Ontology-mediated query answering over temporal and inconsistent data

Camille Bourgaux, Patrick Koopmann, Anni-Yasmin Turhan

▶ **To cite this version:**

Camille Bourgaux, Patrick Koopmann, Anni-Yasmin Turhan. Ontology-mediated query answering over temporal and inconsistent data. Semantic Web – Interoperability, Usability, Applicability, IOS Press, 2019, 10.3233/SW-180337 . hal-02111428

HAL Id: hal-02111428

https://hal.archives-ouvertes.fr/hal-02111428

Submitted on 26 Apr 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Ontology-mediated query answering over temporal and inconsistent data

Camille Bourgaux [a,*], Patrick Koopmann [b] and Anni-Yasmin Turhan [b]

[a] *Laboratoire Traitement et Communication de l'Information, Télécom ParisTech, France*
E-mail: camille.bourgaux@telecom-paristech.fr
[b] *Institute of Theoretical Computer Science, Technische Universität Dresden, Germany*
E-mails: patrick.koopmann@tu-dresden.de, anni-yasmin.turhan@tu-dresden.de

Abstract. Stream-based reasoning systems process data stemming from different sources that are received over time. In this kind of applications, reasoning needs to cope with the temporal dimension and should be resilient against inconsistencies in the data. Motivated by such settings, this paper addresses the problem of handling inconsistent data in a temporal version of ontology-mediated query answering. We consider a recently proposed temporal query language that combines conjunctive queries with operators of propositional linear temporal logic (LTL), and consider these under three inconsistency-tolerant semantics that have been introduced for querying inconsistent description logic knowledge bases. We investigate their complexity for temporal $\mathcal{EL}_\perp$ and DL-Lite$_\mathcal{R}$ knowledge bases. In particular, we consider two different cases, depending on the presence of negations in the query. Furthermore, we complete the complexity picture for the consistent case. We also provide two approaches toward practical algorithms for inconsistency-tolerant temporal query answering.

Keywords: temporal query answering, inconsistency-tolerant reasoning, description logics, computational complexity

1. Introduction

For applications that rely on sensor data, such as context-aware applications, ontologies can enrich and abstract the (numerical) stream data by means of background knowledge. This richer view on the data often results in more query results than over the data alone. Since the collected data usually provides an incomplete description of the observed system, the closed world assumption employed by database systems, where facts not present are assumed to be false, is not appropriate. Most applications that rely on sensor streams observe some kind of running system over time. In order to be able to react to the behaviour of the observed system, they need to employ some representation of temporal information and a query mechanism that can reference this temporal information. If the sources of the collected data are not reliable, as it might be in case of faulty sensors, the internal representation of the observations may contain inconsistencies. In such cases, query mechanisms that rely on logical reasoning are effectively useless, as everything would follow from an inconsistent knowledge base. As a counter measure to this effect, inconsistency-tolerant semantics for answering ontology-mediated queries have been devised. In this paper, we investigate combinations of inconsistency-tolerant and temporal query answering w.r.t. ontologies. This setting is fairly abstract and deliberately neglects implementation details

---

*Corresponding author. E-mail: camille.bourgaux@telecom-paristech.fr.

as it provides foundations on the combination of three aspects vital to most stream reasoning applications. First, stream reasoning requires background knowledge either on the streams used or on the context in which the information provided by the streams is to be handled. Both kinds of background knowledge can be represented by ontologies and be exploited by ontology reasoning systems. Second, stream reasoning inherently has to cope with the temporal dimension. This dimension is given by the sequence of observations that individual items in the streams provide. Thus, traditional reasoning services that are oblivious of the temporal dimension are hardly of use to stream reasoning applications. Naive combinations of such atemporal reasoning services can compute some forms of temporal consequences, but these combinations need not be complete and thus could miss consequences. To stream reasoning applications relying on such ad-hoc systems, it is not discernible when the naive reasoner does or does not provide a reliable answer. Third, and most importantly, the different streams or data sources need not be free of errors due to the methods by which the data is gathered or provided to the system. Moreover, an automatic combination of open stream sources to which a stream reasoning system can subscribe simply cannot guarantee a consistent combination of such sources. Thus, in general, the combination of the data used by a stream reasoning system can simply contain contradictory information. The combination of these three aspects is so far not investigated in the literature. This is why our setting abstracts from many application specific details and variations of stream reasoning systems, in order to concentrate on the combination of the three fundamental conditions inherent to most such systems. Our goal is to provide an initial orientation for selecting between the many design choices by which each of these factors can be addressed in a stream reasoning system, such as expressivity of the ontology and query language, treatment of temporal information, and different inconsistency-tolerant semantics. To this end we map out the landscape of the computational complexity for many combinations of the three design factors for stream reasoning systems in this paper. In preparation for this, we examine the three design aspects in more detail.

In many stream reasoning systems, the collected data is transformed into an abstract logical representation, and situation recognition is performed by some kind of logical inference over the abstract logical representation. There are stream reasoning approaches based on rules, such as answer set programming [1–3], (datalog) rules and approaches based on ontology languages [4–7]. The ontology-based approaches mostly employ the framework of *ontology-mediated queries*, where forms of conjunctive queries are answered over data that is enriched by an ontology, to perform situation recognition. The ontology languages that are investigated for situation recognition are usually those where reasoning is of lower computational complexity in order to obtain systems with low execution times.

In this paper, we investigate the lightweight description logics (DLs) $\mathcal{EL}_\perp$ and DL-Lite$_\mathcal{R}$, for which answering conjunctive queries is tractable (respectively in P and AC^0 w.r.t. the size of the data). The low complexity for query answering in DL-Lite$_\mathcal{R}$ made it the choice for the OWL 2 QL profile [8] in the latest version of OWL [9], the W3C-standardized ontology language for the Semantic Web. For similar reasons, the logic $\mathcal{EL}$ was picked as the core of the OWL 2 EL profile. Both DL-Lite$_\mathcal{R}$ and $\mathcal{EL}_\perp$ admit to use database systems to answer conjunctive queries and are thus good candidates for implementing ontology-based stream reasoning. In DL-Lite$_\mathcal{R}$, the query can be rewritten using the information from the ontology such that the resulting query can be evaluated directly on the data, i.e. stored in a database [10]. For query answering in $\mathcal{EL}$, there are several approaches. In the combined approach [11] the data is augmented in a query-independent way to build a canonical model by adding information form the terminology. The query is then evaluated over this model and unsound answers are filtered out. Alternatively, a translation to datalog variants can be used [12] for query answering in $\mathcal{EL}$ dialects.

In stream reasoning approaches in general, the temporal information is often represented by associating data with the time point at which it was collected. Regarding the language in which queries can be formulated, many variations that capture the temporal aspect have been studied in recent research [2–4, 13]. Window-based approaches admit to concentrate on recent continuous substreams when answering queries over the data, and are the most prominent in implemented systems [2–4]. Ontology-based approaches mostly cover classical temporal logics such as linear temporal logic (LTL) [14] (see [5, 15–18]) or metric variants of temporal logics [19, 20] to enrich the query language. For a recent overview on temporal ontology-mediated querying see [20, 21]. Note that window-based and ontology-based approaches can be

combined. For example, it might often be convenient to first extract a time window of interest and then apply ontology-based query answering to its content. Or alternatively, it is possible to refer to subsequences of the time line using LTL operators, though with a semantics that differs from a standard window operator (see Section 2 for an example.)

Ontology-based approaches for stream reasoning often admit the use of temporal operators only in the query language and use classical ontologies without any temporal operators together with sequences of datasets. Each dataset in such a sequence contains data collected at the same time point. The ontology together with the sequence of datasets constitute the *temporal knowledge base*. Queries can then refer to the different time points by means of temporal operators. This kind of setting has been intensively investigated for *temporal conjunctive queries*, that is, queries with temporal operators from LTL appearing in front of Boolean combinations of atoms, for expressive DLs in [5, 22], for DL-Lite$_\mathcal{R}$ in [6], and for $\mathcal{EL}$ in [23]. We base our study in this paper on this general setting.

For stream reasoning systems, erroneous data can be a severe problem, as for instance pointed out in [24]. If inconsistencies arise in the knowledge base, the logical reasoning mechanisms are rendered useless. There are several directions of research to cope with this problem. While some employ non-monotonic reasoning techniques [25, 26], others try to resolve the inconsistencies directly [27] or perform reasoning with respect to inconsistency-tolerant semantics (see [28] for a recent overview). We follow the latter road, since the techniques developed there are tailored to ontology-mediated queries and often of lower complexity than the other approaches for resolving inconsistencies.

A prominent approach for inconsistency-tolerant reasoning is to consider repairs of the knowledge base, i.e., maximal consistent subsets of the data, and then to perform query answering with respect to these subsets. Arguably the most natural and well-known inconsistency-tolerant semantics is the *AR semantics* [29, 30], inspired by consistent query answering in the database setting [31], which considers the queries that hold in *every repair*. However, AR query answering is intractable even for very simple ontologies [32], which leads the authors of [29, 30] to propose an approximation of AR tractable for DL-Lite$_\mathcal{R}$, namely the *IAR semantics*, which queries the *intersection of the repairs*. Beside its better computational properties, this semantics is more cautious, since it provides answers

supported by facts that are not involved in any contradictions. It may therefore be interesting in our setting when the observed system should change its behaviour only if some situation has been recognized with a very high confidence. Finally, the *brave semantics* [33] returns every answer that holds in *some repair*, so is supported by some consistent set of facts. This less cautious semantics may be relevant for context recognition, when critical situations must imperatively be handled.

For the two DLs to be investigated in this paper, answering of (atemporal) conjunctive queries under these inconsistency-tolerant semantics has already been investigated for DL-Lite$_\mathcal{R}$ in [29, 30, 33, 34] and for $\mathcal{EL}_\perp$ in [28, 35]. Attention has then turned to the problem of designing algorithms and implementing these alternative semantics. Most work has focused on the IAR semantics and dialects of DL-Lite, due to the aforementioned tractability result [30, 36, 37]. A notable exception is the CQAPri system, which implements all three mentioned semantics—AR, IAR and brave—for DL-Lite$_\mathcal{R}$ knowledge bases [38, 39].

So far, inconsistency-tolerant semantics have not been investigated in combination with temporal reasoning. In this paper, we lift inconsistency-tolerant semantics to the case of answering temporal conjunctive queries over lightweight DL temporal knowledge bases.

1.1. Contributions

This article extends the conference paper [40] on temporal query answering in DL-Lite$_\mathcal{R}$ over inconsistent data, where the complexity of answering queries with LTL operators, but without negation, over DL-Lite$_\mathcal{R}$ temporal knowledge bases was investigated. The considered ontologies admit the use of rigid predicates, which are predicates that do not change their interpretation over time. The initial results were obtained for the three inconsistency-tolerant semantics AR, IAR and brave and with respect to three cases of rigid predicates: no rigid predicates, rigid concepts only, and rigid concepts together with rigid roles.

Compared to the conference version, the present article includes new complexity results (all results for $\mathcal{EL}_\perp$, as well as some results for DL-Lite$_\mathcal{R}$). It also extends the set of temporal operators, distinguishing bounded and unbounded variants of the future LTL operators, in order to cover the two different settings that have been investigated for temporal query answering

in the literature, where temporal knowledge bases are interpreted w.r.t. finite or infinite sequences of interpretations. Furthermore, we investigate both cases: the one where negation is admitted in the query language and the one where it is not.

The complexity upper bounds are often obtained by non-deterministic procedures that require, for instance, to guess the right repairs which may not be feasibly computed in practice. Thus algorithms that lend themselves to implementation are still to be devised. We make two contributions toward practical algorithms for temporal inconsistency-tolerant query answering. The first is a polynomial reduction of reasoning in the presence of rigid predicates to reasoning without such predicates by propagating the rigid facts in the sequence of datasets. The second is to identify cases where in the absence of rigid predicates the well-known methods for classical temporal query answering and (atemporal) inconsistency-tolerant query answering can straightforwardly be combined. We show that for the IAR semantics, this yields a sound and complete algorithm. For the AR semantics, such a combination of the algorithms always yields a sound approximation, and additionally yields a sound and complete procedure if the query contains only a restricted set of operators.

This paper is structured as follows. In the next section, we introduce the basic notions of DLs, query answering, inconsistency-tolerant semantics for atemporal knowledge bases and the temporal setting. Furthermore, we discuss known complexity results for both inconsistency-tolerant and temporal query answering in the logics considered. In Section 3, we lift the introduced inconsistency-tolerant semantics to temporal query answering over inconsistent data. Section 4 gives an overview over the complexity results obtained. General versions of algorithms for testing (non-)entailment of temporal conjunctive queries under the different semantics are described in Section 5 in preparation of the complexity analysis. Section 6 shows data and combined complexity of inconsistency-tolerant temporal query answering for DL-Lite$_\mathcal{R}$ and $\mathcal{EL}_\perp$ for the case where the query language admits negation. In Section 7, we complete the complexity picture of temporal query answering under classical semantics by investigating the case where the query does not contain negation. We then built on these results to provide the complexity of inconsistency-tolerant temporal query answering for queries without negation in Section 8. Finally, Section 9 investigates

two approaches for practical implementations that allow to employ well-known methods. The article finishes with a section on conclusions and future work.

To improve readability, some of the proofs have been moved to the appendix and are only sketched in the main text.

2. Preliminaries

We briefly recall the syntax and semantics of DLs and the three inconsistency-tolerant semantics that we consider, and then we introduce our setting of temporal query answering.

Syntax. A DL *knowledge base (KB)* $\mathcal{K}$ consists of an ABox $\mathcal{A}$ and a TBox $\mathcal{T}$, both constructed from three countably infinite sets: a set $\mathsf{N_C}$ of *concept names* (unary predicates), a set $\mathsf{N_R}$ of *role names* (binary predicates), and a set $\mathsf{N_I}$ of *individual names* (constants). The *ABox* (dataset) is a finite set of *concept assertions* $A(a)$ and *role assertions* $R(a, b)$, where $A \in \mathsf{N_C}$, $R \in \mathsf{N_R}$, $a, b \in \mathsf{N_I}$. The *TBox* (ontology) is a finite set of axioms whose form depends on the particular DL.

In DL-Lite$_\mathcal{R}$, TBox axioms are either *concept inclusions* $B \sqsubseteq C$ or *role inclusions* $P \sqsubseteq S$, built according to the following syntax, where $A \in \mathsf{N_C}$ and $R \in \mathsf{N_R}$:

$$B := A \mid \exists P \quad C := B \mid \neg B$$
$$P := R \mid R^- \quad S := P \mid \neg P.$$

Inclusions of the form $B_1 \sqsubseteq B_2$ or $P_1 \sqsubseteq P_2$ are called *positive inclusions* (PI), those of the form $B_1 \sqsubseteq \neg B_2$ or $P_1 \sqsubseteq \neg P_2$ are called *negative inclusions* (NI).

In $\mathcal{EL}_\perp$, the TBox contains concept inclusions of the form $C_1 \sqsubseteq C_2$, where C_1 and C_2 are built according to the following syntax, where $A \in \mathsf{N_C}$ and $R \in \mathsf{N_R}$:

$$C := \top \mid \perp \mid A \mid \exists R.C \mid C \sqcap C.$$

An $\mathcal{EL}_\perp$ inclusion of the form $C_1 \sqcap C_2 \sqsubseteq \perp$ can also be written in the form of a negative inclusion $C_1 \sqsubseteq \neg C_2$.

Semantics. An *interpretation* is a tuple of the form $\mathcal{I} = \langle \Delta^\mathcal{I}, \cdot^\mathcal{I} \rangle$, where $\Delta^\mathcal{I}$ is a non-empty set and $\cdot^\mathcal{I}$ maps each $a \in \mathsf{N_I}$ to $a^\mathcal{I} \in \Delta^\mathcal{I}$, each $A \in \mathsf{N_C}$ to $A^\mathcal{I} \subseteq \Delta^\mathcal{I}$, and each $R \in \mathsf{N_R}$ to $R^\mathcal{I} \subseteq \Delta^\mathcal{I} \times \Delta^\mathcal{I}$. We adopt the unique name assumption, i.e., for all $a, b \in \mathsf{N_I}$, we require $a^\mathcal{I} \neq b^\mathcal{I}$ if $a \neq b$. The function $\cdot^\mathcal{I}$ is straightforwardly extended to general concepts and roles, e.g., $\top^\mathcal{I} = \Delta^\mathcal{I}$, $\perp^\mathcal{I} = \emptyset$, $(R^-)^\mathcal{I} = \{(d, e) \mid (e, d) \in R^\mathcal{I}\}$,

$(\neg R)^{\mathcal{I}} = \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \setminus R^{\mathcal{I}}$, $(\exists P)^{\mathcal{I}} = \{d \mid \exists e : (d,e) \in P^{\mathcal{I}}\}$, $(\exists P.C)^{\mathcal{I}} = \{d \mid \exists e : (d,e) \in P^{\mathcal{I}}, e \in C^{\mathcal{I}}\}$ $(C_1 \sqcap C_2)^{\mathcal{I}} = C_1^{\mathcal{I}} \cap C_2^{\mathcal{I}}$.

An interpretation $\mathcal{I}$ *satisfies* an inclusion $G \sqsubseteq H$ if $G^{\mathcal{I}} \subseteq H^{\mathcal{I}}$; it satisfies $A(a)$ (resp. $R(a,b)$) if $a^{\mathcal{I}} \in A^{\mathcal{I}}$ (resp. $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}}$). We call $\mathcal{I}$ a *model* of a KB $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ if $\mathcal{I}$ satisfies all axioms in $\mathcal{T}$ and all assertions in $\mathcal{A}$. A KB is *consistent* if it has a model, and we say that an ABox $\mathcal{A}$ is $\mathcal{T}$-*consistent* (or simply *consistent* if $\mathcal{T}$ is clear from the context), if the KB $\langle \mathcal{T}, \mathcal{A} \rangle$ is consistent.

Queries. A *conjunctive query* (CQ) takes the form $q = \exists \vec{y}.\psi(\vec{x}, \vec{y})$, where ψ is a conjunction of atoms of the form $A(t)$ or $R(t, t')$, with t, t' individual names or variables from $\vec{x} \cup \vec{y}$. We call the variables in $\vec{x}$ the *free variables in q*. A CQ is called *Boolean* (BCQ) if it has no free variables (i.e., $\vec{x} = \emptyset$). A BCQ q is satisfied by an interpretation $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$, written $\mathcal{I} \models q$, if there is a homomorphism π mapping the variables and individual names of q into $\Delta^{\mathcal{I}}$ such that: $\pi(a) = a^{\mathcal{I}}$ for every $a \in \mathsf{N_I}$, $\pi(t) \in A^{\mathcal{I}}$ for every concept atom $A(t)$ in ψ, and $(\pi(t), \pi(t')) \in R^{\mathcal{I}}$ for every role atom $R(t, t')$ in ψ. A BCQ q is *entailed* from $\mathcal{K}$, written $\mathcal{K} \models q$, iff q is satisfied by every model of $\mathcal{K}$. Given a CQ q with free variables $\vec{x} = (x_1, \ldots, x_k)$ and a tuple of individuals $\vec{a} = (a_1, \ldots, a_k)$, $\vec{a}$ is a *certain answer* to q over $\mathcal{K}$ if $\mathcal{K} \models q(\vec{a})$, where $q(\vec{a})$ is the BCQ resulting from replacing each x_j in q by a_j.

Inconsistency-tolerant semantics. A *repair* of $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ is an inclusion-maximal subset of $\mathcal{A}$ that is $\mathcal{T}$-consistent. We consider three semantics based on repairs that have been previously introduced in the literature [29, 30, 33]. A tuple $\vec{a}$ is an answer to q over $\mathcal{K}$ under

- *AR semantics*, written $\mathcal{K} \models_{\text{AR}} q(\vec{a})$, iff $\langle \mathcal{T}, \mathcal{A}' \rangle \models q(\vec{a})$ for *every repair* $\mathcal{A}'$ of $\mathcal{K}$;
- *IAR semantics*, written $\mathcal{K} \models_{\text{IAR}} q(\vec{a})$, iff $\langle \mathcal{T}, \mathcal{A}^{\cap} \rangle \models q(\vec{a})$ where $\mathcal{A}^{\cap}$ is the *intersection of all repairs* of $\mathcal{K}$;
- *brave semantics*, written $\mathcal{K} \models_{\text{brave}} q(\vec{a})$, iff $\langle \mathcal{T}, \mathcal{A}' \rangle \models q(\vec{a})$ for *some repair* $\mathcal{A}'$ of $\mathcal{K}$.

Figure 1 summarizes the complexity of BCQ entailment under the different semantics for DL-Lite$_{\mathcal{R}}$ and $\mathcal{EL}_{\perp}$. *Data complexity* is measured in the size of the ABox only, while *combined complexity* is measured in the size of the whole KB and the query. When complexity is measured w.r.t. the size of the KB (ABox and TBox), it is called *KB complexity*. For DL-Lite$_{\mathcal{R}}$ and $\mathcal{EL}_{\perp}$, CQ answering under the classical semantics is in P w.r.t. KB complexity. We refer to Section 4 for a reminder on the definitions of the different complexity classes that appear in this work.

Temporal query answering. We now present our temporal framework inspired from [5] and [18].

Definition 2.1 (TKB). A *temporal knowledge base* (TKB) $\mathcal{K} = \langle \mathcal{T}, (\mathcal{A}_i)_{0 \leqslant i \leqslant n} \rangle$ consists of a TBox $\mathcal{T}$ and a finite sequence of ABoxes $(\mathcal{A}_i)_{0 \leqslant i \leqslant n}$. An infinite sequence $\mathcal{J} = (\mathcal{I}_i)_{i \geqslant 0}$ of interpretations $\mathcal{I}_i = \langle \Delta, \cdot^{\mathcal{I}_i} \rangle$ over a fixed non-empty domain Δ (constant domain assumption) is a *model* of $\mathcal{K}$ iff for every $i \in [0, n]$, $\mathcal{I}_i$ is a model of $\langle \mathcal{T}, \mathcal{A}_i \rangle$, for every $i > n$, $\mathcal{I}_i$ is a model of $\mathcal{T}$, and for every $a \in \mathsf{N_I}$ and all $i, j \geqslant 0$, $a^{\mathcal{I}_i} = a^{\mathcal{I}_j}$ (*rigidity* of individual names). *Rigid predicates* are elements from the set $\mathsf{N_{RC}} \subseteq \mathsf{N_C}$ of *rigid concepts* and the set $\mathsf{N_{RR}} \subseteq \mathsf{N_R}$ of *rigid roles*. A sequence of interpretations $\mathcal{J} = (\mathcal{I}_i)_{i \geqslant 0}$ *respects rigid predicates* iff for every $X \in \mathsf{N_{RC}} \cup \mathsf{N_{RR}}$ and all $i, j \geqslant 0$, $X^{\mathcal{I}_i} = X^{\mathcal{I}_j}$. A TKB is *consistent* if it has a model that respects rigid predicates. A sequence of ABoxes $(\mathcal{A}_i)_{0 \leqslant i \leqslant n}$ is $\mathcal{T}$-*consistent*, or simply consistent, if the TKB $\langle \mathcal{T}, (\mathcal{A}_i)_{0 \leqslant i \leqslant n} \rangle$ is consistent.

Example 2.2. The following $\mathcal{EL}_{\perp}$ TKB gives information useful to monitor a cluster of servers. The TBox $\mathcal{T}$ defines that a server is overloaded when it executes a process with an increasing workload while its CPU is already maximally utilized. The sequence of ABoxes $(\mathcal{A}_i)_{0 \leqslant i \leqslant 2}$ records information on a server s. The set of rigid roles is $\mathsf{N_{RR}} = \{\text{hasCPU}\}$, which models the fact that the CPU of a server does not change over time.

$\mathcal{T} = \{\exists \text{executes.IncreasingWorkload} \sqsubseteq \text{LoadRise},$
$\quad\quad \exists \text{hasCPU.MaxUtilized} \sqsubseteq \text{MaxLoaded},$
$\quad\quad \text{LoadRise} \sqcap \text{MaxLoaded} \sqsubseteq \text{Overloaded}\}$

$\mathcal{A}_0 = \{\text{hasCPU}(s, c), \text{MaxUtilized}(c),$
$\quad\quad \text{executes}(s, p_1), \text{IncreasingWorkload}(p_1)\}$

$\mathcal{A}_1 = \{\text{MaxUtilized}(c), \text{executes}(s, p_1),$
$\quad\quad \text{executes}(s, p_2), \text{IncreasingWorkload}(p_2)\}$

$\mathcal{A}_2 = \{\text{executes}(s, p_2)\}$

It is sometimes convenient to represent a sequence of ABoxes as a set of assertions associated with timestamps, which we call *timed assertions*: $(\mathcal{A}_i)_{0 \leqslant i \leqslant n}$ then becomes $\{(\alpha, i) \mid \alpha \in \mathcal{A}_i, 0 \leqslant i \leqslant n\}$.

