{RR}} = \emptyset$ is in co-NP. As detailed in the proof of Theorem 4.13, it suffices to show that r-satisfiability of $\mathcal{S}$ w.r.t. $\iota$ and $\mathcal{K}$ can be checked in NP.

Similar to the previous sections, we construct conjunctions of CQ-literals of which we want to check satisfiability. The approach is a mixture of those of Sections 4.2.2 and 4.2.3, as we combine several satisfiability tests required for r-satisfiability, but do not go as far as compiling all of them into just one conjunction. More precisely, we consider the conjunctions of CQ-literals $\gamma_i \wedge \chi_{\mathcal{S}}$, $0 \leq i \leq n$, w.r.t. $\langle \mathcal{T}_{\mathcal{S}}, \mathcal{R}_{\mathcal{S}} \rangle$, where

$$\gamma_i := \bigwedge_{\alpha \in \mathcal{A}_i} \alpha^{(\iota(i))}, \quad \chi_{\mathcal{S}} := \bigwedge_{1 \leq i \leq k} \chi_i^{(i)},$$
$$\mathcal{T}_{\mathcal{S}} := \{\beta^{(i)} \mid \beta \in \mathcal{T} \text{ and } 1 \leq i \leq k\},$$
$$\mathcal{R}_{\mathcal{S}} := \{\gamma^{(i)} \mid \gamma \in \mathcal{R} \text{ and } 1 \leq i \leq k\}.$$

However, for r-satisfiability we have to make sure that rigid consequences of the form $A(a)$ for a rigid concept name $A \in N_{\mathrm{RC}}$ and an individual name $a \in N_{\mathrm{I}}$ are shared between all of these conjunctions $\gamma_i \wedge \chi_{\mathcal{S}}$. It suffices to do this for the set $\mathsf{RCon}(\mathcal{T})$ of rigid concept names occurring in $\mathcal{T}$ since those that occur only in ABox assertions cannot affect the entailment of the TCQ $\phi$.

Similar to what was done in Lemma 6.3 of [12], we guess a set $\mathcal{D} \subseteq 2^{\mathsf{RCon}(\mathcal{T})}$ and a function $\tau\colon \mathsf{Ind}(\phi) \cup \mathsf{Ind}(\mathcal{K}) \to \mathcal{D}$. The idea is that $\mathcal{D}$ fixes the combinations of rigid concept names that occur in the models of $\gamma_i \wedge \chi_{\mathcal{S}}$ and $\tau$ assigns to each individual name one such combination. To express this formally, we extend the TBox by the axioms in

$$\mathcal{T}_\tau := \{A_{\tau(a)} \equiv C_{\tau(a)} \mid a \in \mathsf{Ind}(\phi) \cup \mathsf{Ind}(\mathcal{K})\},$$

where $A_{\tau(a)}$ are fresh rigid concept names and, for every $Y \subseteq \mathsf{RCon}(\mathcal{T})$, the concept $C_Y$ is defined as

$$\bigsqcap_{A \in Y} A \ \sqcap \bigsqcap_{A \in \mathsf{RCon}(\mathcal{T}) \setminus Y} \neg A.$$

Correspondingly, we extend the conjunctions $\gamma_i \wedge \chi_{\mathcal{S}}$ by

$$\rho_\tau := \bigwedge_{a \in \mathsf{Ind}(\phi) \cup \mathsf{Ind}(\mathcal{K})} A_{\tau(a)}(a)$$

in order to fix the behavior of the rigid concept names on the named individuals.

We need one more definition to formulate the main lemma of this section. We say that an interpretation $\mathcal{I}$ *respects* $\mathcal{D}$ if

$$\mathcal{D} = \{Y \subseteq \mathsf{RCon}(\mathcal{T}) \mid \text{there is a } d \in \Delta^{\mathcal{I}} \text{ with } d \in (C_Y)^{\mathcal{I}}\},$$

which means that every combination of rigid concept names in $\mathcal{D}$ is realized by a domain element of $\mathcal{I}$, and conversely, the domain elements of $\mathcal{I}$ may only realize those combinations that occur in $\mathcal{D}$.

**Lemma 5.1.** *If $N_{RC} \neq \emptyset$ and $N_{RR} = \emptyset$, then $\mathcal{S}$ is r-satisfiable w.r.t. $\iota$ and $\mathcal{K}$ iff there exist $\mathcal{D} \subseteq 2^{\mathsf{RCon}(\mathcal{T})}$ and $\tau\colon \mathsf{Ind}(\phi) \cup \mathsf{Ind}(\mathcal{K}) \to \mathcal{D}$ such that each $\gamma_i \wedge \chi_{\mathcal{S}} \wedge \rho_\tau$, $0 \leq i \leq n$, has a model w.r.t. $\langle \mathcal{T}_{\mathcal{S}} \cup \mathcal{T}_\tau, \mathcal{R}_{\mathcal{S}} \rangle$ that respects $\mathcal{D}$.*

The proof of this lemma can be found in the appendix.

Observe now that the restriction imposed by $\mathcal{D}$ can equivalently be expressed as the conjunction of CQ-literals

$$\sigma_{\mathcal{D}} := (\neg \exists x.A_{\mathcal{D}}(x)) \wedge \bigwedge_{Y \in \mathcal{D}} \exists x.A_Y(x),$$

where $A_Y$ and $A_{\mathcal{D}}$ are fresh concept names that are restricted by adding the axioms $A_{\mathcal{D}} \equiv \bigsqcap_{Y \in \mathcal{D}} \neg A_Y$ and $A_Y \equiv C_Y$ for each $Y \in \mathcal{D}$ to the TBox.[4] We denote by $\mathcal{T}'_{\mathcal{S}}$ the resulting extension of $\mathcal{T}_{\mathcal{S}} \cup \mathcal{T}_\tau$, and have now reduced the r-satisfiability of $\mathcal{S}$ w.r.t. $\iota$ and $\mathcal{K}$ to the consistency of $\gamma_i \wedge \chi_{\mathcal{S}} \wedge \rho_\tau \wedge \sigma_{\mathcal{D}}$ w.r.t. $\langle \mathcal{T}'_{\mathcal{S}}, \mathcal{R}_{\mathcal{S}} \rangle$.

**Theorem 5.2.** *If $N_{RC} \neq \emptyset$ and $N_{RR} = \emptyset$, TCQ entailment is in* co-NP *w.r.t. data complexity.*

Proof. Following the reduction described above, we guess a set $\mathcal{D} \subseteq 2^{\mathsf{RCon}(\mathcal{T})}$ and a function $\tau\colon \mathsf{Ind}(\phi) \cup \mathsf{Ind}(\mathcal{K}) \to \mathcal{D}$, which can be done in nondeterministic polynomial time since $\mathcal{D}$ only depends on $\mathcal{T}$ and $\tau$ is of size linear in the size of the input ABoxes. Next, we check the satisfiability of the polynomially many conjunctions $\gamma_i \wedge \chi_{\mathcal{S}} \wedge \rho_\tau \wedge \sigma_{\mathcal{D}}$ w.r.t. $\langle \mathcal{T}'_{\mathcal{S}}, \mathcal{R}_{\mathcal{S}} \rangle$. Note that $\chi_{\mathcal{S}}$, $\sigma_{\mathcal{D}}$, $\mathcal{T}'_{\mathcal{S}}$, and $\mathcal{R}_{\mathcal{S}}$ do not depend on the input ABoxes, while $\gamma_i$ and $\rho_\tau$ are of polynomial size. Furthermore, only $\chi_{\mathcal{S}}$ may contain negated CQs, and thus their number does not depend on the size of the input ABoxes. Hence, one can see from the proof of Theorem 4.1 that this satisfiability problem can be also be decided in nondeterministic polynomial time in data complexity.

By Lemma 5.1, r-satisfiability of $\mathcal{S}$ w.r.t $\iota$ and $\mathcal{K}$ can be decided in NP, and thus we can obtain the desired complexity upper bound for TCQ entailment as in the proof of Theorem 4.13. □

## 6. Combined Complexity for the Case of Rigid Concept Names

Unfortunately, the approach used in the previous section does not yield a *combined complexity* of co-NExpTime. The reason is that the conjunctions $\chi_{\mathcal{S}}$ and $\sigma_{\mathcal{D}}$ are of exponential size in the size of $\phi$, and thus Theorem 4.1 only yields an upper bound of 2-ExpTime. In this section, we

[4] We did not add all the axioms $A_Y \equiv C_Y$ earlier since we reuse Lemma 5.1 in the following section about combined complexity, and these additional axioms cause an exponential blowup in the size of the TBox.

describe a different approach with a combined complexity of co-NExpTime.

As a first step, we rewrite the Boolean TCQ $\phi$ into a Boolean TCQ $\psi$ of polynomial size in the size of $\phi$ and $\mathcal{K}$ such that answering $\phi$ at time point $n$ is equivalent to answering $\psi$ at time point 0 w.r.t. a trivial sequence of ABoxes. This is done by compiling the ABoxes into the query and postponing $\phi$ using the $\bigcirc$-operator.

**Lemma 6.1.** *Let $\mathcal{K} = \langle (\mathcal{A}_i)_{0\leq i\leq n}, \mathcal{T}, \mathcal{R}\rangle$ be a TKB and $\phi$ be a Boolean TCQ. Then there is a Boolean TCQ $\psi$ of size polynomial in the size of $\phi$ and $\mathcal{K}$ such that $\mathcal{K} \models \phi$ iff $\langle \emptyset, \mathcal{T}, \mathcal{R}\rangle \models \psi$.*

Proof. We define the Boolean TCQ

$$\psi := (\gamma_0 \wedge \bigcirc\gamma_1 \wedge \ldots \wedge \bigcirc^n\gamma_n) \to \bigcirc^n\phi,$$

where $\gamma_i := \bigwedge_{\alpha\in\mathcal{A}_i} \alpha$ and $\bigcirc^i$ abbreviates $i$ nested $\bigcirc$ operators. Obviously, the size of $\psi$ is polynomial in the size of $\phi$ and $\mathcal{K}$. It remains to prove that $\mathcal{K} \models \phi$ iff $\mathcal{K}' := \langle \emptyset, \mathcal{T}, \mathcal{R}\rangle \models \psi$. We have:

$\mathcal{K} \models \phi$

*iff* $\langle (\mathcal{A}_i)_{0\leq i\leq n}, \mathcal{T}, \mathcal{R}\rangle \models \phi$

*iff* $\mathfrak{I}, n \models \phi$ for all $\mathfrak{I} \models \langle (\mathcal{A}_i)_{0\leq i\leq n}, \mathcal{T}, \mathcal{R}\rangle$

*iff* $\mathfrak{I}, n \models \phi$ for all $\mathfrak{I} \models \mathcal{K}'$ with $\mathfrak{I}, 0 \models \gamma_0$; $\mathfrak{I}, 1 \models \gamma_1$; $\ldots$; $\mathfrak{I}, n \models \gamma_n$

*iff* $\mathfrak{I}, 0 \models \bigcirc^n\phi$ for all $\mathfrak{I} \models \mathcal{K}'$ with $\mathfrak{I}, 0 \models \gamma_0$; $\mathfrak{I}, 0 \models \bigcirc\gamma_1$; $\ldots$; $\mathfrak{I}, 0 \models \bigcirc^n\gamma_n$

*iff* $\mathfrak{I}, 0 \models \psi$ for all $\mathfrak{I} \models \mathcal{K}'$

*iff* $\mathcal{K}' \models \psi$. □

We can thus focus on deciding whether a Boolean TCQ $\phi$ has a model w.r.t. a TKB $\mathcal{K} = \langle \emptyset, \mathcal{T}, \mathcal{R}\rangle$ containing only one empty ABox. Note that this compilation does not yield a low *data complexity* for the entailment problem since, after encoding the ABoxes into $\phi$, the size of $\chi_\mathcal{S}$ as well as that of the generalized Büchi automaton $\mathcal{G}$ are exponential in the size of the ABoxes (cf. Sections 4.2.1 and 5).

We now again analyze the two conditions of Lemma 4.7, this time with the goal of obtaining a combined complexity of NExpTime for the TCQ satisfiability problem. First, observe that guessing $\mathcal{S} = \{X_1, \ldots, X_k\} \subseteq 2^{\{p_1,\ldots,p_m\}}$ and $\iota\colon \{0\} \to \{1, \ldots, k\}$ can be done in nondeterministic exponential time in the size of $\phi$. Furthermore, by Lemma 4.12, the t-satisfiability test can be realized in ExpTime. It remains to determine the complexity of testing r-satisfiability of $\mathcal{S}$ w.r.t. $\iota$ and $\mathcal{K} = \langle \emptyset, \mathcal{T}, \mathcal{R}\rangle$.

We do this in three steps. First, in Section 6.1 we reduce this problem to a variant of the satisfiability problem for conjunctions of CQ-literals w.r.t. a knowledge base and a set $\mathcal{D}$ as in Section 5. This problem is then further reduced in Section 6.2 to the consistency problem for Boolean $\mathcal{SHQ}^\cap$-knowledge bases w.r.t. $\mathcal{D}$. Finally, the latter problem is shown to be decidable in ExpTime in Section 6.3.

### *6.1. Reduction to atemporal queries*

As mentioned above, we start the r-satisfiability test as in Section 5 by guessing a set $\mathcal{D} \subseteq 2^{\mathsf{RCon}(\mathcal{T})}$ and a mapping $\tau\colon \mathsf{Ind}(\phi) \to \mathcal{D}$. Since $\mathcal{D}$ is of size exponential in $\mathcal{T}$ and $\tau$ is of size polynomial in the size of $\phi$ and $\mathcal{T}$, guessing $\mathcal{D}$ and $\tau$ can also be done in NExpTime. Since $\gamma_0 = \mathsf{true}$, by Lemma 5.1 we know that r-satisfiability of $\mathcal{S}$ is independent of $\iota$ and it suffices to test whether $\chi_\mathcal{S} \wedge \rho_\tau$ has a model w.r.t. $\langle \mathcal{T}_\mathcal{S} \cup \mathcal{T}_\tau, \mathcal{R}_\mathcal{S}\rangle$ that respects $\mathcal{D}$. Instead of applying Theorem 4.1 directly to this problem, which would yield a complexity of 2-ExpTime, we split it into separate subproblems for each component $\chi_i$ of $\chi_\mathcal{S}$. The proof of the next lemma can be found in the appendix.

**Lemma 6.2.** *If $N_{RC} \neq \emptyset$ and $N_{RR} = \emptyset$, then $\mathcal{S}$ is r-satisfiable w.r.t. $\mathcal{K} = \langle \emptyset, \mathcal{T}, \mathcal{R}\rangle$ iff there exist $\mathcal{D} \subseteq 2^{\mathsf{RCon}(\mathcal{T})}$ and $\tau\colon \mathsf{Ind}(\phi) \to \mathcal{D}$ such that each $\widehat{\chi}_i := \chi_i \wedge \rho_\tau$, $1 \leq i \leq k$, has a model w.r.t. $\langle \mathcal{T} \cup \mathcal{T}_\tau, \mathcal{R}\rangle$ that respects $\mathcal{D}$.*

Note that the size of each $\widehat{\chi}_i$ is polynomial in the size of $\phi$ and $\mathcal{T}$ and the number $k$ of these conjunctions is exponential in the size of $\phi$. Moreover, the size of $\mathcal{T}_\tau$ is polynomial in the size of $\phi$ and $\mathcal{T}$. We show in Lemma 6.8 below that we can find the required models for $\widehat{\chi}_i$ w.r.t. $\langle \mathcal{T} \cup \mathcal{T}_\tau, \mathcal{R}\rangle$ that respect $\mathcal{D}$ in exponential time in the size of $\widehat{\chi}_i$, $\mathcal{T}$, $\mathcal{T}_\tau$, and $\mathcal{R}$. This yields the desired complexity result for r-satisfiability, and thus the last result of Table 1 for TCQ entailment.

**Theorem 6.3.** *If $N_{RC} \neq \emptyset$ and $N_{RR} = \emptyset$, TCQ entailment is in* co-NExpTime *w.r.t. combined complexity.*

### *6.2. Reduction to Boolean $\mathcal{SHQ}^\cap$-knowledge bases*

We now show that the problem of checking whether there is a model of a conjunction $\psi$ of CQ-literals w.r.t. a knowledge base $\langle \mathcal{T}, \mathcal{R}\rangle$ that respects a set $\mathcal{D} \subseteq 2^{\mathsf{RCon}(\mathcal{T})}$ can be solved in exponential time in the size of $\psi$, $\mathcal{T}$, and $\mathcal{R}$. As in the proof of Theorem 4.1, we first reduce this problem to a non-entailment problem for a union of Boolean CQs: there is a model of $\psi$ and $\langle \mathcal{T}, \mathcal{R}\rangle$ that respects $\mathcal{D}$ iff there is a model of $\langle \mathcal{A}, \mathcal{T}, \mathcal{R}\rangle$ that respects $\mathcal{D}$ and is not a model of $\rho$ (written $\langle \mathcal{A}, \mathcal{T}, \mathcal{R}\rangle \not\models \rho$ w.r.t. $\mathcal{D}$), where $\mathcal{A}$ is an ABox obtained by instantiating the non-negated CQs in $\psi$ with fresh individual names and $\rho$ is a UCQ constructed from the negated CQs in $\psi$. It thus suffices to show that we can decide query non-entailment $\langle \mathcal{A}, \mathcal{T}, \mathcal{R}\rangle \not\models \rho$ w.r.t. $\mathcal{D}$ in time exponential in the size of $\mathcal{A}$, $\mathcal{T}$, $\mathcal{R}$, and $\rho$.

It is known that $\langle \mathcal{A}, \mathcal{T}, \mathcal{R}\rangle \not\models \rho$ iff there is a *forest model* $\mathcal{I}$ of $\mathcal{A}$, $\mathcal{T}$, and $\mathcal{R}$ such that $\mathcal{I} \not\models \rho$ [8, 14]. We define here forest models for the more general case of *Boolean $\mathcal{SHQ}^\cap$-knowledge bases* (recall Definition 2.3) since we need them for the subsequent reductions and in the proof of Lemma 6.14.

**Definition 6.4 (forest model).** A *tree* is a non-empty prefix-closed subset of $\mathbb{N}^*$, where $\mathbb{N}^*$ denotes the set of all finite words over the non-negative integers.

An interpretation $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$ is a *forest base* for a Boolean $\mathcal{SHQ}^\cap$-knowledge base $\mathcal{B} = \langle \Psi, \mathcal{R} \rangle$ if

- $\Delta^\mathcal{I} \subseteq \mathsf{Ind}(\Psi) \times \mathbb{N}^*$ such that for all $a \in \mathsf{Ind}(\Psi)$ the set $\{u \mid (a, u) \in \Delta^\mathcal{I}\}$ is a tree;
- if $((a, u), (b, v)) \in r^\mathcal{I}$, then either $u = v = \varepsilon$, or $a = b$ and $v = u \cdot c$ for some $c \in \mathbb{N}$, where $\cdot$ denotes concatenation; and
- for every $a \in \mathsf{Ind}(\Psi)$, we have $a^\mathcal{I} = (a, \varepsilon)$.

A model $\mathcal{J} = (\Delta^\mathcal{J}, \cdot^\mathcal{J})$ of $\mathcal{B}$ is called a *forest model of* $\mathcal{B}$ if there is a forest base $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$ for $\mathcal{B}$ such that $\Delta^\mathcal{I} = \Delta^\mathcal{J}$, for each $A \in N_\mathrm{C}$, we have $A^\mathcal{I} = A^\mathcal{J}$, for each $a \in N_\mathrm{I}$, we have $a^\mathcal{I} = a^\mathcal{J}$, and for each $r \in N_\mathrm{R}$, we have

$$r^\mathcal{J} = r^\mathcal{I} \cup \bigcup_{\mathcal{R} \models s \sqsubseteq r,\ \mathcal{R} \models \mathsf{trans}(s)} (s^\mathcal{I})^+,$$

where $\cdot^+$ denotes the transitive closure.

Note that $\mathcal{B} = \langle \Psi, \mathcal{R} \rangle$ has a model that respects $\mathcal{D}$ iff $\langle \Psi \wedge A(a), \mathcal{R} \rangle$ has a model that respects $\mathcal{D}$, where $a$ is a fresh individual name and $A$ is a fresh concept name. We thus assume without loss of generality that $\Psi$ always contains at least one individual name. This is necessary to ensure that there is a non-empty forest base for $\mathcal{B}$.

As an example of a forest model, consider Figure 2, where a graphical representation of a forest model is given. It depicts the individual names $a$, $b$, and $c$, which represent the roots $(a, \varepsilon)$, $(b, \varepsilon)$, and $(c, \varepsilon)$ of three trees. Moreover, $s$ is a simple role name, and $r$ is a transitive role name. The solid arrows denote the role connections that are present in the corresponding forest base, and the dashed arrows denote role connections that are introduced due to transitivity.

The construction in the proof of the following lemma is very similar to the one in [14], but we extend the previous result to Boolean knowledge bases, take into account a set $\mathcal{D}$, and provide a full proof in the appendix.

**Lemma 6.5.** *Let $\mathcal{B}$ be a Boolean $\mathcal{SHQ}^\cap$-knowledge base, let $A_1, \dots, A_k$ be concept names occurring in $\mathcal{B}$, and let $\mathcal{D} \subseteq 2^{\{A_1,\dots,A_k\}}$. Then $\mathcal{B}$ has a model that respects $\mathcal{D}$ iff it has a forest model that respects $\mathcal{D}$.*

We can also extend the mentioned result about non-entailment of UCQs from [8, 14] to our setting. In the following, we assume that the UCQ $\rho$ contains only individuals that also occur in the ABox (or Boolean axiom formula). If this is not the case for an individual name $a$, we can simply add $A(a)$ to the ABox, where $A$ is a new concept name.

**Lemma 6.6.** *We have $\langle \mathcal{A}, \mathcal{T}, \mathcal{R} \rangle \not\models \rho$ w.r.t. $\mathcal{D}$ iff there is a forest model $\mathcal{J}$ of $\langle \mathcal{A}, \mathcal{T}, \mathcal{R} \rangle$ that respects $\mathcal{D}$ with $\mathcal{J} \not\models \rho$.*

Recall that we want to decide the existence of such a forest model in time exponential in the size of $\mathcal{A}$, $\mathcal{T}$, $\mathcal{R}$, and $\rho$. To this purpose, we further reduce this problem following an idea from [8]. There, the notion of a *spoiler* is introduced. A spoiler is an $\mathcal{SHQ}^\cap$-knowledge base $\langle \mathcal{A}', \mathcal{T}', \emptyset \rangle$ that states properties that must be satisfied such that a query is not entailed by a knowledge base. The ABox $\mathcal{A}'$ of such a spoiler may also contain *negated* assertions, and can thus be seen as a Boolean knowledge base, but for simplicity we will continue to regard it as a set. Furthermore, a spoiler may contain role conjunctions.

It is shown in [8] that $\langle \mathcal{A}, \mathcal{T}, \mathcal{R} \rangle \not\models \rho$ iff there is a spoiler $\langle \mathcal{A}', \mathcal{T}', \emptyset \rangle$ for $\langle \mathcal{A}, \mathcal{T}, \mathcal{R} \rangle$ such that $\langle \mathcal{A} \cup \mathcal{A}', \mathcal{T} \cup \mathcal{T}', \mathcal{R} \rangle$ is consistent. Additionally, all spoilers can be computed in time exponential in the size of $\langle \mathcal{A}, \mathcal{T}, \mathcal{R} \rangle$ and $\rho$, and each spoiler is of polynomial size. In the proof of these results, one only has to deal with forest models, which furthermore do not need to be modified. More formally, for any forest model $\mathcal{I}$ of $\langle \mathcal{A}, \mathcal{T}, \mathcal{R} \rangle$ that does not satisfy $\rho$ there is a spoiler $\langle \mathcal{A}', \mathcal{T}', \emptyset \rangle$ that also has $\mathcal{I}$ as a model and, conversely, every forest model of the knowledge base $\langle \mathcal{A}, \mathcal{T}, \mathcal{R} \rangle$ that also satisfies a spoiler $\langle \mathcal{A}', \mathcal{T}', \emptyset \rangle$ does not satisfy $\rho$ (see the proof of Lemma 3 in [41]). This implies the following more general result that also takes into account the set $\mathcal{D}$.

**Proposition 6.7.** *We have $\langle \mathcal{A}, \mathcal{T}, \mathcal{R} \rangle \not\models \rho$ w.r.t. $\mathcal{D}$ iff there is a spoiler $\langle \mathcal{A}', \mathcal{T}', \emptyset \rangle$ for $\langle \mathcal{A}, \mathcal{T}, \mathcal{R} \rangle$ such that there is a model of $\langle \mathcal{A} \cup \mathcal{A}', \mathcal{T} \cup \mathcal{T}', \mathcal{R} \rangle$ that respects $\mathcal{D}$.*

It remains to show that the existence of such a model can be checked in exponential time in the size of $\langle \mathcal{A} \cup \mathcal{A}', \mathcal{T} \cup \mathcal{T}', \mathcal{R} \rangle$, and therefore in exponential time in the size of $\psi$, $\mathcal{T}$, and $\mathcal{R}$. We will show a more general result for Boolean knowledge bases in the next section (Theorem 6.15). Together with the reductions described in this section, we obtain the desired complexity result.

**Lemma 6.8.** *The existence of a model of a conjunction of CQ-literals $\psi$ w.r.t. a knowledge base $\langle \mathcal{T}, \mathcal{R} \rangle$ that respects $\mathcal{D}$ can be decided in exponential time in the size of $\psi$, $\mathcal{T}$, and $\mathcal{R}$.*

### *6.3. Consistency of Boolean $\mathcal{SHQ}^\cap$-knowledge bases*

For the final result of this paper, we consider a Boolean $\mathcal{SHQ}^\cap$-knowledge base $\mathcal{B} = \langle \Psi, \mathcal{R} \rangle$, a collection of concept names $A_1, \dots, A_k$ occurring in $\mathcal{B}$, and a subset $\mathcal{D}$ of $2^{\{A_1,\dots,A_k\}}$. We assume here that all GCIs in $\Psi$ are of the form $\top \sqsubseteq C$; this is without loss of generality since any GCI $C \sqsubseteq D$ is equivalent to $\top \sqsubseteq \neg(C \sqcap \neg D)$.

We will show that deciding consistency of $\mathcal{B}$ w.r.t. $\mathcal{D}$, i.e., whether $\mathcal{B}$ has a model that respects $\mathcal{D}$, can be done in exponential time in the size of $\mathcal{B}$. This complexity result is tight since already for classical $\mathcal{SHQ}^\cap$-knowledge bases, the consistency problem (without $\mathcal{D}$) is ExpTime-complete [8, 37]. The complexity of this problem even remains in ExpTime when simple role conjunctions are allowed to occur in number restrictions and non-simple roles are allowed in role conjunctions in existential restrictions [42].

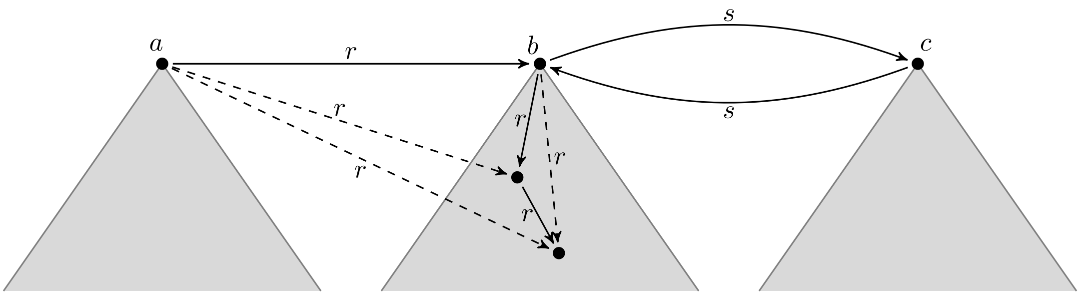


Figure 2: An example of a forest model.

The proof is an adaptation of the proof of Lemma 6.4 in [12], which is again an adaptation of the proof of Theorem 2.27 in [17], which shows that consistency of Boolean $\mathcal{ALC}$-knowledge bases can be decided in exponential time. An earlier version of this proof for $\mathcal{ALC}^{\cap}$ can be found in [29, 34]. There, for role conjunctions, additional concept names are introduced that function as so-called *pebbles* that mark elements that have specific role predecessors, an idea borrowed from [43–45]. In this paper, we employ instead systems of equations over non-negative integers to deal with role conjunctions, transitivity axioms, role inclusions, and number restrictions simultaneously.

For the subsequent construction, we extend the notion of a *quasimodel* from [12], which is an abstract description of a model that characterizes domain elements by the concepts they satisfy. We first introduce several auxiliary notions.

We define $\mathsf{Con}(\Psi)$ as the set of all concepts occurring in $\Psi$, and $\mathsf{Con}(\mathcal{B})$ as the closure under negation of the set

$$\mathsf{Con}(\Psi) \cup \{\exists r.C \mid \exists s.C \in \mathsf{Con}(\Psi),\ \mathcal{R} \models r \sqsubseteq s, \text{ and } \mathcal{R} \models \mathsf{trans}(r)\}.$$

The reason that we consider these additional existential restrictions is that they are needed to properly deal with transitive roles (see Definition 6.9).

Similarly, we denote by $\mathsf{Sub}(\Psi)$ the set of all subformulae of $\Psi$, by $\mathsf{Rol}(\mathcal{B})$ the set of all role names occurring in $\mathcal{B}$, and by $\mathsf{Sub}(\mathcal{B})$ the closure under negation of the set

$$\mathsf{Sub}(\Psi) \cup \{r(a,b) \mid r \in \mathsf{Rol}(\mathcal{B}),\ a,b \in \mathsf{Ind}(\mathcal{B})\}.$$

We include all possible role assertions about individuals and role names from $\mathcal{B}$ since we later want to close sets of role assertions under $\mathcal{R}$ to be able to read off all relevant consequences about individuals from such a set (see Definition 6.11).

In the following, we identify $\neg\neg\psi$ with $\psi$ for all concepts and Boolean knowledge bases $\psi$. Thus, all sets introduced above are polynomial in the size of $\mathcal{B}$.

**Definition 6.9 (concept type).** A *concept type* for $\mathcal{B}$ is a set $\mathbf{c} \subseteq \mathsf{Con}(\mathcal{B}) \cup \mathsf{Ind}(\Psi)$ such that:

- $C \sqcap D \in \mathbf{c}$ iff $C, D \in \mathbf{c}$ for all $C \sqcap D \in \mathsf{Con}(\mathcal{B})$;
- $\neg C \in \mathbf{c}$ iff $C \notin \mathbf{c}$ for all $\neg C \in \mathsf{Con}(\mathcal{B})$; and
- $a \in \mathbf{c}$ for $a \in \mathsf{Ind}(\Psi)$ implies $b \notin \mathbf{c}$ for all $b \in \mathsf{Ind}(\Psi)$ with $b \neq a$.

Given two concept types $\mathbf{c}, \mathbf{d}$ and a role name $r$, we say that $\mathbf{c}$ and $\mathbf{d}$ are *$r$-compatible (w.r.t. $\mathcal{R}$)* (written $(\mathbf{c}, \mathbf{d}) \in r^{\mathcal{R}}$) if the following conditions are satisfied:

- for all $\neg(\exists r.D) \in \mathbf{c}$, we have $\neg D \in \mathbf{d}$; and
- for all $s \in N_{\mathrm{R}}$ with $\mathcal{R} \models r \sqsubseteq s$, $\mathcal{R} \models \mathsf{trans}(r)$, and $\neg(\exists s.D) \in \mathbf{c}$, we have $\neg(\exists r.D) \in \mathbf{d}$.

Obviously, the number of concept types is exponential in the size of $\Psi$. The $r$-compatibility of two concept types $\mathbf{c}, \mathbf{d}$ indicates that it is possible to connect them via an $r$-edge without violating the value restrictions in $\mathbf{c}$. These conditions are very similar to the tableau rules $(\forall)$ and $(\forall_+)$ that deal with value restrictions in the presence of role inclusions and transitivity axioms (see, e.g. [36]).

**Definition 6.10 (role type).** A *role type* for $\mathcal{B}$ is a set $\mathbf{r} \subseteq \mathsf{Rol}(\mathcal{B})$ such that

- if $s \sqsubseteq r \in \mathcal{R}$, then $s \in \mathbf{r}$ implies $r \in \mathbf{r}$.

We denote the set of all role types for $\mathcal{B}$ by $\mathfrak{R}(\mathcal{B})$.

For $\mathbf{r} \in \mathfrak{R}(\mathcal{B})$, we say that two concept types $\mathbf{c}, \mathbf{d}$ for $\mathcal{B}$ are *$\mathbf{r}$-compatible (w.r.t. $\mathcal{R}$)* (written $(\mathbf{c}, \mathbf{d}) \in \mathbf{r}^{\mathcal{R}}$) iff they are $r$-compatible w.r.t. $\mathcal{R}$ for every $r \in \mathbf{r}$.

Again, the number of role types for $\mathcal{B}$ is exponential in the size of $\mathcal{B}$.

Finally, a quasimodel also has to determine which of the axioms in $\Psi$ it satisfies.

**Definition 6.11 (formula type).** A *formula type* for $\mathcal{B}$ is a set $\mathbf{f} \subseteq \mathsf{Sub}(\mathcal{B})$ such that:

- $\Psi \in \mathbf{f}$;
- $\neg\psi \in \mathbf{f}$ iff $\psi \notin \mathbf{f}$ for all $\neg\psi \in \mathsf{Sub}(\mathcal{B})$;
- $\psi_1 \wedge \psi_2 \in \mathbf{f}$ iff $\{\psi_1, \psi_2\} \subseteq \mathbf{f}$ for all $\psi_1 \wedge \psi_2 \in \mathsf{Sub}(\mathcal{B})$;
- if $r(a,b) \in \mathbf{f}$ and $\mathcal{R} \models r \sqsubseteq s$, then $s(a,b) \in \mathbf{f}$; and
- if $r(a,b) \in \mathbf{f}$, $r(b,c) \in \mathbf{f}$, and $\mathcal{R} \models \mathsf{trans}(r)$, then $r(a,c) \in \mathbf{f}$.

The number of formula types for $\mathcal{B}$ is exponential in the size of $\mathcal{B}$. Using these definitions, we can now define model candidates, and later refine this notion to quasimodels.

**Definition 6.12 (model candidate).** A *model candidate* for $\mathcal{B}$ is a triple $\mathcal{M} = (\mathcal{S}, \iota, \mathbf{f})$ such that

- $\mathcal{S}$ is a set of concept types for $\mathcal{B}$ such that for any $\mathbf{c}, \mathbf{d} \in \mathcal{S}$ with $\mathbf{c} \neq \mathbf{d}$, we have $\mathbf{c} \cap \mathbf{d} \cap \mathsf{Ind}(\Psi) = \emptyset$;
- $\iota \colon \mathsf{Ind}(\Psi) \to \mathcal{S}$ is a function such that $a \in \iota(a)$ for all $a \in \mathsf{Ind}(\Psi)$; and
- $\mathbf{f}$ is a formula type for $\mathcal{B}$.

Intuitively, the set $\mathcal{S}$ determines the behavior of the domain elements, while $\iota$ fixes the interpretation of the named domain elements, and $\mathbf{f}$ ensures that $\mathcal{B}$ is satisfied. We denote by $\mathcal{S}_u$ the set $\mathcal{S} \setminus \iota(\mathsf{Ind}(\Psi))$, i.e., the set of all those concept types that do not contain an individual name. These types represent the unnamed domain elements of the model candidate. To define quasimodels, we add to the definition of a model candidate several conditions that ensure that the concept types can indeed be assembled into a model of $\mathcal{B}$.

To satisfy the number restrictions in the concept types of a model candidate $\mathcal{M} = (\mathcal{S}, \iota, \mathbf{f})$, we introduce, for each $\mathbf{c} \in \mathcal{S}$, a system of equations $E_{\mathcal{M},\mathbf{c}}$ with variables ranging over the non-negative integers. Below, we consider mostly inequations, which can, however, easily be turned into equations by introducing new slack variables. In $E_{\mathcal{M},\mathbf{c}}$, we use variables of the form $x_{\mathbf{c},\mathbf{r},\mathbf{d}}$ that determine, for an individual of type $\mathbf{c}$, the number of unnamed $\mathbf{r}$-successors of concept type $\mathbf{d}$, where we require that $(\mathbf{c}, \mathbf{d}) \in \mathbf{r}^{\mathcal{R}}$ and $\mathbf{d} \in \mathcal{S}_u$, i.e., $\mathbf{c}$ and $\mathbf{d}$ are $\mathbf{r}$-compatible and $\mathbf{d}$ does not represent a named individual.

Given $\mathbf{c} \in \mathcal{S}$, $C \in \mathsf{Con}(\mathcal{B})$, and $\mathbf{r} \in \mathfrak{R}(\mathcal{B})$, we can now count the number of *unnamed* $\mathbf{r}$-successors of $\mathbf{c}$ that satisfy $C$ using the following expression:

$$\Xi_{\mathcal{M},\mathbf{c},\mathbf{r},C} := \sum_{C \in \mathbf{d} \in \mathcal{S}_u,\ (\mathbf{c},\mathbf{d}) \in \mathbf{r}^{\mathcal{R}}} x_{\mathbf{c},\mathbf{r},\mathbf{d}}.$$

To count the *named* $\mathbf{r}$-successors of $\mathbf{c}$ that satisfy $C$, we define the constant $\Gamma_{\mathcal{M},\mathbf{c},\mathbf{r},C}$ as

$$\begin{cases} |\{b \in \mathsf{Ind}(\Psi) \mid C \in \iota(b), \text{ and} \\ \qquad\qquad r(a,b) \in \mathbf{f} \text{ iff } r \in \mathbf{r}\}| & \text{if } \mathbf{c} = \iota(a) \\ 0 & \text{otherwise.} \end{cases}$$

To ensure that an at-least restriction ${\geq} n\, r.C \in \mathbf{c}$ is satisfied, we construct the following inequation:

$$\sum_{r \in \mathbf{r} \in \mathfrak{R}(\mathcal{B})} (\Xi_{\mathcal{M},\mathbf{c},\mathbf{r},C} + \Gamma_{\mathcal{M},\mathbf{c},\mathbf{r},C}) \geq n. \tag{E1}$$

Similarly, for each $\neg({\geq} n\, r.C) \in \mathbf{c}$, we add

$$\sum_{r \in \mathbf{r} \in \mathfrak{R}(\mathcal{B})} (\Xi_{\mathcal{M},\mathbf{c},\mathbf{r},C} + \Gamma_{\mathcal{M},\mathbf{c},\mathbf{r},C}) \leq n - 1. \tag{E2}$$

For an existential restriction $E = \exists(r_1 \cap \cdots \cap r_\ell).C \in \mathbf{c}$, we introduce the inequation

$$\sum_{r_1,\dots,r_\ell \in \mathbf{r} \in \mathfrak{R}(\mathcal{B})} (\Xi_{\mathcal{M},\mathbf{c},\mathbf{r},C} + \Gamma_{\mathcal{M},\mathbf{c},\mathbf{r},C}) \geq 1. \tag{E3}$$

Finally, for each $\neg(\exists(r_1 \cap \cdots \cap r_\ell).C) \in \mathbf{c}$, we use the equation

$$\sum_{r_1,\dots,r_\ell \in \mathbf{r} \in \mathfrak{R}(\mathcal{B})} (\Xi_{\mathcal{M},\mathbf{c},\mathbf{r},C} + \Gamma_{\mathcal{M},\mathbf{c},\mathbf{r},C}) = 0. \tag{E4}$$

This finishes the description of $E_{\mathcal{M},\mathbf{c}}$. Note that this system contains *exponentially* many variables in the size of $\mathcal{B}$, but only *polynomially* many equations, and thus it can be solved in exponential time, even if the numbers are given in binary encoding [46] (for details, see the proof of Theorem 6.15).

We finally come to the central definition of this section.

**Definition 6.13 (quasimodel).** The model candidate $\mathcal{M} = (\mathcal{S}, \iota, \mathbf{f})$ for $\mathcal{B}$ is a *quasimodel* for $\mathcal{B}$ if it satisfies the following properties:

(a) $\mathcal{S}$ is not empty;

(b) for every $A(a) \in \mathsf{Sub}(\mathcal{B})$, we have $A(a) \in \mathbf{f}$ iff $A \in \iota(a)$;

(c) for every $r(a, b) \in \mathbf{f}$, we have $(\iota(a), \iota(b)) \in r^{\mathcal{R}}$;

(d) for every $\top \sqsubseteq C \in \mathbf{f}$ and every $\mathbf{c} \in \mathcal{S}$, we have $C \in \mathbf{c}$;

(e) for every $\neg(\top \sqsubseteq C) \in \mathbf{f}$, there is a $\mathbf{c} \in \mathcal{S}$ such that $C \notin \mathbf{c}$; and

(f) for every $\mathbf{c} \in \mathcal{S}$, the system of equations $E_{\mathcal{M},\mathbf{c}}$ has a solution over the non-negative integers.

The quasimodel $\mathcal{M} = (\mathcal{S}, \iota, \mathbf{f})$ for $\mathcal{B}$ *respects* $\mathcal{D}$ if it satisfies:

(g) for every $\mathbf{c} \in \mathcal{S}$, there is a set $Y \in \mathcal{D}$ such that $Y = \mathbf{c} \cap \{A_1, \dots, A_k\}$; and

(h) for every $Y \in \mathcal{D}$, there is a concept type $\mathbf{c} \in \mathcal{S}$ such that $Y = \mathbf{c} \cap \{A_1, \dots, A_k\}$.

We show in the appendix that to check consistency of $\mathcal{B}$ w.r.t. $\mathcal{D}$ it suffices to search for quasimodels for $\mathcal{B}$ that respect $\mathcal{D}$.

**Lemma 6.14.** *Let $\mathcal{B}$ be a Boolean $\mathcal{SHQ}^{\cap}$-knowledge base, let $A_1, \dots, A_k$ be concept names occurring in $\mathcal{B}$, and let $\mathcal{D} \subseteq 2^{\{A_1,\dots,A_k\}}$. Then $\mathcal{B}$ is consistent w.r.t. $\mathcal{D}$ iff it has a quasimodel that respects $\mathcal{D}$.*

It remains to show that one can check the existence of a quasimodel for $\mathcal{B}$ that respects $\mathcal{D}$ in time exponential in the size of $\mathcal{B}$. For this, consider the following algorithm. Given $\mathcal{B} = \langle \Psi, \mathcal{R} \rangle$ and $\mathcal{D}$, it enumerates all model candidates $(\mathcal{S}_u \cup \mathcal{S}_\iota, \iota, \mathbf{f})$ for $\mathcal{B}$, where

- $\mathcal{S}_u$ is the set of *all* concept types for $\mathcal{B}$ that are subsets of $\mathsf{Con}(\mathcal{B})$, and

- $\mathcal{S}_\iota := \{\iota(a) \mid a \in \mathsf{Ind}(\Psi),\ \iota(a) \setminus \{a\} \in \mathcal{S}_u\}$.

We denote these candidates by $\mathcal{M}_1, \dots, \mathcal{M}_N$. Note that each of them is of size exponential in the size of $\mathcal{B}$. It should be clear that

$$N \ \leq \ 2^{|\mathsf{Con}(\mathcal{B})|\cdot|\mathsf{Ind}(\Psi)|} \cdot 2^{|\mathsf{Sub}(\mathcal{B})|},$$

and thus the enumeration of $\mathcal{M}_1, \dots, \mathcal{M}_N$ can be done in exponential time since $\mathsf{Con}(\mathcal{B})$ and $\mathsf{Sub}(\mathcal{B})$ are of size polynomial in the size of $\mathcal{B}$.

Now, set $i = 1$ and consider $\mathcal{M}_i = (\mathcal{S}, \iota, \mathbf{f})$.

*Step 1.* Check whether $\mathcal{M}_i$ satisfies (b) and (c).

If it does, continue with Step 2. Otherwise, stop considering $\mathcal{M}_i$ and go to Step 5.

*Step 2.* Check each concept type in $\mathcal{S}$. A concept type $\mathbf{c} \in \mathcal{S}$ is called *defective* if it violates (d) for some $\top \sqsubseteq C \in \mathbf{f}$ or it violates (g).

If a defective $\mathbf{c} \in \mathcal{S} \setminus \mathcal{S}_\iota$ is found, then set $\mathcal{S} := \mathcal{S} \setminus \{\mathbf{c}\}$ and continue with Step 2. If a defective $\mathbf{c} \in \mathcal{S}_\iota$ is found, then stop considering $\mathcal{M}_i$ and go to Step 5. If no defective concept types in $\mathcal{S}$ are found, continue with Step 3.