Semantics	Data complexity		Combined complexity	
	DL-Lite$_{\mathcal{R}}$	$\mathcal{EL}_{\perp}$	DL-Lite$_{\mathcal{R}}$	$\mathcal{EL}_{\perp}$
classical	in AC^0	P-complete	NP-complete	NP-complete
AR	coNP-complete	coNP-complete	Π_2^p-complete	Π_2^p-complete
IAR	in AC^0	coNP-complete	NP-complete	$\Delta_2^p[O(\log n)]$-complete
brave	in AC^0	NP-complete	NP-complete	NP-complete

Figure 1. Complexity of BCQ entailment in DL-Lite$_{\mathcal{R}}$ [29, 33] and $\mathcal{EL}_{\perp}$ [28, 35]

A *rigid concept* is a concept which only uses rigid predicates. A *rigid assertion* is of the form $A(a)$ with $A \in \mathsf{N_{RC}}$ or $R(a,b)$ with $R \in \mathsf{N_{RR}}$. We distinguish three cases depending on which predicates can be rigid: in the first case, we have no rigid predicates ($\mathsf{N_{RC}} = \emptyset$ and $\mathsf{N_{RR}} = \emptyset$), in the second case, we do allow only for rigid concepts ($\mathsf{N_{RC}} \neq \emptyset$ and $\mathsf{N_{RR}} = \emptyset$), and in the last case, we have both, rigid concepts and rigid roles ($\mathsf{N_{RC}} \neq \emptyset$ and $\mathsf{N_{RR}} \neq \emptyset$). Because rigid concepts can be simulated with rigid roles using a pair of concept inclusions of the form $A \sqsubseteq \exists R$, $\exists R \sqsubseteq A$, these three cases cover all interesting combinations.

In the remainder of the paper, we always use n to denote the maximal timestamp of a TKB.

Definition 2.3 (TCQ). *Temporal conjunctive queries* (TCQs) are built from CQs as follows: each CQ is a TCQ, and if ϕ_1 and ϕ_2 are TCQs, then so are $\neg \phi_1$ (negation), $\phi_1 \wedge \phi_2$ (conjunction), $\phi_1 \vee \phi_2$ (disjunction), $\bigcirc \phi_1$ (next), $\bullet^b \phi_1$ (bounded next), $\bigcirc^- \phi_1$ (strong previous), $\bullet^- \phi_1$ (weak previous), $\square \phi_1$ (always), $\square^b \phi_1$ (bounded always), $\square^- \phi_1$ (always in the past), $\diamondsuit \phi_1$ (eventually), $\diamondsuit^b \phi_1$ (bounded eventually), $\diamondsuit^- \phi_1$ (some time in the past), $\phi_1 \mathsf{U} \phi_2$ (until), $\phi_1 \mathsf{U}^b \phi_2$ (bounded until), and $\phi_1 \mathsf{S} \phi_2$ (since). We further may use $\psi_1 \rightarrow \psi_2$ as a shortcut for $\neg \psi_1 \vee \psi_2$.

We impose the constraint that the past operators $\bigcirc^-$, $\square^-$, $\diamondsuit^-$ and S cannot be nested under the unbounded future operators $\bigcirc$, $\square$, $\diamondsuit$ and U (in second position) (see discussion later in this section).

Example 2.4 (Example 2.2 cont'd). The following TCQ returns the list of servers which have been overloaded at least twice.

$$\phi = \diamondsuit^-(\mathsf{Overloaded}(x) \wedge \bigcirc^- \diamondsuit^- \mathsf{Overloaded}(x))$$

Note that we can modify our query to speak about a window of 10 time units as follows:

$$\bigcirc^{-10} \diamondsuit(\mathsf{Overloaded}(x) \wedge \bigcirc \diamondsuit \mathsf{Overloaded}(x))$$

where $\bigcirc^{-10}$ abbreviates a sequence of ten $\bigcirc^-$ operators. Differently to window operators found in other settings however, subexpressions within such a window can refer to time points outside this window.

Given a TCQ ϕ, we refer to the TCQs that occur in ϕ as *subformulas of ϕ*. We sometimes use the notion of the *propositional abstraction* of a Boolean TCQ ϕ, which is the propositional LTL formula obtained by replacing each BCQ in ϕ by a propositional variable (e.g., the propositional abstraction of $\bigcirc \exists x.A(x) \vee \exists x.B(x)$ is the propositional LTL formula $\bigcirc x_1 \vee x_2$).

Remark 2.5 (Choice of operators). The additional LTL operators W (weak until), W^- (weak since), R (release), and R^- (past release) can be expressed by our set of operators as follows:

$$\phi_1 \mathsf{W} \phi_2 \equiv (\phi_1 \mathsf{U} \phi_2) \vee (\square \phi_1),$$
$$\phi_1 \mathsf{W}^- \phi_2 \equiv (\phi_1 \mathsf{S} \phi_2) \vee (\square^- \phi_1),$$
$$\phi_1 \mathsf{R} \phi_2 \equiv \phi_2 \mathsf{W} (\phi_2 \wedge \phi_1), \text{ and}$$
$$\phi_1 \mathsf{R}^- \phi_2 \equiv \phi_2 \mathsf{W}^- (\phi_2 \wedge \phi_1).$$

We will consider in Sections 7, 8 and 9 a special setting where TCQs do not contain negation symbols, which sometimes leads to a lower computational complexity. For this reason, we did not introduce $\square \phi$ as a shortcut for $\neg \diamondsuit \neg \phi$, as it is often done in the literature, but instead treat the operators $\square$ and $\square^b$ as native members of our query language. Similarly, since the top and bottom concepts $\top$ and $\perp$ are not allowed in every DL, $\diamondsuit$ (resp. $\diamondsuit^b$) cannot be defined using U (resp. U^b) as usual in LTL ($\diamondsuit \phi_1 \equiv \mathsf{true} \mathsf{U} \phi_1$), unless we allow for negation (where we can express true using $\exists x.A(x) \vee \neg \exists x.A(x)$). We thus keep all these operators in the set we consider.

Note also that since disjunctions are allowed, TCQs could be defined with unions of conjunctive queries (UCQs) instead of CQs. We use CQs for simplicity.

Definition 2.6 (TCQ answering). Given a TCQ ϕ with free variables $\vec{x} = (x_1, \ldots, x_k)$ and a tuple of individ-

Table 1

Satisfaction of BTCQs by a sequence of interpretations

ϕ	$\mathcal{J}, p \models \phi$ iff
$\exists \vec{y}.\psi(\vec{y})$	$\mathcal{I}_p \models \exists \vec{y}.\psi(\vec{y})$
$\neg\phi_1$	$\mathcal{J}, p \not\models \phi_1$
$\phi_1 \wedge \phi_2$	$\mathcal{J}, p \models \phi_1$ and $\mathcal{J}, p \models \phi_2$
$\phi_1 \vee \phi_2$	$\mathcal{J}, p \models \phi_1$ or $\mathcal{J}, p \models \phi_2$
$\bigcirc\phi_1$	$\mathcal{J}, p+1 \models \phi_1$
$\bullet^b\phi_1$	$p < n$ implies $\mathcal{J}, p+1 \models \phi_1$
$\bigcirc^-\phi_1$	$p > 0$ and $\mathcal{J}, p-1 \models \phi_1$
$\bullet^-\phi_1$	$p > 0$ implies $\mathcal{J}, p-1 \models \phi_1$
$\square\phi_1$	$\forall k, k \geqslant p, \mathcal{J}, k \models \phi_1$
$\square^b\phi_1$	$\forall k, p \leqslant k \leqslant n, \mathcal{J}, k \models \phi_1$
$\square^-\phi_1$	$\forall k, 0 \leqslant k \leqslant p, \mathcal{J}, k \models \phi_1$
$\lozenge\phi_1$	$\exists k, k \geqslant p, \mathcal{J}, k \models \phi_1$
$\lozenge^b\phi_1$	$\exists k, p \leqslant k \leqslant n, \mathcal{J}, k \models \phi_1$
$\lozenge^-\phi_1$	$\exists k, 0 \leqslant k \leqslant p, \mathcal{J}, k \models \phi_1$
$\phi_1 \mathsf{U} \phi_2$	$\exists k, k \geqslant p, \mathcal{J}, k \models \phi_2$ and $\forall j, p \leqslant j < k, \mathcal{J}, j \models \phi_1$
$\phi_1 \mathsf{U}^b \phi_2$	$\exists k, p \leqslant k \leqslant n, \mathcal{J}, k \models \phi_2$ and $\forall j, p \leqslant j < k, \mathcal{J}, j \models \phi_1$
$\phi_1 \mathsf{S} \phi_2$	$\exists k, 0 \leqslant k \leqslant p, \mathcal{J}, k \models \phi_2$ and $\forall j, k < j \leqslant p, \mathcal{J}, j \models \phi_1$

uals $\vec{a} = (a_1, \ldots, a_k)$, $\phi(\vec{a})$ denotes the Boolean TCQ (BTCQ) resulting from replacing each x_j by a_j. A tuple $\vec{a}$ is an answer to ϕ in a sequence of interpretations $\mathcal{J} = (\mathcal{I}_i)_{i \geqslant 0}$ at time point p iff $\mathcal{J}, p \models \phi(\vec{a})$, where the satisfaction of a BTCQ ϕ by a sequence of interpretations $\mathcal{J}$ is defined by induction on its structure as shown in Table 1. A tuple $\vec{a}$ is a *certain answer* to ϕ over $\mathcal{K}$ at time point p, written $\mathcal{K}, p \models \phi(\vec{a})$, iff $\mathcal{J}, p \models \phi(\vec{a})$ for every model $\mathcal{J}$ of $\mathcal{K}$ that respects rigid predicates.

Example 2.7 (Example 2.2 cont'd). It follows from the semantics of TKBs and TCQs that s is a certain answer to the TCQ ϕ of Example 2.4 over the TKB of Example 2.2 at time point 2 and at time point 1, but not at time point 0. Note that the rigidity of the role hasCPU is crucial to get this answer.

In addition to the standard LTL past and unbounded future operators, we introduce four bounded future operators that mimic the semantics based on finite sequences of interpretations used in [18] and similar to that of LTL on finite traces (see e.g., [41]). Indeed, while the standard way of interpreting TKBs is based on infinite sequences of interpretations, it can be relevant to limit the scope of querying to the known time points, especially in the context of data streams. For instance, a user may want to ask whether a server has been running some process since it started ($\lozenge^-(\mathsf{Start}(s) \wedge \square^b \mathsf{executes}(s, p))$), rather than whether it will continue to run this process forever. Moreover, we will see in Section 9.1 that using the bounded se-

mantics can be of practical interest, since it allows us to reduce TCQ answering in the presence of rigid predicates to TCQ answering without rigid predicates. We choose to keep both unbounded and bounded versions of future operators to cover the two settings that have been previously studied for TCQ answering.

The constraint that TCQs should not contain past operators in the scope of unbounded future operators allows us to take advantage of the fact that a TKB entails the same BCQs for every time point $i > n$ to get a lower complexity in the case where there is no negation in the query. Indeed, in this case, the unknown future ($i > n$) can be entirely summarized in one time point $n + 1$. This will also be useful to obtain the data complexity upper bound of brave semantics in the case where there are neither rigid predicates present nor negation in the queries. Moreover, it turns out that for some cases in our analysis, this restriction has no impact on our results. Indeed, [18] shows that Gabbay's separation theorem [42] can be used to rewrite an LTL formula ϕ containing bounded operators into a logically equivalent LTL formula ϕ' that is a Boolean combination of pure-past and pure-future formulas, although with an exponential blow-up. It follows that the restriction we impose does not have any influence over the data complexity of BTCQ entailment, as long as negation is allowed in the query. Moreover, since past operators can still appear in the scope of bounded future operators, all time points that refer to observations having been made so far ($i \leqslant n$) can be referenced by the query language. This query language can thus still express most situations that might be meaningful to detect.

It follows from the definition of certain answers that TCQ answering can be straightforwardly reduced to BTCQ entailment (using polynomially many tests w.r.t. data complexity and exponentially many tests w.r.t. combined complexity). For this reason, we focus on the latter problem.

Figure 2 summarizes the complexity of BTCQ entailment for DL-Lite$_{\mathcal{R}}$ and $\mathcal{EL}_\perp$ in the different cases depending on which kind of predicates are rigid. Our setting is slightly different from those of [6] and [23], because we have additional bounded operators and the restriction that past operators cannot be nested under unbounded future operators. However, the results shown in those papers apply to our setting. Indeed, the proofs for the lower bounds do not use past operators nested under future operators, and for the upper bounds, we argue that it is possible to reduce the en-

Rigid predicates	Data complexity		Combined complexity	
	DL-Lite$_\mathcal{R}$	$\mathcal{EL}_\perp$	DL-Lite$_\mathcal{R}$	$\mathcal{EL}_\perp$
$\mathsf{N_{RC}} = \mathsf{N_{RR}} = \emptyset$	ALOGTIME-complete	P-complete	PSPACE-complete	PSPACE-complete
$\mathsf{N_{RC}} \neq \emptyset, \mathsf{N_{RR}} = \emptyset$	ALOGTIME-complete	coNP-complete	PSPACE-complete	PSPACE-complete
$\mathsf{N_{RC}} \neq \emptyset, \mathsf{N_{RR}} \neq \emptyset$	ALOGTIME-complete	coNP-complete	PSPACE-complete	CONEXPTIME-complete

Figure 2. Complexity of BTCQ entailment in DL-Lite$_\mathcal{R}$ [6] and $\mathcal{EL}_\perp$ [23]

tailment of a BTCQ ϕ that contains bounded operators to the entailment of a BTCQ ϕ' without bounded operators independently from the size of the TKB and linearly w.r.t. the query size. To do that, we add an assertion $\mathsf{end}(a)$ to the last ABox $\mathcal{A}_n$ of the sequence, where end and a are both fresh names, and rewrite the query without unbounded operators using the following equivalences:

$$\bullet^b \phi_1 \equiv \bigcirc\phi_1 \vee \mathsf{end}(a),$$
$$\square^b \phi_1 \equiv \phi_1 \mathsf{U}\,(\mathsf{end}(a) \wedge \phi_1),$$
$$\diamondsuit^b \phi_1 \equiv \neg\mathsf{end}(a)\mathsf{U}\,\phi_1, \text{ and}$$
$$\phi_1 \mathsf{U}^b \phi_2 \equiv (\phi_1 \wedge \neg\mathsf{end}(a))\mathsf{U}\,\phi_2.$$

3. Temporal query answering over inconsistent data

We extend the three inconsistency-tolerant semantics to temporal query answering. The main difference to the atemporal case is that in the presence of rigid concepts or roles, a TKB $\mathcal{K} = \langle \mathcal{T}, (\mathcal{A}_i)_{0 \leqslant i \leqslant n}\rangle$ may be inconsistent even if each KB $\langle \mathcal{T}, \mathcal{A}_i\rangle$ is consistent. Indeed, in this case, there may not be a sequence of interpretations $\mathcal{J} = (\mathcal{I}_i)_{i \geqslant 0}$ such that $\mathcal{I}_i$ is a model of $\langle \mathcal{T}, \mathcal{A}_i\rangle$ for every $i \in [0, n]$ and which also respects rigid predicates. That is why we have to consider as repairs the $\mathcal{T}$-consistent sequences of subsets of the initial ABoxes that are component-wise maximal.

Definition 3.1 (Repair of a TKB). A *repair* of a TKB $\mathcal{K} = \langle \mathcal{T}, (\mathcal{A}_i)_{0 \leqslant i \leqslant n}\rangle$ is a sequence of ABoxes $(\mathcal{A}'_i)_{0 \leqslant i \leqslant n}$ such that $\{(\alpha, i) \mid \alpha \in \mathcal{A}'_i, 0 \leqslant i \leqslant n\}$ is a maximal $\mathcal{T}$-consistent subset of $\{(\alpha, i) \mid \alpha \in \mathcal{A}_i, 0 \leqslant i \leqslant n\}$. We denote the set of repairs of $\mathcal{K}$ by $Rep(\mathcal{K})$.

The next example illustrates the impact which rigid predicates can have on repairs.

Example 3.2. Consider the TKB $\mathcal{K} = \langle \mathcal{T}, (\mathcal{A}_i)_{0 \leqslant i \leqslant 1}\rangle$. The TBox expresses that web servers and application servers are two distinct kinds of servers, and the

ABoxes provide information about a server a that executes two processes b and c.

$$\mathcal{T} = \{\mathsf{WebServer} \sqsubseteq \mathsf{Server}, \ \mathsf{AppServer} \sqsubseteq \mathsf{Server},$$
$$\mathsf{WebServer} \sqsubseteq \neg\mathsf{AppServer}\}$$
$$\mathcal{A}_0 = \{\mathsf{WebServer}(a), \ \mathsf{executes}(a, b)\}$$
$$\mathcal{A}_1 = \{\mathsf{AppServer}(a), \mathsf{WebServer}(a), \mathsf{executes}(a, c)\}$$

Assume that no predicate is rigid. The TKB $\mathcal{K}$ is inconsistent because the timed assertions $(\mathsf{AppServer}(a), 1)$ and $(\mathsf{WebServer}(a), 1)$ violate the negative inclusion in $\mathcal{T}$. Specifically, $\mathsf{AppServer}(a)$ and $\mathsf{WebServer}(a)$ cannot both be true at time point 1. It follows that $\mathcal{K}$ has two repairs that correspond to the two different ways of restoring consistency: $(\mathcal{A}'_i)_{0 \leqslant i \leqslant 1}$ and $(\mathcal{A}''_i)_{0 \leqslant i \leqslant 1}$, where

$$\mathcal{A}'_0 = \mathcal{A}''_0 = \mathcal{A}_0$$
$$\mathcal{A}'_1 = \{\mathsf{AppServer}(a), \mathsf{executes}(a, c)\}$$
$$\mathcal{A}''_1 = \{\mathsf{WebServer}(a), \mathsf{executes}(a, c)\}.$$

Now assume that $\mathsf{AppServer}$ is rigid. There is then a new reason for $\mathcal{K}$ being inconsistent: the timed assertions $(\mathsf{WebServer}(a), 0)$ and $(\mathsf{AppServer}(a), 1)$ violate the negative inclusion of $\mathcal{T}$ due to the rigidity of $\mathsf{AppServer}$, which implies that $\mathsf{AppServer}(a)$ and $\mathsf{WebServer}(a)$ should be both entailed at time point 0. Therefore, $\mathcal{K}$ has now the two repairs $(\mathcal{A}'_i)_{0 \leqslant i \leqslant 1}$ and $(\mathcal{A}''_i)_{0 \leqslant i \leqslant 1}$, where

$$\mathcal{A}'_0 = \{\mathsf{executes}(a, b)\}$$
$$\mathcal{A}'_1 = \{\mathsf{AppServer}(a), \mathsf{executes}(a, c)\}$$
$$\mathcal{A}''_0 = \mathcal{A}_0$$
$$\mathcal{A}''_1 = \{\mathsf{WebServer}(a), \mathsf{executes}(a, c)\}.$$

Note that even though $(\mathcal{A}'_i)_{0 \leqslant i \leqslant 1}$ is maximal (adding $\mathsf{WebServer}(a)$ to $\mathcal{A}'_0$ renders the TKB inconsistent), $\mathcal{A}'_0$ is not a repair of $\langle \mathcal{T}, \mathcal{A}_0\rangle$, because it is not maximal.

We extend the semantics AR, IAR, and brave to the temporal case in the natural way by regarding sequences of ABoxes.

Definition 3.3 (AR, IAR, brave semantics for TCQs). A tuple $\vec{a}$ is an *answer* to a TCQ ϕ over a TKB $\mathcal{K} = \langle \mathcal{T}, (\mathcal{A}_i)_{0 \leqslant i \leqslant n} \rangle$ at time point p under

– AR semantics, written $\mathcal{K}, p \models_{\text{AR}} \phi(\vec{a})$,
  iff $\langle \mathcal{T}, (\mathcal{A}'_i)_{0 \leqslant i \leqslant n} \rangle, p \models \phi(\vec{a})$ for every repair $(\mathcal{A}'_i)_{0 \leqslant i \leqslant n}$ of $\mathcal{K}$;
– IAR semantics, written $\mathcal{K}, p \models_{\text{IAR}} \phi(\vec{a})$,
  iff $\langle \mathcal{T}, (\mathcal{A}_i^{\text{ir}})_{0 \leqslant i \leqslant n} \rangle, p \models \phi(\vec{a})$,
  with $\mathcal{A}_i^{\text{ir}} = \bigcap_{(\mathcal{A}'_j)_{0 \leqslant j \leqslant n} \in Rep(\mathcal{K})} \mathcal{A}'_i$ for all $i \in [0, n]$;
– brave semantics, written $\mathcal{K}, p \models_{\text{brave}} \phi(\vec{a})$,
  iff $\langle \mathcal{T}, (\mathcal{A}'_i)_{0 \leqslant i \leqslant n} \rangle, p \models \phi(\vec{a})$ for some repair $(\mathcal{A}'_i)_{0 \leqslant i \leqslant n}$ of $\mathcal{K}$.

The following relationships between the semantics, which already hold in the atemporal case, are implied by their definition:

$$\mathcal{K}, p \models_{\text{IAR}} \phi(\vec{a}) \Rightarrow \mathcal{K}, p \models_{\text{AR}} \phi(\vec{a}) \Rightarrow \mathcal{K}, p \models_{\text{brave}} \phi(\vec{a})$$

In the following example, we illustrate the effect of the different semantics in the temporal case.

Example 3.4 (Example 3.2 cont'd). Consider the following three temporal conjunctive queries.

$$\phi_1 = \Box^b (\exists y. \text{executes}(x, y))$$

$$\phi_2 = \Box^b (\exists y. \text{Server}(x) \wedge \text{executes}(x, y))$$

$$\phi_3 = \Box^b (\exists y. \text{AppServer}(x) \wedge \text{executes}(x, y))$$

If there are no rigid predicates, the intersection of the repairs is $(\mathcal{A}_i^{\text{ir}})_{0 \leqslant i \leqslant 1}$, with $\mathcal{A}_0^{\text{ir}} = \mathcal{A}_0$ and $\mathcal{A}_1^{\text{ir}} = \{\text{executes}(a, c)\}$. We have $\mathcal{K}, 0 \models_{\text{IAR}} \phi_1(a)$, because in every model of the intersection of the repairs a executes b at time point 0 and c at time point 1. For ϕ_2, $\mathcal{K}, 0 \models_{\text{AR}} \phi_2(a)$, since every model of every repair assigns a to WebServer at time point 0 and to either AppServer (in models of $(\mathcal{A}'_i)_{0 \leqslant i \leqslant 1}$) or WebServer (in models of $(\mathcal{A}''_i)_{0 \leqslant i \leqslant 1}$) at time point 1. However, $\mathcal{K}, 0 \not\models_{\text{IAR}} \phi_2(a)$. Finally, $\mathcal{K}, 0 \not\models_{\text{brave}} \phi_3(a)$, because no repair entails AppServer(a) at time point 0.

If AppServer is rigid, the intersection of the repairs is $(\mathcal{A}_i^{\text{ir}})_{0 \leqslant i \leqslant 1}$ with $\mathcal{A}_0^{\text{ir}} = \{\text{executes}(a, b)\}$ and $\mathcal{A}_1^{\text{ir}} = \{\text{executes}(a, c)\}$. So, still $\mathcal{K}, 0 \models_{\text{IAR}} \phi_1(a)$. Since every model of every repair assigns a to Server at time points 0 and 1 (either because a is a web server or because a is an application server), $\mathcal{K}, 0 \models_{\text{AR}} \phi_2(a)$,

but $\mathcal{K}, 0 \not\models_{\text{IAR}} \phi_2(a)$. Finally, $\mathcal{K}, 0 \models_{\text{brave}} \phi_3(a)$, because every model of $\langle \mathcal{T}, (\mathcal{A}'_i)_{0 \leqslant i \leqslant 1} \rangle$ assigns a to AppServer at any time point by rigidity of AppServer, but $\mathcal{K}, 0 \not\models_{\text{AR}} \phi_3(a)$.

We conclude this section by pointing out some characteristics of the case without rigid predicates that will be useful later. If there are no rigid predicates, the interpretations $\mathcal{I}_i$ of a model $\mathcal{J} = (\mathcal{I}_i)_{i \geqslant 0}$ of $\mathcal{K}$ that respects rigid predicates are independent, besides the interpretation of the individual names. We thus obtain the following proposition.

Proposition 3.5. *If* $\mathsf{N}_{\text{RC}} = \mathsf{N}_{\text{RR}} = \emptyset$, *then a TKB* $\mathcal{K} = \langle \mathcal{T}, (\mathcal{A}_i)_{0 \leqslant i \leqslant n} \rangle$ *is consistent iff every* $\langle \mathcal{T}, \mathcal{A}_i \rangle$ *is consistent. Moreover, if* $\mathcal{K}$ *is consistent, for every* $p \in [0, n]$, $\mathcal{I}'_p$ *is a model of* $\langle \mathcal{T}, \mathcal{A}_p \rangle$ *iff there exists a model* $\mathcal{J} = (\mathcal{I}_i)_{i \geqslant 0}$ *of* $\mathcal{K}$ *such that* $\mathcal{I}_p = \mathcal{I}'_p$.

Proof. If $\mathsf{N}_{\text{RC}} = \mathsf{N}_{\text{RR}} = \emptyset$, a sequence of interpretations $\mathcal{J} = (\mathcal{I}_i)_{i \geqslant 0}$ is a model of $\mathcal{K}$ that respects rigid predicates iff it is a model of $\mathcal{K}$. It follows that $\mathcal{K}$ is consistent iff there exists $\mathcal{J} = (\mathcal{I}_i)_{i \geqslant 0}$ such that for every $i \in [0, n]$, $\mathcal{I}_i$ is a model of $\langle \mathcal{T}, \mathcal{A}_i \rangle$, for every $i > n$, $\mathcal{I}_i$ is a model of $\mathcal{T}$, and for every $a \in \mathsf{N}_{\text{I}}$ and all $i, j \geqslant 0$, $a^{\mathcal{I}_i} = a^{\mathcal{I}_j}$. This already establishes one direction of the proposition. We show that also the other direction holds.