*Step 3.* Consider the model candidate $\mathcal{M}' = (\mathcal{S}', \iota, \mathbf{f})$ obtained from the previous step. For every $\mathbf{c} \in \mathcal{S}'$, check whether $E_{\mathcal{M}',\mathbf{c}}$ has a solution.

If a $\mathbf{c} \in \mathcal{S}'_u$ is found such that $E_{\mathcal{M}',\mathbf{c}}$ has no solution, then remove $\mathbf{c}$ from $\mathcal{S}'$ and redo Step 3. If a $\mathbf{c} \in \mathcal{S}'_\iota$ is found such that $E_{\mathcal{M}',\mathbf{c}}$ has no solution, then go to Step 5. If no such concept type in $\mathcal{S}'$ is found, continue with Step 4.

*Step 4.* Check whether the model candidate $(\mathcal{S}'', \iota, \mathbf{f})$ obtained from Step 3 satisfies (a), (e), and (h).

If it does, stop with output "quasimodel that respects $\mathcal{D}$ found." Otherwise, continue with Step 5.

*Step 5.* Set $i := i + 1$. If $i \leq N$, continue with Step 1. Otherwise, stop with output "no quasimodel that respects $\mathcal{D}$ exists."

We show in the appendix that the algorithm is sound and complete and terminates in exponential time. By Lemma 6.14, we get the following result.

**Theorem 6.15.** *Let $\mathcal{B}$ be a Boolean $\mathcal{SHQ}^\cap$-knowledge base, let $A_1, \dots, A_k$ be concept names occurring in $\mathcal{B}$, and let $\mathcal{D} \subseteq 2^{\{A_1,\dots,A_k\}}$. Then consistency of $\mathcal{B}$ w.r.t. $\mathcal{D}$ can be decided in time exponential in the size of $\mathcal{B}$.*

## 7. Conclusions

We have introduced a new temporal query language that extends the temporal DL $\mathcal{ALC}$-LTL to $\mathcal{SHQ}$ and uses simple conjunctive queries as atoms. Our complexity results on the entailment problem for such queries w.r.t. temporal knowledge bases are summarized in Table 1. Without any rigid names, we observed that entailment of TCQs is as hard as entailment of CQs w.r.t. atemporal $\mathcal{ALC}$- and $\mathcal{SHQ}$-knowledge bases, i.e., in this case adding temporal operators to the query language does not increase the complexity. However, if we allow rigid concept names (but no rigid role names), the picture changes. While the data complexity remains the same as in the atemporal case, the combined complexity of query entailment increases to co-NExpTime, i.e., the non-entailment problem is as hard as satisfiability in $\mathcal{ALC}$-LTL. If we further add rigid role names, the combined complexity (of non-entailment) again increases in accordance with the complexity of satisfiability in $\mathcal{ALC}$-LTL. For data complexity, it is still unclear whether adding rigid role names results in an increase. We have shown an upper bound of ExpTime (which is one exponential better than the combined complexity), but the only lower bound we have is the trivial one of co-NP.

Further work will include trying to close this gap. Moreover, it would be interesting to find out what effect the addition of inverse roles has on the complexity of query entailment in the temporal case. Given the results for $\mathcal{ALCI}$ and $\mathcal{SHIQ}$ in the atemporal case, where query entailment is 2-ExpTime-complete w.r.t. combined complexity [8] and co-NP-complete w.r.t. data complexity [13], there is the possibility that the problem remains co-NP-complete w.r.t. data complexity also in the temporal case, and 2-ExpTime-complete w.r.t. combined complexity for all three settings considered in this paper (i.e., without rigid names, without rigid role names, with rigid names). But showing this will require considerable extensions of the proof techniques employed until now since the presence of inverse roles creates additional problems. We have also left open the complexity of the entailment problem for non-simple TCQs, which is already 2-ExpTime-hard in $\mathcal{SH}$ [33].


### Acknowledgments

This work was partially supported by the DFG in the Collaborative Research Center 912 (HAEC) and in the Research Training Group 1763 (QuantLA). We also thank the anonymous reviewers for their suggestions for improving this paper.



[1] F. Baader, A. Bauer, P. Baumgartner, A. Cregan, A. Gabaldon, K. Ji, K. Lee, D. Rajaratnam, R. Schwitter, A novel architecture for situation awareness systems, in: M. Giese, A. Waaler (Eds.), Proceedings of the 18th International Conference on Automated Reasoning with Analytic Tableaux and Related Methods (TABLEAUX 2009), Vol. 5607 of Lecture Notes in Computer Science, Springer-Verlag, 2009, pp. 77–92.

[2] M. R. Endsley, Toward a theory of situation awareness in dynamic systems, Human Factors 37 (1) (1995) 32–64.

[3] S. Abiteboul, R. Hull, V. Vianu, Foundations of Databases, Addison-Wesley, 1995.

[4] S. Decker, M. Erdmann, D. Fensel, R. Studer, Ontobroker: Ontology based access to distributed and semi-structured information, in: R. Meersman, Z. Tari, S. M. Stevens (Eds.), Proceedings of the 8th Working Conference on Database Semantics (DS-8), Vol. 138 of IFIP Conference Proceedings, Kluwer, Rotorua, New Zealand, 1999, pp. 351–369.

[5] A. Poggi, D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, R. Rosati, Linking data to ontologies, Journal on Data Semantics X (2008) 133–173.

[6] A. K. Chandra, P. M. Merlin, Optimal implementation of conjunctive queries in relational data bases, in: J. E. Hopcroft, E. P. Friedman, M. A. Harrison (Eds.), Proceedings of the 9th Annual ACM Symposium on Theory of Computing (STOC 1977), ACM Press, Boulder, Colorado, USA, 1977, pp. 77–90.

[7] D. Calvanese, G. De Giacomo, M. Lenzerini, On the decidability of query containment under constraints, in: A. O. Mendelzon, J. Paredaens (Eds.), Proceedings of the 17th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems (PODS'98), ACM Press, Seattle, Washington, USA, 1998, pp. 149—158.

[8] C. Lutz, The complexity of conjunctive query answering in expressive description logics, in: A. Armando, P. Baumgartner, G. Dowek (Eds.), Proceedings of the 4th International Joint Conference on Automated Reasoning (IJCAR 2008), Vol. 5195 of Lecture Notes in Computer Science, Springer-Verlag, Sydney, Australia, 2008, pp. 179–193.

[9] D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, R. Rosati, Data complexity of query answering in description logics, in: P. Doherty, J. Mylopoulos, C. A. Welty (Eds.), Proceedings of the 10th International Conference on Principles of Knowledge Representation and Reasoning (KR 2006), AAAI Press, Lake District of the United Kingdom, 2006, pp. 260–270.

[10] D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, A. Poggi, M. Rodriguez-Muro, R. Rosati, Ontologies and databases: The DL-Lite approach, in: S. Tessaris, E. Franconi, T. Eiter, C. Gutierrez, S. Handschuh, M.-C. Rousset, R. A. Schmidt (Eds.), Reasoning Web, 5th Int. Summer School 2009, Tutorial Lectures, Vol. 5689 of Lecture Notes in Computer Science, Springer-Verlag, Brixen-Bressanone, Italy, 2009, pp. 255–356.

[11] A. Pnueli, The temporal logic of programs, in: Proceedings of the 18th Annual Symposium on Foundations of Computer Science (FOCS 1977), IEEE Computer Society Press, Providence, Rhode Island, USA, 1977, pp. 46–57.

[12] F. Baader, S. Ghilardi, C. Lutz, LTL over description logic axioms, ACM Transactions on Computational Logic 13 (3).

[13] M. Ortiz, D. Calvanese, T. Eiter, Characterizing data complexity for conjunctive query answering in expressive description logics, in: Proceedings of the 21st National Conference on Artificial Intelligence (AAAI 2006) and the 18th Innovative Applications of Artificial Intelligence Conference, AAAI Press, Boston, Massachusetts, USA, 2006, pp. 275–280.

[14] B. Glimm, I. Horrocks, C. Lutz, U. Sattler, Answering conjunctive queries in the $\mathcal{SHIQ}$ description logic, Journal of Artificial Intelligence Research 31 (2008) 150–197. `doi:10.1613/jair.2372`.

[15] P. Wolper, M. Y. Vardi, A. P. Sistla, Reasoning about infinite computation paths, in: Proceedings of the 24th Annual Symposium on Foundations of Computer Science (FOCS 1983), IEEE Computer Society Press, 1983, pp. 185–194.

[16] M. Y. Vardi, P. Wolper, Reasoning about infinite computations, Information and Computation 155 (1) (1994) 1–37. `doi:10.1006/inco.1994.1092`.

[17] D. M. Gabbay, Á. Kurucz, F. Wolter, M. Zakharyaschev, Many-Dimensional Modal Logics: Theory and Applications, Elsevier Science, 2003.

[18] F. Wolter, M. Zakharyaschev, Temporalizing description logics, in: D. Gabbay, M. de Rijke (Eds.), Frontiers of Combining Systems 2, Vol. 7 of Studies in Logic and Computation, Research Studies Press/Wiley, 2000, pp. 379–402.

[19] C. Lutz, F. Wolter, M. Zakharyaschev, Temporal description logics: A survey, in: S. Demri, C. S. Jensen (Eds.), Proceedings of the 15th International Symposium on Temporal Representation and Reasoning (TIME 2008), IEEE Press, 2008, pp. 3–14. `doi:10.1109/TIME.2008.14`.

[20] A. Artale, R. Kontchakov, C. Lutz, F. Wolter, M. Zakharyaschev, Temporalising tractable description logics, in: V. Goranko, X. S. Wang (Eds.), Proceedings of the 14th International Symposium on Temporal Representation and Reasoning (TIME 2007), IEEE Press, 2007, pp. 11–22.

[21] B. Motik, Representing and querying validity time in RDF and OWL: A logic-based approach, Journal of Web Semantics 12–13 (2012) 3–21. `doi:10.1016/j.websem.2011.11.004`.

[22] C. Gutiérrez, C. A. Hurtado, A. A. Vaisman, Temporal RDF, in: A. Gómez-Pérez, J. Euzenat (Eds.), Proceedings of the 2nd European Semantic Web Conference (ESWC'05), Vol. 3532 of Lecture Notes in Computer Science, Springer-Verlag, 2005, pp. 93–107. `doi:10.1007/11431053_7`.

[23] A. Artale, E. Franconi, F. Wolter, M. Zakharyaschev, A temporal description logic for reasoning over conceptual schemas and queries, in: Proceedings of the 8th European Conference on Logics in Artificial Intelligence (JELIA 2002), Springer-Verlag, 2002, pp. 98–110. `doi:10.1007/3-540-45757-7_9`.

[24] A. Artale, R. Kontchakov, F. Wolter, M. Zakharyaschev, Temporal description logic for ontology-based data access, in: F. Rossi (Ed.), Proceedings of the 23rd Interntational Joint Conference on Artificial Intelligence (IJCAI 2013), AAAI Press, 2013, pp. 711–717.
URL `http://ijcai.org/papers13/Papers/IJCAI13-112.pdf`

[25] S. Borgwardt, M. Lippmann, V. Thost, Temporal query answering in the description logic *DL-Lite*, in: P. Fontaine, C. Ringeissen, R. A. Schmidt (Eds.), Proceedings of the 9th International Symposium on Frontiers of Combining Systems (FroCoS 2013), Vol. 8152 of Lecture Notes in Computer Science, Springer-Verlag, 2013, pp. 165–180.

[26] J. Chomicki, Efficient checking of temporal integrity constraints using bounded history encoding, ACM Transactions on Database Systems 20 (2) (1995) 148–186. `doi:10.1145/210197.210200`.

[27] J. Chomicki, D. Toman, Temporal databases, in: M. Fischer, D. Gabbay, L. Vila (Eds.), Handbook of Temporal Reasoning in Artificial Intelligence, Elsevier, 2005, Ch. 14, pp. 429–467.

[28] V. Gutiérrez-Basulto, S. Klarman, Towards a unifying approach to representing and querying temporal data in description logics, in: M. Krötzsch, U. Straccia (Eds.), Proceedings of the 6th International Conference on Web Reasoning and Rule Systems (RR'12), Vol. 7497 of Lecture Notes in Computer Science, Springer-Verlag, 2012, pp. 90–105. `doi:10.1007/978-3-642-33203-6_8`.

[29] F. Baader, S. Borgwardt, M. Lippmann, Temporalizing ontology-based data access, in: M. P. Bonacina (Ed.), Proc. of the 24th Int. Conf. on Automated Deduction (CADE'13), Vol. 7898 of Lecture Notes in Computer Science, Springer-Verlag, 2013, pp. 330–344. `doi:10.1007/978-3-642-38574-2_23`.

[30] B. Suntisrivaraporn, F. Baader, S. Schulz, K. Spackman, Replacing SEP-triplets in SNOMED CT using tractable description logic operators, in: R. Bellazzi, A. Abu-Hanna, J. Hunter (Eds.), Proceedings of the 11th Conference on Artificial Intelligence in Medicine (AIME'07), Vol. 4594 of Lecture Notes in Computer Science, Springer-Verlag, 2007, pp. 287–291.

[31] B. Glimm, I. Horrocks, U. Sattler, Unions of conjunctive queries in $\mathcal{SHOQ}$, in: G. Brewka, J. Lang (Eds.), Proceedings of the 11th International Conference on the Principles of Knowledge Representation and Reasoning (KR'08), AAAI Press, 2008, pp. 252–262.

[32] D. Calvanese, T. Eiter, M. Ortiz, Regular path queries in expressive description logics with nominals, in: C. Boutilier (Ed.), Proceedings of the 21st International Joint Conferences on Artificial Intelligence (IJCAI'09), AAAI Press, 2009, pp. 714–720.

[33] T. Eiter, C. Lutz, M. Ortiz, M. Šimkus, Query answering in description logics with transitive roles, in: C. Boutilier (Ed.), Proceedings of the 21st International Joint Conference on Artifi-

cial Intelligence (IJCAI 2009), AAAI Press, 2009, pp. 759–764.
[34] F. Baader, S. Borgwardt, M. Lippmann, On the complexity of temporal query answering, LTCS-Report 13-01, Technische Universität Dresden, Germany, see `http://lat.inf.tu-dresden.de/research/reports.html`. (2012).
[35] F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi, P. F. Patel-Schneider (Eds.), The Description Logic Handbook: Theory, Implementation, and Applications, 2nd Edition, Cambridge University Press, 2007.
[36] I. Horrocks, U. Sattler, S. Tobies, Practical reasoning for very expressive description logics, Journal of the Interest Group in Pure and Applied Logic 8 (3) (2000) 239–263. `doi:10.1093/jigpal/8.3.239`.
[37] S. Tobies, Complexity results and practical algorithms for logics in knowledge representation, Ph.D. thesis, Fakultät für Mathematik, Informatik und Naturwissenschaften, RWTH Aachen (2001).
[38] A. Schaerf, On the complexity of the instance checking problem in concept languages with existential quantification, Journal of Intelligent Information Systems 2 (3) (1993) 265–278. `doi:10.1007/bf00962071`.
[39] R. Gerth, D. A. Peled, M. Y. Vardi, P. Wolper, Simple on-the-fly automatic verification of linear temporal logic, in: P. Dembinski, M. Sredniawa (Eds.), Proceedings of the 15th IFIP WG6.1 International Symposium on Protocol Specification, Testing and Verification XV, Chapman & Hall, Ltd., London, UK, 1996, pp. 3–18.
[40] C. Baier, J.-P. Katoen, Principles of Model Checking, The MIT Press, Cambridge, Massachusetts, USA, 2008.
[41] C. Lutz, Two upper bounds for conjunctive query answering in $\mathcal{SHIQ}$, in: F. Baader, C. Lutz, B. Motik (Eds.), Proceedings of the 21st International Workshop on Description Logics (DL 2008), Vol. 353 of CEUR Workshop Proceedings, 2008. URL `http://ceur-ws.org/Vol-353/Lutz.pdf`
[42] B. Glimm, Y. Kazakov, Role conjunctions in expressive description logics, in: I. Cervesato, H. Veith, A. Voronkov (Eds.), Proceedings of the 15th International Conference on Logic for Programming, Artificial Intelligence, and Reasoning (LPAR'08), Vol. 5330 of Lecture Notes in Computer Science, Springer-Verlag, 2008, pp. 391–405.
[43] R. Danecki, Nondeterministic propositional dynamic logic with intersection is decidable, in: A. Skowron (Ed.), Proceedings of the 5th Symposium on Computation Theory, Vol. 208 of Lecture Notes in Computer Science, Springer-Verlag, 1984, pp. 34–53.
[44] G. De Giacomo, F. Massacci, Combining deduction and model checking into tableaux and algorithms for Converse-PDL, Information and Computation 162 (1–2) (2000) 117–137. `doi:10.1006/inco.1999.2852`.
[45] F. Massacci, Decision procedures for expressive description logics with intersection, composition, converse of roles and role identity, in: B. Nebel (Ed.), Proceedings of the 17th International Joint Conference on Artificial Intelligence (IJCAI 2001), Morgan Kaufmann, 2001, pp. 193–198.
[46] C. H. Papadimitriou, On the complexity of integer programming, Journal of the ACM 28 (4) (1981) 765–768. `doi:10.1145/322276.322287`.

## Appendix A. Full Proofs

**Lemma 4.11.** *For every infinite word $w = w_0 w_1 \ldots \in \Sigma^\omega$, we have $w \in L_\omega(\mathcal{G})$ iff the LTL-structure $\mathfrak{J} := (w_i)_{i \geq 0}$ satisfies $\mathfrak{J}, n \models \widehat{\phi}_{\mathcal{S}}$ and $w_i = X_{\iota(i)}$ for all $i$, $0 \leq i \leq n$.*

Proof. Assume that the LTL-structure $\mathfrak{J} := (w_i)_{i \geq 0}$ is a model of $\widehat{\phi}_{\mathcal{S}}$ at time point $n$ and we have $w_i = X_{\iota(i)}$ for all $i$, $0 \leq i \leq n$. If we define

$$S_i := \{\psi \mid \mathfrak{J}, i \models \psi, \text{ and } \psi \text{ is a sub-literal of } \widehat{\phi}\}$$

for $i \geq 0$, then

$$(S_0, 0)(S_1, 1) \ldots (S_n, n)(S_{n+1}, n+1)(S_{n+2}, n+1) \ldots$$

is a run on $\mathcal{G}$:

- We have $(S_i, k) \in Q$ for all $i \geq 0$ and $k$, $0 \leq k \leq n+1$:
  - For every sub-formula $\psi$ of $\widehat{\phi}$, we have either $\mathfrak{J}, i \models \psi$ or $\mathfrak{J}, i \models \neg\psi$. Thus, we have $\psi \in S_i$ iff $\neg\psi \notin S_i$.
  - For every sub-formula $\psi_1 \wedge \psi_2$ of $\widehat{\phi}$, we have $\mathfrak{J}, i \models \psi_1 \wedge \psi_2$ iff $\mathfrak{J}, i \models \psi_1$ and $\mathfrak{J}, i \models \psi_2$. Thus, we have $\psi_1 \wedge \psi_2 \in S_i$ iff $\{\psi_1, \psi_2\} \subseteq S_i$.
  - For each world $w_i$, $i \geq 0$, we have $w_i \in \mathcal{S}$ since $\mathfrak{J}$ satisfies $\widehat{\phi}_{\mathcal{S}}$. Thus, we have $S_i \cap \{p_1, \ldots, p_m\} = w_i \in \mathcal{S}$ for all $i \geq 0$.
- We have for every sub-formula $\bigcirc^-\psi$ of $\widehat{\phi}$ that $\mathfrak{J}, 0 \not\models \bigcirc^-\psi$, and thus $\bigcirc^-\psi \notin S_0$. Additionally, we have for every $\psi_1 \mathsf{S} \psi_2 \in S_0$, since $\mathfrak{J}, 0 \models \psi_1 \mathsf{S} \psi_2$ also $\mathfrak{J}, 0 \models \psi_2$. This implies that $(S_0, 0) \in Q_0$.
- We have for all $i$, $0 \leq i \leq n$,
$$((S_i, i),\ w_i,\ (S_{i+1}, i+1)) \ \in\ \Delta,$$
and for all $i \geq n+1$,
$$((S_i, n+1),\ w_i,\ (S_{i+1}, n+1)) \ \in\ \Delta,$$
since:
  - we have $w_i = S_i \cap \{p_1, \ldots, p_m\}$ by the definition of $S_i$;
  - for every sub-formula $\bigcirc\psi$ of $\widehat{\phi}$, we have $\bigcirc\psi \in S_i$ iff $\mathfrak{J}, i \models \bigcirc\psi$ iff $\mathfrak{J}, i+1 \models \psi$ iff $\psi \in S_{i+1}$;
  - for every sub-formula $\bigcirc^-\psi$ of $\widehat{\phi}$, we have $\bigcirc^-\psi \in S_{i+1}$ iff $\mathfrak{J}, i+1 \models \bigcirc^-\psi$ iff $\mathfrak{J}, i \models \psi$ iff $\psi \in S_i$;
  - for every sub-formula $\psi_1 \mathsf{U} \psi_2$ of $\widehat{\phi}$, we have $\psi_1 \mathsf{U} \psi_2 \in S_i$ iff $\mathfrak{J}, i \models \psi_1 \mathsf{U} \psi_2$ iff (i) $\mathfrak{J}, i \models \psi_2$ or (ii) $\mathfrak{J}, i \models \psi_1$ and $\mathfrak{J}, i+1 \models \psi_1 \mathsf{U} \psi_2$ iff (i) $\psi_2 \in S_i$ or (ii) $\psi_1 \in S_i$ and $\psi_1 \mathsf{U} \psi_2 \in S_{i+1}$;
  - for every sub-formula $\psi_1 \mathsf{S} \psi_2$ of $\widehat{\phi}$, we have $\psi_1 \mathsf{S} \psi_2 \in S_{i+1}$ iff $\mathfrak{J}, i+1 \models \psi_1 \mathsf{S} \psi_2$ iff (i) $\mathfrak{J}, i+1 \models \psi_2$ or (ii) $\mathfrak{J}, i+1 \models \psi_1$ and $\mathfrak{J}, i \models \psi_1 \mathsf{S} \psi_2$ iff (i) $\psi_2 \in S_{i+1}$ or (ii) $\psi_1 \in S_{i+1}$ and $\psi_1 \mathsf{S} \psi_2 \in S_i$;
  - $i < n+1$ implies $w_i = X_{\iota(i)}$ by assumption;
  - for $i = n$ we have $\mathfrak{J}, n \models \widehat{\phi}_{\mathcal{S}}$, which implies $\mathfrak{J}, n \models \widehat{\phi}$, and thus $\widehat{\phi}_{\mathcal{S}} \in S_n$;
  - the condition for incrementing the second component of a state (until $n+1$ is reached) is obviously also satisfied.

Moreover, the above run is accepting. We prove this by contradiction. Suppose that for some sub-formula $\psi_1 \mathsf{U} \psi_2$ of $\widehat{\phi}$, the set $\{i \geq 0 \mid S_i \in F_{\psi_1 \mathsf{U} \psi_2}\}$ is finite. Then there exists a $k \geq 0$ such that $S_\ell \notin F_{\psi_1 \mathsf{U} \psi_2}$ for all $\ell \geq k$. This means $\psi_1 \mathsf{U} \psi_2 \in S_\ell$ and $\psi_2 \notin S_\ell$ for all $\ell \geq k$. Hence, $\mathfrak{J}, k \models \psi_1 \mathsf{U} \psi_2$ and $\mathfrak{J}, \ell \not\models \psi_2$ for all $\ell \geq k$. This contradicts the semantics of $\mathsf{U}$.

For the converse direction, assume that $w \in L_\omega(\mathcal{G})$, i.e., there is an accepting run

$$(S_0, 0)(S_1, 1) \dots (S_n, n)(S_{n+1}, n+1)(S_{n+2}, n+1) \dots$$

of $\mathcal{G}$ on $w$.

By the definition of $\Delta$, we have $w_i = X_{\iota(i)}$ for all $i$, $0 \leq i \leq n$. To show that $\mathfrak{J} := (w_i)_{i \geq 0}$ is a model of $\widehat{\phi}_{\mathcal{S}}$ at time point $n$, observe that for each $i \geq 0$ we have $w_i = S_i \cap \{p_1, \dots, p_m\} \in \mathcal{S}$ by definition of the state set $Q$. Thus, the conjunct

$$\Box^- \Box \left( \bigvee_{X \in \mathcal{S}} \left( \bigwedge_{p \in X} p \wedge \bigwedge_{p \notin X} \neg p \right) \right)$$

of $\widehat{\phi}_{\mathcal{S}}$ is clearly satisfied by $\mathfrak{J}$ (at any time point).

Furthermore, we have that $\widehat{\phi} \in S_n$ again by the definition of $\Delta$, and thus it is now enough to show that $\psi \in S_i$ iff $\mathfrak{J}, i \models \psi$ for each $i \geq 0$. This can be shown by induction on the structure of $\psi$.

- If $\psi$ is a propositional variable, we have $\psi \in S_i$ iff $\psi \in w_i$ iff $w_i \models \psi$ iff $\mathfrak{J}, i \models \psi$.

- If $\psi = \neg\chi$, we have $\neg\chi \in S_i$ iff $\chi \notin S_i$ iff $\mathfrak{J}, i \not\models \chi$ iff $\mathfrak{J}, i \models \neg\chi$.

- If $\psi = \chi_1 \wedge \chi_2$, we have $\chi_1 \wedge \chi_2 \in S_i$ iff $\{\chi_1, \chi_2\} \subseteq S_i$ iff $\mathfrak{J}, i \models \chi_1$ and $\mathfrak{J}, i \models \chi_2$ iff $\mathfrak{J}, i \models \chi_1 \wedge \chi_2$.

- If $\psi = \bigcirc\chi$, we have $\bigcirc\chi \in S_i$ iff $\chi \in S_{i+1}$ iff $\mathfrak{J}, i+1 \models \chi$ iff $\mathfrak{J}, i \models \bigcirc\chi$.

- If $\psi = \bigcirc^-\chi$, we have $\bigcirc^-\chi \in S_i$ iff $i > 0$ and $\chi \in S_{i-1}$ iff $i > 0$ and $\mathfrak{J}, i-1 \models \chi$ iff $\mathfrak{J}, i \models \bigcirc^-\chi$. The first iff holds because of the definition of $Q_0$.

- If $\psi = \chi_1 \mathsf{U} \chi_2$, we prove $\chi_1 \mathsf{U} \chi_2 \in S_i$ iff $\mathfrak{J}, i \models \chi_1 \mathsf{U} \chi_2$ as follows.

  ($\Longleftarrow$) Assume $\mathfrak{J}, i \models \chi_1 \mathsf{U} \chi_2$. Then there exists a $k \geq i$ such that $\mathfrak{J}, k \models \chi_2$ and $\mathfrak{J}, \ell \models \chi_1$ for all $\ell$, $i \leq \ell < k$. We show by induction on $j$ that $\chi_1 \mathsf{U} \chi_2 \in S_{k-j}$ for $j \leq k-i$.

  For $j = 0$, we have: $\mathfrak{J}, k \models \chi_2$ implies $\chi_2 \in S_k$ by the outer induction hypothesis, and the definition of $\Delta$ yields $\chi_1 \mathsf{U} \chi_2 \in S_k$.

  For $j > 0$, we have: $\mathfrak{J}, k-j \models \chi_1$ implies $\chi_1 \in S_{k-j}$ by the outer induction hypothesis. By the inner induction hypothesis, we have $\chi_1 \mathsf{U} \chi_2 \in S_{k-j+1}$. Thus, by the definition of $\Delta$, it follows that $\chi_1 \mathsf{U} \chi_2 \in S_{k-j}$.

  ($\Longrightarrow$) Assume $\chi_1 \mathsf{U} \chi_2 \in S_i$. Since states of $F_{\chi_1 \mathsf{U} \chi_2}$ occur infinitely often among $S_0, S_1, S_2 \dots$, there is a $k \geq i$ such that $S_k \in F_{\chi_1 \mathsf{U} \chi_2}$. Let $k$ be the smallest index with that property. Then it follows that $\chi_1 \mathsf{U} \chi_2 \in S_\ell$ and $\chi_2 \notin S_\ell$ for all $\ell$, $i \leq \ell < k$.

  $\chi_1 \mathsf{U} \chi_2 \in S_\ell$ and $\chi_2 \notin S_\ell$ for all $\ell$, $i \leq \ell < k$, yield $\chi_1 \in S_\ell$ because of the definition of $\Delta$. Thus, $\mathfrak{J}, \ell \models \chi_1$ for all $\ell$, $i \leq \ell < k$ $(*)$.

  $\chi_1 \mathsf{U} \chi_2 \in S_{k-1}$ and $\chi_2 \notin S_{k-1}$ imply $\chi_1 \mathsf{U} \chi_2 \in S_k$ because of the definition of $\Delta$. This yields $\chi_2 \in S_k$ since $S_k \in F_{\chi_1 \mathsf{U} \chi_2}$, and thus $\mathfrak{J}, k \models \chi_2$ $(**)$.

  $(*)$ and $(**)$ yield that $\mathfrak{J}, i \models \chi_1 \mathsf{U} \chi_2$ by the semantics of $\mathsf{U}$.

- If $\psi = \chi_1 \mathsf{S} \chi_2$, we prove $\chi_1 \mathsf{S} \chi_2 \in S_i$ iff $\mathfrak{J}, i \models \chi_1 \mathsf{S} \chi_2$ as follows.

  ($\Longleftarrow$) Assume $\mathfrak{J}, i \models \chi_1 \mathsf{S} \chi_2$. Then there exists a $k$, $0 \leq k \leq i$ such that $\mathfrak{J}, k \models \chi_2$ and $\mathfrak{J}, \ell \models \chi_1$ for all $\ell$, $k < \ell \leq i$. We show by induction on $j$ that $\chi_1 \mathsf{S} \chi_2 \in S_{k+j}$ for $j \leq i-k$.

  For $j = 0$, we have: $\mathfrak{J}, k \models \chi_2$ implies $\chi_2 \in S_k$ by the outer induction hypothesis, and the definition of $\Delta$ yields $\chi_1 \mathsf{S} \chi_2 \in S_k$.

  For $j > 0$, we have: $\mathfrak{J}, k+j \models \chi_1$ $\chi_1 \in S_{k+j}$ by the outer induction hypothesis. By the inner induction hypothesis, we have $\chi_1 \mathsf{S} \chi_2 \in S_{k+j-1}$. Thus, by the definition of $\Delta$, it follows that $\chi_1 \mathsf{S} \chi_2 \in S_{k+j}$.

  ($\Longrightarrow$) Assume $\chi_1 \mathsf{S} \chi_2 \in S_i$. There are two cases: either $i = 0$ or $i > 0$.

  For $i = 0$, we have: $\chi_1 \mathsf{S} \chi_2 \in S_0$ implies $\chi_2 \in S_0$ by the definition of $Q_0$. This yields $\mathfrak{J}, 0 \models \chi_2$, and thus $\mathfrak{J}, 0 \models \chi_1 \mathsf{S} \chi_2$.

  For $i > 0$, we have again two cases: either $\chi_2 \in S_i$ or $\chi_1 \in S_i$ and $\chi_1 \mathsf{S} \chi_2 \in S_{i-1}$. For the case where $\chi_1 \in S_i$, it directly follows that $\mathfrak{J}, i \models \chi_1 \mathsf{S} \chi_2$. For the other case where $\chi_1 \in S_i$ and $\chi_1 \mathsf{S} \chi_2 \in S_{i-1}$, we have by the inner induction hypothesis: $\mathfrak{J}, i-1 \models \chi_1 \mathsf{S} \chi_2$. Thus, there is a $k$, $0 \leq k \leq i-1$, such that $\mathfrak{J}, k \models \chi_2$ and $\mathfrak{J}, j \models \chi_1$ for all $j$, $k < j \leq i-1$. Since we have by the outer induction hypothesis also that $\mathfrak{J}, i \models \chi_1$, it follows that there is a $k$, $0 \leq k \leq i$, such that $\mathfrak{J}, k \models \chi_2$ and $\mathfrak{J}, j \models \chi_1$ for all $j$, $k < j \leq i$. Hence, $\mathfrak{J}, i \models \chi_1 \mathsf{S} \chi_2$. □

**Lemma 4.14.** *The set $\mathcal{S}$ is r-satisfiable w.r.t. $\iota$ and $\mathcal{K}$ iff $\chi_{\mathcal{S},\iota}$ is satisfiable w.r.t. $\langle \mathcal{T}_{\mathcal{S},\iota}, \mathcal{R}_{\mathcal{S},\iota} \rangle$.*

Proof. Let $\mathcal{J}_1, \dots, \mathcal{J}_k, \mathcal{I}_0, \dots, \mathcal{I}_n$ be the interpretations required by Definition 4.5 for the r-satisfiability of $\mathcal{S}$ w.r.t. $\iota$ and $\mathcal{K}$. We construct the interpretation $\mathcal{J}$ as follows:

- the domain of $\mathcal{J}$ is the shared domain of the above interpretations;

- the rigid names are interpreted as in the above interpretations;
- the $i$-th copy, $1 \leq i \leq k$, of each flexible name is interpreted like the original name in $\mathcal{J}_i$; and
- the $i$-th copy, $k+1 \leq i \leq k+n+1$, of each flexible name is interpreted like the original name in $\mathcal{I}_{i-k-1}$.

It is easy to verify that $\mathcal{J}$ is a model of $\chi_{\mathcal{S},\iota}$, $\mathcal{T}_{\mathcal{S},\iota}$, and $\mathcal{R}_{\mathcal{S},\iota}$.

For the other direction, let $\mathcal{J}$ be a model of $\chi_{\mathcal{S},\iota}$ w.r.t. $\langle \mathcal{T}_{\mathcal{S},\iota}, \mathcal{R}_{\mathcal{S},\iota}\rangle$. We obtain the interpretations $\mathcal{J}_1, \dots, \mathcal{J}_k$, $\mathcal{I}_0, \dots, \mathcal{I}_n$ by the inverse construction to the one above:

- the domain of all these interpretations is the domain of $\mathcal{J}$;
- the rigid names are interpreted by these interpretations as in $\mathcal{J}$;
- every flexible name is interpreted in $\mathcal{J}_i$, $1 \leq i \leq k$, as its $i$-th copy is interpreted in $\mathcal{J}$; and
- every flexible name is interpreted in $\mathcal{I}_i$, $0 \leq i \leq n$, as it $k+i+1$-st copy is interpreted in $\mathcal{J}$.

Again, it is easy to verify that these interpretations satisfy the conditions of Definition 4.5. □

**Lemma 5.1.** *If $N_{RC} \neq \emptyset$ and $N_{RR} = \emptyset$, then $\mathcal{S}$ is r-satisfiable w.r.t. $\iota$ and $\mathcal{K}$ iff there exist $\mathcal{D} \subseteq 2^{\mathsf{RCon}(\mathcal{T})}$ and $\tau\colon \mathsf{Ind}(\phi) \cup \mathsf{Ind}(\mathcal{K}) \to \mathcal{D}$ such that each $\gamma_i \wedge \chi_{\mathcal{S}} \wedge \rho_\tau$, $0 \leq i \leq n$, has a model w.r.t. $\langle \mathcal{T}_{\mathcal{S}} \cup \mathcal{T}_\tau, \mathcal{R}_{\mathcal{S}}\rangle$ that respects $\mathcal{D}$.*

Proof. For the "if" direction, assume that $\mathcal{I}_i$, $0 \leq i \leq n$, are the required models for $\gamma_i \wedge \chi_{\mathcal{S}} \wedge \rho_\tau$ w.r.t. $\langle \mathcal{T}_{\mathcal{S}} \cup \mathcal{T}_\tau, \mathcal{R}_{\mathcal{S}}\rangle$. Similar to the proof of Lemma 6.3 in [12], we can assume w.l.o.g. that their domains $\Delta_i$ are countably infinite and for each $Y \in \mathcal{D}$ there are countably infinitely many elements $d \in (C_Y)^{\mathcal{I}_i}$. This is a consequence of the Löwenheim-Skolem theorem and the fact that the countably infinite disjoint union of $\mathcal{I}_i$ with itself is again a model of $\gamma_i \wedge \chi_{\mathcal{S}} \wedge \rho_\tau$ and $\langle \mathcal{T}_{\mathcal{S}} \cup \mathcal{T}_\tau, \mathcal{R}_{\mathcal{S}}\rangle$. The latter follows from the observation that for any CQ there is a homomorphism into $\mathcal{I}_i$ iff there is a homomorphism into the disjoint union of $\mathcal{I}_i$ with itself. One direction is trivial, while whenever there is a homomorphism into the disjoint union, we can construct a homomorphism into $\mathcal{I}_i$ by replacing the elements in the image of this homomorphism by the corresponding elements of $\Delta_i$. It is easy to see that the resulting homomorphism still satisfies all atoms of the CQ.

Consequently, we can partition the domains $\Delta_i$ into the countably infinite sets $\Delta_i(Y) := \{d \in \Delta_i \mid d \in (C_Y)^{\mathcal{I}_i}\}$ for $Y \in \mathcal{D}$. By the assumptions above and the fact that all $\mathcal{I}_i$ satisfy $\rho_\tau$ and $\mathcal{T}_\tau$, there are bijections $\pi_i\colon \Delta_0 \to \Delta_i$, $1 \leq i \leq n$, such that

- $\pi_i(\Delta_0(Y)) = \Delta_i(Y)$ for all $Y \in \mathcal{D}$ and
- $\pi_i(a^{\mathcal{I}_0}) = a^{\mathcal{I}_i}$ for all $a \in \mathsf{Ind}(\phi) \cup \mathsf{Ind}(\mathcal{K})$.

Thus, we can assume in the following that the models $\mathcal{I}_i$ actually share the same domain and interpret the rigid names in $\mathsf{RCon}(\mathcal{T})$ and $\mathsf{Ind}(\phi) \cup \mathsf{Ind}(\mathcal{K})$ in the same way. We can now construct the models required by Definition 4.5 by appropriately relating the flexible names and their copies. For example, interpreting the rigid concept names as in $\mathcal{I}_i$ and the flexible names as their $\iota(i)$-th copies in $\mathcal{I}_i$ yields a model of $\chi_{\iota(i)}$ w.r.t. $\langle \mathcal{A}_i, \mathcal{T}, \mathcal{R}\rangle$, and similarly for the models of $\chi_j$ w.r.t. $\langle \mathcal{T}, \mathcal{R}\rangle$ for $1 \leq j \leq k$. These models share the same domain and respect the rigid names in $\mathsf{RCon}(\mathcal{T})$ and $\mathsf{Ind}(\phi) \cup \mathsf{Ind}(\mathcal{K})$. Note that the interpretation of the names in $N_{\mathrm{RC}}$ and $N_{\mathrm{I}}$ that occur neither in $\mathcal{K}$ nor in $\phi$ is irrelevant and can be fixed arbitrarily, as long as the UNA is satisfied.

Thus, it remains to consider those rigid concept names $A$ occurring in $(\mathcal{A}_i)_{0 \leq i \leq n}$, but not in $\mathcal{T}$. Since they are not constrained by the TBox, it suffices to interpret them in such a way that they satisfy all ABox assertions. But since these assertions can only occur positively in the ABoxes, the set $\{a^{\mathcal{I}_0} \mid A(a) \in \mathcal{A}_i,\ 0 \leq i \leq n\}$ fulfills this restriction.

For the "only if" direction, it is easy to see that one can combine the interpretations $\mathcal{I}_i$, $\mathcal{J}_1, \dots, \mathcal{J}_k$ from Definition 4.5 to a model $\mathcal{I}'_i$ of $\gamma_i \wedge \chi_{\mathcal{S}}$ w.r.t. $\langle \mathcal{T}_{\mathcal{S}}, \mathcal{R}_{\mathcal{S}}\rangle$ by interpreting the $\iota(i)$-th copy of a flexible name as the original name in $\mathcal{I}_i$, and the $j$-th copy of a flexible name as the original name in $\mathcal{J}_j$, for each $j$, $1 \leq j \leq k$, with $j \neq \iota(i)$. Obviously, the interpretations $\mathcal{I}'_i$ share the same domain, interpret individual names in the same way, and respect rigid concept names.

For $a \in \mathsf{Ind}(\phi) \cup \mathsf{Ind}(\mathcal{K})$, we define $\tau(a) := Y \subseteq \mathsf{RCon}(\mathcal{T})$ iff $a \in (C_Y)^{\mathcal{I}_0}$, which ensures that the interpretations $\mathcal{I}'_i$ can be extended to models of $\rho_\tau$ and $\mathcal{T}_\tau$ by appropriately interpreting the new concept names $T_{\tau(a)}$. Furthermore, we let $\mathcal{D}$ contain all those sets $Y \subseteq \mathsf{RCon}(\mathcal{T})$ such that there is a $d \in (C_Y)^{\mathcal{I}'_i}$ for some $0 \leq i \leq n$. Since we have $(C_Y)^{\mathcal{I}'_i} = (C_Y)^{\mathcal{I}'_j}$ for all $0 \leq i,j \leq n$ and all $Y \in \mathcal{D}$, the interpretations $\mathcal{I}'_i$ respect $\mathcal{D}$. Hence, we obtain models of $\gamma_i \wedge \chi_{\mathcal{S}} \wedge \rho_\tau$ w.r.t. $\langle \mathcal{T}_{\mathcal{S}} \cup \mathcal{T}_\tau, \mathcal{R}_{\mathcal{S}}\rangle$ that respect $\mathcal{D}$. □

**Lemma 6.2.** *If $N_{RC} \neq \emptyset$ and $N_{RR} = \emptyset$, then $\mathcal{S}$ is r-satisfiable w.r.t. $\mathcal{K} = \langle \emptyset, \mathcal{T}, \mathcal{R}\rangle$ iff there exist $\mathcal{D} \subseteq 2^{\mathsf{RCon}(\mathcal{T})}$ and $\tau\colon \mathsf{Ind}(\phi) \to \mathcal{D}$ such that each $\widehat{\chi}_i := \chi_i \wedge \rho_\tau$, $1 \leq i \leq k$, has a model w.r.t. $\langle \mathcal{T} \cup \mathcal{T}_\tau, \mathcal{R}\rangle$ that respects $\mathcal{D}$.*

Proof. By Lemma 5.1, $\mathcal{S}$ is r-satisfiable w.r.t. $\mathcal{K}$ iff there exist $\mathcal{D} \subseteq 2^{\mathsf{RCon}(\mathcal{T})}$ and $\tau\colon \mathsf{Ind}(\phi) \to \mathcal{D}$ such that $\chi_{\mathcal{S}} \wedge \rho_\tau$ has a model w.r.t. $\langle \mathcal{T}_{\mathcal{S}} \cup \mathcal{T}_\tau, \mathcal{R}_{\mathcal{S}}\rangle$ that respects $\mathcal{D}$.

For the "if" direction, let $\mathcal{D} \subseteq 2^{\mathsf{RCon}(\mathcal{T})}$, $\tau\colon \mathsf{Ind}(\phi) \to \mathcal{D}$, and $\mathcal{I}_i$ be models of $\chi_i \wedge \rho_\tau$ and $\langle \mathcal{T} \cup \mathcal{T}_\tau, \mathcal{R}\rangle$ that respect $\mathcal{D}$. As in the proof of Lemma 5.1, we can ensure that they share the same domain and interpret the rigid names in $\mathsf{RCon}(\mathcal{T})$ and $\mathsf{Ind}(\phi)$ in the same way. Similar to before, we can construct a model $\mathcal{J}$ of $\chi_{\mathcal{S}} \wedge \rho_\tau$ and $\langle \mathcal{T}_{\mathcal{S}} \cup \mathcal{T}_\tau, \mathcal{R}_{\mathcal{S}}\rangle$ over the shared domain of $\mathcal{I}_1, \dots \mathcal{I}_k$ as follows: interpret the $i$-th copy of a flexible name as the original name in $\mathcal{I}_i$, and every rigid name as in $\mathcal{I}_1$. Since the interpretations of the names in $\mathsf{RCon}(\mathcal{T})$ are not changed, $\mathcal{J}$ also respects $\mathcal{D}$.