Assume that every $\langle \mathcal{T}, \mathcal{A}_i \rangle$ is consistent, i.e., every $\langle \mathcal{T}, \mathcal{A}_i \rangle$ has a model. Let $p \in [0, n]$ be any fixed index and $\mathcal{I}'_p = \langle \Delta^{\mathcal{I}'_p}, \cdot^{\mathcal{I}'_p} \rangle$ be a model of $\langle \mathcal{T}, \mathcal{A}_p \rangle$. Since we adopt the unique name assumption and N_{I} is unbounded, $\Delta^{\mathcal{I}'_p}$ is an infinite set. Moreover, for every $i \in [0, n]$, since $\mathcal{A}_i$ is finite, by the Löwenheim-Skolem theorem, there exists a model $\mathcal{I}''_i = \langle \Delta^{\mathcal{I}''_i}, \cdot^{\mathcal{I}''_i} \rangle$ of $\langle \mathcal{T}, \mathcal{A}_i \rangle$ such that $\Delta^{\mathcal{I}''_i}$ is countable. Thus $\Delta^{\mathcal{I}''_i}$ can be mapped to $\Delta^{\mathcal{I}'_p}$ by an injective function f_i. For every i, we can then build a model $\mathcal{I}'_i = \langle \Delta^{\mathcal{I}'_i}, \cdot^{\mathcal{I}'_i} \rangle$ of $\langle \mathcal{T}, \mathcal{A}_i \rangle$ such that $\Delta^{\mathcal{I}'_i} = \Delta^{\mathcal{I}'_p}$. For this, let $\Delta = \Delta^{\mathcal{I}'_p} \backslash f_i(\Delta^{\mathcal{I}''_i})$ and select some domain element $d_1 \in \Delta^{\mathcal{I}''_i}$. We define a mapping $m_i : \Delta^{\mathcal{I}''_i} \rightarrow 2^{\Delta^{\mathcal{I}'_p}}$ by setting $m_i(d) = \{f_i(d)\}$ for $d \neq d_1$ and $m_i(d_1) = \{f_i(d_1)\} \cup \Delta$. It remains to set $a^{\mathcal{I}'_i} = f_i(a^{\mathcal{I}''_i})$ for all $a \in \mathsf{N}_{\text{I}}$, $A^{\mathcal{I}'_i} = \bigcup_{d \in A^{\mathcal{I}''_i}} m_i(d)$, and $R^{\mathcal{I}'_i} = \{(d_1, d_2) \in \Delta^{\mathcal{I}'_i} \times \Delta^{\mathcal{I}'_i} \mid (d'_1, d'_2) \in R^{\mathcal{I}''_i}, d_1 \in m_i(d'_1), d_2 \in m_i(d'_2)\}$. Since $\Delta^{\mathcal{I}'_i} = \Delta^{\mathcal{I}'_p}$ for every $i \in [0, n]$, there exists a bijection $b_i : \Delta^{\mathcal{I}'_i} \rightarrow \Delta^{\mathcal{I}'_p}$ such that for all $a \in \mathsf{N}_{\text{I}}$, $b_i(a^{\mathcal{I}'_i}) = a^{\mathcal{I}'_p}$.

Using these bijections, we can construct a model $\mathcal{J} = (\mathcal{I}_i)_{i \geqslant 0}$ of $\mathcal{K}$ based on the interpretations $\mathcal{I}'_0, \ldots \mathcal{I}'_n$ as follows. For every $i > n$, $\mathcal{I}_i = \mathcal{I}'_p$,

$\mathcal{I}_p = \mathcal{I}'_p$, and for every $i \in [0,n]$ such that $i \neq p$, $\mathcal{I}_i = \langle \Delta^{\mathcal{I}'_p}, \cdot^{\mathcal{I}_i} \rangle$, where $\cdot^{\mathcal{I}_i}$ is defined by $a^{\mathcal{I}_i} = a^{\mathcal{I}'_p}$ for every $a \in \mathsf{N}_\mathsf{I}$, $A^{\mathcal{I}_i} = \{b_i(d) \mid d \in A^{\mathcal{I}'_i}\}$ for every $A \in \mathsf{N}_\mathsf{C}$, and $R^{\mathcal{I}_i} = \{(b_i(d_1), b_i(d_2)) \mid (d_1, d_2) \in R^{\mathcal{I}'_i}\}$. Note that this construction ensures $a^{\mathcal{I}_i} = a^{\mathcal{I}_j}$ for all $i, j \geqslant 0$. Furthermore, one easily establishes that for all $i \in [0,n]$, $\mathcal{I}_i$ is a model of $\langle \mathcal{T}, \mathcal{A}_i \rangle$, and for $i > n$, $\mathcal{I}_i$ is a model of $\mathcal{T}$. We obtain that $\mathcal{J}$ is a model of $\mathcal{K}$ such that $\mathcal{I}_p = \mathcal{I}'_p$. $\square$

It follows that, if $\mathsf{N}_\mathsf{RC} = \mathsf{N}_\mathsf{RR} = \emptyset$, CQs can be answered at time point p by answering them over the KB $\langle \mathcal{T}, \mathcal{A}_p \rangle$.

Proposition 3.6. *If* $\mathsf{N}_\mathsf{RC} = \mathsf{N}_\mathsf{RR} = \emptyset$*, then for every BCQ* $q = \exists \vec{y}.\psi(\vec{y})$ *and* $p \in [0,n]$*,* $\mathcal{K}, p \models q$ *iff* $\langle \mathcal{T}, \mathcal{A}_p \rangle \models q$*.*

Proof. $\mathcal{K}, p \models q$ iff for every model $\mathcal{J} = (\mathcal{I}_i)_{i \geqslant 0}$ of $\mathcal{K}$ that respects rigid predicates, $\mathcal{I}_p \models q$. By Proposition 3.5, this is the case iff for every model $\mathcal{I}_p$ of $\langle \mathcal{T}, \mathcal{A}_p \rangle$, $\mathcal{I}_p \models q$, which is equivalent to $\langle \mathcal{T}, \mathcal{A}_p \rangle \models q$. $\square$

4. Complexity analysis overview

In the next four sections, we investigate the complexity of inconsistency-tolerant BTCQ entailment in DL-Lite$_\mathcal{R}$ and $\mathcal{EL}_\perp$. Apart from the different DLs, we also consider two settings of query languages: in the first setting, all TCQs as defined in Section 2 are considered, in the second setting, we analyze the complexity for TCQs that do not use the negation operator. Furthermore, we investigate complexity depending on whether rigid symbols are present or absent and which of the 4 semantics are used (classical, AR, IAR or brave). All these investigations are carried out for combined or data complexity.

For classical semantics, some complexities have been investigated earlier for the different settings we consider. For the case where *negations are allowed* in the queries, the complexity of BTCQ entailment under the classical semantics has been studied in [6] for DL-Lite$_\mathcal{R}$ and in [23] for $\mathcal{EL}$ (cf. Section 2, Figure 2). Furthermore, it has also been shown in [18, 43] that in DL-Lite$_\mathcal{R}$, TCQs *without negation* (and with a bounded future semantics) can be rewritten into FO-queries for temporal databases, but this was shown only in a restricted setting without rigid names and only for *rooted* TCQs. In this case, rootedness of a CQ means that it contains at least a constant or an answer

variable and all its terms are connected via chains of role atoms. A TCQ is rooted if every CQ in it is rooted.

Most of our complexity upper bounds are based on a set of general algorithms for BTCQ entailment under the different inconsistency-tolerant semantics, which we present in Section 5. Those allow us to obtain complexity upper bounds for the different settings based on the complexity of BTCQ entailment under classical semantics, on the complexity of recognizing repairs, and on the complexity of consistency checking. In Section 6, we establish the complexity of these basic tasks, and give complexity results for our two DLs of interest, $\mathcal{EL}_\perp$ and DL-Lite$_\mathcal{R}$, regarding both data and combined complexity. In the cases where the general algorithms are insufficient to give tight bounds, we try to provide specialized algorithms.

We then restrict the query language and study the complexity of the entailment of BTCQs without negation. In Section 7, we first investigate this case under classical semantics, and observe that in some cases, disallowing negation leads to lower worst case complexities, even if we alleviate the limitations imposed in [18, 43]. In Section 8 these lower complexities allow us to also improve the complexity bounds for inconsistency-tolerant reasoning when there is no negation in the TCQs. Furthermore, we take advantage of the absence of negation to tighten an upper bound for brave semantics without rigid predicates.

Synopsis of complexity classes. We recall the definitions of the complexity classes that appear in the following sections.

- P: problems solvable in polynomial time.
- NP: problems solvable in non-deterministic polynomial time.
- coNP: problems whose complement is in NP.
- $\Delta_2^p[O(\log n)]$: problems solvable in polynomial time with at most logarithmically many calls to an NP oracle.
- Σ_2^p: problems solvable in non-deterministic polynomial time with an NP oracle.
- Π_2^p: problems whose complement is in Σ_2^p.
- AC^0: problems that can be solved by a uniform family of circuits of constant depth and polynomial size, with unbounded-fanin AND and OR gates. We have $\mathsf{AC}^0 \subseteq \mathsf{P}$.
- ALogTime: problems solvable in logarithmic time by a random access alternating Turing machine. We have $\mathsf{AC}^0 \subseteq$ ALogTime $\subseteq \mathsf{P}$.
- PSPACE: problems solvable in polynomial space.
- ExpTime: problems solvable in exponential time.

- NEXPTIME: problems solvable in non-deterministic exponential time.
- CONEXPTIME: problems whose complement is in NEXPTIME.

For the remainder of this paper, $\mathcal{L}$ is a DL which is interpreted w.r.t. standard interpretations, as defined in Section 2. We will consider in particular the cases $\mathcal{L} = \text{DL-Lite}_{\mathcal{R}}$ and $\mathcal{L} = \mathcal{EL}_{\perp}$. Furthermore, we assume $\mathcal{K} = \langle \mathcal{T}, (\mathcal{A}_i)_{0 \leqslant i \leqslant n} \rangle$ to be the TKB we evaluate our query against, and ϕ to be the considered query.

5. General algorithms for inconsistency-tolerant BTCQ entailment

Our complexity bounds are based on a set of general algorithms for deciding BTCQ entailment under the different semantics, which are inspired from known algorithms for inconsistency-tolerant BCQ entailment in the atemporal case (see e.g., [35]). Depending on the semantics under consideration, it might be more natural to describe either a procedure for entailment or one for non-entailment, which is why these algorithms focus on different directions of the problem.

Non-entailment under AR semantics. The procedure ARNonEntailment decides whether ϕ is *not* entailed by $\mathcal{K}$ at time point p under AR semantics, and is defined as follows.

1. Guess a sequence $(\mathcal{A}'_i)_{0 \leqslant i \leqslant n} \subseteq (\mathcal{A}_i)_{0 \leqslant i \leqslant n}$ of ABoxes.
2. Verify that $(\mathcal{A}'_i)_{0 \leqslant i \leqslant n}$ is a repair of $\mathcal{K}$ and that $\langle \mathcal{T}, (\mathcal{A}'_i)_{0 \leqslant i \leqslant n} \rangle, p \not\models \phi$.

It is common to focus on non-entailment when studying the complexity of AR semantics, since this procedure needs to consider only one repair instead of all of them.

Entailment under brave semantics. The procedure braveEntailment decides whether ϕ is entailed by $\mathcal{K}$ at time point p under brave semantics, and is defined as follows.

1. Guess a sequence $(\mathcal{A}'_i)_{0 \leqslant i \leqslant n} \subseteq (\mathcal{A}_i)_{0 \leqslant i \leqslant n}$ of ABoxes.
2. Verify that $(\mathcal{A}'_i)_{0 \leqslant i \leqslant n}$ is a repair of $\mathcal{K}$ and that $\langle \mathcal{T}, (\mathcal{A}'_i)_{0 \leqslant i \leqslant n} \rangle, p \models \phi$.

Non-entailment under IAR semantics. The procedure IARNonEntailment decides whether ϕ is *not* entailed by $\mathcal{K}$ at time point p under IAR semantics, and is defined as follows.

1. Guess
    (a) a set $\mathcal{B} = \{(\alpha_1, i_1), \ldots, (\alpha_m, i_m)\} \subseteq (\mathcal{A}_i)_{0 \leqslant i \leqslant n}$ of timed assertions, together with
    (b) m subsets of the data $(\mathcal{A}'^1_i)_{0 \leqslant i \leqslant n} \subseteq (\mathcal{A}_i)_{0 \leqslant i \leqslant n}$, $\ldots, (\mathcal{A}'^m_i)_{0 \leqslant i \leqslant n} \subseteq (\mathcal{A}_i)_{0 \leqslant i \leqslant n}$ such that for every $j \in [1, m]$, $\alpha_j \notin \mathcal{A}'^j_{i_j}$.
2. Verify that
    (a) for every $j \in [1, m]$, $(\mathcal{A}'^j_i)_{0 \leqslant i \leqslant n}$ is a repair of $\mathcal{K}$, and
    (b) $\langle \mathcal{T}, (\mathcal{A}_i)_{0 \leqslant i \leqslant n} \backslash \mathcal{B} \rangle, p \not\models \phi$.

Note that $m \leqslant |(\mathcal{A}_i)_{0 \leqslant i \leqslant n}|$, Step 2a therefore has to verify only a linear number of repairs. We show that the algorithm decides non-entailment under IAR semantics. Indeed, if for every (α_j, i_j) there exists a repair $(\mathcal{A}'^j_i)_{0 \leqslant i \leqslant n}$ of $\mathcal{K}$ that does not contain (α_j, i_j), then (α_j, i_j) is not in the intersection of the repairs of $\mathcal{K}$. Thus $(\mathcal{A}_i)_{0 \leqslant i \leqslant n} \backslash \{(\alpha_1, i_1), \ldots, (\alpha_m, i_m)\}$ is a superset of the intersection $(\mathcal{A}^{\text{ir}}_i)_{0 \leqslant i \leqslant n}$ of the repairs of $\mathcal{K}$. It follows that if $\langle \mathcal{T}, (\mathcal{A}_i)_{0 \leqslant i \leqslant n} \backslash \mathcal{B} \rangle, p \not\models \phi$, then $\mathcal{K}, p \not\models_{\text{IAR}} \phi$. For the other direction, assume that $\mathcal{K}, p \not\models_{\text{IAR}} \phi$, and let $\mathcal{B} = \{(\alpha_1, i_1), \ldots, (\alpha_m, i_m)\} = (\mathcal{A}_i)_{0 \leqslant i \leqslant n} \backslash (\mathcal{A}^{\text{ir}}_i)_{0 \leqslant i \leqslant n}$. For each (α_j, i_j), there exists a repair $(\mathcal{A}'^j_i)_{0 \leqslant i \leqslant n}$ of $\mathcal{K}$ that does not contain the timed assertion (α_j, i_j), and $\langle \mathcal{T}, (\mathcal{A}_i)_{0 \leqslant i \leqslant n} \backslash \mathcal{B} \rangle, p \not\models \phi$.

Similarly to the AR case, we use this procedure for non-entailment instead of a procedure for entailment. Deciding non-entailment allows us to consider a linear number of repairs instead of all of them, which can be exponentially many. However, for the IAR semantics, a direct procedure for IAR entailment is useful for showing combined complexity results for the DLs we consider. This procedure is based on the computation of the set of timed assertions that belong to some minimal inconsistent subset of the TKB.

Entailment under IAR semantics. We give an alternative procedure for IAR, IAREntailment, which decides whether ϕ is entailed under IAR semantics at time point p.

1. For every $(\alpha, j) \in (\mathcal{A}_i)_{0 \leqslant i \leqslant n}$, check whether (α, j) belongs to a minimal inconsistent subset of $\mathcal{K}$ by asking an oracle whether there exists a $\mathcal{T}$-inconsistent set of timed assertions $\mathcal{B} \subseteq (\mathcal{A}_i)_{0 \leqslant i \leqslant n}$ such that $\mathcal{B} \backslash \{(\alpha, j)\}$ is consistent.

2. Call the oracle to determine whether ϕ is entailed at time point p by the TKB from which all timed assertions that belong to some minimal inconsistent subset have been removed.

We show that the intersection of the repairs of $\mathcal{K}$ is obtained by removing the minimal inconsistent subsets of $\mathcal{K}$. Let $\mathcal{B} \subseteq (\mathcal{A}_i)_{0 \leqslant i \leqslant n}$ be a minimal inconsistent subset of $\mathcal{K}$ and $(\alpha, i) \in \mathcal{B}$. Since $\mathcal{B} \backslash \{(\alpha, i)\}$ is consistent, (α, i) is not in the repairs that contain $\mathcal{B} \backslash \{(\alpha, i)\}$. In the other direction, if a timed assertion (α, i) does not appear in some repair $(\mathcal{A}'_i)_{0 \leqslant i \leqslant n}$ of $\mathcal{K}$, since the repairs are maximal, $(\mathcal{A}'_i)_{0 \leqslant i \leqslant n} \cup \{(\alpha, i)\}$ is inconsistent, and (α, i) occurs in some minimal inconsistent subset of $\mathcal{K}$.

6. Complexity of inconsistency-tolerant BTCQ entailment with negation in the query

In this section, we investigate the complexity of BTCQ entailment for general BTCQs, that is, BTCQs that may contain negation. For this, we first establish the complexities of consistency checking and repair recognition, i.e., the task of deciding whether a sequence of ABoxes is a repair of $\mathcal{K}$. We then build on these results to prove the complexity of inconsistency-tolerant temporal query entailment using the general algorithms described in the last section, while showing matching lower bounds. We thus obtain the following theorem.

Theorem 6.1. *The results in Figure 3 hold.*

6.1. Consistency checking and repair recognition for TKBs

We reduce these tasks to the atemporal case by defining an atemporal KB $\widetilde{\mathcal{K}}$ based on $\mathcal{K}$. For $\mathcal{K} = \langle \mathcal{T}, (\mathcal{A}_i)_{0 \leqslant i \leqslant n} \rangle$, $\widetilde{\mathcal{K}} = \langle \widetilde{\mathcal{T}}, \widetilde{\mathcal{A}} \rangle$ is defined in Figure 4. We first show a correspondence between the models of $\mathcal{K}$ that respect rigid predicates and the models of $\widetilde{\mathcal{K}}$.

Lemma 6.2. $\mathcal{K}$ *is consistent iff* $\widetilde{\mathcal{K}}$ *is consistent.*

Proof. ($\Leftarrow$) We construct a function temp from the models of $\widetilde{\mathcal{K}}$ to those of $\mathcal{K}$ that respect rigid predicates. Assume $\widetilde{\mathcal{K}}$ is consistent, and let $\widetilde{\mathcal{I}}$ be a model of $\widetilde{\mathcal{K}}$. We define $\mathsf{temp}(\widetilde{\mathcal{I}}) = \mathcal{J} = (\mathcal{I}_i)_{i \geqslant 0}$ as follows. For every $i \in [0, n]$, we set

– $a^{\mathcal{I}_i} = a^{\widetilde{\mathcal{I}}}$ for every $a \in \mathsf{N_I}$,

– $A^{\mathcal{I}_i} = A^{\widetilde{\mathcal{I}}}$ for every $A \in \mathsf{N_{RC}}$,
– $R^{\mathcal{I}_i} = R^{\widetilde{\mathcal{I}}}$ for every $R \in \mathsf{N_{RR}}$,
– $A^{\mathcal{I}_i} = A_i^{\widetilde{\mathcal{I}}}$ for every $A \in \mathsf{N_C} \backslash \mathsf{N_{RC}}$, and
– $R^{\mathcal{I}_i} = R_i^{\widetilde{\mathcal{I}}}$ for every $R \in \mathsf{N_R} \backslash \mathsf{N_{RR}}$,

and for every $i > n$, we set

– $a^{\mathcal{I}_i} = a^{\widetilde{\mathcal{I}}}$ for every $a \in \mathsf{N_I}$,
– $A^{\mathcal{I}_i} = A^{\widetilde{\mathcal{I}}}$ for every $A \in \mathsf{N_{RC}}$,
– $R^{\mathcal{I}_i} = R^{\widetilde{\mathcal{I}}}$ for every $R \in \mathsf{N_{RR}}$,
– $A^{\mathcal{I}_i} = A_{n+1}^{\widetilde{\mathcal{I}}}$ for every $A \in \mathsf{N_C} \backslash \mathsf{N_{RC}}$, and
– $R^{\mathcal{I}_i} = R_{n+1}^{\widetilde{\mathcal{I}}}$ for every $R \in \mathsf{N_R} \backslash \mathsf{N_{RR}}$.

We show that the sequence $\mathsf{temp}(\widetilde{\mathcal{I}})$ of interpretations is a model of $\mathcal{K}$ that respects rigid predicates.

1. For every time-point $i \in [0, n]$, $\mathcal{I}_i$ is a model of $\mathcal{A}_i$. If $A(a) \in \mathcal{A}_i$, then either $A \in \mathsf{N_{RC}}$ and $A(a) \in \widetilde{\mathcal{A}}$, or $A \notin \mathsf{N_{RC}}$ and $A_i(a) \in \widetilde{\mathcal{A}}$. In both cases, $a^{\mathcal{I}_i} = a^{\widetilde{\mathcal{I}}} \in A^{\mathcal{I}_i}$. The same argument holds for the role assertions in $\mathcal{A}_i$.
2. For every $i \in [0, n + 1]$, $\mathcal{I}_i$ is a model of $\mathcal{T}$. Slightly abusing notation and viewing an interpretation as an infinite set of assertions, we denote by $\mathsf{RenameNotRig}(\mathcal{I}_i, i)$ the interpretation obtained from $\mathcal{I}_i$ by renaming every non-rigid predicate X by X_i. The interpretations of all rigid predicates and of all A_i and R_i are the same in $\mathsf{RenameNotRig}(\mathcal{I}_i, i)$ and $\widetilde{\mathcal{I}}$. Since $\widetilde{\mathcal{I}}$ is a model of $\widetilde{\mathcal{T}}$, and $\widetilde{\mathcal{T}}$ does not contain any axiom that involves two non-rigid predicates X_i and X_j with $i \neq j$, $\mathsf{RenameNotRig}(\mathcal{I}_i, i)$ is a model of $\widetilde{\mathcal{T}}$. Moreover $\mathsf{RenameNotRig}(\mathcal{T}, i) \subseteq \widetilde{\mathcal{T}}$, and therefore $\mathsf{RenameNotRig}(\mathcal{I}_i, i)$ is a model of $\mathsf{RenameNotRig}(\mathcal{T}, i)$. Hence, $\mathcal{I}_i$ is a model of $\mathcal{T}$.
3. For every $i > n + 1$, $\mathcal{I}_i = \mathcal{I}_{n+1}$ is a model of $\mathcal{T}$.
4. For every $i \geqslant 0$, for every $A \in \mathsf{N_{RC}}$, $A^{\mathcal{I}_i} = A^{\widetilde{\mathcal{I}}}$, and for every $R \in \mathsf{N_{RC}}$, $R^{\mathcal{I}_i} = R^{\widetilde{\mathcal{I}}}$. Therefore, $\mathcal{J}$ respects rigid predicates.

We obtain that $\mathsf{temp}(\widetilde{\mathcal{I}})$ is a model of $\mathcal{K}$ that respects rigid predicates.

($\Rightarrow$) For the other direction, we construct a function atemp from the models of $\mathcal{K}$ that respect rigid predicates to those of $\widetilde{\mathcal{K}}$. Assume $\mathcal{K}$ is consistent, and let $\mathcal{J} = (\mathcal{I}_i)_{i \geqslant 0}$ be a model of $\mathcal{K}$ that respects rigid predicates. We define $\mathsf{atemp}(\mathcal{J}) = \widetilde{\mathcal{I}}$ as follows.