For the "only if" direction, let $\mathcal{J}$ be a model of $\chi_\mathcal{S} \wedge \rho_\mathcal{T}$ and $\langle \mathcal{T}_\mathcal{S} \cup \mathcal{T}_\tau, \mathcal{R}_\mathcal{S} \rangle$ that respects $\mathcal{D}$. As before, a model $\mathcal{I}_i$ of $\chi_i \wedge \rho_\mathcal{T}$ and $\langle \mathcal{T} \cup \mathcal{T}_\tau, \mathcal{R} \rangle$ can be constructed by interpreting the rigid names as in $\mathcal{J}$ and the flexible names as their $i$-th copies in $\mathcal{J}$. Again, these models still respect $\mathcal{D}$. $\square$

**Lemma 6.5.** *Let $\mathcal{B}$ be a Boolean $\mathcal{SHQ}^\cap$-knowledge base, let $A_1, \ldots, A_k$ be concept names occurring in $\mathcal{B}$, and let $\mathcal{D} \subseteq 2^{\{A_1,\ldots,A_k\}}$. Then $\mathcal{B}$ has a model that respects $\mathcal{D}$ iff it has a forest model that respects $\mathcal{D}$.*

Proof. The "if" direction is trivial. For the "only if" direction, assume that $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$ is a model of $\mathcal{B} = \langle \Phi, \mathcal{R} \rangle$ that respects $\mathcal{D}$. Moreover, we assume that $\Delta^\mathcal{I}$ is countable, which is w.l.o.g. due to the downward Löwenheim-Skolem theorem. We can thus assume that $\Delta^\mathcal{I} \subseteq \mathbb{N}$. We define now a forest base $\mathcal{J} = (\Delta^\mathcal{J}, \cdot^\mathcal{J})$ for $\mathcal{B}$ with domain

$$\begin{aligned}\Delta^\mathcal{J} := \big\{(a, d_1 \ldots d_m) \mid {} & a \in \mathsf{Ind}(\Psi),\ m \geq 0,\\ & d_1, \ldots, d_m \in \Delta^\mathcal{I}, \text{ there is no}\\ & b \in \mathsf{Ind}(\Psi) \text{ such that } d_1 = b^\mathcal{I}\big\}\end{aligned}$$

as follows:

- $a^\mathcal{J} := (a, \varepsilon)$ for all $a \in \mathsf{Ind}(\Psi)$;
- $b^\mathcal{J}$ for $b \in N_\mathrm{I} \setminus \mathsf{Ind}(\Psi)$ can be fixed arbitrarily, as long as the UNA is satisfied;
- $A^\mathcal{J} := \{(a, \varepsilon) \mid a^\mathcal{I} \in A^\mathcal{I}\} \cup \{(a, d_1 \ldots d_m) \mid d_m \in A^\mathcal{I}\}$; and
- $$\begin{aligned} r^\mathcal{J} := {} & \{((a,\varepsilon),(b,\varepsilon)) \mid (a^\mathcal{I}, b^\mathcal{I}) \in r^\mathcal{I}\} \cup \\ & \{((a,\varepsilon),(a,d)) \mid (a^\mathcal{I}, d) \in r^\mathcal{I}\} \cup \\ & \{((a, d_1 \ldots d_m), (a, d_1 \ldots d_m d_{m+1})) \mid \\ & \qquad m > 0,\ (d_m, d_{m+1}) \in r^\mathcal{I}\}.\end{aligned}$$

Obviously, $\mathcal{J}$ satisfies the conditions for a forest base for $\mathcal{B}$. We construct now a forest model $\widehat{\mathcal{J}} = (\Delta^{\widehat{\mathcal{J}}}, \cdot^{\widehat{\mathcal{J}}})$ for $\mathcal{B}$. For that, we define $\Delta^{\widehat{\mathcal{J}}} := \Delta^\mathcal{J}$, for each $A \in N_\mathrm{C}$, $A^{\widehat{\mathcal{J}}} := A^\mathcal{J}$, for each $a \in N_\mathrm{I}$, $a^{\widehat{\mathcal{J}}} := a^\mathcal{J}$, and for each $r \in N_\mathrm{R}$:

$$r^{\widehat{\mathcal{J}}} := r^\mathcal{J} \cup \bigcup_{\mathcal{R} \models s \sqsubseteq r,\ \mathcal{R} \models \mathsf{trans}(s)} (s^\mathcal{J})^+.$$

To prove that this indeed defines a forest model, we first show the following claim by structural induction.

**Claim 1.** For every $(a, d_1 \ldots d_m) \in \Delta^{\widehat{\mathcal{J}}}$ and concept $C$, we have $(a, d_1 \ldots d_m) \in C^{\widehat{\mathcal{J}}}$ iff either $m = 0$ and $a^\mathcal{I} \in C^\mathcal{I}$, or $d_m \in C^\mathcal{I}$.

For the base case, $C$ being a concept name, the claim is directly implied by the definition.

For the case where $C$ is of the form $\neg D$, we have

$(a, d_1 \ldots d_m) \in (\neg D)^{\widehat{\mathcal{J}}}$

*iff* $(a, d_1 \ldots d_m) \notin D^{\widehat{\mathcal{J}}}$

*iff* either $m = 0$ and $a^\mathcal{I} \notin D^\mathcal{I}$, or $d_m \notin D^\mathcal{I}$

*iff* either $m = 0$ and $a^\mathcal{I} \in (\neg D)^\mathcal{I}$, or $d_m \in (\neg D)^\mathcal{I}$.

For the case where $C$ is of the form $D \sqcap E$, we have

$(a, d_1 \ldots d_m) \in (D \sqcap E)^{\widehat{\mathcal{J}}}$

*iff* $(a, d_1 \ldots d_m) \in D^{\widehat{\mathcal{J}}}$ and $(a, d_1 \ldots d_m) \in E^{\widehat{\mathcal{J}}}$

*iff* either $m = 0$ and $a^\mathcal{I} \in D^\mathcal{I}$ and $a^\mathcal{I} \in E^\mathcal{I}$, or $d_m \in D^\mathcal{I}$ and $d_m \in E^\mathcal{I}$

*iff* either $m = 0$ and $a^\mathcal{I} \in (D \sqcap E)^\mathcal{I}$, or $d_m \in (D \sqcap E)^\mathcal{I}$.

For the case where $C$ is of the form $\exists(r_1 \cap \cdots \cap r_\ell).D$ with $\ell > 1$, we have $r_1^{\widehat{\mathcal{J}}} \cap \cdots \cap r_\ell^{\widehat{\mathcal{J}}} = r_1^\mathcal{J} \cap \cdots \cap r_\ell^\mathcal{J}$ since $r_1, \ldots, r_\ell$ are simple role names, and thus

$(a, d_1 \ldots d_m) \in (\exists(r_1 \cap \cdots \cap r_\ell).D)^{\widehat{\mathcal{J}}}$

*iff* either $m = 0$ and

- there is a $(b, \varepsilon) \in D^{\widehat{\mathcal{J}}}$ such that $((a,\varepsilon),(b,\varepsilon))$ is in $r_1^\mathcal{J} \cap \cdots \cap r_\ell^\mathcal{J}$, or
- there is a $(a, d) \in D^{\widehat{\mathcal{J}}}$ such that $((a,\varepsilon),(a,d))$ is in $r_1^\mathcal{J} \cap \cdots \cap r_\ell^\mathcal{J}$;

or there is a $(a, d_1 \ldots d_m d_{m+1}) \in D^{\widehat{\mathcal{J}}}$ such that $((a, d_1 \ldots d_m), (a, d_1 \ldots d_m d_{m+1}))$ is in $r_1^\mathcal{J} \cap \cdots \cap r_\ell^\mathcal{J}$

*iff* either $m = 0$ and there is a $d \in D^\mathcal{I}$ such that $(a^\mathcal{I}, d) \in r_1^\mathcal{I} \cap \cdots \cap r_\ell^\mathcal{I}$, or there is a $d \in D^\mathcal{I}$ such that $(d_m, d) \in r_1^\mathcal{I} \cap \cdots \cap r_\ell^\mathcal{I}$

*iff* either $m = 0$ and $a^\mathcal{I} \in (\exists(r_1 \cap \cdots \cap r_\ell).D)^\mathcal{I}$, or $d_m \in (\exists(r_1 \cap \cdots \cap r_\ell).D)^\mathcal{I}$.

For the case where $C$ is of the form $\exists r.D$, we have

$(a, d_1 \ldots d_m) \in (\exists r.D)^{\widehat{\mathcal{J}}}$

*iff* there is $x \in D^{\widehat{\mathcal{J}}}$ with either $((a, d_1 \ldots d_m), x) \in r^\mathcal{J}$ or there is a role name $s$ with $\mathcal{R} \models s \sqsubseteq r$, $\mathcal{R} \models \mathsf{trans}(s)$, and $((a, d_1 \ldots d_m), x) \in (s^\mathcal{J})^+$

*iff* either $m = 0$ and

- there is a $(b, \varepsilon) \in D^{\widehat{\mathcal{J}}}$ with $((a,\varepsilon),(b,\varepsilon)) \in r^\mathcal{J}$,
- there is a $(a, d) \in D^{\widehat{\mathcal{J}}}$ with $((a,\varepsilon),(a,d)) \in r^\mathcal{J}$, or
- there is a role name $s$ with $\mathcal{I} \models s \sqsubseteq r$ and $\mathcal{I} \models \mathsf{trans}(s)$, and a sequence $(a_0, \varepsilon)$, $(a_1, \varepsilon)$, …, $(a_n, \varepsilon)$, $(a_n, e_1)$, …, $(a_n, e_1 \ldots e_k)$ of elements of $\Delta^{\widehat{\mathcal{J}}}$ such that $a_0 = a$, $(a_n, e_1 \ldots e_k) \in D^{\widehat{\mathcal{J}}}$, and each two consecutive elements of this sequence are connected via $s^\mathcal{J}$;

or there is a sequence $(a, d_1 \dots d_m)$, $(a, d_1 \dots d_{m+1})$, $\dots, (a, d_1 \dots d_{m+n})$ of elements of $\Delta^{\widehat{\mathcal{J}}}$ such that $n \geq 1$, $(a, d_1 \dots d_{m+n}) \in D^{\widehat{\mathcal{J}}}$, and each two consecutive elements of this sequence are connected via $s^{\mathcal{J}}$, where $s$ is a role name such that either $n = 1$ and $s = r$, or $\mathcal{I} \models s \sqsubseteq r$ and $\mathcal{I} \models \mathsf{trans}(s)$,

*iff* either $m = 0$ and

- there is a $d \in D^{\mathcal{I}}$ such that $(a^{\mathcal{I}}, d) \in r^{\mathcal{I}}$, or
- there is $s \in N_{\mathrm{R}}$ with $\mathcal{I} \models s \sqsubseteq r$ and $\mathcal{I} \models \mathsf{trans}(s)$, and an $e_k \in \Delta^{\mathcal{I}}$ such that $(a^{\mathcal{I}}, e_k) \in s^{\mathcal{I}} \subseteq r^{\mathcal{I}}$ and $e_k \in D^{\mathcal{I}}$;

or there is a $d \in D^{\mathcal{I}}$ such that $(d_m, d) \in s^{\mathcal{I}} \subseteq r^{\mathcal{I}}$, where $s$ is a role name such that either $s = r$, or $\mathcal{I} \models s \sqsubseteq r$ and $\mathcal{I} \models \mathsf{trans}(s)$,

*iff* either $m = 0$ and $a^{\mathcal{I}} \in (\exists r.D)^{\mathcal{I}}$, or $d_m \in (\exists r.D)^{\mathcal{I}}$.

For the case where $C$ is of the form $\geq n\, r.D$ for a simple role name $r$, we again have $r^{\widehat{\mathcal{J}}} = r^{\mathcal{J}}$, and thus

$(a, d_1 \dots d_m) \in (\geq n\, r.D)^{\widehat{\mathcal{J}}}$

*iff* there is a subset $X \subseteq D^{\widehat{\mathcal{J}}}$ with $|X| = n$ such that $((a, \varepsilon), x) \in r^{\mathcal{J}}$ for each $x \in X$, and either

- $m = 0$ and each $x \in X$ is either of the form $(b, \varepsilon)$ or $(a, d)$, or
- each $x \in X$ is of the form $(a, d_1 \dots d_m d_{m+1})$

*iff* there is a subset $Y \subseteq D^{\mathcal{I}}$ with $|Y| = n$ such that $m = 0$ and $(a^{\mathcal{I}}, y) \in r^{\mathcal{I}}$ for each $y \in Y$, or $(d_m, y) \in r^{\mathcal{I}}$ for each $y \in Y$

*iff* either $m = 0$ and $a^{\mathcal{I}} \in (\geq n\, r.D)^{\mathcal{I}}$, or $d_m \in (\geq n\, r.D)^{\mathcal{I}}$.

The second equivalence holds since each $r^{\mathcal{I}}$-successor of a named individual $a^{\mathcal{I}}$ is represented by exactly one $r^{\widehat{\mathcal{J}}}$-successor of $(a, \varepsilon)$ since domain elements of the form $(a, b^{\mathcal{I}})$ for $b \in \mathsf{Ind}(\Psi)$ are not allowed. This finishes the proof of Claim 1.

It remains only to show that $\widehat{\mathcal{J}}$ is indeed a model of $\mathcal{B}$. For this, we prove first the following claim by structural induction.

**Claim 2.** For all $\psi \in \mathsf{Sub}(\Psi)$, we have $\widehat{\mathcal{J}} \models \psi$ iff $\mathcal{I} \models \psi$.

For the first base case, assume that $\psi$ is of the form $A(a)$ for some $A \in N_{\mathrm{C}}$ and $a \in N_{\mathrm{I}}$. We have $a^{\mathcal{I}} \in A^{\mathcal{I}}$ iff $a^{\widehat{\mathcal{J}}} = a^{\mathcal{J}} = (a, \varepsilon) \in A^{\mathcal{J}} = A^{\widehat{\mathcal{J}}}$ by definition.

For the second base case, assume that $\psi$ is of the form $r(a, b)$ for $a, b \in N_{\mathrm{I}}$ and $r \in N_{\mathrm{R}}$. If $\mathcal{I} \models r(a, b)$, then $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$, and thus

$$(a^{\widehat{\mathcal{J}}}, b^{\widehat{\mathcal{J}}}) = (a^{\mathcal{J}}, b^{\mathcal{J}}) = ((a, \varepsilon), (b, \varepsilon)) \in r^{\mathcal{J}} \subseteq r^{\widehat{\mathcal{J}}}.$$

Conversely, if $((a, \varepsilon), (b, \varepsilon)) \in r^{\widehat{\mathcal{J}}}$, then there is a role name $s$ and a sequence $(a_0, \varepsilon), \dots, (a_n, \varepsilon)$, $n \geq 1$, of elements of $\Delta^{\widehat{\mathcal{J}}}$ such that $a_0 = a$, $a_n = b$, each two consecutive elements of this sequence are connected via $s^{\mathcal{J}}$, and either $n = 1$ and $s = r$, or $\mathcal{R} \models s \sqsubseteq r$ and $\mathcal{R} \models \mathsf{trans}(s)$. By definition of $s^{\mathcal{J}}$, the properties of $s$, and since $\mathcal{I} \models \mathcal{R}$, we can infer that $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$, and thus $\mathcal{I} \models r(a, b)$.

For the third base case, assume that $\psi$ is of the form $C \sqsubseteq D$. For the "if" direction, assume that $\mathcal{I} \models C \sqsubseteq D$ and thus $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$. Suppose that there is a $(a, d_1 \dots d_m) \in C^{\widehat{\mathcal{J}}}$ with $(a, d_1 \dots d_m) \notin D^{\widehat{\mathcal{J}}}$. By Claim 1, either $m = 0$ and we have $a^{\mathcal{I}} \in C^{\mathcal{I}}$ and $a^{\mathcal{I}} \notin D^{\mathcal{I}}$, or $d_m \in C^{\mathcal{I}}$ and $d_m \notin D^{\mathcal{I}}$, which contradicts our assumption that $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$.

For the "only if" direction, assume that $C^{\widehat{\mathcal{J}}} \subseteq D^{\widehat{\mathcal{J}}}$. Suppose that there is a $d \in C^{\mathcal{I}}$ with $d \notin D^{\mathcal{I}}$. By the definition of $\Delta^{\widehat{\mathcal{J}}}$, we have $(a, d'd) \in \Delta^{\widehat{\mathcal{J}}}$ for any $a \in \mathsf{Ind}(\Psi)$ and $d' \in \Delta^{\mathcal{I}}$ such that there is no $b \in \mathsf{Ind}(\Psi)$ with $d' = b^{\mathcal{I}}$. By Claim 1, we get $(a, d'd) \in C^{\widehat{\mathcal{J}}}$ and $(a, d'd) \notin D^{\widehat{\mathcal{J}}}$, which again yields a contradiction.

For the induction step, assume first that $\psi$ is of the form $\neg\psi'$. We have that $\widehat{\mathcal{J}} \models \neg\psi'$ iff $\widehat{\mathcal{J}} \not\models \psi'$ iff $\mathcal{I} \not\models \psi'$ iff $\mathcal{I} \models \neg\psi'$. Assume now that $\psi$ is of the form $\psi_1 \wedge \psi_2$. We have that $\widehat{\mathcal{J}} \models \psi_1 \wedge \psi_2$ iff $\widehat{\mathcal{J}} \models \psi_1$ and $\widehat{\mathcal{J}} \models \psi_2$ iff $\mathcal{I} \models \psi_1$ and $\mathcal{I} \models \psi_2$ iff $\mathcal{I} \models \psi_1 \wedge \psi_2$.

This finishes the proof of the claim. Since $\Psi \in \mathsf{Sub}(\Psi)$, this shows that $\widehat{\mathcal{J}}$ is indeed a model of $\Psi$. We show that $\widehat{\mathcal{J}}$ is also a model of $\mathcal{R}$ in the following claim.

**Claim 3.** For all $\alpha \in \mathcal{R}$, we have $\widehat{\mathcal{J}} \models \alpha$.

Assume first that $\alpha$ is of the form $r \sqsubseteq s$. Since $\mathcal{I} \models \mathcal{R}$, we have $\mathcal{I} \models r \sqsubseteq s$ and thus $r^{\mathcal{I}} \subseteq s^{\mathcal{I}}$. We first show that $r^{\mathcal{J}} \subseteq s^{\mathcal{J}}$. For this, take $(x, y) \in r^{\mathcal{J}}$. There are three cases to consider:

- If $x = (a, \varepsilon)$ and $y = (b, \varepsilon)$ with $a, b \in \mathsf{Ind}(\Psi)$, we have $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$ and thus $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in s^{\mathcal{I}}$. Hence, the definition of $s^{\mathcal{J}}$ yields that $(x, y) \in s^{\mathcal{J}}$.
- If $x = (a, \varepsilon)$ and $y = (a, d)$ with $a \in \mathsf{Ind}(\Psi), d \in \Delta^{\mathcal{I}}$, we have $(a^{\mathcal{I}}, d) \in r^{\mathcal{I}}$ and thus $(a^{\mathcal{I}}, d) \in s^{\mathcal{I}}$. Again, the definition of $s^{\mathcal{J}}$ yields that $(x, y) \in s^{\mathcal{J}}$.
- If $x = (a, d_1 \dots d_m)$ and $y = (a, d_1 \dots d_m d_{m+1})$ with $a \in \mathsf{Ind}(\Psi)$, $m > 0$, $d_1, \dots, d_{m+1} \in \Delta^{\mathcal{I}}$, we have $(d_m, d_{m+1}) \in r^{\mathcal{I}}$ and thus $(d_m, d_{m+1}) \in s^{\mathcal{I}}$. Again, the definition of $s^{\mathcal{J}}$ yields that $(x, y) \in s^{\mathcal{J}}$.

To show that $r^{\widehat{\mathcal{J}}} \subseteq s^{\widehat{\mathcal{J}}}$, take $(x, y) \in r^{\widehat{\mathcal{J}}}$. If $(x, y) \in r^{\mathcal{J}}$, we have $(x, y) \in s^{\mathcal{J}}$ and thus $(x, y) \in s^{\widehat{\mathcal{J}}}$. Otherwise, we have that $(x, y) \in (t^{\mathcal{J}})^+$ with $\mathcal{R} \models t \sqsubseteq r$ and $\mathcal{R} \models \mathsf{trans}(t)$. Since $r \sqsubseteq s \in \mathcal{R}$, we have obviously $\mathcal{R} \models r \sqsubseteq s$. It is easy to see that this implies $\mathcal{R} \models t \sqsubseteq s$. Then the definition of $s^{\widehat{\mathcal{J}}}$ yields that $(t^{\mathcal{J}})^+ \subseteq s^{\widehat{\mathcal{J}}}$. Hence $(x, y) \in s^{\widehat{\mathcal{J}}}$.

Assume now that $\psi$ is of the form $\mathsf{trans}(r)$. Since $\mathcal{I} \models \mathcal{R}$, we have $\mathcal{I} \models \mathsf{trans}(r)$ and thus $r^{\mathcal{I}} \circ r^{\mathcal{I}} \subseteq r^{\mathcal{I}}$. By the same arguments as above, we have that for each $t$ with $t^{\mathcal{I}} \subseteq r^{\mathcal{I}}$, we have $t^{\mathcal{J}} \subseteq r^{\mathcal{J}}$, and thus $(t^{\mathcal{J}})^+ \subseteq (r^{\mathcal{J}})^+$ since the transitive closure is monotonic. Since $r^{\mathcal{I}} \subseteq r^{\mathcal{I}}$, we have also $\mathcal{I} \models r \sqsubseteq r$. The definition of $r^{\widehat{\mathcal{J}}}$ yields now that $r^{\widehat{\mathcal{J}}} = (r^{\mathcal{J}})^+$, and hence $\widehat{\mathcal{J}}$ is a model of $\mathsf{trans}(r)$.

Claim 2 and Claim 3 yield that $\widehat{\mathcal{I}}$ is indeed a model of $\mathcal{B}$. It only remains to be shown that $\widehat{\mathcal{J}}$ respects $\mathcal{D}$. Since $\mathcal{I}$ respects $\mathcal{D}$, we have

$$\mathcal{D} = \{Y \subseteq \{A_1, \ldots, A_k\} \mid \exists d \in \Delta^{\mathcal{I}} \text{ with } d \in (C_Y)^{\mathcal{I}}\}.$$

We now define

$$\mathcal{D}' := \{Y \subseteq \{A_1, \ldots, A_k\} \mid \exists x \in \Delta^{\widehat{\mathcal{J}}} \text{ with } x \in (C_Y)^{\widehat{\mathcal{J}}}\}.$$

and show that $\mathcal{D} = \mathcal{D}'$. Since $\widehat{\mathcal{J}}$ respects $\mathcal{D}'$, this implies that $\widehat{\mathcal{J}}$ respects $\mathcal{D}$.

For the direction ($\subseteq$), assume that $Y \in \mathcal{D}$, and thus there is a $d \in (C_Y)^{\mathcal{I}}$. By Claim 1 and the definition of $\Delta^{\widehat{\mathcal{J}}}$, there is a $(a, d'd) \in (C_Y)^{\widehat{\mathcal{J}}}$, and hence $Y \in \mathcal{D}'$. For the direction ($\supseteq$), assume that $Y \in \mathcal{D}'$, i.e., there is a $(a, d_1 \ldots d_m) \in (C_Y)^{\widehat{\mathcal{J}}}$. By Claim 1 and the definition of $\Delta^{\widehat{\mathcal{J}}}$, there is a $d \in (C_Y)^{\mathcal{I}}$, where for $m = 0$, we can set $d := a^{\mathcal{I}}$, and for $m > 0$, we can take $d := d_m$. Hence, $Y \in \mathcal{D}$. □

**Lemma 6.6.** *We have $\langle \mathcal{A}, \mathcal{T}, \mathcal{R}\rangle \not\models \rho$ w.r.t. $\mathcal{D}$ iff there is a forest model $\mathcal{J}$ of $\langle \mathcal{A}, \mathcal{T}, \mathcal{R}\rangle$ that respects $\mathcal{D}$ with $\mathcal{J} \not\models \rho$.*

Proof. The "if" direction is trivial. For the "only if" direction, assume that there is a model $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ of $\langle \mathcal{A}, \mathcal{T}, \mathcal{R}\rangle$ that respects $\mathcal{D}$ such that $\mathcal{I} \not\models \rho$. As shown in the proof of Lemma 6.5, $\mathcal{I}$ can be transformed into a forest model $\widehat{\mathcal{J}} = (\Delta^{\widehat{\mathcal{J}}}, \cdot^{\widehat{\mathcal{J}}})$ that respects $\mathcal{D}$. Assume that $\mathcal{J}, \widehat{\mathcal{J}}$ are obtained from $\mathcal{I}$ as in the proof of Lemma 6.5. It is left to show that then $\widehat{\mathcal{J}} \not\models \rho$.

Assume to the contrary that $\widehat{\mathcal{J}} \models \rho$. Then there is a Boolean CQ $\rho_i$ in the UCQ $\rho$ such that there is a homomorphism $\pi$ from $\rho_i$ into $\widehat{\mathcal{J}}$. We define a homomorphism $\pi'$ from $\rho_i$ into $\mathcal{I}$ as follows: $\pi'(a) := a^{\mathcal{I}}$ for all individual names $a$ occurring in the input; and for all $v \in \mathsf{Var}(\rho_i)$, we define $\pi'(v) := a^{\mathcal{I}}$ if $\pi(v) = (a, \varepsilon)$ for $a \in \mathsf{Ind}(\mathcal{A})$, and $\pi'(v) = d_m$ if $\pi(v) = (a, d_1 \ldots d_m)$ with $m > 0$. We now show that $\pi'$ is indeed a homomorphism from $\rho_i$ into $\mathcal{I}$.

Consider first a concept atom $A(a) \in \mathsf{At}(\rho_i)$. Since $(a, \varepsilon) = a^{\widehat{\mathcal{J}}} \in A^{\widehat{\mathcal{J}}}$, we get $a^{\mathcal{I}} \in A^{\mathcal{I}}$ by Claim 1.

For an atom $A(v) \in \mathsf{At}(\rho_i)$ with $v \in \mathsf{Var}(\rho_i)$, we get $\pi(v) \in A^{\widehat{\mathcal{J}}}$, and thus $\pi'(v) \in A^{\mathcal{I}}$ again by Claim 1.

For $r(a, b) \in \mathsf{At}(\rho_i)$, we can show $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$ as in the proof of Claim 2.

Assume now that there is a role atom of the form $r(a, v)$ in $\mathsf{At}(\rho_i)$, i.e., $((a, \varepsilon), \pi(v)) \in r^{\widehat{\mathcal{J}}}$. If $((a, \varepsilon), \pi(v)) \in r^{\mathcal{J}}$, then $(a, \pi'(v)) \in r^{\mathcal{I}}$ by the definitions of $\mathcal{J}$ and $\pi'$. Otherwise, there must be a role name $s$ such that $\mathcal{R} \models s \sqsubseteq r$, $\mathcal{R} \models \mathsf{trans}(s)$, and $((a, \varepsilon), \pi(v)) \in (s^{\mathcal{J}})^+$. This implies the existence of a sequence $(a_0, \varepsilon)$, $(a_1, \varepsilon)$, $\ldots$, $(a_n, \varepsilon)$, $(a_n, e_1)$, $\ldots$, $(a_n, e_1 \ldots e_k)$ in $\Delta^{\widehat{\mathcal{J}}}$ such that $a_0 = a$, $\pi(v) = (a_n, e_1 \ldots e_k)$, and each two consecutive elements of this sequence are connected via $s^{\mathcal{J}}$. By the definition of $s^{\mathcal{J}}$, we get $(a^{\mathcal{I}}, \pi'(v)) \in s^{\mathcal{I}} \subseteq r^{\mathcal{I}}$.

For any role atom $r(v, a) \in \mathsf{At}(\rho_i)$, we know that $(\pi(v), (a, \varepsilon)) \in r^{\widehat{\mathcal{J}}}$. By the definition of $r^{\widehat{\mathcal{J}}}$, this implies that there is a sequence $(a_0, \varepsilon)$, $\ldots$, $(a_n, \varepsilon)$ in $\Delta^{\widehat{\mathcal{J}}}$ such that $a_n = a$, $\pi(v) = (a_0, \varepsilon)$, and each two consecutive elements of this sequence are connected via $s^{\mathcal{J}}$, where $s$ is a role name such that either $n = 1$ and $s = r$, or $\mathcal{R} \models s \sqsubseteq r$ and $\mathcal{R} \models \mathsf{trans}(s)$. By the definition of $s^{\mathcal{J}}$, the properties of $s$, and since $\mathcal{I} \models \mathcal{R}$, this implies that $(\pi'(v), a^{\mathcal{I}}) = (a_0^{\mathcal{I}}, a_n^{\mathcal{I}}) \in r^{\mathcal{I}}$.

Finally, consider a role atom of the form $r(v, v')$ in $\mathsf{At}(\rho_i)$. We have $(\pi(v), \pi(v')) \in r^{\widehat{\mathcal{J}}}$. If $\pi(v) = (a, \varepsilon)$ for some $a \in \mathsf{Ind}(\mathcal{A})$, then we can show as in the case of $r(a, v)$ that $(\pi'(v), \pi'(v')) = (a^{\mathcal{I}}, \pi'(v')) \in r^{\mathcal{I}}$. Otherwise, we have $\pi(v) = (a, d_1 \ldots d_m)$ for $m > 0$ and there is a sequence $(a, d_1 \ldots d_m)$, $(a, d_1 \ldots d_{m+1})$, $\ldots$, $(a, d_1 \ldots d_{m+n})$ in $\Delta^{\widehat{\mathcal{J}}}$ such that $n \geq 1$, $\pi(v') = (a, d_1 \ldots d_{m+n})$, and each two consecutive elements of this sequence are connected via $s^{\mathcal{J}}$, where $s$ is a role name such that either $n = 1$ and $s = r$, or $\mathcal{R} \models s \sqsubseteq r$ and $\mathcal{R} \models \mathsf{trans}(s)$. This implies that $(\pi'(v), \pi'(v')) = (d_m, d_{m+n}) \in s^{\mathcal{I}} \subseteq r^{\mathcal{I}}$.

Hence, $\mathcal{I} \models \rho_i$, and thus $\mathcal{I} \models \rho$, which contradicts our assumption that $\mathcal{I} \not\models \rho$. □

**Lemma 6.14.** *Let $\mathcal{B}$ be a Boolean $\mathcal{SHQ}^{\cap}$-knowledge base, let $A_1, \ldots, A_k$ be concept names occurring in $\mathcal{B}$, and let $\mathcal{D} \subseteq 2^{\{A_1, \ldots, A_k\}}$. Then $\mathcal{B}$ is consistent w.r.t. $\mathcal{D}$ iff it has a quasimodel that respects $\mathcal{D}$.*

Proof. For the "if" direction, suppose that $\mathcal{M} = (\mathcal{S}, \iota, \mathbf{f})$ is a quasimodel for $\mathcal{B} = \langle \Psi, \mathcal{R}\rangle$ that respects $\mathcal{D}$. Then by condition (f), for each $\mathbf{c} \in \mathcal{S}$, $E_{\mathcal{M}, \mathbf{c}}$ has a solution $\nu_{\mathbf{c}}$ that maps the variables of $E_{\mathcal{M}, \mathbf{c}}$ into the non-negative integers. Let $z_{\mathcal{M}}$ be the greatest non-negative integer that occurs in any of these solutions. Let $\mathfrak{Z}$ denote the set $\{1, \ldots, z_{\mathcal{M}}\}$.

We define an interpretation $\mathcal{J} = (\Delta^{\mathcal{J}}, \cdot^{\mathcal{J}})$ as follows:

- $\Delta^{\mathcal{J}} := \mathsf{Anon} \cup \mathsf{Ind}(\Psi)$, where $\mathsf{Anon} := \mathcal{S}_u \times \mathfrak{Z} \times \mathfrak{R}(\mathcal{B})$;
- $a^{\mathcal{J}} := a$ for all $a \in \mathsf{Ind}(\Psi)$;[5]
- $A^{\mathcal{J}} := \{(\mathbf{c}, i, \mathbf{r}) \in \mathsf{Anon} \mid A \in \mathbf{c}\} \cup$ $\{a \in \mathsf{Ind}(\Psi) \mid A \in \iota(a)\}$ for all $A \in N_{\mathrm{C}}$; and
- for all role names $r \in N_{\mathrm{R}}$, $(\mathbf{c}, i, \mathbf{r}), (\mathbf{d}, j, \mathbf{s}) \in \mathsf{Anon}$, and $a, b \in \mathsf{Ind}(\Psi)$, we define
  - $(a, b) \in r^{\mathcal{J}}$ iff $r(a, b) \in \mathbf{f}$;

[5] We ignore for now the individual names in $N_{\mathrm{I}} \setminus \mathsf{Ind}(\Psi)$ since they are irrelevant when dealing with $\mathcal{B}$. After constructing the model $\mathcal{I}$ below, one can ensure that it respects the UNA by constructing the countably infinite disjoint union of $\mathcal{I}$ with itself to allow for different interpretations of each of these individual names.

– $(a,(\mathbf{d},j,\mathbf{s})) \in r^{\mathcal{J}}$ iff $r \in \mathbf{s}$, $(\iota(a),\mathbf{d}) \in \mathbf{s}^{\mathcal{R}}$, and $\nu_{\iota(a)}(x_{\iota(a),\mathbf{s},\mathbf{d}}) \geq j$;
– $((\mathbf{c},i,\mathbf{r}),(\mathbf{d},j,\mathbf{s})) \in r^{\mathcal{J}}$ iff $r \in \mathbf{s}$, $(\mathbf{c},\mathbf{d}) \in \mathbf{s}^{\mathcal{R}}$, and $\nu_{\mathbf{c}}(x_{\mathbf{c},\mathbf{s},\mathbf{d}}) \geq j$;
– $((\mathbf{c},i,\mathbf{r}),b) \notin r^{\mathcal{J}}$.

Now we construct a model $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ of $\mathcal{B}$ by defining $\Delta^{\mathcal{I}} := \Delta^{\mathcal{J}}$, for each $A \in N_{\mathrm{C}}$, $A^{\mathcal{I}} := A^{\mathcal{J}}$, for each $a \in \mathsf{Ind}(\Psi)$, $a^{\mathcal{I}} := a^{\mathcal{J}}$, and for each $r \in N_{\mathrm{R}}$,

$$r^{\mathcal{I}} := r^{\mathcal{J}} \cup \bigcup_{\mathcal{R}\models s\sqsubseteq r,\ \mathcal{R}\models\mathsf{trans}(s)} (s^{\mathcal{J}})^{+}.$$

We prove the following claim by structural induction.

**Claim 4.** For all concepts $C \in \mathsf{Con}(\mathcal{B})$, we have:

$$C^{\mathcal{I}} = \{(\mathbf{c},i,\mathbf{r}) \in \mathsf{Anon} \mid C \in \mathbf{c}\} \cup \{a \in \mathsf{Ind}(\Psi) \mid C \in \iota(a)\}.$$

For the base case, $C$ being a concept name, the definition of $\mathcal{I}$ immediately implies the claim. For the case that $C$ is of the form $\neg D$, we have by the semantics of $\mathcal{SHQ}^{\sqcap}$, the induction hypothesis, the definition of $\mathcal{I}$, and the definition of concept types that

- for all $(\mathbf{c},i,\mathbf{r}) \in \mathsf{Anon}$, we have $(\mathbf{c},i,\mathbf{r}) \in (\neg D)^{\mathcal{I}}$ iff $D \notin \mathbf{c}$ iff $\neg D \in \mathbf{c}$; and
- for all $a \in \mathsf{Ind}(\Psi)$, we have $a \in (\neg D)^{\mathcal{I}}$ iff $D \notin \iota(a)$ iff $\neg D \in \iota(a)$.

For the case that $C$ is of the form $D \sqcap E$, we have by similar arguments that following:

- for all $(\mathbf{c},i,\mathbf{r}) \in \mathsf{Anon}$, we have $(\mathbf{c},i,\mathbf{r}) \in (D \sqcap E)^{\mathcal{I}}$ iff $D \in \mathbf{c}$ and $E \in \mathbf{c}$ iff $D \sqcap E \in \mathbf{c}$; and
- for all $a \in \mathsf{Ind}(\Psi)$, we have $a \in (D \sqcap E)^{\mathcal{I}}$ iff $D \in \iota(a)$ and $E \in \iota(a)$ iff $D \sqcap E \in \iota(a)$.

For the case that $C$ is of the form $\exists(r_1 \cap \cdots \cap r_\ell).D$, we have by similar arguments the following:

$$\begin{aligned}
&(\exists(r_1 \cap \cdots \cap r_\ell).D)^{\mathcal{I}}\\
&= \{d \in \Delta^{\mathcal{I}} \mid \text{there is an } e \in \Delta^{\mathcal{I}} \text{ with}\\
&\qquad (d,e) \in r_1^{\mathcal{I}} \cap \cdots \cap r_\ell^{\mathcal{I}} \text{ and } e \in D^{\mathcal{I}}\}\\
&= \{d \in \Delta^{\mathcal{I}} \mid \text{there is a } (\mathbf{d},j,\mathbf{s}) \in D^{\mathcal{I}} \text{ with}\\
&\qquad (d,(\mathbf{d},j,\mathbf{s})) \in r_1^{\mathcal{I}} \cap \cdots \cap r_\ell^{\mathcal{I}}\} \cup\\
&\quad \{a \in \mathsf{Ind}(\Psi) \mid \text{there is a } b \in \mathsf{Ind}(\Psi) \text{ with}\\
&\qquad (a,b) \in r_1^{\mathcal{I}} \cap \cdots \cap r_\ell^{\mathcal{I}} \text{ and } b \in D^{\mathcal{I}}\}\\
&= \{d \in \Delta^{\mathcal{I}} \mid \text{there is a } (\mathbf{d},j,\mathbf{s}) \in \mathsf{Anon} \text{ with}\\
&\qquad (d,(\mathbf{d},j,\mathbf{s})) \in r_1^{\mathcal{I}} \cap \cdots \cap r_\ell^{\mathcal{I}} \text{ and } D \in \mathbf{d}\} \cup\\
&\quad \{a \in \mathsf{Ind}(\Psi) \mid \text{there is a } b \in \mathsf{Ind}(\Psi) \text{ with}\\
&\qquad (a,b) \in r_1^{\mathcal{I}} \cap \cdots \cap r_\ell^{\mathcal{I}} \text{ and } D \in \iota(b)\}\\
&\overset{*}{=} \{(\mathbf{c},i,\mathbf{r}) \in \mathsf{Anon} \mid \exists(r_1 \cap \cdots \cap r_\ell).D \in \mathbf{c}\} \cup\\
&\quad \{a \in \mathsf{Ind}(\Psi) \mid \exists(r_1 \cap \cdots \cap r_\ell).D \in \iota(a)\}.
\end{aligned}$$

The starred equality $\overset{*}{=}$ holds due to the following arguments. Assume, for the direction ($\supseteq$), that $(\mathbf{c},i,\mathbf{r}) \in \mathsf{Anon}$ and $\exists(r_1\cap\cdots\cap r_\ell).D \in \mathbf{c}$. Since $\nu_{\mathbf{c}}$ solves (E3), there are $\mathbf{d} \in \mathcal{S}_u$ and $\mathbf{s} \in \mathfrak{R}(\mathcal{B})$ such that $D \in \mathbf{d}$, $r_1,\ldots,r_\ell \in \mathbf{s}$, $(\mathbf{c},\mathbf{d}) \in \mathbf{s}^{\mathcal{R}}$, and $\nu_{\mathbf{c}}(x_{\mathbf{c},\mathbf{s},\mathbf{d}}) \geq 1$. By definition of $r_1^{\mathcal{J}},\ldots,r_\ell^{\mathcal{J}}$, we obtain $((\mathbf{c},i,\mathbf{r}),(\mathbf{d},1,\mathbf{s})) \in r_1^{\mathcal{J}} \cap \cdots \cap r_\ell^{\mathcal{J}} \subseteq r_1^{\mathcal{I}} \cap \cdots \cap r_\ell^{\mathcal{I}}$. For the remaining part of the direction ($\supseteq$), assume that $a \in \mathsf{Ind}(\Psi)$ and $\exists(r_1\cap\cdots\cap r_\ell).D \in \iota(a)$. Since $\nu_{\mathbf{c}}$ is a solution of (E3), there is an $\mathbf{s} \in \mathfrak{R}(\mathcal{B})$ such that $r_1,\ldots,r_\ell \in \mathbf{s}$ and

- there is a $\mathbf{d} \in \mathcal{S}_u$ with $D \in \mathbf{d}$, $(\iota(a),\mathbf{d}) \in \mathbf{s}^{\mathcal{R}}$, and $\nu_{\iota(a)}(x_{\iota(a),\mathbf{s},\mathbf{d}}) \geq 1$; or
- there is a $b \in \mathsf{Ind}(\Psi)$ such that $D \in \iota(b)$ and $\{r_1(a,b),\ldots r_\ell(a,b)\} \subseteq \mathbf{f}$.

In the first case, we can infer that $(a,(\mathbf{d},1,\mathbf{s})) \in r_1^{\mathcal{I}}\cap\cdots\cap r_\ell^{\mathcal{I}}$ as above. In the second case, by the definition of $\mathcal{J}$, we get $(a,b) \in r_1^{\mathcal{J}} \cap \cdots \cap r_\ell^{\mathcal{J}} \subseteq r_1^{\mathcal{I}} \cap \cdots \cap r_\ell^{\mathcal{I}}$.

For the other direction ($\subseteq$), consider a $d \in \Delta^{\mathcal{I}}$ and $(\mathbf{d},j,\mathbf{s}) \in \mathsf{Anon}$ such that $(d,(\mathbf{d},j,\mathbf{s})) \in r_1^{\mathcal{I}} \cap \cdots \cap r_\ell^{\mathcal{I}}$ and $D \in \mathbf{d}$. We consider first the case that $d = (\mathbf{c},i,\mathbf{r}) \in \mathsf{Anon}$ and show that $C = \exists(r_1 \cap \cdots \cap r_\ell).D \in \mathbf{c}$. Assume to the contrary that $C \notin \mathbf{c}$, and thus $\neg C \in \mathbf{c}$.

- For the case $\ell > 1$, we have that $r_1,\ldots,r_\ell$ are simple role names, and thus $((\mathbf{c},i,\mathbf{r}),(\mathbf{d},j,\mathbf{s})) \in r_1^{\mathcal{J}}\cap\cdots\cap r_\ell^{\mathcal{J}}$. By definition of $\mathcal{J}$, we have $r_1,\ldots,r_\ell \in \mathbf{s}$, $(\mathbf{c},\mathbf{d}) \in \mathbf{s}^{\mathcal{R}}$, and $\nu_{\mathbf{c}}(x_{\mathbf{c},\mathbf{s},\mathbf{d}}) \geq j \geq 1$. Since $\nu_{\mathbf{c}}$ is a solution of (E4), we must have $\nu_{\mathbf{c}}(x_{\mathbf{c},\mathbf{s},\mathbf{d}}) = 0$, which is a contradiction.
- For the case $\ell = 1$, by the definition of $r_1^{\mathcal{I}}$, we have $((\mathbf{c},i,\mathbf{r}),(\mathbf{d},j,\mathbf{s})) \in r_1^{\mathcal{J}}$ or $((\mathbf{c},i,\mathbf{r}),(\mathbf{d},j,\mathbf{s})) \in (s^{\mathcal{J}})^{+}$ for some $s \in N_{\mathrm{R}}$ with $\mathcal{R} \models s \sqsubseteq r_1$ and $\mathcal{R} \models \mathsf{trans}(s)$. The first case can be handled as in the case for $\ell > 1$, while in the second case there is a sequence $(\mathbf{c}_0,i_0,\mathbf{r}_0)$, $\ldots$, $(\mathbf{c}_n,i_n,\mathbf{r}_n)$ in $\mathsf{Anon}$ such that
  – $n \geq 1$;
  – $(\mathbf{c}_0,i_0,\mathbf{r}_0) = (\mathbf{c},i,\mathbf{r})$;
  – $(\mathbf{c}_n,i_n,\mathbf{r}_n) = (\mathbf{d},j,\mathbf{s})$; and
  – for all $k$, $0 \leq k \leq n-1$, we have $s \in \mathbf{r}_{k+1}$, $(\mathbf{c}_k,\mathbf{c}_{k+1}) \in \mathbf{r}_{k+1}^{\mathcal{R}}$, and $\nu_{\mathbf{c}_k}(x_{\mathbf{c}_k,\mathbf{r}_{k+1},\mathbf{c}_{k+1}}) \geq i_{k+1}$.

  If $n = 1$, then $\mathbf{c}_1 = \mathbf{d}$, $\mathbf{r}_1 = \mathbf{s}$, and $(\mathbf{c},\mathbf{d}) \in \mathbf{s}^{\mathcal{R}}$. Since $\mathbf{s}$ is a role type, $s \in \mathbf{s}$, and $\mathcal{R} \models s \sqsubseteq r_1$, we also have $r_1 \in \mathbf{s}$. This implies that $(\mathbf{c},\mathbf{d}) \in \mathbf{s}^{\mathcal{R}} \subseteq r_1^{\mathcal{R}}$. Since we assumed that $\neg(\exists r_1.D) \in \mathbf{c}$, we obtain $\neg D \in \mathbf{d}$, which yields a contradiction.