– $a^{\widetilde{\mathcal{I}}} = a^{\mathcal{I}_0}$ ($= a^{\mathcal{I}_i}$ for every $i \geqslant 0$) for every $a \in \mathsf{N_I}$,
– $A^{\widetilde{\mathcal{I}}} = A^{\mathcal{I}_0}$ for every $A \in \mathsf{N_{RC}}$,
– $R^{\widetilde{\mathcal{I}}} = R^{\mathcal{I}_0}$ for every $R \in \mathsf{N_{RR}}$,
– $A_i^{\widetilde{\mathcal{I}}} = A^{\mathcal{I}_i}$ for every $A \in \mathsf{N_C} \backslash \mathsf{N_{RC}}$ and $i \in [0, n]$, and

	Data complexity				Combined complexity			
	classical	AR	IAR	brave	classical	AR	IAR	brave
$\mathcal{EL}_\perp$								
$N_{RC} = N_{RR} = \emptyset$	P	coNP	coNP	NP	PSPACE	PSPACE	PSPACE	PSPACE
$N_{RC} \neq \emptyset, N_{RR} = \emptyset$	coNP	coNP	coNP	Σ_2^p	PSPACE	PSPACE	PSPACE	PSPACE
$N_{RC} \neq \emptyset, N_{RR} \neq \emptyset$	coNP	coNP	coNP	Σ_2^p	CONEXPTIME	CONEXPTIME	CONEXPTIME	CONEXPTIME
DL-Lite$_\mathcal{R}$								
$N_{RC} = N_{RR} = \emptyset$	ALOGTIME	coNP	in P	in NP	PSPACE	PSPACE	PSPACE	PSPACE
$N_{RC} \neq \emptyset, N_{RR} = \emptyset$	ALOGTIME	coNP	in P	NP	PSPACE	PSPACE	PSPACE	PSPACE
$N_{RC} \neq \emptyset, N_{RR} \neq \emptyset$	ALOGTIME	coNP	in P	NP	PSPACE	PSPACE	PSPACE	PSPACE

Figure 3. Data [left] and combined [right] complexity of BTCQ entailment for BTCQs with negations. All complexities are tight, except those preceded by "in", which are upper bounds. The results for the classical semantics are shown in [23] for $\mathcal{EL}_\perp$ and in [6] for DL-Lite$_\mathcal{R}$.

$$\widetilde{\mathcal{T}} = \bigcup_{i=0}^{n+1} \mathsf{RenameNotRig}(\mathcal{T}, i)$$

$$\widetilde{\mathcal{A}} = \bigcup_{i=0}^{n} \mathsf{RenameNotRig}(\mathcal{A}_i, i)$$

where for every set of axioms $\mathcal{O}$, the function $\mathsf{RenameNotRig}(\mathcal{O}, i)$ substitutes every non-rigid predicate X by X_i in every axiom $\alpha \in \mathcal{O}$.

Figure 4. KB $\widetilde{\mathcal{K}} = \langle \widetilde{\mathcal{T}}, \widetilde{\mathcal{A}} \rangle$ representing $\mathcal{K} = \langle \mathcal{T}, (\mathcal{A}_i)_{0 \leqslant i \leqslant n} \rangle$.

– $R_i^{\widetilde{\mathcal{I}}} = R^{\mathcal{I}_i}$ for every $R \in N_R \backslash N_{RR}$ and $i \in [0, n]$.

Again, we show that $\widetilde{\mathcal{I}}$ is a model of $\widetilde{\mathcal{K}}$ by considering the ABox and the TBox separately.

1. $\widetilde{\mathcal{I}}$ is a model of $\widetilde{\mathcal{A}}$. If $A(a) \in \widetilde{\mathcal{A}}$ with $A \in N_{RC}$, then $a^{\widetilde{\mathcal{I}}} = a^{\mathcal{I}_0} \in A^{\widetilde{\mathcal{I}}}$, and if $A_i(a) \in \widetilde{\mathcal{A}}$ for some $A \notin N_{RC}$, then $A(a) \in \mathcal{A}_i$ and $a^{\widetilde{\mathcal{I}}} = a^{\mathcal{I}_i} \in A^{\widetilde{\mathcal{I}}}$. The situation is the same for the role assertions in $\widetilde{\mathcal{A}}$.

2. $\widetilde{\mathcal{I}}$ is a model of $\widetilde{\mathcal{T}}$. If we rename the non-rigid predicates, $\mathsf{RenameNotRig}(\mathcal{I}_i, i)$ coincides with $\widetilde{\mathcal{I}}$ on the interpretation of all rigid predicates and all A_i and R_i. Since each $\mathcal{I}_i$ is a model of $\mathcal{T}$, each interpretation $\mathsf{RenameNotRig}(\mathcal{I}_i, i)$ is a model of $\mathsf{RenameNotRig}(\mathcal{T}, i)$, and since $\widetilde{\mathcal{T}}$ does not contain any axiom that involves two non-rigid predicates X_i and X_j with $i \neq j$, each $\mathsf{RenameNotRig}(\mathcal{I}_i, i)$ is a model of $\widetilde{\mathcal{T}}$. It follows that $\widetilde{\mathcal{I}}$ is a model of $\widetilde{\mathcal{T}}$.

We thus obtain a direct correspondence between the models of $\mathcal{K}$ and those of $\widetilde{\mathcal{K}}$, and that $\mathcal{K}$ is satisfiable iff $\widetilde{\mathcal{K}}$ is satisfiable. □

It follows that consistency checking of TKBs can be polynomially reduced to consistency checking of KBs.

Lemma 6.3. *If for a DL $\mathcal{L}$, consistency checking of $\mathcal{L}$-KBs is in P, then consistency checking of $\mathcal{L}$-TKBs is in P as well.*

Proof. By Lemma 6.2, the TKB $\mathcal{K}$ is consistent iff the KB $\widetilde{\mathcal{K}}$ is consistent. If consistency checking is in P for $\mathcal{L}$-KBs, the consistency of $\mathcal{K}$ can then be checked in time polynomial in the size of $\widetilde{\mathcal{T}}$ and $\widetilde{\mathcal{A}}$. Since the size of $\widetilde{\mathcal{T}}$ is polynomial in $|\mathcal{T}|$ and n, and the size of $\widetilde{\mathcal{A}}$ is at most the size of $(\mathcal{A}_i)_{0 \leqslant i \leqslant n}$, we obtain that TKB consistency checking is in P. □

We next show that repair recognition can be done with a polynomial number of consistency checks.

Lemma 6.4. *If for a DL $\mathcal{L}$, consistency checking of $\mathcal{L}$-TKBs is in P, then repair recognition, i.e., deciding whether a sequence of ABoxes $(\mathcal{A}'_i)_{0 \leqslant i \leqslant n}$ is a repair of $\mathcal{K}$, is in P.*

Proof. Assume consistency checking of $\mathcal{L}$-TKBs is in P. Then, we can verify in P whether a sequence of ABoxes $(\mathcal{A}'_i)_{0 \leqslant i \leqslant n}$ is a repair of $\mathcal{K}$ as follows.

1. For every i, check that $\mathcal{A}'_i \subseteq \mathcal{A}_i$.
2. Check that $(\mathcal{A}'_i)_{0 \leqslant i \leqslant n}$ is $\mathcal{T}$-consistent.
3. For every $(\alpha, j) \in (\mathcal{A}_i)_{0 \leqslant i \leqslant n} \backslash (\mathcal{A}'_i)_{0 \leqslant i \leqslant n}$, check that $(\mathcal{A}'_i)_{0 \leqslant i \leqslant n} \cup \{(\alpha, j)\}$ is $\mathcal{T}$-inconsistent. □

Note that Lemmas 6.3 and 6.4 apply to DL-Lite$_\mathcal{R}$ and $\mathcal{EL}_\perp$ alike.

6.2. Combined complexity

We are now ready to establish the complexity of BTCQ entailment under inconsistency-tolerant semantics. We start with the combined complexity. The following upper bounds follow straightforwardly from the procedures described in Section 5.

Proposition 6.5. *If repair recognition is in* P *and BTCQ entailment under the classical semantics is in* PSPACE *w.r.t. combined complexity, then BTCQ entailment under AR, IAR, and brave semantics is in* PSPACE *w.r.t. combined complexity.*

Proof. If verifying that a sequence of ABoxes is a repair is in P and verifying the entailment, and thus also verifying the non-entailment, of a BTCQ is in PSPACE, then the procedures ARNonEntailment, IARNonEntailment and braveEntailment all run in NPSPACE=PSPACE. □

Proposition 6.5 applies to DL-Lite$_\mathcal{R}$ and $\mathcal{EL}_\perp$ in all cases except for $\mathcal{EL}_\perp$ with rigid roles, for which BTCQ entailment under classical semantics is CO-NEXPTIME-hard [23].

Proposition 6.6. *BTCQ entailment from an $\mathcal{EL}_\perp$-TKB under AR, IAR, and brave semantics is in* CONEXP-TIME *w.r.t. combined complexity, even if* $N_{RR} \neq \emptyset$.

Proof. For the AR and IAR semantics, we modify the procedures ARNonEntailment and IARNonEntailment described in Section 5 so that they also guess a certificate of the non-entailment of ϕ in the first step. Then, in the second step, the non-entailment of ϕ can be decided by simply verifying this certificate. The certificate can be checked in EXPTIME, since the non-entailment of ϕ can be decided in NEXPTIME.

For the brave semantics' upper bound, we give a NEXPTIME procedure to decide $\mathcal{K}, p \not\models_{\text{brave}} \phi$. For every subset $(\mathcal{A}'_i)_{0 \leqslant i \leqslant n}$ of $(\mathcal{A}_i)_{0 \leqslant i \leqslant n}$, guess either "not a repair" or a certificate of the non-entailment of ϕ from $\langle \mathcal{T}, (\mathcal{A}'_i)_{0 \leqslant i \leqslant n} \rangle$ at time point p. Note that there are $2^{|(\mathcal{A}_i)_{0 \leqslant i \leqslant n}|}$ such subsets. For every such subset, verify in EXPTIME whether it is indeed not a repair, or whether $\langle \mathcal{T}, (\mathcal{A}'_i)_{0 \leqslant i \leqslant n} \rangle, p \not\models \phi$. □

The matching PSPACE and CONEXPTIME combined complexity lower bounds for $\mathcal{EL}_\perp$ and DL-Lite$_\mathcal{R}$ follow from the consistent case (cf. Section 2).

6.3. Data complexity for $\mathcal{EL}_\perp$-TKBs

We now prove the data complexity results, starting with $\mathcal{EL}_\perp$. We first consider the case without rigid predicates.

Proposition 6.7. *BTCQ entailment from an $\mathcal{EL}_\perp$-TKB with* $N_{RC} = N_{RR} = \emptyset$ *is*

- *co*NP-*complete w.r.t. data complexity under AR and IAR semantics, and*
- NP-*complete w.r.t. data complexity under brave semantics.*

Proof. The upper bounds follow from the procedures described in Section 5: since verifying that a sequence of ABoxes is a repair as well as verifying the non-entailment and entailment of a BTCQ take polynomial time w.r.t. data complexity, the procedures ARNonEntailment, IARNonEntailment, and braveEntailment run in NP w.r.t. data complexity. The lower bounds follow from the atemporal case. □

Next, we prove the complexity of BTCQ entailment with rigid predicates. The following proposition establishes the upper bounds for the case where both rigid concepts and rigid roles are allowed.

Proposition 6.8. *BTCQ entailment from an $\mathcal{EL}_\perp$-TKB with* $N_{RC} \neq \emptyset$ *and* $N_{RR} \neq \emptyset$ *is*

- *in co*NP *w.r.t. data complexity under AR and IAR semantics, and*
- *in* Σ_2^p *w.r.t. data complexity under brave semantics.*

Proof. For AR and IAR semantics, we modify the procedures described in Section 5 to also guess a certificate for the non-entailment of ϕ. This certificate can be checked in P, since the non-entailment of ϕ can be decided in NP. The upper bound for brave semantics is obtained using the procedure braveEntailment described in Section 5. □

We show that these results are tight even if we only have rigid concepts.

Proposition 6.9. *BTCQ entailment from an $\mathcal{EL}_\perp$-TKB with* $N_{RC} \neq \emptyset$ *is*

- *co*NP-*hard w.r.t. data complexity under AR and IAR semantics, and*
- Σ_2^p-*hard w.r.t. data complexity under brave semantics.*

Proof (Sketch). The lower bounds for AR and IAR semantics follow from the atemporal case. For brave semantics, we show that the complement of brave TCQ entailment is Π_2^p-hard by reduction from $\text{QBF}_{2,\forall}$.

Let $\Phi = \forall x_1 \ldots x_m \exists y_1 \ldots y_r \varphi$ be a $\text{QBF}_{2,\forall}$-formula, where $\varphi = \bigwedge_{i=0}^{h} \ell_i^0 \vee \ell_i^1 \vee \ell_i^2$ is a 3-CNF formula over the propositional variables $\{x_1, \ldots, x_m, y_1, \ldots, y_r\}$. Based on Φ, define the TKB $\mathcal{K} = \langle \mathcal{T}, (\mathcal{A}_i)_{0 \leqslant i \leqslant 3h+2} \rangle$ and the TCQ ϕ as follows, where $\mathsf{N}_{\mathsf{RC}} = \{\mathsf{T}\}$.

$$\mathcal{T} = \{\exists \mathsf{Pos}.\mathsf{T} \sqsubseteq \mathsf{Sat}, \exists \mathsf{Neg}.\mathsf{F} \sqsubseteq \mathsf{Sat},$$
$$\exists \mathsf{FromPos}.\mathsf{Sat} \sqsubseteq \mathsf{T}, \exists \mathsf{FromNeg}.\mathsf{Sat} \sqsubseteq \mathsf{F},$$
$$\exists \mathsf{FromY}.\mathsf{Sat} \sqsubseteq \mathsf{T}, \mathsf{T} \sqcap \mathsf{F} \sqsubseteq \bot,$$
$$\mathsf{T} \sqcap \exists \mathsf{ValY}.\mathsf{T} \sqsubseteq \bot\}$$
$$\phi = \neg \square^b (\mathsf{NotFirst}(c) \vee \mathsf{Sat}(c) \vee$$
$$\bigcirc \mathsf{Sat}(c) \vee \bigcirc\bigcirc \mathsf{Sat}(c))$$

For each clause $\ell_i^0 \vee \ell_i^1 \vee \ell_i^2$, i.e., for every $i \in [0, h]$, we define the following three ABoxes $\mathcal{A}_{3i+k}$ ($0 \leqslant k \leqslant 2$):

$$\mathcal{A}_{3i+0} = \mathcal{B} \cup \mathcal{B}_{3i+0}$$
$$\mathcal{A}_{3i+k} = \mathcal{B} \cup \mathcal{B}_{3i+k} \cup \{\mathsf{NotFirst}(c)\}, 1 \leqslant k \leqslant 2,$$

where

$$\mathcal{B} = \{\mathsf{T}(x_j), \mathsf{F}(x_j) \mid 1 \leqslant j \leqslant m\} \cup$$
$$\{\mathsf{ValY}(y_j, \neg y_j), \mid 1 \leqslant j \leqslant r\}$$
$$\mathcal{B}_{3i+k} = \{\mathsf{Pos}(c, x_j), \mathsf{FromPos}(x_j, c)\} \text{ if } \ell_i^k = x_j$$
$$\mathcal{B}_{3i+k} = \{\mathsf{Neg}(c, x_j), \mathsf{FromNeg}(x_j, c)\} \text{ if } \ell_i^k = \neg x_j$$
$$\mathcal{B}_{3i+k} = \{\mathsf{FromY}(y_j, c)\} \text{ if } \ell_i^k = y_j$$
$$\mathcal{B}_{3i+k} = \{\mathsf{FromY}(\neg y_j, c)\} \text{ if } \ell_i^k = \neg y_j.$$

We use the following claim, which we prove in detail in the appendix, and give the intuition behind the construction here.

Claim. Φ is valid iff $\mathcal{K}, 0 \not\models_{\text{brave}} \phi$.

Recall that (i) Φ is valid iff for every valuation of the x_j, there exists a valuation of the y_j that satisfies φ, and that (ii) $\mathcal{K}, 0 \not\models_{\text{brave}} \phi$ iff every repair of $\mathcal{K}$ has a model in which ϕ does not hold. The correspondence between valuations of variables of Φ and models of repairs of $\mathcal{K}$ is ensured by $\mathcal{B}$. Intuitively, the rigid concept T is used to encode that a variable is true. The disjoint concept F encodes that a variable is false and is

used to enforce that the repairs of $\mathcal{K}$ correspond to the valuations of the x_j. The role ValY connects every pair of $y_j, \neg y_j$, which ensures together with the inclusion $\mathsf{T} \sqcap \exists \mathsf{ValY}.\mathsf{T} \sqsubseteq \bot$ that no model of a repair of $\mathcal{K}$ assigns both y_j and $\neg y_j$ to the rigid concept T. As a consequence, each model of a repair of $\mathcal{K}$ corresponds to a valuation of the y_j.

Each clause $\ell_i^0 \vee \ell_i^1 \vee \ell_i^2$ is represented by the three consecutive ABoxes $\mathcal{A}_{3i+0}, \mathcal{A}_{3i+1}$ and $\mathcal{A}_{3i+2}$, which each encode a literal of the clause in the $\mathcal{B}_{3i+k}$ part. The concept $\mathsf{NotFirst}$ is used to indicate that the literal represented is not the first of the clause. The concept Sat encodes the satisfaction of a clause, and the query expresses that it is not the case that every clause is satisfied (the disjunction is true iff the clause is satisfied by at least one of its three literals). Finally, the TBox expresses the conditions under which a clause is satisfied. In particular, the first two inclusions enforce that a clause is always satisfied if it contains a literal x_j (resp. $\neg x_j$) assigned to true (resp. to false) in the repair, and the next three inclusions ensure that a clause is satisfied only if it contains a literal which is assigned to true (for the $\neg x_j$, since the literals x_j and $\neg x_j$ are represented with only one individual x_j, the condition is that x_j is assigned to false). $\qquad \square$

6.4. Data complexity for DL-Lite$_\mathcal{R}$-TKBs

It remains to show the data complexity results for DL-Lite$_\mathcal{R}$. For AR and brave semantics, the upper bounds follow from the guess and check procedures described in Section 5. (Recall that the complexity of BTCQ entailment under classical semantics is ALOG-TIME-complete for this DL, and thus in P.)

The lower bound for AR follows from the atemporal case, which establishes a tight coNP-bound even if $\mathsf{N}_{\mathsf{RC}} = \mathsf{N}_{\mathsf{RR}} = \emptyset$.

In contrast, for brave semantics, BCQ entailment is tractable in the atemporal setting. However, we cannot directly extend this result to the temporal case. Indeed, the data complexity upper bound for brave CQ answering in DL-Lite$_\mathcal{R}$ relies on the fact that the size of the minimal sets of assertions that support the query is bounded by the query size. This is not true in the temporal setting, as can already be seen by the query $\phi = \square^- A(a)$, whose entailment at time point p can depend on p assertions in the TKB. In fact, we show that in the presence of rigid concepts, brave BTCQ entailment becomes NP-hard.

Proposition 6.10. *If $\mathsf{N}_{\mathsf{RC}} \neq \emptyset$, then brave BTCQ entailment from DL-Lite$_\mathcal{R}$-TKBs is NP-hard w.r.t. data*

complexity. This already holds for BTCQs that do not contain negation.

Proof. We show NP-hardness of brave BTCQ entailment from DL-Lite$_\mathcal{R}$-TKBs by reduction from SAT. Let $c_0 \wedge ... \wedge c_n$ be a CNF formula over variables $x_1, ..., x_m$. We define the following problem of BTCQ entailment under brave semantics, with two rigid concepts T and F. Let $\mathcal{K} = \langle \mathcal{T}, (\mathcal{A}_i)_{0 \leqslant i \leqslant n} \rangle$ be such that:

$$\mathcal{T} = \{\exists \mathsf{Pos} \sqsubseteq \mathsf{Sat}, \exists \mathsf{Neg} \sqsubseteq \mathsf{Sat},$$
$$\exists \mathsf{Pos}^- \sqsubseteq \mathsf{T}, \exists \mathsf{Neg}^- \sqsubseteq \mathsf{F}, \mathsf{T} \sqsubseteq \neg \mathsf{F}\}$$
$$\mathcal{A}_i = \{\mathsf{Pos}(a, x_j) \mid x_j \in c_i\} \cup$$
$$\{\mathsf{Neg}(a, x_j) \mid \neg x_j \in c_i\} \text{ for } 0 \leqslant i \leqslant n$$

Let $\phi = \square^- \mathsf{Sat}(a)$. We show that $c_0 \wedge ... \wedge c_n$ is satisfiable iff $\mathcal{K}, n \models_{\mathrm{brave}} \phi$. Note that ϕ does not contain negation. Indeed, since T and F are rigid, a repair $(\mathcal{A}'_i)_{0 \leqslant i \leqslant n}$ of $\mathcal{K}$ is such that each x_j has either only Pos or only Neg incoming edges in $(\mathcal{A}'_i)_{0 \leqslant i \leqslant n}$, i.e., x_j occurs either only in assertions of the form $\mathsf{Pos}(a, x_j)$ or only in assertions of the form $\mathsf{Neg}(a, x_j)$. We can thus define a valuation ν of the variables such that $\nu(x_j) = \mathsf{true}$ if $(\mathcal{A}'_i)_{0 \leqslant i \leqslant n}$ does not contain a timed assertion of the form $(\mathsf{Neg}(c, x_j), k)$, and $\nu(x_j) = \mathsf{false}$ otherwise. The clause c_i is satisfied by ν iff there exists x_j such that either $x_j \in c_i$ and $\nu(x_j) = \mathsf{true}$ or $\neg x_j \in c_i$ and $\nu(x_j) = \mathsf{false}$, that is, iff there exists x_j such that either $\mathsf{Pos}(a, x_j) \in \mathcal{A}'_i$ or $\mathsf{Neg}(a, x_j) \in \mathcal{A}'_i$, which holds exactly iff $\langle \mathcal{T}, (\mathcal{A}'_i)_{0 \leqslant i \leqslant n}\rangle, i \models \mathsf{Sat}(a)$. It follows that $c_0 \wedge ... \wedge c_n$ is satisfiable iff there exists a repair $(\mathcal{A}'_i)_{0 \leqslant i \leqslant n}$ of $\mathcal{K}$ that entails ϕ at time point n. $\square$

For the case $\mathsf{N}_{\mathsf{RC}} = \mathsf{N}_{\mathsf{RR}} = \emptyset$, we have an ALOG-TIME lower bound from the classical semantics, and it is open whether the NP upper bound for brave semantics can be improved. The challenge in obtaining a polynomial upper bound for BTCQ entailment under brave semantics lies intuitively in the non-trivial interaction between the brave semantics and the negation in the query (see also end of Section 8). While under brave semantics, it is more natural to focus on the positive problem of query entailment (one has to construct some subset of a repair), for entailment of BTCQs, it is more natural to focus on the negative problem of query non-entailment (one tries to find some counterexample for the entailment of the query [6]). While for both problems polynomial algorithms have been found, naive combinations of these approaches are not possible. However, since we did not manage to prove

an NP lower bound either, it might still be possible to solve this problem in polynomial time.

In contrast, for IAR semantics, we can give a tractable upper bound even if $\mathsf{N}_{\mathsf{RR}} \neq \emptyset$. The reason is that, in DL-Lite$_\mathcal{R}$-TKBs, the size of a minimal inconsistent subset is at most two, as in the atemporal case.

Fact 6.11 ([29], Lemma 1). Due to the DL-Lite$_\mathcal{R}$ syntax, the following holds: for every DL-Lite$_\mathcal{R}$ TBox $\mathcal{T}$, the size of a minimal $\mathcal{T}$-inconsistent set of (timed) assertions is at most two.

Therefore, we can always compute the intersection of all repairs in polynomial time.

Proposition 6.12. *BTCQ entailment from a DL-Lite$_\mathcal{R}$-TKB under IAR semantics is in P w.r.t. data complexity, even if $\mathsf{N}_{\mathsf{RR}} \neq \emptyset$ and $\mathsf{N}_{\mathsf{RC}} \neq \emptyset$.*

Proof. The size of the minimal $\mathcal{T}$-inconsistent subsets of $(\mathcal{A}_i)_{0 \leqslant i \leqslant n}$ is bounded by 2. We can thus skip the first step of the procedure IAREntailment described in Section 5 and compute the minimal inconsistent subsets in P by checking the consistency of every timed assertion and pair of timed assertions (with a quadratic number of consistency checks), and then verify the entailment of the query in P w.r.t. data complexity over the TKB from which they have been removed. $\square$

7. BTCQ entailment under classical semantics without negation in the query

This section completes the complexity picture for BTCQ entailment under the classical semantics by investigating the case where TCQs do not use negation. We show that the absence of negation in the query induces a complexity drop in several cases. These results are based on a more general property: we show that for any DL $\mathcal{L}$, if $\mathcal{L}$ has the canonical model property for CQ answering over KBs, then $\mathcal{L}$ has also the canonical model property for TCQ answering over TKBs for TCQs without negation. We use the canonical model to prove that for the classical semantics, the complexity upper bounds of the atemporal case transfer to the temporal case.

We first show that BCQ entailment from a TKB $\mathcal{K}$ can be reduced to BCQ entailment from the KB $\widetilde{\mathcal{K}}$ defined Section 6, Figure 4. For this, we define a similar transformation for BCQs as we did for TKBs. Let

$q = \exists \vec{y}.\psi(\vec{y})$ be a BCQ and $p \geqslant 0$ be a time point. Consider

$$\widetilde{q}_p = \mathsf{RenameNotRig}(q, p)$$

where $\mathsf{RenameNotRig}(q, p)$ replaces every non-rigid predicate X in q by X_p if $p \leqslant n$, and by X_{n+1} otherwise.

Lemma 7.1. $\mathcal{K}, p \models q$ iff $\widetilde{\mathcal{K}} \models \widetilde{q}_p$.

Proof. The following proof is written for the case where $p \in [0, n + 1]$. For the case $p > n + 1$, p is replaced by $n + 1$ in the predicates names.

Assume that $\mathcal{K}, p \models q$, and let $\widetilde{\mathcal{I}}$ be a model of $\widetilde{\mathcal{K}}$. Let $\mathcal{J} = (\mathcal{I}_i)_{i \geqslant 0} = \mathsf{temp}(\widetilde{\mathcal{I}})$ be the corresponding model of $\mathcal{K}$ that respects rigid predicates, as defined in the proof for Lemma 6.2. For any BCQ ψ without existential variables, we denote by $\mathsf{atoms}(\psi)$ the set of (ground) atoms of ψ. Since $\mathcal{I}_p \models q$, there then exists a mapping π from the set of constants and variables that appear in ψ into Δ such that for every $X(\vec{d}) \in \mathsf{atoms}(\psi_\pi)$, where ψ_π is the BCQ obtained by replacing the terms of ψ by their image by π, we have $\vec{d} \in X^{\mathcal{I}_p}$. It follows that for every $X(\vec{d}) \in \mathsf{atoms}(\psi_\pi)$, if X is rigid then $\vec{d} \in X^{\widetilde{\mathcal{I}}}$, and otherwise $\vec{d} \in X_p^{\widetilde{\mathcal{I}}}$. Thus, $\widetilde{\mathcal{I}} \models \mathsf{RenameNotRig}(q, p)$, i.e., $\widetilde{\mathcal{I}} \models \widetilde{q}_p$. Hence $\widetilde{\mathcal{K}} \models \widetilde{q}_p$.

For the other direction, assume that $\widetilde{\mathcal{K}} \models \widetilde{q}_p$ and let $\mathcal{J} = (\mathcal{I}_i)_{i \geqslant 0}$ be a model of $\mathcal{K}$ that respects rigid predicates. Let $\widetilde{\mathcal{I}} = \mathsf{atemp}(\mathcal{J})$ be as defined in the proof for Lemma 6.2. Since $\widetilde{\mathcal{I}} \models \widetilde{q}_p$, there exists a mapping π from the set of constants and variables that appear in ψ into Δ such that for every $X(\vec{d}) \in \mathsf{atoms}(\mathsf{RenameNotRig}(\psi_\pi, p))$, we have $\vec{d} \in X^{\widetilde{\mathcal{I}}}$. It follows that for every $X(\vec{d}) \in \mathsf{atoms}(\mathsf{RenameNotRig}(\psi_\pi, p))$ such that X is rigid, $\vec{d} \in X^{\mathcal{I}_p}$. Furthermore, we have $\vec{d} \in X^{\mathcal{I}_p}$ for every $X_p(\vec{d}) \in \mathsf{atoms}(\mathsf{RenameNotRig}(\psi_\pi, p))$ such that X is not rigid. Thus, $\mathcal{I}_p \models q$, and we obtain $\mathcal{K}, p \models q$. □

Moreover, the size of $\widetilde{q}_p$ is the same as q. We thus obtain the following lemma.

Lemma 7.2. *If BCQ entailment from an $\mathcal{L}$-KB is in P w.r.t. KB complexity, respectively in NP w.r.t. combined complexity, then so is BCQ entailment from an $\mathcal{L}$-TKB.*

Proof. If deciding whether $\widetilde{\mathcal{K}} \models \widetilde{q}_p$ is polynomial both in $|\widetilde{\mathcal{T}}|$ and in $|\widetilde{\mathcal{A}}|$, then it is polynomial in $|\mathcal{T}|$ and

$|(\mathcal{A}_i)_{0 \leqslant i \leqslant n}|$. It follows that deciding whether $\mathcal{K}, p \models q$ is in P w.r.t. KB complexity.

Moreover, if deciding whether $\widetilde{\mathcal{K}} \models \widetilde{q}_p$ is in NP w.r.t. $|\widetilde{\mathcal{T}}|$, $|\widetilde{\mathcal{A}}|$ and $|\widetilde{q}_p|$, then verifying a certificate that $\widetilde{\mathcal{K}} \models \widetilde{q}_p$ can be done in polynomial time w.r.t. $|\widetilde{\mathcal{T}}|$, $|\widetilde{\mathcal{A}}|$ and $|\widetilde{q}_p|$, so in polynomial time w.r.t. $|\mathcal{T}|$, n, $|(\mathcal{A}_i)_{0 \leqslant i \leqslant n}|$ and $|q|$. It follows that deciding whether $\mathcal{K}, p \models q$ is in NP w.r.t. combined complexity. □

Next, we define the notion of canonical model property for BCQ entailment and for entailment of BTCQs without negation.

Definition 7.3 (Canonical model property). A DL $\mathcal{L}$ has the *canonical model property for BCQ entailment* iff for every consistent $\mathcal{L}$-KB $\langle \mathcal{T}, \mathcal{A} \rangle$, there exists a model $\mathcal{I}_{\langle \mathcal{T}, \mathcal{A} \rangle}$ such that for every BCQ q, $\langle \mathcal{T}, \mathcal{A} \rangle \models q$ iff $\mathcal{I}_{\langle \mathcal{T}, \mathcal{A} \rangle} \models q$. We call $\mathcal{I}_{\langle \mathcal{T}, \mathcal{A} \rangle}$ the *canonical model* of KB $\langle \mathcal{T}, \mathcal{A} \rangle$.

A DL $\mathcal{L}$ has the *canonical model property for entailment of BTCQs without negation* iff for any $\mathcal{L}$-TKB $\langle \mathcal{T}, (\mathcal{A}_i)_{0 \leqslant i \leqslant n} \rangle$, there exists a model $\mathcal{J}_{\langle \mathcal{T}, (\mathcal{A}_i)_{0 \leqslant i \leqslant n} \rangle}$ such that for every BTCQ without negation ϕ and every time point p, holds that

$$\langle \mathcal{T}, (\mathcal{A}_i)_{0 \leqslant i \leqslant n} \rangle, p \models \phi \text{ iff } \mathcal{J}_{\langle \mathcal{T}, (\mathcal{A}_i)_{0 \leqslant i \leqslant n} \rangle}, p \models \phi.$$

We call this model $\mathcal{J}_{\langle \mathcal{T}, (\mathcal{A}_i)_{0 \leqslant i \leqslant n} \rangle}$ the *canonical model* of TKB $\langle \mathcal{T}, (\mathcal{A}_i)_{0 \leqslant i \leqslant n} \rangle$.

Note that it is justified to speak of *the* canonical model of a KB or TKB $\mathcal{K}$ because such a model can be homomorphically mapped into any other model of $\mathcal{K}$. Indeed, for every assertion α built over $\mathsf{N_I}$, $\mathsf{N_C}$ and $\mathsf{N_R}$, if α holds in the canonical model of $\mathcal{K}$ then it also holds in every model of $\mathcal{K}$.

The following theorem gives the relation between the canonical model property for BCQ entailment and the one for BTCQ entailment, and shows why the presence or absence of negation in the query matters.

Theorem 7.4. *If $\mathcal{L}$ has the canonical model property for BCQ entailment, then $\mathcal{L}$ also has the canonical model property for entailment of BTCQs without negation.*

Proof (Sketch). Let $\widetilde{\mathcal{I}}_{\widetilde{\mathcal{K}}}$ be the canonical model of $\widetilde{\mathcal{K}}$ and $\mathcal{J}_{\mathcal{K}} = (\mathcal{I}_i)_{i \geqslant 0} = \mathsf{temp}(\widetilde{\mathcal{I}}_{\widetilde{\mathcal{K}}})$. We show that $\mathcal{J}_{\mathcal{K}}$ is the canonical model of $\mathcal{K}$ for BTCQs without negation, that is, for every BTCQ ϕ that does not contain any negation, $\mathcal{K}, p \models \phi$ iff $\mathcal{J}_{\mathcal{K}}, p \models \phi$.

Since $\mathcal{J}_{\mathcal{K}}$ is a model of $\mathcal{K}$ that respects rigid predicates, if $\mathcal{K}, p \models \phi$ then $\mathcal{J}_{\mathcal{K}}, p \models \phi$. For the other di-

rection, we show in the appendix by induction on the structure of ϕ that if $\mathcal{J}_{\mathcal{K}}, p \models \phi$, then $\mathcal{K}, p \models \phi$. The interesting case is when ϕ is a BCQ. For a BCQ q, by Lemma 7.1, $\mathcal{K}, p \models q$ iff $\widetilde{\mathcal{K}} \models \widetilde{q}_p$, which is exactly the case iff $\widetilde{\mathcal{I}}_{\widetilde{\mathcal{K}}} \models \widetilde{q}_p$. By construction of $\mathcal{J}_{\mathcal{K}}$, it follows that $\mathcal{K}, p \models q$ iff $\mathcal{J}_{\mathcal{K}}, p \models q$. $\qquad\square$

Remark 7.5. If ϕ contains negation, the preceding argument fails and $\mathcal{J}_{\mathcal{K}}$ is not a canonical model for TCQ answering, i.e., $\mathcal{J}_{\mathcal{K}}, p \models \neg\phi$ does not guarantee that $\mathcal{J}, p \not\models \phi$ for every model $\mathcal{J}$ that respects rigid predicates. As an example, consider the empty TKB, i.e. $\mathcal{T} = \emptyset$ and $\mathcal{A}_i = \emptyset$ for every $i \in [0, n]$. We have $\mathcal{J}_{\mathcal{K}}, 0 \not\models \exists x. A(x)$, but we can easily construct a model $\mathcal{J}$ for $\langle \mathcal{T}, (\mathcal{A}_i)_{0 \leqslant i \leqslant n} \rangle$ such that $\mathcal{J}, 0 \models \exists x. A(x)$, so that $\mathcal{K} \not\models \neg\exists x. A(x)$.

The following proposition is a direct consequence of the existence of a canonical model for entailment of BTCQs without negation.

Proposition 7.6. *Assume $\mathcal{L}$ has the canonical model property for BCQ entailment. Then, for every $\mathcal{L}$-TKBs $\mathcal{K}$ and $\mathcal{K}'$, if $\mathcal{K}$ and $\mathcal{K}'$ coincide for BCQ entailment, then $\mathcal{K}$ and $\mathcal{K}'$ coincide for entailment of BTCQs without negation. I.e., if for every time point p and BCQ q, $\mathcal{K}, p \models q$ iff $\mathcal{K}', p \models q$, then for every time point p and BTCQ ϕ without negation, $\mathcal{K}, p \models \phi$ iff $\mathcal{K}', p \models \phi$.*

Proof (Sketch). We show this result in the appendix by induction on the structure of ϕ. We use the fact that by Theorem 7.4, $\mathcal{L}$ has the canonical model property for entailment of BTCQs without negation, to apply the definitions of BTCQ satisfaction of Table 1 to the canonical models of $\mathcal{K}$ and $\mathcal{K}'$. $\qquad\square$

We now prove a central proposition for TCQ answering over TKBs in DLs that have the canonical model property for entailment of BTCQs without negation. It amounts to reducing the entailment of BTCQs with unbounded future operators to the entailment of BTCQs with only bounded future operators. These can then be answered by considering only a finite number of time points.

Let $\mathcal{K}^*$ be the following TKB:

$$\mathcal{K}^* = \langle \mathcal{T}, (\mathcal{A}_i)_{0 \leqslant i \leqslant n} \cup (\mathcal{A}_{n+1}) \rangle \text{ where}$$

$$\mathcal{A}_{n+1} = \{A(a) \mid A \in \mathsf{N}_{\mathsf{RC}}, A(a) \in \widetilde{\mathcal{A}}\} \cup$$

$$\{R(a, b) \mid R \in \mathsf{N}_{\mathsf{RR}}, R(a, b) \in \widetilde{\mathcal{A}}\}.$$

Proposition 7.7. *If $\mathcal{L}$ has the canonical model property for BCQ entailment, the relations in Table 2 hold for any $\mathcal{L}$-TKB $\mathcal{K}$.*

Proof. It is easy to see that $\mathcal{J}_{\mathcal{K}^*} = \mathcal{J}_{\mathcal{K}}$ holds by construction. We obtain that $\mathcal{K}, p \models \phi$ iff $\mathcal{K}^*, p \models \phi$ for every BTCQ without negation ϕ.

All relations in Table 2 but those for the operators $\square$, $\lozenge$ and U are straightforwardly obtained by applying the definitions of BTCQ satisfaction of Table 1 to this canonical model. To show the three remaining relations, we rely on the fact that $\mathcal{J}_{\mathcal{K}^*}$ is such that for every $i > n$, $\mathcal{I}_i = \mathcal{I}_{n+1}$, and there are no past operators nested under unbounded future operators by definition of TCQs. Indeed, if a BTCQ ϕ_1 does not contain any past operators and $i > n$, $\mathcal{J}_{\mathcal{K}^*}, i \models \phi_1$ iff $\mathcal{J}_{\mathcal{K}^*}, n + 1 \models \phi_1$. Hence we can show the following, from which the relations from Table 2 then follow straightforwardly.

- $\mathcal{K}^*, p \models \square\phi_1$ iff for every $k \geqslant p$, $\mathcal{K}^*, k \models \phi_1$. Hence $\mathcal{K}^*, p \models \square\phi_1$ iff $\forall k, p \leqslant k \leqslant n$, $\mathcal{K}^*, k \models \phi_1$ and $\mathcal{K}^*, n + 1 \models \phi_1$.
- $\mathcal{K}^*, p \models \lozenge\phi_1$ iff there exists $k \geqslant p$, $\mathcal{K}^*, k \models \phi_1$. Hence $\mathcal{K}^*, p \models \lozenge\phi_1$ iff $\exists k, p \leqslant k \leqslant n$, $\mathcal{K}^*, k \models \phi_1$ or $\mathcal{K}^*, n + 1 \models \phi_1$.
- $\mathcal{K}^*, p \models \phi_1 \mathsf{U} \phi_2$ iff $\exists k \geqslant p$, $\mathcal{K}^*, k \models \phi_2$ and $\forall j$, $p \leqslant j < k, \mathcal{K}^*, j \models \phi_1$. Hence $\mathcal{K}^*, p \models \phi_1 \mathsf{U} \phi_2$ iff $\exists k, p \leqslant k \leqslant n$, $\mathcal{K}^*, k \models \phi_2$ and $\forall j, p \leqslant j < k, \mathcal{K}^*, j \models \phi_1$, or $\mathcal{K}^*, n + 1 \models \phi_2$ and $\forall j, p \leqslant j < n + 1, \mathcal{K}^*, j \models \phi_1$. $\qquad\square$

In the next theorem, we transfer complexity upper bounds from the atemporal case to the temporal case, even with rigid predicates, for queries without negation and DLs that have the canonical model property for BCQ entailment. We consider DLs for which BCQ entailment is in P w.r.t. KB complexity and in NP w.r.t. combined complexity, such as DL-Lite$_{\mathcal{R}}$ and $\mathcal{EL}_{\perp}$.

Theorem 7.8. *If $\mathcal{L}$ has the canonical model property for BCQ entailment and is such that BCQ entailment from KBs is in P w.r.t. KB complexity, respectively in NP w.r.t. combined complexity, then the entailment of BTCQs without negation from $\mathcal{L}$-TKBs is in P w.r.t. KB complexity, respectively in NP w.r.t. combined complexity.*

Proof. By Lemma 7.2, it is possible to decide whether $\mathcal{K}^*, p \models q$ in P w.r.t. KB complexity for any BCQ q. Based on this, we can show by induction on the structure of ϕ that $\mathcal{K}^*, p \models \phi$ can be decided in P w.r.t. KB complexity. Assume that for two BTCQs ϕ_1, ϕ_2

Table 2

Entailment under classical semantics for DLs with the canonical model property for BCQ entailment

ϕ	$\mathcal{K}, p \models \phi$ iff $\mathcal{K}^*, p \models \phi$ iff
$\exists \vec{y}.\psi(\vec{y})$	$\mathcal{K}^*, p \models \exists \vec{y}.\psi(\vec{y})$
$\phi_1 \wedge \phi_2$	$\mathcal{K}^*, p \models \phi_1$ and $\mathcal{K}^*, p \models \phi_2$
$\phi_1 \vee \phi_2$	$\mathcal{K}^*, p \models \phi_1$ or $\mathcal{K}^*, p \models \phi_2$
$\bigcirc \phi_1$	$\mathcal{K}^*, p+1 \models \phi_1$
$\bullet^b \phi_1$	$p < n$ implies $\mathcal{K}^*, p+1 \models \phi_1$
$\bigcirc^- \phi_1$	$p > 0$ and $\mathcal{K}^*, p-1 \models \phi_1$
$\bullet^- \phi_1$	$p > 0$ implies $\mathcal{K}^*, p-1 \models \phi_1$
$\Box \phi_1$	$\forall k, p \leqslant k \leqslant n+1, \mathcal{K}^*, k \models \phi_1$
$\Box^b \phi_1$	$\forall k, p \leqslant k \leqslant n, \mathcal{K}^*, k \models \phi_1$
$\Box^- \phi_1$	$\forall k, 0 \leqslant k \leqslant p, \mathcal{K}^*, k \models \phi_1$
$\Diamond \phi_1$	$\exists k, p \leqslant k \leqslant n+1, \mathcal{K}^*, k \models \phi_1$
$\Diamond^b \phi_1$	$\exists k, p \leqslant k \leqslant n, \mathcal{K}^*, k \models \phi_1$
$\Diamond^- \phi_1$	$\exists k, 0 \leqslant k \leqslant p, \mathcal{K}^*, k \models \phi_1$
$\phi_1 \mathsf{U} \phi_2$	$\exists k, p \leqslant k \leqslant n+1, \mathcal{K}^*, k \models \phi_2$ and $\forall j, p \leqslant j < k, \mathcal{K}^*, j \models \phi_1$
$\phi_1 \mathsf{U}^b \phi_2$	$\exists k, p \leqslant k \leqslant n, \mathcal{K}^*, k \models \phi_2$ and $\forall j, p \leqslant j < k, \mathcal{K}^*, j \models \phi_1$
$\phi_1 \mathsf{S} \phi_2$	$\exists k, 0 \leqslant k \leqslant p, \mathcal{K}^*, k \models \phi_2$ and $\forall j, k < j \leqslant p, \mathcal{K}^*, j \models \phi_1$

and any $p \geqslant 0$, it is possible to decide in P whether $\mathcal{K}^*, p \models \phi_i$. Using the relations in Table 2, we can prove the following.

– $\mathcal{K}^*, p \models \phi_1 \wedge \phi_2$ iff $\mathcal{K}^*, p \models \phi_1$ and $\mathcal{K}^*, p \models \phi_2$. Therefore, deciding whether $\mathcal{K}^*, p \models \phi_1 \wedge \phi_2$ can be done in P by checking that $\mathcal{K}^*, p \models \phi_1$ and $\mathcal{K}^*, p \models \phi_2$.

– $\mathcal{K}^*, p \models \phi_1 \vee \phi_2$ iff $\mathcal{K}^*, p \models \phi_1$ or $\mathcal{K}^*, p \models \phi_2$. Therefore, deciding whether $\mathcal{K}^*, p \models \phi_1 \vee \phi_2$ can be done in P by deciding whether $\mathcal{K}^*, p \models \phi_1$ and $\mathcal{K}^*, p \models \phi_2$ and checking that at least one is true.

– $\mathcal{K}^*, p \models \bigcirc \phi_1$ iff $\mathcal{K}^*, p+1 \models \phi_1$. Therefore, deciding whether $\mathcal{K}^*, p \models \bigcirc \phi_1$ can be done in P by checking whether $\mathcal{K}^*, p+1 \models \phi_1$.

– $\mathcal{K}^*, p \models \bullet^b \phi_1$ iff $p < n$ implies $\mathcal{K}^*, p+1 \models \phi_1$. Therefore, deciding whether $\mathcal{K}^*, p \models \bullet^b \phi_1$ can be done in P by checking whether $p \geqslant n$ or $\mathcal{K}^*, p+1 \models \phi_1$.

– We show in the same way that we can decide in P whether $\mathcal{K}^*, p \models \bigcirc^- \phi_1$ and whether $\mathcal{K}^*, p \models \bullet^- \phi_1$.

– $\mathcal{K}^*, p \models \Box \phi_1$ iff for every $k, p \leqslant k \leqslant n+1, \mathcal{K}^*, k \models \phi_1$. Therefore, deciding whether $\mathcal{K}^*, p \models \Box \phi_1$ can be done in P by checking for each $p \leqslant k \leqslant n+1$ that $\mathcal{K}^*, k \models \phi_1$.

– We show in the same way that we can decide in P whether $\mathcal{K}^*, p \models \Box^b \phi_1$ and whether $\mathcal{K}^*, p \models \Box^- \phi_1$.

– $\mathcal{K}^*, p \models \Diamond \phi_1$ iff there exists $k, p \leqslant k \leqslant n+1, \mathcal{K}^*, k \models \phi_1$. Therefore, deciding whether $\mathcal{K}^*, p \models \Diamond \phi_1$ can be done in P by deciding for each $p \leqslant k \leqslant$

$n+1$ whether $\mathcal{K}^*, k \models \phi_1$, and checking that this is true for at least one k.

– We show in the same way that we can decide in P whether $\mathcal{K}^*, p \models \Diamond^b \phi_1$ and whether $\mathcal{K}^*, p \models \Diamond^- \phi_1$.

– $\mathcal{K}^*, p \models \phi_1 \mathsf{U} \phi_2$ iff there exists $k, p \leqslant k \leqslant n+1$, $\mathcal{K}^*, k \models \phi_2$, and for every $j, p \leqslant j < k, \mathcal{K}^*, j \models \phi_1$. Therefore, deciding whether $\mathcal{K}^*, p \models \phi_1 \mathsf{U} \phi_2$ can be done in P by deciding for each $p \leqslant k \leqslant n+1$ whether $\mathcal{K}^*, k \models \phi_1$ and whether $\mathcal{K}^*, k \models \phi_2$, and checking that the condition holds.

– We show in the same way that we can decide in P whether $\mathcal{K}^*, p \models \phi_1 \mathsf{U}^b \phi_2$ and whether $\mathcal{K}^*, p \models \phi_1 \mathsf{S} \phi_2$.

The number of subqueries in ϕ is linear w.r.t. the size of ϕ, and independent from the TKB size. It follows that the total number of polynomial checks is also polynomially bounded. Therefore, we obtain that for every BTCQ ϕ without negation, $\mathcal{K}^*, p \models \phi$ can be decided in P w.r.t. the size of $\mathcal{K}^*$. Since $\mathcal{K}, p \models \phi$ iff $\mathcal{K}^*, p \models \phi$ and the size of $\mathcal{K}^*$ is polynomial in the size of $\mathcal{K}$, deciding whether $\mathcal{K}, p \models \phi$ is in P w.r.t. KB complexity.

For the NP membership of entailment of BTCQs without negation w.r.t. combined complexity, we describe how to guess a certificate that $\mathcal{K}, p \models \phi$ that can be checked in P. This certificate consists of:

– a sequence of functions $(\nu_i)_{0 \leqslant i \leqslant n+1}$ that associate to each BCQ q of ϕ true or false, and

– for each BCQ q of ϕ and time point $i \in [0, n+1]$ such that $\nu_i(q) = \text{true}$: a certificate that $\widetilde{q}_i = \text{RenameNotRig}(q, i)$ is entailed from $\widetilde{\mathcal{K}}$.

There are polynomially many pairs of a time point and a BCQ, and the certificate that $\widetilde{q}_i$ is entailed from $\widetilde{\mathcal{K}}$ can be checked in polynomial time, since BCQ entailment is in NP. Moreover, we can show that, since ϕ contains neither negation nor past operators nested under unbounded future operators, deciding whether the propositional abstraction of ϕ is satisfied by the sequence of truth assignments that assign the propositional abstraction of q to $\nu_i(q)$ for every $i \in [0, n+1]$ and to $\nu_{n+1}(q)$ for every $i > n+1$ can be done in polynomial time w.r.t. the size of the query and the length of the sequence of ABoxes. Indeed, identify ϕ and the BCQs in ϕ with their propositional abstractions, and denote by $w = w^0 w^1 ... w^n w^{n+1} ...$ the trace over $2^{BCQ(\phi)}$, where $BCQ(\phi)$ denotes the set of BCQs of ϕ, such that $w^i = \{q \mid \nu_i(q) = \text{true}\}$ for $i \leqslant n+1$, and $w^i = w^{n+1}$ for $i > n+1$. Since $w^i = w^{n+1}$, for

$i > n + 1$, we can show similar relations as those in Table 2 for the entailment of LTL formulas without past operators nested under unbounded future operators or negations from w. We can then use a similar induction as we used in the proof for the data complexity to show that $w, p \models \phi$ can be decided by checking which queries are in w^i. For this, the number of queries to be tested is polynomial in n and the size of ϕ. $\qquad\square$

As a consequence of Theorems 7.4 and 7.8, and since $\mathcal{EL}_\perp$ and DL-Lite$_\mathcal{R}$ have the canonical model property for BCQ entailment (cf. [11] for $\mathcal{EL}_\perp$, and [44] for DL-Lite$_\mathcal{R}$), we obtain the following theorem.

Theorem 7.9. *For DL-Lite$_\mathcal{R}$ and $\mathcal{EL}_\perp$, entailment of BTCQs without negation is in* P *w.r.t. KB complexity and in* NP *w.r.t. combined complexity, even if* $\mathsf{N}_{\mathsf{RR}} \neq \emptyset$.

Besides these results for DL-Lite$_\mathcal{R}$ and $\mathcal{EL}_\perp$, Theorems 7.4 and 7.8 hold for all Horn-DLs satisfying the complexity constraints in the precondition of Theorem 7.8. For instance, they also hold for DL-Lite$_{horn}^{\mathcal{N}}$ [45].

8. Complexity of inconsistency-tolerant BTCQ entailment without negation in the query

The following proposition gives general complexity upper bounds for BTCQ entailment under the AR, IAR, and brave semantics. By Theorem 7.9, they hold in particular for $\mathcal{L} = \mathcal{EL}_\perp$ and $\mathcal{L} = $ DL-Lite$_\mathcal{R}$, provided no negation is used in the TCQs.