  If $n > 1$, then, since $\neg(\exists r_1.D) \in \mathbf{c}$ and $(\mathbf{c},\mathbf{c}_1) \in s^{\mathcal{R}}$, we have $\neg(\exists s.D) \in \mathbf{c}_1$. By similar arguments, we can infer that $\neg(\exists s.D) \in \mathbf{c}_{n-1}$. Since $(\mathbf{c}_{n-1},\mathbf{d}) \in s^{\mathcal{R}}$, we again conclude the contradictory $\neg D \in \mathbf{d}$.

For the second part of the direction ($\subseteq$), consider the case that $d = a \in \mathsf{Ind}(\Psi)$. We show $C = \exists(r_1\cap\cdots\cap r_\ell).D \in \iota(a)$ by similar arguments as above. Assume that $\neg C \in \iota(a)$.

- For the case $\ell > 1$, we have $(a, (\mathbf{d}, j, \mathbf{s})) \in r_1^{\mathcal{J}} \cap \dots \cap r_\ell^{\mathcal{J}}$. It follows from the definition of $\mathcal{J}$ that $r_1, \dots, r_\ell \in \mathbf{s}$, $(\iota(a), \mathbf{d}) \in \mathbf{s}^{\mathcal{R}}$, and $\nu_{\iota(a)}(x_{\iota(a),\mathbf{s},\mathbf{d}}) \geq j \geq 1$. As before, this contradicts the fact that $\nu_{\iota(a)}$ is a solution of (E4).

- For the case $\ell = 1$, we have $(a, (\mathbf{d}, j, \mathbf{s})) \in r_1^{\mathcal{J}}$ or $(a, (\mathbf{d}, j, \mathbf{s})) \in (s^{\mathcal{J}})^+$ for some $s \in N_{\mathrm{R}}$ with $\mathcal{R} \models s \sqsubseteq r_1$ and $\mathcal{R} \models \mathsf{trans}(s)$. The first case is again the same as for the case $\ell > 1$, while in the second case, there is a sequence $a_0, \dots, a_n, (\mathbf{c}_0, i_0, \mathbf{r}_0), \dots, (\mathbf{c}_m, i_m, \mathbf{r}_m)$ in $\Delta^{\mathcal{I}}$ such that

  - $n, m \geq 0$;
  - $a_0 = a$;
  - $(\mathbf{c}_m, i_m, \mathbf{r}_m) = (\mathbf{d}, j, \mathbf{s})$;
  - for all $k$, $0 \leq k \leq n-1$, we have $s(a_k, a_{k+1}) \in \mathbf{f}$;
  - $\nu_{\iota(a_n)}(x_{\iota(a_n),\mathbf{r}_0,\mathbf{c}_0}) \geq i_0$, $(\iota(a_n), \mathbf{c}_0) \in \mathbf{r}_0^{\mathcal{R}}$, and $s \in \mathbf{r}_0$; and
  - for all $k$, $0 \leq k \leq m-1$, we have $s \in r_{k+1}$, $(\mathbf{c}_k, \mathbf{c}_{k+1}) \in r_{k+1}^{\mathcal{R}}$, and $\nu_{\mathbf{c}_k}(x_{\mathbf{c}_k,\mathbf{r}_{k+1},\mathbf{c}_{k+1}}) \geq i_{k+1}$.

  We first consider the case that $n = m = 0$. Then $a = a_n$, $\mathbf{c}_0 = \mathbf{d}$, $\mathbf{r}_0 = \mathbf{s}$, and $(\iota(a), \mathbf{d}) \in \mathbf{s}^{\mathcal{R}}$. Since $\mathbf{s}$ is a role type, $s \in \mathbf{s}$, and $\mathcal{R} \models s \sqsubseteq r_1$, we also have $r_1 \in \mathbf{s}$, and thus $(\iota(a), \mathbf{d}) \in r_1^{\mathcal{R}}$. Since $\neg(\exists r_1.D) \in \iota(a)$, we obtain $\neg D \in \mathbf{d}$, which is a contradiction.

  If $n = 0$ and $m > 0$, then we have $(\iota(a), \mathbf{c}_0) \in s^{\mathcal{R}}$ since $s \in \mathbf{r}_0$. Since $\neg(\exists r_1.D) \in \iota(a)$, we obtain $\neg(\exists s.D) \in \mathbf{c}_0$, and similarly $\neg(\exists s.D) \in \mathbf{c}_{m-1}$, and thus $\neg D \in \mathbf{c}_m = \mathbf{d}$. This is a contradiction.

  If $n > 0$, then $s(a, a_1) \in \mathbf{f}$. By condition (c), this implies that $(\iota(a), \iota(a_1)) \in s^{\mathcal{R}}$. Since $\neg(\exists r_1.D) \in \iota(a)$, we obtain $\neg(\exists s.D) \in \iota(a_1)$. By similar arguments, we can infer that $\neg(\exists s.D) \in \iota(a_n)$, and finally $\neg D \in \mathbf{d}$, which contradicts our assumption that $D \in \mathbf{d}$.

For the last part of the direction ($\subseteq$), let $a, b \in \mathsf{Ind}(\Psi)$ with $(a, b) \in r_1^{\mathcal{I}} \cap \dots \cap r_\ell^{\mathcal{I}}$ and $D \in \iota(b)$. For the last time, we assume that $C = \exists(r_1 \cap \dots \cap r_\ell).D \notin \iota(a)$ and make a case distinction on $\ell$.

- If $\ell > 1$, then $(a, b) \in r_1^{\mathcal{J}} \cap \dots \cap r_\ell^{\mathcal{J}}$, and thus $\{r_1(a, b), \dots, r_\ell(a, b)\} \subseteq \mathbf{f}$. Since $\mathbf{f}$ is a formula type, the set $\{r \in \mathsf{Rol}(\mathcal{B}) \mid r(a, b) \in \mathbf{f}\}$ is a role type that contains $r_1, \dots, r_\ell$. Since $D \in \iota(b)$, we know that $\Gamma_{\mathcal{M},\iota(a),\mathbf{r},D} \geq 1$. This contradicts our assumption that (E4) has a solution.

- If $\ell = 1$, then $(a, b) \in r_1^{\mathcal{J}}$ or $(a, b) \in (s^{\mathcal{J}})^+$ for some $s \in N_{\mathrm{R}}$ with $\mathcal{R} \models s \sqsubseteq r_1$ and $\mathcal{R} \models \mathsf{trans}(s)$. The first case is impossible by the same arguments as above, and in the second case, there is a sequence $a_0, \dots, a_n$ in $\mathsf{Ind}(\Psi)$ such that

  - $n \geq 1$;
  - $a_0 = a$;
  - $a_n = b$; and
  - for all $k$, $0 \leq k \leq n-1$, we have $s(a_k, a_{k+1}) \in \mathbf{f}$.

  If $n = 1$, then $s(a, b) \in \mathbf{f}$, and thus $(\iota(a), \iota(b)) \in s^{\mathcal{R}}$ by condition (c). Since $\neg(\exists r_1.D) \in \iota(a)$, we again obtain $\neg D \in \iota(b)$, and thus a contradiction.

  If $n > 1$, then $r_1(a, a_1) \in \mathbf{f}$ since $\mathbf{f}$ is a formula type for $\mathcal{B}$ and $\mathcal{R} \models s \sqsubseteq r_1$. By condition (c), we obtain $(\iota(a), \iota(a_1)) \in r_1^{\mathcal{R}}$, and thus $\neg(\exists s.D) \in \iota(a_1)$ since $\neg(\exists r_1.D) \in \iota(a)$. Similarly, we can infer that $\neg(\exists s.D) \in \iota(a_{n-1})$, and finally $\neg D \in \iota(b)$. This contradicts our assumption that $D \in \iota(b)$.

Finally, consider the case that $C$ is of the form $\geq n\, r.D$. Recall that $r$ must be simple, and thus $r^{\mathcal{I}} = r^{\mathcal{J}}$. We first count, for any element $d \in \Delta^{\mathcal{I}}$, the number of *unnamed* $r^{\mathcal{J}}$-successors that satisfy $D$. Let $\mathbf{c}$ be a concept type such that either $d = (\mathbf{c}, i, \mathbf{r}) \in \mathsf{Anon}$, or $d = a \in \mathsf{Ind}(\Psi)$ and $\mathbf{c} = \iota(a)$. For a fixed role type $\mathbf{s} \in \mathfrak{R}(\mathcal{B})$ and concept type $\mathbf{d} \in \mathcal{S}_u$ with $r \in \mathbf{s}$, $D \in \mathbf{d}$, and $(\mathbf{c}, \mathbf{d}) \in \mathbf{s}^{\mathcal{R}}$, we have by definition of $\mathcal{J}$ that $((\mathbf{c}, i, \mathbf{r}), (\mathbf{d}, j, \mathbf{s})) \in r^{\mathcal{J}}$ iff $\nu_{\mathbf{c}}(x_{\mathbf{c},\mathbf{s},\mathbf{d}}) \geq j$. Thus, the number of $r^{\mathcal{J}}$-successors of $d$ that are of the form $(\mathbf{d}, j, \mathbf{s})$ is exactly $\nu_{\mathbf{c}}(x_{\mathbf{c},\mathbf{s},\mathbf{d}})$. By induction, we obtain

$$
\begin{aligned}
&|\{(\mathbf{d}, j, \mathbf{s}) \in \mathsf{Anon} \mid (d, (\mathbf{d}, j, \mathbf{s})) \in r^{\mathcal{J}},\ (\mathbf{d}, j, \mathbf{s}) \in D^{\mathcal{I}}\}| \\
&\quad = |\{(\mathbf{d}, j, \mathbf{s}) \in \mathsf{Anon} \mid (d, (\mathbf{d}, j, \mathbf{s})) \in r^{\mathcal{J}},\ D \in \mathbf{d}\}| \\
&\quad = \sum_{\substack{r \in \mathbf{s} \in \mathfrak{R}(\mathcal{B}) \\ D \in \mathbf{d} \in \mathcal{S}_u,\ (\mathbf{c},\mathbf{d}) \in \mathbf{s}^{\mathcal{R}}}} |\{j \in \mathfrak{Z} \mid (d, (\mathbf{d}, j, \mathbf{s})) \in r^{\mathcal{J}}\}| \\
&\quad = \sum_{\substack{r \in \mathbf{s} \in \mathfrak{R}(\mathcal{B}), \\ D \in \mathbf{d} \in \mathcal{S}_u,\ (\mathbf{c},\mathbf{d}) \in \mathbf{s}^{\mathcal{R}}}} \nu_{\mathbf{c}}(x_{\mathbf{c},\mathbf{s},\mathbf{d}}) \\
&\quad = \sum_{r \in \mathbf{s} \in \mathfrak{R}(\mathcal{B})} \nu_{\mathbf{c}}(\Xi_{\mathcal{M},\mathbf{c},\mathbf{s},D}). \qquad (1)
\end{aligned}
$$

To similarly count the *named* successors of $d \in \Delta^{\mathcal{I}}$, we only have to consider the case that $d = a \in \mathsf{Ind}(\Psi)$ since unnamed domain elements can only have unnamed $r^{\mathcal{J}}$-successors. By the definitions of role types and formula types, for every $b \in \mathsf{Ind}(\Psi)$ there is a unique role type $\mathbf{s} \in \mathfrak{R}(\mathcal{B})$ such that $s(a, b) \in \mathbf{f}$ iff $s \in \mathbf{s}$. By definition of $\mathcal{J}$, $s(a, b) \in \mathbf{f}$ is equivalent to $(a, b) \in s^{\mathcal{J}}$, and thus we have

$$
\begin{aligned}
&|\{b \in \mathsf{Ind}(\Psi) \mid (a, b) \in r^{\mathcal{J}},\ b \in D^{\mathcal{I}}\}| \\
&= |\{b \in \mathsf{Ind}(\Psi) \mid (a, b) \in r^{\mathcal{J}},\ D \in \iota(b)\}| \\
&= \sum_{r \in \mathbf{s} \in \mathfrak{R}(\mathcal{B})} |\{b \in \mathsf{Ind}(\Psi) \mid (a, b) \in s^{\mathcal{J}} \text{ iff } s \in \mathbf{s},\ D \in \iota(b)\}| \\
&= \sum_{r \in \mathbf{s} \in \mathfrak{R}(\mathcal{B})} |\{b \in \mathsf{Ind}(\Psi) \mid s(a, b) \in \mathbf{f} \text{ iff } s \in \mathbf{s},\ D \in \iota(b)\}| \\
&= \sum_{r \in \mathbf{s} \in \mathfrak{R}(\mathcal{B})} \Gamma_{\mathcal{M},\iota(a),\mathbf{s},D}. \qquad (2)
\end{aligned}
$$

For every $(\mathbf{c}, i, \mathbf{r}) \in \mathsf{Anon}$, we know that $\nu_{\mathbf{c}}$ solves the inequations in (E1) and (E2). Thus, we have $\geq n\, r.D \in \mathbf{c}$

iff the values in (1) are all $\geq n$ iff $(\mathbf{c}, i, \mathbf{r}) \in (\geq n\, r.D)^{\mathcal{I}}$. Similarly, for $a \in \mathsf{Ind}(\Psi)$ it follows that $\geq n\, r.D \in \iota(a)$ iff the sum of (1) and (2) is $\geq n$ iff $a \in (\geq n\, r.D)^{\mathcal{I}}$.

This finishes the proof of Claim 4. To show that $\mathcal{I}$ is indeed a model of $\mathcal{B}$, we first show the following claim by structural induction.

**Claim 5.** For all $\psi \in \mathsf{Sub}(\mathcal{B})$, we have $\psi \in \mathbf{f}$ iff $\mathcal{I} \models \psi$.

For the first base case, assume that $\psi$ is of the form $A(a)$ for $A \in N_{\mathrm{C}}$ and $a \in N_{\mathrm{I}}$. We have $A(a) \in \mathbf{f}$ iff $A \in \iota(a)$ by condition (b). Thus, $A(a) \in \mathbf{f}$ iff $a^{\mathcal{I}} = a^{\mathcal{J}} = a \in A^{\mathcal{J}} = A^{\mathcal{I}}$ iff $\mathcal{I} \models A(a)$.

For the second base case, assume that $\psi$ is of the form $r(a, b)$ for $a, b \in N_{\mathrm{I}}$ and $r \in N_{\mathrm{R}}$. If $r(a, b) \in \mathbf{f}$, we have $(a, b) \in r^{\mathcal{J}}$ by the definition of $r^{\mathcal{J}}$. Since $r^{\mathcal{J}} \subseteq r^{\mathcal{I}}$, $a = a^{\mathcal{I}}$, and $b = b^{\mathcal{I}}$, we obtain $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$, and thus $\mathcal{I} \models r(a, b)$. Conversely, if $\mathcal{I} \models r(a, b)$, we have by the definition of $r^{\mathcal{I}}$ that $(a, b) \in r^{\mathcal{J}}$ or $(a, b) \in (s^{\mathcal{J}})^+$ for some $s \in N_{\mathrm{R}}$ with $\mathcal{R} \models s \sqsubseteq r$ and $\mathcal{R} \models \mathsf{trans}(s)$. If $(a, b) \in r^{\mathcal{J}}$, the definition of $r^{\mathcal{J}}$ implies that $r(a, b) \in \mathbf{f}$. Otherwise, there are $d_1, \ldots, d_m \in \Delta^{\mathcal{I}}$ such that $(a, d_1) \in s^{\mathcal{J}}$, $(d_1, d_2) \in s^{\mathcal{J}}$, ..., and $(d_m, b) \in s^{\mathcal{J}}$. By the definition of $s^{\mathcal{J}}$, we know that $d_1, \ldots, d_m \in \mathsf{Ind}(\Psi)$, and thus $s(a, d_1) \in \mathbf{f}$, $s(d_1, d_2) \in \mathbf{f}$, ..., and $s(d_m, b) \in \mathbf{f}$. The definition of a formula type yields that $s(a, b) \in \mathbf{f}$ and $r(a, b) \in \mathbf{f}$.

For the third base case, assume that $\psi$ is of the form $\top \sqsubseteq C$. If $\top \sqsubseteq C \in \mathbf{f}$, then for every $\mathbf{c} \in \mathcal{S}$, we have $C \in \mathbf{c}$ by condition (d). Claim 4 yields together with the fact that $\iota$ maps into $\mathcal{S}$ that $C^{\mathcal{I}} = \mathsf{Anon} \cup \mathsf{Ind}(\Psi) = \Delta^{\mathcal{I}}$. For the converse direction, if $\top \sqsubseteq C \notin \mathbf{f}$, then by the definition of a formula type, $\neg(\top \sqsubseteq C) \in \mathbf{f}$. Then, by condition (e), there is a $\mathbf{c} \in \mathcal{S}$ such that $C \notin \mathbf{c}$, which implies $\neg C \in \mathbf{c}$, because $\mathbf{c}$ is a concept type. Claim 4 yields that either $\{\mathbf{c}\} \times \mathfrak{Z} \times \mathfrak{R} \subseteq (\neg C)^{\mathcal{I}}$ or there is an $a \in \mathsf{Ind}(\Psi)$ such that $\mathbf{c} = \iota(a)$ and $a \in (\neg C)^{\mathcal{I}}$. Thus, we have $C^{\mathcal{I}} \neq \mathsf{Anon} \cup \mathsf{Ind}(\Psi) = \Delta^{\mathcal{I}}$.

For the induction step, assume first that $\psi$ is of the form $\neg\psi'$. By induction, we have $\psi \in \mathbf{f}$ iff $\psi' \notin \mathbf{f}$ iff $\mathcal{I} \not\models \psi'$ iff $\mathcal{I} \models \neg\psi'$. Similarly, if $\psi$ is of the form $\psi_1 \wedge \psi_2$, then $\psi \in \mathbf{f}$ iff $\{\psi_1, \psi_2\} \subseteq \mathbf{f}$ iff $\mathcal{I} \models \psi_1$ and $\mathcal{I} \models \psi_2$ iff $\mathcal{I} \models \psi_1 \wedge \psi_2$.

This finishes the proof of Claim 5. Since $\mathbf{f}$ is a formula type for $\Psi$, we have $\Psi \in \mathbf{f}$, and thus together with Claim 5 that $\mathcal{I} \models \Psi$. We now show that $\mathcal{I}$ is also a model of $\mathcal{R}$.

**Claim 6.** For all $\alpha \in \mathcal{R}$, we have $\mathcal{I} \models \alpha$.

Assume first that $\alpha$ is of the form $r \sqsubseteq s$. We first show that $r^{\mathcal{J}} \subseteq s^{\mathcal{J}}$. For this, take $(x, y) \in r^{\mathcal{J}}$. There are three cases to consider:

- If $x, y \in \mathsf{Ind}(\Psi)$, we have $r(x, y) \in \mathbf{f}$. Since $r \sqsubseteq s \in \mathcal{R}$, we have also $\mathcal{R} \models r \sqsubseteq s$, which yields $s(x, y) \in \mathbf{f}$ since $\mathbf{f}$ is a formula type. The definition of $s^{\mathcal{J}}$ yields that $(x, y) \in s^{\mathcal{J}}$.
- If $x \in \mathsf{Ind}(\Psi)$ and $y = (\mathbf{d}, j, \mathbf{s}) \in \mathsf{Anon}$, we have $r \in \mathbf{s}$, $(\iota(x), \mathbf{d}) \in \mathbf{s}^{\mathcal{R}}$, and $\nu_{\iota(x)}(x_{\iota(x), \mathbf{s}, \mathbf{d}}) \geq j$. By the definition of a role type, we have $s \in \mathbf{s}$. Hence, $(x, (\mathbf{d}, j, \mathbf{s})) \in s^{\mathcal{J}}$.
- If $x = (\mathbf{c}, i, \mathbf{r}) \in \mathsf{Anon}$ and $y = (\mathbf{d}, j, \mathbf{s}) \in \mathsf{Anon}$, we have $r \in \mathbf{s}$, $(\mathbf{c}, \mathbf{d}) \in \mathbf{s}^{\mathcal{R}}$, and $\nu_{\mathbf{c}}(x_{\mathbf{c}, \mathbf{s}, \mathbf{d}}) \geq j$. By the definition of a role type, we have $s \in \mathbf{s}$. Hence $((\mathbf{c}, i, \mathbf{r}), (\mathbf{d}, j, \mathbf{s})) \in s^{\mathcal{J}}$.

To show that $r^{\mathcal{I}} \subseteq s^{\mathcal{I}}$, take $(x, y) \in r^{\mathcal{I}}$. If $(x, y) \in r^{\mathcal{J}}$, we have $(x, y) \in s^{\mathcal{J}}$ and thus $(x, y) \in s^{\mathcal{I}}$. Otherwise, we have $(x, y) \in (t^{\mathcal{J}})^+$ with $\mathcal{R} \models t \sqsubseteq r$ and $\mathcal{R} \models \mathsf{trans}(t)$. Since $r \sqsubseteq s \in \mathcal{R}$, we also have $\mathcal{R} \models t \sqsubseteq s$. The definition of $s^{\mathcal{I}}$ yields that $(t^{\mathcal{J}})^+ \subseteq s^{\mathcal{I}}$, and hence $(x, y) \in s^{\mathcal{I}}$.

Assume now that $\psi$ is of the form $\mathsf{trans}(r)$. Since $\mathsf{trans}(r) \in \mathcal{R}$, we have also $\mathcal{R} \models \mathsf{trans}(r)$, and obviously also $\mathcal{R} \models r \sqsubseteq r$. By the same arguments as above, we have that for each $t$ with $\mathcal{R} \models t \sqsubseteq r$ that $t^{\mathcal{J}} \subseteq r^{\mathcal{J}}$, and thus $(t^{\mathcal{J}})^+ \subseteq (r^{\mathcal{J}})^+$ since the transitive closure is monotonic. This yields that $r^{\mathcal{I}} = (r^{\mathcal{J}})^+$, and thus $\mathcal{I} \models \mathsf{trans}(r)$.

This finishes the proof of Claim 6. Together with Claim 5, this implies that $\mathcal{I}$ is indeed a model of $\mathcal{B}$. It only remains to be shown that $\mathcal{I}$ respects $\mathcal{D}$. By condition (g) and Claim 4, we have for every $d \in \Delta^{\mathcal{I}}$ a set $Y \in \mathcal{D}$ such that $d \in (C_Y)^{\mathcal{I}}$. By condition (h) and Claim 4, we also have for every $Y \in \mathcal{D}$ a $d \in \Delta^{\mathcal{I}}$ such that $d \in (C_Y)^{\mathcal{I}}$. This shows that $\mathcal{I}$ respects $\mathcal{D}$.