Proposition 8.1. *If* $\mathcal{L}$ *is such that consistency checking of a* $\mathcal{L}$-*TKB is in* P *and BTCQ entailment from a* $\mathcal{L}$-*TKB is in* P *w.r.t. data complexity and in* NP *w.r.t. combined complexity, then BTCQ entailment from a* $\mathcal{L}$-*TKB*

- *under AR semantics is in* coNP *w.r.t. data complexity and in* Π_2^p *w.r.t. combined complexity;*
- *under IAR semantics is in* coNP *w.r.t. data complexity and in* $\Delta_2^p[O(\log n)]$ *w.r.t. combined complexity;*
- *under brave semantics is in* NP *w.r.t. data complexity and in* NP *w.r.t. combined complexity.*

Proof. The data complexities follow from the procedures described in Section 5: since verifying that a sequence of ABoxes is a repair as well as non-entailment and entailment of a BTCQ can be decided in polynomial time, ARNonEntailment, IARNonEntailment and braveEntailment take non-deterministic polynomial time.

For the combined complexity of brave BTCQ entailment, a certificate that $\langle \mathcal{T}, (\mathcal{A}_i')_{0 \leqslant i \leqslant n} \rangle, p \models \phi$ can be guessed together with $(\mathcal{A}_i')_{0 \leqslant i \leqslant n}$, and verified in P.

For the combined complexity of IAR, we use the procedure IAREntailment from Section 5 with an NP oracle to decide whether a timed assertion belongs to some minimal inconsistent subset in the first step, and to decide the entailment of ϕ in the last step. Since the oracle calls can be structured as a tree, this procedure gives membership in $\Delta_2^p[O(\log n)]$ [46]. $\qquad\square$

For $\mathcal{EL}_\perp$, matching lower bounds for all semantics follow from the atemporal case [28, 35].

For DL-Lite$_\mathcal{R}$, we can obtain matching lower bounds from the atemporal case for the combined complexity of all semantics as well as for the data complexity of the AR semantics [29, 33]. Regarding IAR semantics, entailment of BTCQs with negations under IAR semantics is already in P (see Figure 3), so this better upper bound applies. Finally, we show that for brave semantics and DL-Lite$_\mathcal{R}$, in the case where there are no rigid predicates, we can improve the NP upper bound of Figure 3 to a P bound. Recall that the lower bound established by Proposition 6.10 for brave semantics with rigid predicates already applies to BTCQs without negations, which means that no improvement of our bounds is possible in the presence of rigid predicates. The only remaining case is entailment under brave semantics where $\mathsf{N}_{\mathsf{RC}} = \mathsf{N}_{\mathsf{RR}} = \emptyset$.

We describe a method for brave entailment of BTCQ without negation when $\mathsf{N}_{\mathsf{RC}} = \mathsf{N}_{\mathsf{RR}} = \emptyset$. Our procedure proceeds by type elimination over a set of tuples built from the query and that represent the TCQs that are entailed at each time point.

First, we define the structure on which the method operates. We consider the set $BCQ(\phi)$ of *leaves* of ϕ, that is, the set of all BCQs in ϕ, and the set $F(\phi)$ of *subformulas* of ϕ. In what follows, we identify the BCQs of $BCQ(\phi)$ and the BTCQs of $F(\phi)$ with their *propositional abstractions* as defined in Section 2: if we write that a KB or a TKB entails some elements of $BCQ(\phi)$ or $F(\phi)$, we consider them as BCQs or BTCQs, and if we write that some elements of $BCQ(\phi)$ or $F(\phi)$ entail others, we consider the elements of $BCQ(\phi)$ as propositional variables and those of $F(\phi)$ as propositional LTL formulas built over these variables.

Definition 8.2. A *brave-justification structure* J for the BTCQ without negation ϕ in the TKB $\mathcal{K}$ is a set of tuples of the form $(i, L_{\mathrm{now}}, F_{\mathrm{now}}, F_{\mathrm{prev}}, F_{\mathrm{next}})$, where

$0 \leqslant i \leqslant n$, $L_{\text{now}} \subseteq BCQ(\phi)$, $F_{\text{now}} \subseteq F(\phi)$, $F_{\text{prev}} \subseteq F(\phi)$, and $F_{\text{next}} \subseteq F(\phi)$.

Note that the size of a brave-justification structure for ϕ in $\mathcal{K} = \langle \mathcal{T}, (\mathcal{A}_i)_{0 \leqslant i \leqslant n} \rangle$ is linearly bounded in n and independent of the size of the ABoxes. A tuple $(i, L_{\text{now}}, F_{\text{now}}, F_{\text{prev}}, F_{\text{next}})$ is *justified in J* iff it fulfils all of the following conditions.

1. $\langle \mathcal{T}, \mathcal{A}_i \rangle \models_{\text{brave}} \bigwedge_{q \in L_{\text{now}}} q$.
2. If $i > 0$, there exists $(i-1, L'_{\text{now}}, F'_{\text{now}}, F'_{\text{prev}}, F'_{\text{next}}) \in J$ such that $F_{\text{prev}} = F'_{\text{now}}$ and $F_{\text{now}} = F'_{\text{next}}$.
3. If $i < n$, there exists $(i+1, L'_{\text{now}}, F'_{\text{now}}, F'_{\text{prev}}, F'_{\text{next}}) \in J$ such that $F_{\text{next}} = F'_{\text{now}}$ and $F_{\text{now}} = F'_{\text{prev}}$.
4. For every $\psi \in BCQ(\phi)$, if $F_{\text{now}} \models \psi$, then $\psi \in L_{\text{now}}$.
5. For every $\psi \in F(\phi)$, if $F_{\text{now}} \models \psi$, then $\psi \in F_{\text{now}}$.
6. For every $\psi \in F(\phi)$, if $\bigwedge_{q \in L_{\text{now}}} q \wedge \bigcirc^{-}(\bigwedge_{\chi \in F_{\text{prev}}} \chi) \wedge \bigcirc(\bigwedge_{\chi \in F_{\text{next}}} \chi) \models \psi$, then $\psi \in F_{\text{now}}$.
7. For every $\psi, \psi' \in F(\phi)$:
   if $\psi \vee \psi' \in F_{\text{now}}$, then either $\psi \in F_{\text{now}}$ or $\psi' \in F_{\text{now}}$,
   if $\Diamond \psi \in F_{\text{now}}$, then either $\psi \in F_{\text{now}}$ or $\Diamond \psi \in F_{\text{next}}$,
   if $\Diamond^{b} \psi \in F_{\text{now}}$, then either $\psi \in F_{\text{now}}$ or $\Diamond^{b} \psi \in F_{\text{next}}$,
   if $\Diamond^{-} \psi \in F_{\text{now}}$, then either $\psi \in F_{\text{now}}$ or $\Diamond^{-} \psi \in F_{\text{prev}}$,
   if $\psi' \mathsf{U} \psi \in F_{\text{now}}$, then either $\psi \in F_{\text{now}}$ or $\psi' \in F_{\text{now}}$ and $\psi' \mathsf{U} \psi \in F_{\text{next}}$,
   if $\psi' \mathsf{U}^{b} \psi \in F_{\text{now}}$, then either $\psi \in F_{\text{now}}$ or $\psi' \in F_{\text{now}}$ and $\psi' \mathsf{U}^{b} \psi \in F_{\text{next}}$,
   if $\psi' \mathsf{S} \psi \in F_{\text{now}}$, then either $\psi \in F_{\text{now}}$ or $\psi' \in F_{\text{now}}$ and $\psi' \mathsf{S} \psi \in F_{\text{prev}}$, and
   if ψ is of the form $\Box \varphi$, then either $\psi \notin F_{\text{now}}$ or $\langle \emptyset, (\emptyset)_{0 \leqslant i \leqslant n} \rangle, n+1 \models \Box \varphi$.
8. If $i = n$, then
   for all $\psi \in F(\phi)$ that are of the form $\bullet^{b} \varphi$, $\psi \in F_{\text{now}}$,
   for all $\psi \in F(\phi)$ that are of the form $\bigcirc \varphi$ and such that $\langle \emptyset, (\emptyset)_{0 \leqslant i \leqslant n} \rangle, n+1 \not\models \Box \varphi$, we have $\psi \notin F_{\text{now}}$,
   for all $\psi \in F(\phi)$ that are of the form $\Diamond^{b} \varphi, \Box^{b} \varphi$, or $\varphi' \mathsf{U}^{b} \varphi$, $\psi \in F_{\text{now}}$ iff $\varphi \in F_{\text{now}}$,
   for all $\psi \in F(\phi)$ that are of the form $\Diamond \varphi, \varphi' \mathsf{U} \varphi$ and such that $\langle \emptyset, (\emptyset)_{0 \leqslant i \leqslant n} \rangle, n+1 \not\models \Box \varphi$, $\psi \in F_{\text{now}}$ iff $\varphi \in F_{\text{now}}$.
9. If $i = 0$, then
   for all $\psi \in F(\phi)$ that are of the form $\bullet^{-} \varphi$, $\psi \in F_{\text{now}}$,
   for all $\psi \in F(\phi)$ that are of the form $\bigcirc^{-} \varphi$, $\psi \notin F_{\text{now}}$, and
   for all $\psi \in F(\phi)$ that are of the form $\Diamond^{-} \varphi, \Box^{-} \varphi, \varphi' \mathsf{S} \varphi$, $\psi \in F_{\text{now}}$ iff $\varphi \in F_{\text{now}}$.

We give the intuition behind the elements of the tuples that fulfil these conditions. The first element i is the time point we are considering, L_{now} is a set of BCQs whose conjunction is entailed under brave semantics by $\langle \mathcal{T}, \mathcal{A}_i \rangle$ (Condition 1), and F_{now} is the set of formulas that can be entailed together with L_{now}, depending on what is entailed in the previous and next time points, this information being stored in F_{prev} and F_{next} respectively (Condition 6). Conditions 2 and 3 ensure that there is a sequence of tuples representing every time point from 0 to n such that this information is coherent between consecutive tuples. Condition 4 expresses that L_{now} is exactly the set of BCQs contained in F_{now} and Condition 5 that F_{now} is maximal in the sense that it contains its consequences. Condition 7 enforces that F_{now}, F_{prev} and F_{next} respect the semantics of LTL operators and Conditions 8 and 9 enforce this semantics at both ends of the finite sequence. (Note that we use here the fact that past operators cannot be nested under unbounded future operators, and that no BCQ can be entailed under brave semantics after time point n in the absence of rigid predicates.)

A brave-justification structure J is *correct* if every tuple is justified, and ϕ is *justified at time point p by J* if there is $(p, L_{\text{now}}, F_{\text{now}}, F_{\text{prev}}, F_{\text{next}}) \in J$ such that $\phi \in F_{\text{now}}$. We show that ϕ is entailed from $\mathcal{K}$ at time point p under brave semantics iff there is a correct brave-justification structure for ϕ in $\mathcal{K}$ that justifies ϕ at time point p. The main idea of the proof is to link the tuples of a sequence $((i, L_{\text{now}}, F_{\text{now}}, F_{\text{prev}}, F_{\text{next}}))_{0 \leqslant i \leqslant n}$ to a consistent TKB $\mathcal{K}' = \langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle$ such that for every i, we have $\mathcal{C}_i \subseteq \mathcal{A}_i$ and $\langle \mathcal{T}, \mathcal{C}_i \rangle \models \bigwedge_{q \in L_{\text{now}}} q$. We show in the appendix that there is such a $\mathcal{K}'$ such that $\mathcal{K}', p \models \phi$ iff there is such a sequence of tuples that is a correct brave-justification structure for ϕ in $\mathcal{K}$ and justifies ϕ at time point p.

The data complexity of brave entailment of BTCQ without negation when there are no rigid predicates follows from the characterization of brave BTCQ entailment with brave-justification structures.

Proposition 8.3. *For DL-Lite$_{\mathcal{R}}$, if $\mathsf{N}_{\text{RC}} = \mathsf{N}_{\text{RR}} = \emptyset$, then entailment of BTCQs without negation under brave semantics is in* P *w.r.t. data complexity.*

Proof. We describe a polynomial procedure that decides the existence of a brave-justification structure for ϕ in $\mathcal{K}$ that justifies ϕ at time point p. We start with a brave-justification structure J for ϕ in $\mathcal{K}$ that contains all possible tuples. Note that the size of this initial structure J is linearly bounded in the number n of

ABoxes, and otherwise independent of the data size. The reason is that, except for the time point, the elements of each justification tuple, namely the sets L_{now}, F_{now}, F_{prev} and F_{next}, only depend on the query.

From the initial, maximal, brave-justification structure, we remove the unjustified tuples one after the other as follows. (i) Remove every tuple that does not satisfy Conditions 1, 4, 5, 6, 7, 8 or 9, and (ii) repeat the following steps until a fix-point has been reached: iterate over the tuples from time point 0 to n, eliminating those which do not satisfy Condition 3, and then iterate from n to 0 eliminating those which do not satisfy Condition 2. For the resulting brave-justification structure, we check whether it contains a tuple $(p, L_{\text{now}}, F_{\text{now}}, F_{\text{prev}}, F_{\text{next}})$ such that $\phi \in F_{\text{now}}$. If yes, we return "*entailed at time point p*", otherwise, we return "*not entailed at time point p*". Since the size of J is linear in n, this process requires at most quadratically many steps. The verification that a given tuple is justified requires polynomial time w.r.t. data complexity (the verification of Condition 3 and Condition 2 is linear in n, and only the brave entailment of a BCQ from a DL-Lite$_{\mathcal{R}}$-KB for Condition 1 depends on the size of the ABox, which can be performed in AC^0 w.r.t. data complexity). Therefore, the complete procedure runs in polynomial time w.r.t. data complexity.   □

Note that it is vital for this procedure that the BTCQ does not contain negation, as this allows to fully focus on the positive entailments of BCQs. In contrast, BTCQs with negation may require a case analysis over query entailments, as exemplified by the tautological BTCQ $\square(A(a) \vee \neg A(a))$.

The following theorem summarizes the complexity results for the case without negation in the TCQ.

Theorem 8.4. *The results in Figure 5 hold.*

9. Toward practical algorithms

Until now, work on TCQ answering has primarily focused on complexity analysis for different DL languages [5, 6, 22]. Attempts towards practical algorithms or implementations are as of now scarce [43, 47]. The only attempt toward more practical algorithms close to our scenario that we are aware of has been made for DL-Lite$_{\mathcal{R}}$ and TCQs without negation in [43], and partially implemented [48]. Some of the results have then been generalized in [18] to query languages that are rewritable in the atemporal case.

In this section, we mainly focus on DL-Lite$_{\mathcal{R}}$-TKBs and TCQs without negation, building on this previous work. However, some of our results also apply to other DLs and we discuss the case of $\mathcal{EL}_{\perp}$.

Three different algorithms for answering TCQs without negation over DL-Lite$_{\mathcal{R}}$-TKBs without rigid predicates are provided in [18, 43]. The first algorithm rewrites the TCQ into a query in ATSQL [49], an SQL variant for temporal databases. The second algorithm first rewrites the TCQ into an equivalent TCQ that does not contain future operators, and then iteratively computes the answers for each time point. The third algorithm computes the answers of the TCQ iteratively as well, but does not eliminate the future operators beforehand. For this, it uses a data structure called *answer formulas*, which represents the TCQs in which some parts have already been evaluated. This structure contains sets of already computed answers to subqueries, as well as variables that serve as place-holders for subqueries that have to be evaluated at the next time point.

Our first contribution is a method for handling rigid predicates (both concepts and roles) in polynomial time for TCQ answering over DL-Lite$_{\mathcal{R}}$-TKBs under the classical semantics. Indeed, [18, 43] consider only rigid concepts (but not rigid roles) for which they provide a method that is restricted to TCQs that are rooted, i.e., in which each CQ contains an individual or an answer variable that is connected to all the other terms through roles. As a second contribution, we show that in the absence of rigid predicates, it is sometimes possible to combine the algorithms for inconsistency-tolerant query answering in the atemporal case with algorithms for temporal query answering in the consistent case in order to perform inconsistency-tolerant temporal query answering.

9.1. TCQ answering under classical semantics with rigid predicates for DL-Lite$_{\mathcal{R}}$ and TCQs without negation or unbounded future operators

In this section, we show how TCQ answering with rigid predicates can be reduced to TCQ answering without rigid predicates, enabling us to use the algorithms that have been proposed for this latter case. Throughout the section, $\mathcal{K}$ is a DL-Lite$_{\mathcal{R}}$-TKB and ϕ a TCQ without negation and unbounded future operators ($\bigcirc$, $\square$, $\lozenge$, U). This restriction amounts to using the setting of [18, 43] in which the semantics is defined w.r.t. finite sequences of interpretations, and is necessary to reduce TCQ answering with rigid predicates to

	Data complexity				Combined complexity			
	classical	AR	IAR	brave	classical	AR	IAR	brave
$\mathcal{EL}_\perp$								
$N_{RC} = N_{RR} = \emptyset$	P	coNP	coNP	NP	**NP**	Π_2^p	$\Delta_2^p[O(\log n)]$	**NP**
$N_{RC} \neq \emptyset, N_{RR} = \emptyset$	**P**	coNP	coNP	**NP**	**NP**	Π_2^p	$\Delta_2^p[O(\log n)]$	**NP**
$N_{RC} \neq \emptyset, N_{RR} \neq \emptyset$	**P**	coNP	coNP	**NP**	**NP**	Π_2^p	$\Delta_2^p[O(\log n)]$	**NP**
DL-Lite$_\mathcal{R}$								
$N_{RC} = N_{RR} = \emptyset$	in ALOGTIME	coNP	in P	**in P**	**NP**	Π_2^p	NP	NP
$N_{RC} \neq \emptyset, N_{RR} = \emptyset$	in ALOGTIME	coNP	in P	NP	**NP**	Π_2^p	NP	NP
$N_{RC} \neq \emptyset, N_{RR} \neq \emptyset$	in ALOGTIME	coNP	in P	NP	**NP**	Π_2^p	NP	NP

Figure 5. Data [left] and combined [right] complexity of BTCQ entailment for BTCQs without negation. All results are tight but those preceded by "in" which are upper bounds. The complexities lower than in the case of BTCQs with negation are in bold.

TCQ answering without rigid predicates. Indeed, consider for instance the query $\Box A(a)$. Such a query can be entailed with rigid predicates, e.g., if A is rigid, but not without rigid predicates since for $p > n$, the interpretation of every predicate is empty in the pth component of the canonical model of a TKB without rigid predicates.

To the best of our knowledge, the only algorithm that has been proposed for TCQ answering with rigid predicates and aims at practicality is described as well in [18, 43], and deals only with rigid concepts and rooted TCQs. We briefly describe this algorithm, which aims at handling streaming data by computing the answers to the query at the last available time point. The key idea is to check all sets of potentially rigid concept assertions and test their compatibility with each of the ABoxes from the sequence together with the TBox. Unfortunately, the original algorithm omits the test whether the checked set of rigid concept assertions covers also the rigid information from the tested ABox together with the TBox. As we found this small flaw in the original algorithm, we present here a mended variant.

In the following, let $N_C^\mathcal{K}$, $N_R^\mathcal{K}$, $N_{RC}^\mathcal{K}$, $N_{RR}^\mathcal{K}$, and $N_I^\mathcal{K}$ denote respectively the sets of concepts, roles, rigid concepts, rigid roles, and individuals that occur in the TKB $\mathcal{K}$. The algorithm first constructs every possible set $\mathcal{R}$ of assertions built from the rigid concepts and individuals in the TKB $\mathcal{K}$. Note that there are $2^{|N_{RC}^\mathcal{K}| * |N_I^\mathcal{K}|}$ such sets. It then runs, in parallel, for each such set $\mathcal{R}$ an instance of the algorithm for TCQ answering without rigid predicates on the TKB that is obtained by adding the assertions in $\mathcal{R}$ to every ABox of the TKB. For each time point i, it takes into account the new

dataset available by eliminating the incompatible instances, i.e., those for which

1. $\langle \mathcal{T}, \mathcal{A}_i \cup \mathcal{R} \rangle$ is inconsistent, or
2. a rigid concept assertion entailed by $\langle \mathcal{T}, \mathcal{A}_i \rangle$ does not belong to $\mathcal{R}$.[1]

The answers at time point i are then obtained by taking the intersection of the answers returned by all active instances.

We follow a similar idea in the sense that we also add assertions to the TKB that propagate the effects of the rigid predicates. We show that this way, for DL-Lite$_\mathcal{R}$, TCQ answering with rigid predicates can be reduced to TCQ answering without rigid predicates in *polynomial time*.

In order to show that TCQ answering with rigid predicates can be reduced to TCQ answering without, we construct in polynomial time a set $\mathcal{R}$ of assertions that captures all information about rigid concepts and roles that is relevant for consistency checking and TCQ answering. Then, TCQ answering over $\mathcal{K}$ with $N_{RC} \neq \emptyset$, $N_{RR} \neq \emptyset$ can be performed by TCQ answering over $\langle \mathcal{T}, (\mathcal{A}_i \cup \mathcal{R})_{0 \leqslant i \leqslant n} \rangle$ with $N_{RC} = N_{RR} = \emptyset$. Without any restriction on the TBox, $\mathcal{R}$ may be infinite, as illustrated in the following example.

Example 9.1. Consider the TKB $\mathcal{K}$ with the TBox $\mathcal{T} = \{\exists R^- \sqsubseteq \exists R, R \sqsubseteq S\}$, where S is rigid, and the ABox sequence consisting of $\mathcal{A}_0 = \{R(a, b)\}$, and $\mathcal{A}_i = \emptyset$ for $i \in [1, n]$.

---

[1]This condition is new and added after consultation with the authors of [18, 43].

Every model of $\mathcal{K}$ that respects rigid predicates satisfies $\phi = \exists x_1...x_{k+1}.S(x_1,x_2) \wedge ... \wedge S(x_k,x_{k+1})$ for every $k > 0$ and at every time point. Since with $N_{RC} = N_{RR} = \emptyset$, $\mathcal{K}$ entails such a query only at time point 0, $\mathcal{R}$ should be such that $\langle \mathcal{T}, \mathcal{R} \rangle$ entails such a query, so that $\langle \mathcal{T}, (\mathcal{A}_i \cup \mathcal{R})_{0 \leqslant i \leqslant n} \rangle$ entails it at every time point. Moreover, there exist models of $\mathcal{K}$ that respect rigid predicates and for which neither $\exists x_1...x_k.S(x_1,x_2) \wedge ... \wedge S(x_k,x_1)$ nor $\exists xy.R(x,y)$ hold at any time point $i > 0$. Therefore, $\mathcal{R}$ cannot contain cycles of S, nor R-assertions . Consequently, $\mathcal{R}$ has to contain an infinite chain of S-assertions.

This problem motivates us to disallow rigid roles that have non-rigid sub-roles. In other words, we restrict ourselves in the following to TBoxes $\mathcal{T}$ that entail no role inclusions of the form $P_1 \sqsubseteq P_2$ with $P_1 := R_1|R_1^-$, $R_1 \in N_R \backslash N_{RR}$ and $P_2 := R_2|R_2^-$, $R_2 \in N_{RR}$. This condition avoids chains of rigid roles in the anonymous part of the canonical model $\mathcal{J}_\mathcal{K}$ that cannot be entailed by a single rigid assertion. In the example above, if rigid roles are only allowed to have rigid sub-roles, then R has to be rigid. In this case, adding the single assertion $R(x,y)$ to every $\mathcal{A}_i$ is sufficient for $\exists x_1...x_{k+1}.R(x_1,x_2) \wedge ... \wedge R(x_k,x_{k+1})$ to be entailed at every time point and for every $k > 0$.

As a first step, we explicitly construct the canonical model $\mathcal{J}_\mathcal{K}$ of the DL-Lite$_\mathcal{R}$-TKB $\mathcal{K}$. This model will be used to prove that $\mathcal{K}$ with $N_{RC} \neq \emptyset$, $N_{RR} \neq \emptyset$ and $\langle \mathcal{T}, (\mathcal{A}_i \cup \mathcal{R})_{0 \leqslant i \leqslant n} \rangle$ with $N_{RC} = N_{RR} = \emptyset$ entail the same BTCQs that do not have negation or unbounded future operators.

We build a sequence of (possibly infinite) ABoxes $(chase_{rig}^\mathcal{K}(\mathcal{A}_i))_{0 \leqslant i \leqslant n+1}$ similar to the chase presented in [50] for KBs. Let $\mathcal{S}$ be a set of DL-Lite$_\mathcal{R}$ assertions. We say a positive inclusion (PI) α is *applicable in $\mathcal{S}$ to an assertion $\beta \in \mathcal{S}$* if one of the following conditions is satisfied:

- $\alpha = A_1 \sqsubseteq A_2, \beta = A_1(a)$ and $A_2(a) \notin \mathcal{S}$,
- $\alpha = A \sqsubseteq \exists P, \beta = A(a)$ and there exists no b such that $P(a,b) \in \mathcal{S}$,
- $\alpha = \exists P \sqsubseteq A, \beta = P(a,b)$ and $A(a) \notin \mathcal{S}$
- $\alpha = \exists P_1 \sqsubseteq \exists P_2, \beta = P_1(a_1,a_2)$ and there exists no b such that $P_2(a_1,b) \in \mathcal{S}$, or
- $\alpha = P_1 \sqsubseteq P_2, \beta = P_1(a_1,a_2)$, and $P_2(a_1,a_2) \notin \mathcal{S}$.

A PI α is *applied to an assertion β* by adding a new assertion β_{new} to $\mathcal{S}$ such that α is not applicable to β in $\mathcal{S} \cup \{\beta_{new}\}$ anymore.

Definition 9.2 (Rigid chase of a TKB). Let $\mathcal{K} = \langle \mathcal{T}, (\mathcal{A}_i)_{0 \leqslant i \leqslant n} \rangle$ be a DL-Lite$_\mathcal{R}$-TKB. Let $(\mathcal{A}_i')_{0 \leqslant i \leqslant n+1}$ be such that $\mathcal{A}_i' = \mathcal{A}_i \cup \{\beta \mid \exists k, \beta \in \mathcal{A}_k$ and β is rigid$\}$ for $i \in [0,n]$ and $\mathcal{A}_{n+1}' = \emptyset$. Finally, let $\mathcal{T}_p$ be the set of positive inclusions in $\mathcal{T}$, and N_i be the number of assertions in $\mathcal{A}_i'$. Assume that the assertions in each $\mathcal{A}_i'$ are enumerated from $N_1 + \cdots + N_{i-1} + 1$ to $N_1 + \cdots + N_i$ following their lexicographic order. Consider the sequences of sets $\mathcal{S}^j = (\mathcal{S}_i^j)_{0 \leqslant i \leqslant n+1}$ of assertions defined by

$$\mathcal{S}^0 = (\mathcal{A}_i')_{0 \leqslant i \leqslant n+1}$$
$$\mathcal{S}^{j+1} = \mathcal{S}^j \cup \mathcal{S}^{new} = (\mathcal{S}_i^j \cup \mathcal{S}_i^{new})_{0 \leqslant i \leqslant n+1},$$

where $\mathcal{S}^{new}$ is defined in terms of the assertion β_{new} obtained as follows: let $\beta \in \mathcal{S}_{i_\beta}^j$ be the first assertion in $\mathcal{S}^j$ such that there exists a PI in $\mathcal{T}_p$ applicable in $\mathcal{S}_{i_\beta}^j$ to β and let α be the lexicographically first PI applicable in $\mathcal{S}_{i_\beta}^j$ to β. β_{new} is then defined based on the syntactictal form of α and β. Specifically,

- if $\alpha = A_1 \sqsubseteq A_2$ and $\beta = A_1(a)$, then $\beta_{new} = A_2(a)$,
- if $\alpha = A \sqsubseteq \exists P$ and $\beta = A(a)$, then $\beta_{new} = P(a,a_{new})$,
- if $\alpha = \exists P \sqsubseteq A$ and $\beta = P(a,b)$, then $\beta_{new} = A(a)$,
- if $\alpha = \exists P_1 \sqsubseteq \exists P$ and $\beta = P_1(a,b)$, then $\beta_{new} = P(a,a_{new})$, and
- if $\alpha = P_1 \sqsubseteq P_2$ and $\beta = P_1(a_1,a_2)$, then $\beta_{new} = P_2(a_1,a_2)$,

where a_{new} is constructed from α and β as follows:

- if $a \in N_I^\mathcal{K}$, then $a_{new} = x_{aP}^{i_\beta}$, and
- otherwise $a \notin N_I^\mathcal{K}$, then a is of the form $x_{a'P_1...P_l}^{i_1...i_l}$ and we define $a_{new} = x_{a'P_1...P_lP}^{i_1...i_li_\beta}$.

If β_{new} is rigid, then $\mathcal{S}^{new} = (\{\beta_{new}\})_{0 \leqslant i \leqslant n+1}$, otherwise, $\mathcal{S}^{new} = (\mathcal{S}_i^{new})_{0 \leqslant i \leqslant n+1}$ with $\mathcal{S}_{i_\beta}^{new} = \{\beta_{new}\}$ and $\mathcal{S}_i^{new} = \emptyset$ for $i \neq i_\beta$.

Let N be the total number of assertions in $\mathcal{S}^j$. The assertion(s) added are numbered as follows: if β_{new} is not rigid, β_{new} is numbered by $N + 1$, otherwise for every $i \in [0,n+1]$, the assertion $\beta_{new} \in \mathcal{S}_i^{new}$ added to $\mathcal{S}_i^j$ is numbered by $N + 1 + i$.

We call the rigid chase of the TKB $\mathcal{K}$, denoted by $chase_{rig}(\mathcal{K}) = (chase_{rig}^\mathcal{K}(\mathcal{A}_i))_{0 \leqslant i \leqslant n+1}$, the sequence of sets of assertions obtained as the infinite union of all $\mathcal{S}^j$, i.e.,

$$(chase_{rig}^\mathcal{K}(\mathcal{A}_i))_{0 \leqslant i \leqslant n+1} = (\bigcup_{j \in \mathbb{N}} \mathcal{S}_i^j)_{0 \leqslant i \leqslant n+1}.$$

Based on the rigid chase of $\mathcal{K}$, we construct the sequence of interpretations $\mathcal{J}_{\mathcal{K}} = (\mathcal{I}_i)_{i \geqslant 0}$, where $\mathcal{I}_i = \langle \Delta, \cdot^{\mathcal{I}_i} \rangle$ is defined as follows.

- $\Delta = \mathsf{N}_\mathsf{I}^{\mathcal{K}} \cup \Gamma_N$, where Γ_N is the set of individuals that appear in $chase_{\mathrm{rig}}(\mathcal{K})$ and not in $\mathcal{K}$.
- For every $a \in \Delta$, $a^{\mathcal{I}_i} = a$.
- For every $A \in \mathsf{N}_\mathsf{C}$, $A^{\mathcal{I}_i} = \{a \mid A(a) \in chase_{\mathrm{rig}}^{\mathcal{K}}(\mathcal{A}_i)\}$ if $i \leqslant n$, $A^{\mathcal{I}_i} = \{a \mid A(a) \in chase_{\mathrm{rig}}^{\mathcal{K}}(\mathcal{A}_{n+1})\}$ if $i > n$.
- For every $R \in \mathsf{N}_\mathsf{R}$, $R^{\mathcal{I}_i} = \{(a,b) \mid R(a,b) \in chase_{\mathrm{rig}}^{\mathcal{K}}(\mathcal{A}_i)\}$ if $i \leqslant n$, $R^{\mathcal{I}_i} = \{(a,b) \mid R(a,b) \in chase_{\mathrm{rig}}^{\mathcal{K}}(\mathcal{A}_{n+1})\}$ if $i > n$.

We show that $\mathcal{J}_{\mathcal{K}}$ is a model of $\mathcal{K}$ that respects rigid predicates, and that for any BTCQ without negation ϕ such that $\mathsf{N}_\mathsf{I}^{\phi} \subseteq \mathsf{N}_\mathsf{I}^{\mathcal{K}}$, $\mathcal{K}, p \models \phi$ iff $\mathcal{J}_{\mathcal{K}}, p \models \phi$.

Lemma 9.3. *If $\mathcal{K}$ is consistent, then $\mathcal{J}_{\mathcal{K}}$ is a model of $\mathcal{K}$ that respects rigid predicates.*

Proof (Sketch). Since for every $i \in [0,n]$, $\mathcal{A}_i \subseteq chase_{\mathrm{rig}}^{\mathcal{K}}(\mathcal{A}_i)$, we directly obtain $\mathcal{I}_i \models \mathcal{A}_i$. We can show that for every i, $\mathcal{I}_i$ further satisfies every PI in $\mathcal{T}$ with similar arguments as those used in [50]. Indeed, every PI applicable to an assertion β in $\mathcal{S}_i^j$ at step j of the construction of the rigid chase becomes not applicable to β in $\mathcal{S}_i^k$ for some $k \geqslant j$, because there are neither infinitely many assertions before β, nor infinitely many PIs applied to some assertion that precedes β. Finally, we show that $\mathcal{I}_i$ satisfies every negative inclusion of $\mathcal{T}$ because otherwise $\mathcal{K}$ would be inconsistent. Moreover, the model $\mathcal{J}_{\mathcal{K}}$ respects rigid predicates because, if an assertion β of $chase_{\mathrm{rig}}^{\mathcal{K}}(\mathcal{A}_i)$ is rigid, either $\beta \in \mathcal{A}_i$ and by construction $\beta \in \mathcal{S}_k^0 = \mathcal{A}_k'$ for every k, or β has been derived at some step j by applying some PI to an assertion of $\mathcal{S}^j$ and $\beta \in \mathcal{S}_k^{j+1}$ for every k. We obtain that in both cases, $\beta \in chase_{\mathrm{rig}}^{\mathcal{K}}(\mathcal{A}_k)$ for every k. $\square$

Next, we show that $\mathcal{J}_{\mathcal{K}}$ is the canonical model of $\mathcal{K}$ for entailment of BTCQ without negation.

Lemma 9.4. *If $\mathcal{K}$ is consistent, then for every BTCQ without negation ϕ such that $\mathsf{N}_\mathsf{I}^{\phi} \subseteq \mathsf{N}_\mathsf{I}^{\mathcal{K}}$, $\mathcal{K}, p \models \phi$ iff $\mathcal{J}_{\mathcal{K}}, p \models \phi$.*

Proof (Sketch). Since $\mathcal{J}_{\mathcal{K}} = (\mathcal{I}_i)_{i \geqslant 0}$ with $\mathcal{I}_i = \langle \Delta, \cdot^{\mathcal{I}_i} \rangle$ is a model of $\mathcal{K}$ that respects rigid predicates, the first direction is straightforward, and we only need to show that $\mathcal{J}_{\mathcal{K}}, p \models \phi$ implies $\mathcal{K}, p \models \phi$. Let $\mathcal{J} = (\mathcal{I}_i')_{i \geqslant 0}$ with $\mathcal{I}_i' = \langle \Delta', \cdot^{\mathcal{I}_i'} \rangle$ be a model of $\mathcal{K}$ that respects rigid predicates. We show by structural induction on ϕ that if $\mathcal{J}_{\mathcal{K}}, p \models \phi$, then $\mathcal{J}, p \models \phi$. For the case where ϕ

is a CQ $\exists \vec{y}.\psi(\vec{y})$, we show that if there exists a homomorphism π of $\exists \vec{y}.\psi(\vec{y})$ into $\mathcal{I}_p$, then $\mathcal{I}_p' \models \exists \vec{y}.\psi(\vec{y})$, by defining a homomorphism h from Δ into Δ'. $\square$

We are now ready to introduce the set $\mathcal{R}$ that, if added to every ABox of the TKB, allows us to reduce TCQ answering with rigid predicates to TCQ answering without.

Proposition 9.5. *Let $\mathcal{R}$ be as defined in Figure 6. Then, $\mathcal{R}$ is computable in polynomial time and such that*

1. *$\mathcal{K}$ is consistent iff $\mathcal{K}_{\mathcal{R}} = \langle \mathcal{T}, (\mathcal{A}_i \cup \mathcal{R})_{0 \leqslant i \leqslant n} \rangle$ is consistent with $\mathsf{N}_\mathsf{RC} = \mathsf{N}_\mathsf{RR} = \emptyset$, and*
2. *for any BTCQ ϕ without negation and unbounded future operators and such that $\mathsf{N}_\mathsf{I}^{\phi} \subseteq \mathsf{N}_\mathsf{I}^{\mathcal{K}}$, $\mathcal{K}, p \models \phi$ iff $\mathcal{K}_{\mathcal{R}}, p \models \phi$ with $\mathsf{N}_\mathsf{RC} = \mathsf{N}_\mathsf{RR} = \emptyset$.*

The size of $\mathcal{R}$ is polynomial in the size of $\mathsf{N}_\mathsf{C}^{\mathcal{K}}, \mathsf{N}_\mathsf{R}^{\mathcal{K}}$, and $\mathsf{N}_\mathsf{I}^{\mathcal{K}}$, and since query answering as well as subsumption checking are in P, $\mathcal{R}$ can be computed in polynomial time. The first three parts of $\mathcal{R}$ contain information about the participation of individuals of $\mathsf{N}_\mathsf{I}^{\mathcal{K}}$ in rigid predicates. The last two witness the participation in rigid predicates of the role-successors w.r.t. non-rigid roles, and thus take into account also anonymous individuals that are created in $chase_{\mathrm{rig}}(\mathcal{K})$ when applying PIs whose right-hand side is an existential restriction of a non-rigid role. The individuals created in $chase_{\mathrm{rig}}(\mathcal{K})$ when applying such a PI with a rigid role are witnessed by the individuals x_{aP} or $x_{P_1 P_2}$ if they do not follow from a rigid role assertion, and do not need to be witnessed otherwise, since the assertion $P_2(x_{P_1}, x_{P_1 P_2})$ is sufficient to trigger the generation of the whole anonymous part implied by the fact that $x_{P_1 P_2}$ is in the range of P_2.

We split the proof of Proposition 9.5 into several lemmas.

Lemma 9.6. *$\mathcal{K}$ is consistent iff $\mathcal{K}_{\mathcal{R}}$ is consistent with $\mathsf{N}_\mathsf{RC} = \mathsf{N}_\mathsf{RR} = \emptyset$.*

Proof. By Proposition 3.5, $\mathcal{K}_{\mathcal{R}}$ is consistent with $\mathsf{N}_\mathsf{RC} = \mathsf{N}_\mathsf{RR} = \emptyset$ iff each $\langle \mathcal{T}, \mathcal{A}_i \cup \mathcal{R} \rangle$ is consistent. We show that also $\mathcal{K}$ is consistent iff each $\langle \mathcal{T}, \mathcal{A}_i \cup \mathcal{R} \rangle$ is consistent.

For the case where $\mathcal{K}$ is not consistent, let $\mathcal{B}$ be a minimal inconsistent subset of $\mathcal{K}$. Recall that by Fact 6.11, $\mathcal{B}$ contains at most 2 assertions. Then $\mathcal{B}$ is either internal to some $\mathcal{A}_i$, or is of the form $\mathcal{B} = \{(\alpha, i), (\beta, j)\}$ with $i \neq j$. In the former case, $\langle \mathcal{T}, \mathcal{A}_i \cup \mathcal{R} \rangle$ is clearly inconsistent. In the latter case,

$$\mathcal{R} = \{A(a) \mid A \in \mathsf{N}_{\mathsf{RC}}^{\mathcal{K}}, \ a \in \mathsf{N}_{\mathsf{I}}^{\mathcal{K}}, \ \text{and} \ \exists i, \langle \mathcal{T}, \mathcal{A}_i \rangle \models A(a)\} \ \cup$$

$$\{R(a,b) \mid R \in \mathsf{N}_{\mathsf{RR}}^{\mathcal{K}}, \ a,b \in \mathsf{N}_{\mathsf{I}}^{\mathcal{K}}, \ \text{and} \ \exists i, \langle \mathcal{T}, \mathcal{A}_i \rangle \models R(a,b)\} \ \cup$$

$$\{P(a, x_{aP}) \mid P := R \mid R^-, \ R \in \mathsf{N}_{\mathsf{RR}}^{\mathcal{K}}, \ a \in \mathsf{N}_{\mathsf{I}}^{\mathcal{K}}, \ \text{and} \ \exists i, \langle \mathcal{T}, \mathcal{A}_i \rangle \models \exists x.P(a,x)\} \ \cup$$

$$\{A(x_{P_1}) \mid A \in \mathsf{N}_{\mathsf{RC}}^{\mathcal{K}}, \ P_1 := S \mid S^-, \ S \in \mathsf{N}_{\mathsf{R}}^{\mathcal{K}} \backslash \mathsf{N}_{\mathsf{RR}}^{\mathcal{K}}, \ \mathcal{T} \models \exists P_1^- \sqsubseteq A, \ \text{and} \ \exists i, \langle \mathcal{T}, \mathcal{A}_i \rangle \models \exists xy.P_1(x,y)\} \ \cup$$

$$\{P_2(x_{P_1}, x_{P_1 P_2}) \mid P_2 := R \mid R^-, \ R \in \mathsf{N}_{\mathsf{RR}}^{\mathcal{K}}, \ P_1 := S \mid S^-, \ S \in \mathsf{N}_{\mathsf{R}}^{\mathcal{K}} \backslash \mathsf{N}_{\mathsf{RR}}^{\mathcal{K}}, \ \mathcal{T} \models \exists P_1^- \sqsubseteq \exists P_2 \ \text{and}$$

$$\exists i, \langle \mathcal{T}, \mathcal{A}_i \rangle \models \exists xy.P_1(x,y)\}$$

Figure 6. Set of rigid assertions added to every ABox of $\mathcal{K}$

$\{\alpha, \beta\}$ violates some negative inclusion in the closure of the TBox that involves at least a rigid concept A or a rigid role R by assigning an individual a (or two individuals a, b) to two disjoint concepts (or roles). We can then assume w.l.o.g. that we are in one of the following cases: (i) $\langle \mathcal{T}, \alpha \rangle \models A(a)$, (ii) $\langle \mathcal{T}, \alpha \rangle \models \exists x.R(a,x)$, (iii) $\langle \mathcal{T}, \alpha \rangle \models \exists x.R(x,a)$, or (iv) $\langle \mathcal{T}, \alpha \rangle \models R(a,b)$. It follows that respectively (i) $\langle \mathcal{T}, \mathcal{A}_i \rangle \models A(a)$, (ii) $\langle \mathcal{T}, \mathcal{A}_i \rangle \models \exists x.R(a,x)$, (iii) $\langle \mathcal{T}, \mathcal{A}_i \rangle \models \exists x.R(x,a)$, or (iv) $\langle \mathcal{T}, \mathcal{A}_i \rangle \models R(a,b)$. By construction of $\mathcal{R}$, we then conclude that (i) $A(a) \in \mathcal{R}$, (ii) $R(a, x_{aR}) \in \mathcal{R}$, (iii) $R(x_{aR^-}, a) \in \mathcal{R}$, or (iv) $R(a,b) \in \mathcal{R}$ respectively, and therefore that $\langle \mathcal{T}, \mathcal{A}_j \cup \mathcal{R} \rangle$ is inconsistent.

For the other direction, assume there exists $i \in [0,n]$, such that $\langle \mathcal{T}, \mathcal{A}_i \cup \mathcal{R} \rangle$ is inconsistent, and let $\mathcal{B}$ be a minimal inconsistent subset of $\langle \mathcal{T}, \mathcal{A}_i \cup \mathcal{R} \rangle$. If $\mathcal{B}$ is internal to $\mathcal{A}_i$, $\mathcal{K}$ is clearly inconsistent. Otherwise, $\mathcal{B}$ is of the form $\{\alpha, \beta\}$ and involves at least one assertion from $\mathcal{R}$. The assertions α and β assign an individual x to two disjoint concepts C_1, C_2, or a pair of individuals x, y to two disjoint roles R_1, R_2. We distinguish three cases.

1. In the case where $x = x_{aP}$ (resp. $x = x_{P_1 P_2}$), since $P(a, x_{aP})$ (resp. $P_2(x_{P_1}, x_{P_1 P_2})$) is the only assertion of $\mathcal{R}$ that contains x, we obtain that $\langle \mathcal{T}, \{P(a, x_{aP})\}\rangle$ (resp. $\langle \mathcal{T}, \{P_2(x_{P_1}, x_{P_1 P_2})\}\rangle$) is inconsistent, which implies that P (resp. P_2) is unsatisfiable, i.e., has an empty interpretation in every model of $\mathcal{T}$. Since there exists j such that $\langle \mathcal{T}, \mathcal{A}_j \rangle \models \exists x.P(a,x)$ (resp. $\langle \mathcal{T}, \mathcal{A}_j \rangle \models \exists xy.P_1(x,y)$ and $\mathcal{T} \models \exists P_1^- \sqsubseteq \exists P_2$), it follows that $\mathcal{A}_j$ is inconsistent.
2. In the case where $x = x_{P_1}$, since x_{P_1} appears only in concepts that subsume $\exists P_1^-$, the fact that x is assigned to two disjoint concepts implies that $\exists P_1^-$ is

unsatisfiable. Thus, and since there exists j such that $\langle \mathcal{T}, \mathcal{A}_j \rangle \models \exists xy.P_1(x,y)$, $\mathcal{A}_j$ is inconsistent.
3. Finally, in the case where $x \in \mathsf{N}_{\mathsf{I}}^{\mathcal{K}}$, since α or β is in $\mathcal{R}$, at least one of C_1, C_2 (or R_1, R_2) is rigid. As the case where some ABox $\mathcal{A}_j$ is inconsistent is trivial, we assume every $\mathcal{A}_j$ to be consistent. If $\alpha \notin \mathcal{A}_i$, let $c_\alpha \in \mathcal{A}_{j_\alpha}$ be an assertion responsible for the entailment that triggered the addition of α to $\mathcal{R}$, and otherwise let $(c_\alpha, j_\alpha) = (\alpha, i)$. If $\beta \notin \mathcal{A}_i$, let $c_\beta \in \mathcal{A}_{j_\beta}$ be an assertion responsible for the entailment that triggered the addition of β to $\mathcal{R}$, and otherwise $(c_\beta, j_\beta) = (\beta, i)$. Then $\{(c_\alpha, j_\alpha), (c_\beta, j_\beta)\}$ is inconsistent because c_α and c_β lead to a (or a, b) being assigned to two disjoint concepts (or disjoint roles) such that at least one of them is rigid. As a result, $\mathcal{K}$ is inconsistent as well.

We obtain that in every case, if $\mathcal{K}_{\mathcal{R}}$ is inconsistent with $\mathsf{N}_{\mathsf{RC}} = \mathsf{N}_{\mathsf{RR}} = \emptyset$, then $\mathcal{K}$ is inconsistent with rigid predicates. $\square$

We now assume that $\mathcal{K}$ and $\mathcal{K}_{\mathcal{R}}$ are consistent. Note that if this is not the case, they both trivially entail any BTCQ. The following two lemmas show that if a Boolean conjunctive query $q = \exists \vec{y}.\psi(\vec{y})$ is such that $\mathsf{N}_{\mathsf{I}}^q \subseteq \mathsf{N}_{\mathsf{I}}^{\mathcal{K}}$, then for every $p \in [0,n]$, $\mathcal{K}_{\mathcal{R}}, p \models q$ iff $\mathcal{K}, p \models q$.

Lemma 9.7. *Let $q = \exists \vec{y}.\psi(\vec{y})$ be such that $\mathsf{N}_{\mathsf{I}}^q \subseteq \mathsf{N}_{\mathsf{I}}^{\mathcal{K}}$. Then, for every $p \in [0,n]$, if $\mathcal{K}_{\mathcal{R}}, p \models q$ then $\mathcal{K}, p \models q$.*

Proof (Sketch). By Lemma 9.4, it suffices to focus on the canonical model $\mathcal{J}_{\mathcal{K}} = (\mathcal{I}_i)_{i \geqslant 0}$ for testing entailments of $\mathcal{K}$. The lemma can be shown by defining a homomorphism from the canonical model of $\langle \mathcal{T}, \mathcal{A}_p \cup \mathcal{R} \rangle$ into $\mathcal{I}_p$, the pth component of $\mathcal{J}_{\mathcal{K}}$. $\square$

Lemma 9.8. *Let $q = \exists \vec{y}.\psi(\vec{y})$ be such that $\mathsf{N}_\mathsf{I}^q \subseteq \mathsf{N}_\mathsf{I}^{\mathcal{K}}$. For every $p \in [0, n]$, if $\mathcal{K}, p \models q$ then $\mathcal{K}_{\mathcal{R}}, p \models q$.*

Proof (Sketch). Again, we make use of Lemma 9.4 and use the canonical model $\mathcal{J}_{\mathcal{K}} = (\mathcal{I}_i)_{i \geqslant 0}$. The lemma can then be shown by considering some model $\mathcal{I}_p^{\mathcal{R}}$ of $\langle \mathcal{T}, \mathcal{A}_p \cup \mathcal{R} \rangle$, and defining a homomorphism of $\mathcal{I}_p$ into $\mathcal{I}_p^{\mathcal{R}}$. □

Since by Lemmas 9.7 and 9.8, $\mathcal{K}$ and $\mathcal{K}_{\mathcal{R}}$ with $\mathsf{N}_{\mathsf{RC}} = \mathsf{N}_{\mathsf{RR}} = \emptyset$ coincide on the entailment of BCQs for every time point $p \in [0, n]$, we can show as in Proposition 7.6 that they coincide on entailment of BTCQs without negation nor unbounded future operators.