This finishes the proof of the "if" direction of the lemma. For the "only if" direction, assume that there is a model $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ of $\mathcal{B} = \langle \Psi, \mathcal{R} \rangle$ that respects $\mathcal{D}$. Due to Lemma 6.5, we can assume w.l.o.g. that $\mathcal{I}$ is a forest model. We denote by $\Delta_u^{\mathcal{I}}$ the set $\{d \in \Delta^{\mathcal{I}} \mid d \neq a^{\mathcal{I}} \text{ for all } a \in N_{\mathrm{I}}\}$ of unnamed domain elements, and by $\Delta_n^{\mathcal{I}}$ the set $\Delta^{\mathcal{I}} \setminus \Delta_u^{\mathcal{I}}$ of named domain elements. We now construct a quasimodel for $\mathcal{B}$.

Let $\tau(d) := \{C \in \mathsf{Con}(\mathcal{B}) \mid d \in C^{\mathcal{I}}\}$ for $d \in \Delta^{\mathcal{I}}$. We define $\mathcal{M} = (\mathcal{S}, \iota, \mathbf{f})$ as follows:

- $\mathcal{S} := \{\tau(d) \mid d \in \Delta_u^{\mathcal{I}}\} \cup \{\tau(a^{\mathcal{I}}) \cup \{a\} \mid a \in \mathsf{Ind}(\Psi)\}$;
- $\iota(a) := \tau(a^{\mathcal{I}}) \cup \{a\}$ for all $a \in \mathsf{Ind}(\Psi)$; and
- $\mathbf{f} := \{\psi \in \mathsf{Sub}(\mathcal{B}) \mid \mathcal{I} \models \psi\}$.

Obviously, $\mathcal{S}$ is a set of concept types for $\mathcal{B}$, $\mathbf{f}$ is a formula type for $\mathcal{B}$, and we have also that for any $\mathbf{c}, \mathbf{d} \in \mathcal{S}$ with $\mathbf{c} \neq \mathbf{d}$ that $\mathbf{c} \cap \mathbf{d} \cap \mathsf{Ind}(\Psi) = \emptyset$. By definition, $a \in \iota(a)$ for all $a \in \mathsf{Ind}(\Psi)$. Hence, $\mathcal{M}$ is a model candidate for $\mathcal{B}$. We continue showing the following claim.

**Claim 7.** For all $d, e \in \Delta^{\mathcal{I}}$ and $r \in N_{\mathrm{R}}$, we have that $(d, e) \in r^{\mathcal{I}}$ implies $(\tau(d), \tau(e)) \in r^{\mathcal{R}}$.

Assume that $(d, e) \in r^{\mathcal{I}}$. For the first condition of $r$-compatibility, take any $\neg(\exists r.D) \in \tau(d)$, which implies that $d \in (\neg\exists r.D)^{\mathcal{I}}$. By the semantics of $\mathcal{SHQ}^{\cap}$, we have $e \in (\neg D)^{\mathcal{I}}$, and thus $\neg D \in \tau(e)$. For the second condition of $r$-compatibility, take any $s \in N_{\mathrm{R}}$ with $\mathcal{R} \models r \sqsubseteq s$, $\mathcal{R} \models \mathsf{trans}(r)$, and $\neg(\exists s.D) \in \tau(d)$. Since $\mathcal{I}$ is a model of $\mathcal{R}$, we have $r^{\mathcal{I}} \subseteq s^{\mathcal{I}}$ and $r^{\mathcal{I}}$ is transitive. Suppose that

$\neg(\exists r.D) \notin \tau(e)$, and thus $\exists r.D \in \tau(e)$. Then there is an $e' \in \Delta^{\mathcal{I}}$ with $e' \in D^{\mathcal{I}}$ and $(e, e') \in r^{\mathcal{I}}$. Since $r^{\mathcal{I}}$ is transitive, we have also $(d, e') \in r^{\mathcal{I}}$, and thus $(d, e') \in s^{\mathcal{I}}$, which yields a contradiction to $\neg(\exists s.D) \in \tau(d)$.

This finishes the proof of Claim 7. We can now use this claim to show that $\mathcal{M}$ is also a quasimodel for $\mathcal{B}$ that respects $\mathcal{D}$.

Condition (a) is easily verified, because $\Delta^{\mathcal{I}} \neq \emptyset$ by definition.

For Condition (b), we have $A(a) \in \mathbf{f}$ iff $\mathcal{I} \models A(a)$ iff $a^{\mathcal{I}} \in A^{\mathcal{I}}$ iff $A \in \tau(a^{\mathcal{I}}) \cup \{a\} = \iota(a)$.

For Condition (c), assume that $r(a,b) \in \mathbf{f}$. Then, $\mathcal{I} \models r(a,b)$, and thus $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$. Claim 7 yields that $(\tau(a^{\mathcal{I}}), \tau(b^{\mathcal{I}})) \in r^{\mathcal{R}}$. Obviously, we also have that $(\iota(a), \iota(b)) \in r^{\mathcal{R}}$.

For Condition (d), take $\mathbf{c} \in \mathcal{S}$ and $\top \sqsubseteq C \in \mathbf{f}$. The definition of $\mathbf{f}$ yields $\mathcal{I} \models \top \sqsubseteq C$, and thus $C^{\mathcal{I}} = \Delta^{\mathcal{I}}$. Hence, $C \in \tau(d)$ for any $d \in \Delta^{\mathcal{I}}$, which yields by the definition of $\mathcal{S}$ that $C \in \mathbf{c}$.

For Condition (e), take $\neg(\top \sqsubseteq C) \in \mathbf{f}$. By the definition of $\mathbf{f}$, this implies $\mathcal{I} \not\models \top \sqsubseteq C$. Thus, there is a $d \in \Delta^{\mathcal{I}}$ with $d \notin C^{\mathcal{I}}$. Thus, we have either $C \notin \tau(d) \in \mathcal{S}$ or $C \notin \tau(d) \cup \{a\} \in \mathcal{S}$ for some $a \in \mathsf{Ind}(\Psi)$.

For Condition (f), take any $\mathbf{c} \in \mathcal{S}$. We construct a solution $\nu_{\mathbf{c}}$ of the system of equations $E_{\mathcal{M},\mathbf{c}}$. Since $\mathbf{c} \in \mathcal{S}$, there is a $d \in \Delta^{\mathcal{I}}$ with $\mathbf{c} = \tau(d)$ if $d \in \Delta_u^{\mathcal{I}}$ and $\mathbf{c} = \tau(d) \cup \{a\}$ if $d = a^{\mathcal{I}}$ for some $a \in \mathsf{Ind}(\Psi)$. Let $z$ denote the maximal integer that occurs in any number restriction in $\mathcal{B}$. We first consider the variables $x_{\mathbf{c},\mathbf{r},\mathbf{d}}$. Take any $\mathbf{r} \in \mathfrak{R}(\mathcal{B})$ and any $\mathbf{d} \in \mathcal{S}_u$ such that $(\mathbf{c}, \mathbf{d}) \in \mathbf{r}^{\mathcal{R}}$. Then we define

$$\nu_{\mathbf{c}}(x_{\mathbf{c},\mathbf{r},\mathbf{d}}) := \min\big\{z,\ |\{e \in \Delta_u^{\mathcal{I}} \mid \tau(e) = \mathbf{d},\ (d,e) \in s^{\mathcal{I}} \text{ iff } s \in \mathbf{r}\}|\big\}.$$

We set $\nu_{\mathbf{c}}(x_{\mathbf{c},\mathbf{r},\mathbf{d}})$ to at most $z$ to ensure that this value is finite.

Consider now any $\geq n\, r.C \in \mathsf{Con}(\mathcal{B})$. We show that

$$\geq n\, r.C \in \mathbf{c} \text{ iff } \sum_{r \in \mathbf{r} \in \mathfrak{R}(\mathcal{B})} (\nu_{\mathbf{c}}(\Xi_{\mathcal{M},\mathbf{c},\mathbf{r},C}) + \Gamma_{\mathcal{M},\mathbf{c},\mathbf{r},C}) \geq n, \tag{3}$$

which implies that the inequations of the form (E1) and (E2) are satisfied.

Assume first that there are $\mathbf{d} \in \mathcal{S}_u$ and $\mathbf{r} \in \mathfrak{R}(\mathcal{B})$ such that $C \in \mathbf{d}$, $r \in \mathbf{r}$, $(\mathbf{c}, \mathbf{d}) \in \mathbf{r}^{\mathcal{R}}$, and $\nu_{\mathbf{c}}(x_{\mathbf{c},\mathbf{r},\mathbf{d}}) = z \geq n$. Then by definition of $\nu_{\mathbf{c}}$, there are at least $n$ unnamed domain elements $e \in \Delta_u^{\mathcal{I}}$ with $C \in \mathbf{d} = \tau(e)$ and $(d, e) \in r^{\mathcal{I}}$, which implies that $d \in (\geq n\, r.C)^{\mathcal{I}}$, and thus $\geq n\, r.C \in \mathbf{c}$. Additionally, $\nu_{\mathbf{c}}(\Xi_{\mathcal{M},\mathbf{c},\mathbf{r},C}) \geq z \geq n$, which shows that (3) holds. We assume in the following that for all $\mathbf{d} \in \mathcal{S}_u$ and $\mathbf{r} \in \mathfrak{R}(\mathcal{B})$ with $C \in \mathbf{d}$, $r \in \mathbf{r}$, and $(\mathbf{c}, \mathbf{d}) \in \mathbf{r}^{\mathcal{R}}$, we have $\nu_{\mathbf{c}}(x_{\mathbf{c},\mathbf{r},\mathbf{d}}) = |\{e \in \Delta_u^{\mathcal{I}} \mid \tau(e) = \mathbf{d},\ (d, e) \in s^{\mathcal{I}} \text{ iff } s \in \mathbf{r}\}|$.

It now follows that, for any $\mathbf{r} \in \mathfrak{R}(\mathcal{B})$, we have

$$\begin{aligned}
&\nu_{\mathbf{c}}(\Xi_{\mathcal{M},\mathbf{c},\mathbf{r},C}) \\
&= \sum_{C \in \mathbf{d} \in \mathcal{S}_u,\ (\mathbf{c},\mathbf{d}) \in \mathbf{r}^{\mathcal{R}}} \nu_{\mathbf{c}}(x_{\mathbf{c},\mathbf{r},\mathbf{d}}) \\
&= \sum_{C \in \mathbf{d} \in \mathcal{S}_u,\ (\mathbf{c},\mathbf{d}) \in \mathbf{r}^{\mathcal{R}}} |\{e \in \Delta_u^{\mathcal{I}} \mid \tau(e) = \mathbf{d},\ (d,e) \in s^{\mathcal{I}} \text{ iff } s \in \mathbf{r}\}| \\
&= |\{e \in C^{\mathcal{I}} \cap \Delta_u^{\mathcal{I}} \mid (d,e) \in s^{\mathcal{I}} \text{ iff } s \in \mathbf{r}\}|,
\end{aligned} \tag{4}$$

where the third equality follows by Claim 7. Thus,

$$\sum_{r \in \mathbf{r} \in \mathfrak{R}(\mathcal{B})} \nu_{\mathbf{c}}(\Xi_{\mathcal{M},\mathbf{c},\mathbf{r},C}) = |\{e \in C^{\mathcal{I}} \cap \Delta_u^{\mathcal{I}} \mid (d,e) \in r^{\mathcal{I}}\}|. \tag{5}$$

If $d \in \Delta_n^{\mathcal{I}}$, then $d = a^{\mathcal{I}}$ and $\mathbf{c} = \tau(a^{\mathcal{I}}) \cup \{a\}$ for some $a \in \mathsf{Ind}(\Psi)$. Thus,

$$\begin{aligned}
&\sum_{r \in \mathbf{r} \in \mathfrak{R}(\mathcal{B})} \Gamma_{\mathcal{M},\mathbf{c},\mathbf{r},C} \\
&= \sum_{r \in \mathbf{r} \in \mathfrak{R}(\mathcal{B})} |\{b \in \mathsf{Ind}(\Psi) \mid C \in \iota(b),\ s(a,b) \in \mathbf{f} \text{ iff } s \in \mathbf{r}\}| \\
&= |\{b \in \mathsf{Ind}(\Psi) \mid C \in \iota(b),\ r(a,b) \in \mathbf{f}\}| \\
&= |\{b \in \mathsf{Ind}(\Psi) \mid b^{\mathcal{I}} \in C^{\mathcal{I}},\ (a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}\}| \\
&= |\{e \in C^{\mathcal{I}} \cap \Delta_n^{\mathcal{I}} \mid (d,e) \in r^{\mathcal{I}}\}|.
\end{aligned} \tag{6}$$

If $d \in \Delta_u^{\mathcal{I}}$, then $\mathbf{c} = \tau(d) \in \mathcal{S}_u$, and therefore we have $\Gamma_{\mathcal{M},\mathbf{c},\mathbf{r},C} = 0$ for all $\mathbf{r} \in \mathfrak{R}(\mathcal{B})$ with $r \in \mathbf{r}$. Since $\mathcal{I}$ is a forest model, $d$ cannot have named $r^{\mathcal{I}}$-successors, and thus also $|\{e \in C^{\mathcal{I}} \cap \Delta_n^{\mathcal{I}} \mid (d,e) \in r^{\mathcal{I}}\}| = 0$, which shows that (6) holds for all $d \in \Delta^{\mathcal{I}} = \Delta_u^{\mathcal{I}} \cup \Delta_n^{\mathcal{I}}$.

Since $\{\Delta_u^{\mathcal{I}}, \Delta_n^{\mathcal{I}}\}$ partitions $\Delta^{\mathcal{I}}$, we thus have $\geq n\, r.C \in \mathbf{c}$ iff $d \in (\geq n\, r.C)^{\mathcal{I}}$ iff $|\{e \in C^{\mathcal{I}} \mid (d,e) \in r^{\mathcal{I}}\}| \geq n$ iff

$$\sum_{r \in \mathbf{r} \in \mathfrak{R}(\mathcal{B})} (\nu_{\mathbf{c}}(\Xi_{\mathcal{M},\mathbf{c},\mathbf{r},C}) + \Gamma_{\mathcal{M},\mathbf{c},\mathbf{r},C}) \geq n$$

by (5) and (6), which shows that (3) holds.

Consider now any $E = \exists(r_1 \cap \cdots \cap r_\ell).C \in \mathsf{Con}(\mathcal{B})$. As above, the existence of $\mathbf{d} \in \mathcal{S}_u$ and $\mathbf{r} \in \mathfrak{R}(\mathcal{B})$ such that $C \in \mathbf{d}$, $r_1, \ldots, r_\ell \in \mathbf{r}$, $(\mathbf{c}, \mathbf{d}) \in \mathbf{r}^{\mathcal{R}}$, and $\nu_{\mathbf{c}}(x_{\mathbf{c},\mathbf{r},\mathbf{d}}) = z \geq 1$ implies that both $E \in \mathbf{c}$ and $\nu_{\mathbf{c}}(\Xi_{\mathcal{M},\mathbf{c},\mathbf{r},C}) \geq z \geq 1$, which shows that the corresponding inequation of the form (E3) is satisfied.

Therefore, in the following we can make the same assumption as in the previous case, i.e., that none of these variables is assigned the value $z$. Then (4) holds as before, and thus

$$\sum_{r_1, \ldots, r_\ell \in \mathbf{r} \in \mathfrak{R}(\mathcal{B})} \nu_{\mathbf{c}}(\Xi_{\mathcal{M},\mathbf{c},\mathbf{r},C}) = |\{e \in C^{\mathcal{I}} \cap \Delta_u^{\mathcal{I}} \mid (d,e) \in r_1^{\mathcal{I}} \cap \cdots \cap r_\ell^{\mathcal{I}}\}|.$$

We also have

$$\sum_{r_1,\dots,r_\ell\in\mathbf{r}\in\mathfrak{R}(\mathcal{B})} \Gamma_{\mathcal{M},\mathbf{c},\mathbf{r},C}$$
$$= |\{e \in C^{\mathcal{I}} \cap \Delta_n^{\mathcal{I}} \mid (d,e) \in r_1^{\mathcal{I}} \cap \dots \cap r_\ell^{\mathcal{I}}\}|$$

by similar arguments as in the previous case.

Again, it follows that $E \in \mathbf{c}$ iff $d \in E^{\mathcal{I}}$ iff there is at least one $e \in C^{\mathcal{I}}$ with $(d,e) \in r_1^{\mathcal{I}} \cap \dots \cap r_\ell^{\mathcal{I}}$ iff

$$\sum_{r_1,\dots,r_\ell\in\mathbf{r}\in\mathfrak{R}(\mathcal{B})} (\nu_{\mathbf{c}}(\Xi_{\mathcal{M},\mathbf{c},\mathbf{r},C}) + \Gamma_{\mathcal{M},\mathbf{c},\mathbf{r},C}) \geq 1,$$

which shows that the (in-)equations of the forms (E3) and (E4) are satisfied, and thus $\mathcal{M}$ satisfies Condition (f).

For Condition (g), let $\mathbf{c} \in \mathcal{S}$. Then there must be a $d \in \Delta^{\mathcal{I}}$ with $\tau(d) \subseteq \mathbf{c}$. Since $\mathcal{I}$ respects $\mathcal{D}$, there must be a set $Y \in \mathcal{D}$ such that $d \in (C_Y)^{\mathcal{I}}$. Hence, by definition of $\tau(d)$, we have $Y = \mathbf{c} \cap \{A_1,\dots,A_k\}$.

For Condition (h), let $Y \in \mathcal{D}$. Since $\mathcal{I}$ respects $\mathcal{D}$, there must be a $d \in (C_Y)^{\mathcal{I}}$. Hence, by definition of $\tau(d)$, we have either $Y = \tau(d) \cap \{A_1,\dots,A_k\}$ with $\tau(d) \in \mathcal{S}$ or $Y = (\tau(d) \cup \{a\}) \cap \{A_1,\dots,A_k\}$ with $\tau(d) \cup \{a\} \in \mathcal{S}$ for some $a \in \mathsf{Ind}(\Psi)$. □

**Theorem 6.15.** *Let $\mathcal{B}$ be a Boolean $\mathcal{SHQ}^{\cap}$-knowledge base, let $A_1,\dots,A_k$ be concept names occurring in $\mathcal{B}$, and let $\mathcal{D} \subseteq 2^{\{A_1,\dots,A_k\}}$. Then consistency of $\mathcal{B}$ w.r.t. $\mathcal{D}$ can be decided in time exponential in the size of $\mathcal{B}$.*

Proof. By Lemma 6.14, it suffices to show that the algorithm described in Section 6.3 to find quasimodels for $\mathcal{B}$ that respect $\mathcal{D}$ is sound, complete, and terminates in time exponential in the size of $\mathcal{B}$.

If the algorithm has constructed a model candidate $\mathcal{M} = (\mathcal{S}, \iota, \mathbf{f})$ that passed all tests, then $\mathcal{M}$ obviously satisfies Conditions (a)–(h) of Definition 6.13.

Conversely, if $\mathcal{M} = (\mathcal{S}, \iota, \mathbf{f})$ is a quasimodel of $\mathcal{B}$ that respects $\mathcal{D}$, then $\iota$ and $\mathbf{f}$ must be enumerated by the algorithm at some point. Since $\iota$ and $\mathbf{f}$ satisfy Conditions (b) and (c), they pass the tests in Step 1. In Step 2, a model candidate $\mathcal{M}' := (\mathcal{S}', \iota, \mathbf{f})$ with $\mathcal{S} \subseteq \mathcal{S}'$ is constructed since the concept types in $\mathcal{S}$ satisfy (d) and (g) by assumption. We continue with Step 3, where a model candidate $\mathcal{M}'' := (\mathcal{S}'', \iota, \mathbf{f})$ with $\mathcal{S}'' \subseteq \mathcal{S}'$ is constructed. The systems of equations $E_{\mathcal{M}'',\mathbf{c}}$ for $\mathbf{c} \in \mathcal{S}$ have the same solutions as $E_{\mathcal{M},\mathbf{c}}$—the additional variables for the concept types in $\mathcal{S}'' \setminus \mathcal{S}$ can simply be evaluated to 0. Thus, we know that $\mathcal{S} \subseteq \mathcal{S}''$ and we continue with Step 4. Finally, observe that the concept types needed to satisfy Conditions (a), (e), and (h) are contained in $\mathcal{S}$, and therefore in $\mathcal{S}''$. This shows that the algorithm detects the existence of a quasimodel of $\mathcal{B}$ that respects $\mathcal{D}$.

To analyze the time complexity of the algorithm, observe first that $r$-compatibility w.r.t. $\mathcal{R}$ can be checked in polynomial time since this only involves inclusion tests for sets of polynomial size and entailment tests of role axioms w.r.t. $\mathcal{R}$.

As mentioned before, the number $N$ of model candidates is at most exponential, while each model candidate $(\mathcal{S}_u \cup \mathcal{S}_\iota, \iota, \mathbf{f})$ is of exponential size. For each of these exponentially many model candidates, the checks in Step 1 can be done in polynomial time and the checks in Step 2 are done at most exponentially often since each time one of the exponentially many concept types in $\mathcal{S}$ is removed. Each of these checks can be done in exponential time since the following conditions are checked for at most exponentially many concept types $\mathbf{c}$:

- for (d) we check for inclusion of polynomially many concepts in $\mathbf{c}$;
- for (g), we enumerate all (at most exponentially many) elements of $\mathcal{D}$ and do a simple check.

By similar arguments as above, Step 3 is executed at most exponentially often. Each time this step is performed, for exponentially many concept types $\mathbf{c} \in \mathcal{S}'$ it must be checked whether $E_{\mathcal{M}',\mathbf{c}}$ has a solution. Consider now a concept type $\mathbf{c} \in \mathcal{S}'$, and denote by $n$ the number of variables and by $m$ the number of equations in $E_{\mathcal{M}',\mathbf{c}}$. Note that $n$ may be exponential in the size of $\mathcal{B}$ since there are exponentially many possible concept types and role types. However, $m$ is polynomial since we have one equation per at-least and existential restriction occurring in $\Psi$. In [46], it was shown that $E_{\mathcal{M}',\mathbf{c}}$ can be solved in time $O(n^{2m+2}(ma)^{(m+1)(2m+1)})$, where $a$ is the value of the largest number appearing in the equations. Thus, even if the numbers in at-least restrictions are given in binary encoding, Condition (f) can also be checked in exponential time in the size of $\mathcal{B}$.

Finally, checking (a), (e), and (h) in Step 4 can be done in exponential time by similar arguments as for Step 2. □