Lemma 9.9. *Let ϕ be a BTCQ with no negation or unbounded future operators such that $\mathsf{N}_\mathsf{I}^\phi \subseteq \mathsf{N}_\mathsf{I}^{\mathcal{K}}$. Then, $\mathcal{K}, p \models \phi$ iff $\mathcal{K}_{\mathcal{R}}, p \models \phi$ with $\mathsf{N}_{\mathsf{RC}} = \mathsf{N}_{\mathsf{RR}} = \emptyset$.*

It follows that TCQs can be answered in $\mathcal{K}$ with rigid predicates by answering TCQs in $\mathcal{K}_{\mathcal{R}}$ without rigid predicates and pruning answers that contain individual names not in $\mathsf{N}_\mathsf{I}^{\mathcal{K}}$. Note that every model of $\mathcal{K}_{\mathcal{R}}$ is a model of $\mathcal{K}$, but does not respect rigid predicates in general. We can reduce BTCQ entailment over $\mathcal{K}$ with rigid predicates to BTCQ entailment over $\mathcal{K}_{\mathcal{R}}$ without rigid predicates only because our TCQs do not allow LTL operators to be nested in existential quantifications. This prevents existentially quantified variables to link different time points. To see this, consider the query $\exists xy.\square^b(R(a, x) \wedge R(x, y))$ and the TKB $\mathcal{K} = \langle \mathcal{T}, (\mathcal{A}_i)_{0 \leqslant i \leqslant n} \rangle$ with $\mathcal{T} = \{B \sqsubseteq \exists R, \exists R^- \sqsubseteq \exists R\}$, $R \in \mathsf{N}_{\mathsf{RR}}$ and $\mathcal{A}_i = \{B(a)\}$. For this TKB, we would have $\mathcal{R} = \{R(a, x_{aR})\}$, and therefore x_{aR} could have a different R-successors in each interpretation of a model of $\mathcal{K}_{\mathcal{R}}$, thus y cannot be mapped to the same object at every time point.

Remark 9.10. In the case of streaming data, if we want to take into account a newly available dataset, we do not need to fully recompute $\mathcal{R}$: we only need to add the new rigid assertions that can be derived from the new dataset. Moreover, if we only reason over a window of n time points from our stream, we can annotate the assertions in $\mathcal{R}$ with a counter that is initialised with n and decremented with each new time point. If an existing assertion is derived again, it is reset to n. Assertions are then removed from $\mathcal{R}$ if their counter reaches 0.

Remark 9.11. The main goal of the approaches presented in [18, 43] for TCQ answering in DL-Lite$_{\mathcal{R}}$ is to obtain the query answers at the last time point without storing all the data for all previous time points. Their algorithm uses a *bounded history encoding*, which means that the space required by the algorithm is constant w.r.t. the number n of previous time points: only the current dataset and some auxiliary relations required for computing the query answers are stored and updated at each time point.

Unfortunately, with rigid predicates present, our approach does not achieve bounded history encoding, since the answers of the subqueries of ϕ at previous time points may change when new rigid assertions are derived from the last dataset. However, if the algorithm of [18, 43] has this property, it requires exponential space w.r.t. $\mathsf{N}_{\mathsf{RC}}^{\mathcal{K}}$ and $\mathsf{N}_\mathsf{I}^{\mathcal{K}}$ which can also be problematic, while our algorithm requires only polynomial space and time. To achieve bounded history encoding (but in exponential time w.r.t. $\mathsf{N}_{\mathsf{RC}}^{\mathcal{K}}$, $\mathsf{N}_{\mathsf{RR}}^{\mathcal{K}}$ and $\mathsf{N}_\mathsf{I}^{\mathcal{K}}$), we could adapt the algorithm of [18, 43] to support rigid roles. We would consider all possible sets $\mathcal{R}$ built from $\mathsf{N}_{\mathsf{RC}}^{\mathcal{K}}$, $\mathsf{N}_{\mathsf{RR}}^{\mathcal{K}}$ and $\mathsf{N}_\mathsf{I}^{\mathcal{K}}$ following the form of Figure 6, then verify at each time point whether $\mathcal{R}$ is consistent with $\mathcal{A}_i$ and $\mathcal{T}$ and contains all rigid assertions that can be derived from $\mathcal{A}_i$ as described in Figure 6.

Our procedure only applies to TBoxes in which rigid roles do not have non-rigid subroles. A possible direction to alleviate these restrictions would be to use ideas similar to those developed in [44] for CQ answering over DL-Lite$_{\mathcal{R}}$-KB using the combined approach. This CQ answering approach saturates the data by adding to the ABox every assertion that can be derived, introducing individual names to witness existential role restrictions, and then uses a special rewriting to prune spurious answers. In our setting, we could model infinite chains of rigid roles by adding cycles of rigid roles to $\mathcal{R}$, then prune the spurious answers resulting from these cycles.

Regarding $\mathcal{EL}_\perp$, we conjecture that we could have a similar approach for rigid predicates. The main difference would be that since in $\mathcal{EL}_\perp$ several assertions may be needed to derive one, $\mathcal{R}$ would have to be computed iteratively, taking into account its own assertions to derive new ones until a fix-point is reached. Moreover, the problem of infinite chains of rigid roles that cannot be entailed by a polynomial set of assertions would appear as soon as $\mathsf{N}_{\mathsf{RR}} \neq \emptyset$. The combined approach for $\mathcal{EL}$ [11] could provide ideas to overcome this difficulty.

Table 3

Entailment under classical or IAR semantics from a $\mathcal{L}$-TKB without rigid predicates and such that $\mathcal{L}$ has the canonical model property for BCQ entailment

ϕ	$\mathcal{K}, p \models_S \phi$ iff
$\exists \vec{y}.\psi(\vec{y})$	$p \leqslant n$ and $\langle \mathcal{T}, \mathcal{A}_p \rangle \models_S \exists \vec{y}.\psi(\vec{y})$
$\phi_1 \wedge \phi_2$	$\mathcal{K}, p \models_S \phi_1$ and $\mathcal{K}, p \models_S \phi_2$
$\phi_1 \vee \phi_2$	$\mathcal{K}, p \models_S \phi_1$ or $\mathcal{K}, p \models_S \phi_2$
$\bigcirc \phi_1$	$\mathcal{K}, p+1 \models_S \phi_1$
$\bullet^b \phi_1$	$p < n$ implies $\mathcal{K}, p+1 \models_S \phi_1$
$\bigcirc^- \phi_1$	$p > 0$ and $\mathcal{K}, p-1 \models_S \phi_1$
$\bullet^- \phi_1$	$p > 0$ implies $\mathcal{K}, p-1 \models_S \phi_1$
$\Box \phi_1$	$\forall k, k \geqslant p, \mathcal{K}, k \models \phi_1$
$\Box^b \phi_1$	$\forall k, p \leqslant k \leqslant n, \mathcal{K}, k \models \phi_1$
$\Box^- \phi_1$	$\forall k, 0 \leqslant k \leqslant p, \mathcal{K}, k \models_S \phi_1$
$\Diamond \phi_1$	$\exists k, k \geqslant p, \mathcal{K}, k \models \phi_1$
$\Diamond^b \phi_1$	$\exists k, p \leqslant k \leqslant n, \mathcal{K}, k \models \phi_1$
$\Diamond^- \phi_1$	$\exists k, 0 \leqslant k \leqslant p, \mathcal{K}, k \models_S \phi_1$
$\phi_1 \mathsf{U} \phi_2$	$\exists k, k \geqslant p, \mathcal{K}, k \models \phi_2$ and $\forall j, p \leqslant j < k, \mathcal{K}, j \models \phi_1$
$\phi_1 \mathsf{U}^b \phi_2$	$\exists k, p \leqslant k \leqslant n, \mathcal{K}, k \models_S \phi_2$ and $\forall j, p \leqslant j < k, \mathcal{K}, j \models_S \phi_1$
$\phi_1 \mathsf{S} \phi_2$	$\exists k, 0 \leqslant k \leqslant p, \mathcal{K}, k \models_S \phi_2$ and $\forall j, k < j \leqslant p, \mathcal{K}, j \models_S \phi_1$

9.2. Inconsistency-tolerant TCQ answering without rigid predicates

In this section $\mathcal{K}$ is a $\mathcal{L}$-TKB and ϕ a TCQ without negation.

When $\mathsf{N_{RC}} = \mathsf{N_{RR}} = \emptyset$, an important consequence of Proposition 3.5 is that the repairs of $\mathcal{K}$ are all possible sequences $(\mathcal{A}_i')_{0 \leqslant i \leqslant n}$ where $\mathcal{A}_i'$ is a repair of $\langle \mathcal{T}, \mathcal{A}_i \rangle$, so the intersection of the repairs of $\mathcal{K}$ is $(\mathcal{A}_i^\cap)_{0 \leqslant i \leqslant n}$ where $\mathcal{A}_i^\cap$ is the intersection of the repairs of $\langle \mathcal{T}, \mathcal{A}_i \rangle$. This allows us to show that the entailment (resp. IAR entailment) of a BTCQ without negation from a consistent (resp. possibly inconsistent) TKB in a DL $\mathcal{L}$ that has the canonical model property for BCQ entailment can be equivalently defined w.r.t. the entailment (resp. IAR entailment) of the BCQs it contains as follows.

Proposition 9.12. *If $\mathcal{L}$ has the canonical model property for BCQ entailment and $\mathsf{N_{RC}} = \mathsf{N_{RR}} = \emptyset$, then the entailments shown in Table 3 hold for $S = $ classical when $\mathcal{K}$ is consistent, and for $S = $ IAR.*

Proof. For the consistent case, all relations in Table 3 but the first one are straightforwardly obtained by applying the definitions of BTCQ satisfaction of Table 1 in Section 2 to the canonical model of $\mathcal{K}$. Moreover, by Proposition 3.6, if $p \leqslant n$, then $\mathcal{K}, p \models \exists \vec{y}.\psi(\vec{y})$ iff $\langle \mathcal{T}, \mathcal{A}_p \rangle \models \exists \vec{y}.\psi(\vec{y})$. Finally, $\mathcal{K}, p \not\models \exists \vec{y}.\psi(\vec{y})$ if $p > n$, because there exists a model of $\mathcal{K}$ whose p^{th} component interprets every predicate as the empty set.

For IAR semantics, let $(\mathcal{A}_i^{\mathrm{ir}})_{0 \leqslant i \leqslant n}$ denote the intersection of the repairs of $\mathcal{K}$ and $\mathcal{A}_i^\cap$ denote the intersection of the repairs of $\langle \mathcal{T}, \mathcal{A}_i \rangle$.

- $\mathcal{K}, p \models_{\mathrm{IAR}} \exists \vec{y}.\psi(\vec{y})$ iff $\langle \mathcal{T}, (\mathcal{A}_i^{\mathrm{ir}})_{0 \leqslant i \leqslant n} \rangle, p \models \exists \vec{y}.\psi(\vec{y})$, i.e., iff $p \leqslant n$ and $\langle \mathcal{T}, \mathcal{A}_p^{\mathrm{ir}} \rangle \models \exists \vec{y}.\psi(\vec{y})$ because $(\mathcal{A}_i^{\mathrm{ir}})_{0 \leqslant i \leqslant n}$ is consistent. Since the repairs of $\mathcal{K}$ are the sequences of the repairs of the $\langle \mathcal{T}, \mathcal{A}_i \rangle$, $\mathcal{A}_p^{\mathrm{ir}} = \mathcal{A}_p^\cap$, so $\mathcal{K}, p \models_{\mathrm{IAR}} \exists \vec{y}.\psi(\vec{y})$ iff $p \leqslant n$ and $\langle \mathcal{T}, \mathcal{A}_p \rangle \models_{\mathrm{IAR}} \exists \vec{y}.\psi(\vec{y})$.
- $\mathcal{K}, p \models_{\mathrm{IAR}} \phi_1 \wedge \phi_2$ iff $\langle \mathcal{T}, (\mathcal{A}_i^{\mathrm{ir}})_{0 \leqslant i \leqslant n} \rangle, p \models \phi_1 \wedge \phi_2$, i.e., iff $\langle \mathcal{T}, (\mathcal{A}_i^{\mathrm{ir}})_{0 \leqslant i \leqslant n} \rangle, p \models \phi_1$ and $\langle \mathcal{T}, (\mathcal{A}_i^{\mathrm{ir}})_{0 \leqslant i \leqslant n} \rangle, p \models \phi_2$ because $(\mathcal{A}_i^{\mathrm{ir}})_{0 \leqslant i \leqslant n}$ is consistent. It follows that $\mathcal{K}, p \models_{\mathrm{IAR}} \phi_1 \wedge \phi_2$ iff $\mathcal{K}, p \models_{\mathrm{IAR}} \phi_1$ and $\mathcal{K}, p \models_{\mathrm{IAR}} \phi_2$.
- We show all remaining relations in the same way, applying the definition of IAR semantics and using the fact that $(\mathcal{A}_i^{\mathrm{ir}})_{0 \leqslant i \leqslant n}$ is consistent. $\square$

This is an important result, since it implies that answering temporal CQs under IAR semantics can be done with the algorithms developed for the consistent case (see [18, 43] for algorithms for DL-Lite$_\mathcal{R}$ without unbounded future operators) by replacing classical CQ answering by IAR CQ answering (see [30, 37, 38] for algorithms for DL-Lite$_\mathcal{R}$). The following example shows that this is unfortunately not true for brave or AR semantics.

Example 9.13. Consider the TKB $\mathcal{K} = \langle \mathcal{T}, (\mathcal{A}_i)_{0 \leqslant i \leqslant n} \rangle$ and the TCQ ϕ defined as follows.

$$\mathcal{T} = \{T \sqsubseteq \neg F\}$$
$$\mathcal{A}_i = \{T(a), F(a)\} \text{ for } 0 \leqslant i \leqslant n$$
$$\phi = \Box^- (T(a) \wedge \bullet^- F(a))$$

Now, $\mathcal{K}, k \models_{\mathrm{brave}} T(a) \wedge \bullet^- F(a)$ for every $0 \leqslant k \leqslant n$, but $\mathcal{K}, n \not\models_{\mathrm{brave}} \phi$. This is because the same repair cannot entail $T(a) \wedge \bullet^- F(a)$ both at time point k and $k+1$, since it would contain both $(T(a), k)$ and $(F(a), k)$ which is not possible. For AR semantics, consider $\phi = T(a) \vee F(a)$ over the TKB $\mathcal{K}$: while ϕ holds under AR semantics at each time point, neither $T(a)$ nor $F(a)$ does.

However, if the operators allowed in the TCQ are restricted to $\wedge, \bigcirc, \bullet^b, \bigcirc^-, \bullet^-, \Box, \Box^b$, and $\Box^-$, then AR TCQ answering can be done with the algorithms developed for the consistent case by simply replacing classical CQ answering by AR CQ answering (see [38] for algorithms for DL-Lite$_\mathcal{R}$). Indeed, for these operators, the relations of Proposition 9.12 hold for $S = $ AR.

- $\mathcal{K}, p \models_{\mathrm{AR}} \exists \vec{y}.\psi(\vec{y})$ iff for every repair $(\mathcal{A}'_i)_{0 \leqslant i \leqslant n}$ of $\mathcal{K}$, $\langle \mathcal{T}, (\mathcal{A}'_i)_{0 \leqslant i \leqslant n} \rangle, p \models \exists \vec{y}.\psi(\vec{y})$, i.e., iff for every repair $(\mathcal{A}'_i)_{0 \leqslant i \leqslant n}$ of $\mathcal{K}$, $p \leqslant n$ and $\langle \mathcal{T}, \mathcal{A}'_p \rangle \models \exists \vec{y}.\psi(\vec{y})$ because $(\mathcal{A}'_i)_{0 \leqslant i \leqslant n}$ is consistent. Since the repairs of $\mathcal{K}$ are the sequences of the repairs of the $\langle \mathcal{T}, \mathcal{A}_i \rangle$, this is the case iff $p \leqslant n$ and for every repair $\mathcal{A}'_p$ of $\langle \mathcal{T}, \mathcal{A}_p \rangle$, $\langle \mathcal{T}, \mathcal{A}'_p \rangle \models \exists \vec{y}.\psi(\vec{y})$, i.e., iff $p \leqslant n$ and $\langle \mathcal{T}, \mathcal{A}_p \rangle \models_{\mathrm{AR}} \exists \vec{y}.\psi(\vec{y})$.

- $\mathcal{K}, p \models_{\mathrm{AR}} \phi_1 \wedge \phi_2$ iff for every repair $(\mathcal{A}'_i)_{0 \leqslant i \leqslant n}$ of $\mathcal{K}$, $\langle \mathcal{T}, (\mathcal{A}'_i)_{0 \leqslant i \leqslant n} \rangle, p \models \phi_1 \wedge \phi_2$, i.e., iff for every repair $(\mathcal{A}'_i)_{0 \leqslant i \leqslant n}$ of $\mathcal{K}$, $\langle \mathcal{T}, (\mathcal{A}'_i)_{0 \leqslant i \leqslant n} \rangle, p \models \phi_1$ and $\langle \mathcal{T}, (\mathcal{A}'_i)_{0 \leqslant i \leqslant n} \rangle, p \models \phi_2$ because $(\mathcal{A}'_i)_{0 \leqslant i \leqslant n}$ is consistent. It follows that $\mathcal{K}, p \models_{\mathrm{AR}} \phi_1 \wedge \phi_2$ iff $\mathcal{K}, p \models_{\mathrm{AR}} \phi_1$ and $\mathcal{K}, p \models_{\mathrm{AR}} \phi_2$.

- We show all remaining relations in the same way, applying the definition of AR semantics and using the fact that TKB repairs are consistent.

The following counter-examples show that this is not the case for the other operators: $\vee, \Diamond, \Diamond^b, \Diamond^-, \mathsf{U}, \mathsf{U}^b$, and S.

- $\mathcal{K}, 0 \models_{\mathrm{AR}} \phi_1 \vee \phi_2$ but $\mathcal{K}, 0 \not\models_{\mathrm{AR}} \phi_1$ and $\mathcal{K}, 0 \not\models_{\mathrm{AR}} \phi_2$:

$$\mathcal{T} = \{A \sqsubseteq \neg B\} \qquad \mathcal{A}_0 = \{A(a), B(a)\}$$
$$\phi_1 = A(a) \qquad \phi_2 = B(a)$$

- $\mathcal{K}, 0 \models_{\mathrm{AR}} \Diamond \phi_1$ (resp. $\mathcal{K}, 0 \models_{\mathrm{AR}} \Diamond^b \phi_1$) but for every k (resp. such that $0 \leqslant k \leqslant 2$), $\mathcal{K}, k \not\models_{\mathrm{AR}} \phi_1$:

$$\mathcal{T} = \{A \sqsubseteq \neg B\} \qquad \mathcal{A}_0 = \{A(a)\}$$
$$\mathcal{A}_1 = \{A(a), B(a)\} \qquad \mathcal{A}_2 = \{B(a)\}$$
$$\phi_1 = A(a) \wedge \bigcirc B(a)$$

- $\mathcal{K}, 0 \models_{\mathrm{AR}} \phi_1 \mathsf{U} \phi_2$ (resp. $\mathcal{K}, 0 \models_{\mathrm{AR}} \phi_1 \mathsf{U}^b \phi_2$) but for every k (resp. such that $0 \leqslant k \leqslant 2$), either $\mathcal{K}, k \not\models_{\mathrm{AR}} \phi_2$ or there exists j, such that $0 \leqslant j < k$ and $\mathcal{K}, j \not\models_{\mathrm{AR}} \phi_1$:

$$\mathcal{T} = \{A \sqsubseteq \neg B\} \qquad \mathcal{A}_0 = \{A(a)\}$$
$$\mathcal{A}_1 = \{A(a), B(a)\} \qquad \mathcal{A}_2 = \{B(a)\}$$
$$\phi_1 = A(a) \qquad \phi_2 = B(a)$$

- We can construct similar counter-examples for $\Diamond^-$ and S.

Interestingly, contrary to the brave semantics, even for general TCQs the "if" direction of Proposition 9.12 is true.

- If $\mathcal{K}, p \models_{\mathrm{AR}} \phi_1$ or $\mathcal{K}, p \models_{\mathrm{AR}} \phi_2$, then $\mathcal{K}, p \models_{\mathrm{AR}} \phi_1 \vee \phi_2$.

- If there exists $k \geqslant p$ such that $\mathcal{K}, k \models_{\mathrm{AR}} \phi_1$, then $\mathcal{K}, p \models_{\mathrm{AR}} \Diamond \phi_1$.

- If there exists k such that $p \leqslant k \leqslant n$ and $\mathcal{K}, k \models_{\mathrm{AR}} \phi_1$, then $\mathcal{K}, p \models_{\mathrm{AR}} \Diamond^b \phi_1$.

- If there exists k such that $0 \leqslant k \leqslant p$ and $\mathcal{K}, k \models_{\mathrm{AR}} \phi_1$, then $\mathcal{K}, p \models_{\mathrm{AR}} \Diamond^- \phi_1$.

- If there exists $k \geqslant p$ such that $\mathcal{K}, k \models_{\mathrm{AR}} \phi_2$ and for every j such that $p \leqslant j < k$, $\mathcal{K}, j \models_{\mathrm{AR}} \phi_1$, then $\mathcal{K}, p \models_{\mathrm{AR}} \phi_1 \mathsf{U} \phi_2$.

- If there exists k such that $p \leqslant k \leqslant n$, $\mathcal{K}, k \models_{\mathrm{AR}} \phi_2$ and for every j such that $p \leqslant j < k$, $\mathcal{K}, j \models_{\mathrm{AR}} \phi_1$, then $\mathcal{K}, p \models_{\mathrm{AR}} \phi_1 \mathsf{U}^b \phi_2$.

- If there exists k such that $0 \leqslant k \leqslant p$, $\mathcal{K}, k \models_{\mathrm{AR}} \phi_2$ and for every j such that $k < j \leqslant p$, $\mathcal{K}, j \models_{\mathrm{AR}} \phi_1$, then $\mathcal{K}, p \models_{\mathrm{AR}} \phi_1 \mathsf{S} \phi_2$.

It follows that even for unrestricted TCQs, combining algorithms for TCQ answering with algorithms for AR query answering will provide *a sound approximation* of AR answers.

For brave semantics, it would be useful to characterize the queries for which this method would be correct. Indeed, for many pairs of a TBox and a query, the minimal subsets of the TKB such that the query can be mapped into them cannot be inconsistent. For instance, for DL-Lite$_{\mathcal{R}}$-TKBs, this is the case if no pair of predicates that may be involved at the same time point appears in an NI entailed by the TBox. Consider for instance $\mathcal{T} = \{A \sqsubseteq \neg C, B \sqsubseteq \neg C\}$ and $\phi = \exists x.A(x) \wedge \Diamond(\exists x.B(x) \wedge \bigcirc(\exists x.C(x)))$. For ϕ to be entailed at time point p, $\exists x.A(x)$ should hold at p, $\exists x.B(x)$ at time point $i \geqslant p$ and $\exists x.C(x)$ at $i + 1 > p$, so there cannot be a conflict between the C and the A or B timed assertions used to satisfy the different CQs.

10. Conclusions and future work

For stream reasoning, handling the temporal dimension of the collected data and being resilient against errors in the data are expedient requirements. In the presence of erroneous data, handling inconsistencies is indispensable for logic-based approaches to stream reasoning. In this paper, we have lifted the standard inconsistency-tolerant semantics AR, IAR and brave to a temporal query answering setting that has been widely studied in the literature—namely, where the data is associated with time points and only the query language admits the use of temporal operators

from LTL. We have presented complexity results and techniques to combine temporal with inconsistency-tolerant query answering over lightweight DL temporal knowledge bases suited for ontology-mediated situation recognition.

Our main contribution is a complexity analysis of the three semantics, focusing on the DLs $\mathcal{EL}_\perp$ and DL-Lite$_\mathcal{R}$, where we distinguished the cases based on whether rigid concept or role names occur in the TKB, and on whether the query contains negation. We provided general algorithms that allow us to derive the complexity of temporal inconsistency-tolerant query answering from the complexities of consistency checking and classical entailment of temporal conjunctive queries. We furthermore completed the complexity picture for the classical semantics for TCQs without negations. Indeed, for the case where the query language does not provide negation, we devised a general approach to assess the complexity by the use of the canonical model property for B(T)CQ answering and thus not only limited to a particular DL. This approach allows us to derive the complexity of temporal query answering from the complexity of conjunctive query entailment for DLs that have this canonical model property.

Encouragingly, our analysis shows that either with or without negation in the query, in most cases, inconsistency-tolerant reasoning and temporal query answering can be combined without increasing the computational complexity. Furthermore, our results show that disallowing negation in the query language results in a drop in the combined complexity of TCQ answering, and, in the case of $\mathcal{EL}_\perp$ with rigid predicates, even in the data complexity. This raises hope that ontology-based stream reasoning applications which are resilient against noise in the data can be feasibly implemented and used in practice.

As a second major contribution, we investigated two techniques useful for developing practical algorithms for inconsistency-tolerant temporal query answering. We first showed that in DL-Lite$_\mathcal{R}$, under the classical semantics and for queries without negation and unbounded temporal operators, rigid predicates can be handled by adding a set of assertions of polynomial size to each ABox from the TKB. However, our approach, which reduces TCQ answering with rigid predicates to TCQ answering without rigid predicates, works only for BTCQ entailment under the classical semantics.

We then showed that in the case without rigid predicates and for queries without negation, TCQ answer-

ing under IAR semantics can be implemented by combining algorithms developed for TCQ answering under the classical semantics with algorithms for CQ answering under IAR semantics over atemporal KBs. Moreover, we showed that when disallowing some of the operators in the queries, the same method can be used for AR semantics, while for full TCQs without negation, it provides for a sound approximation of the AR answers. Unfortunately, this is not the case for brave semantics, which is relevant for practical applications such as recognizing highly critical situations. Thus it would be useful to characterize the queries and TBoxes for which this method is correct. Now, fully fledged practical algorithms still remain to be found for inconsistency-tolerant temporal query answering with rigid predicates.

Another interesting open question is how the complexity changes if we admit not only the qualitative temporal operators from LTL, but also metric temporal operators as found in MTL or LTL$^{\text{bin}}$ [20, 51, 52]. In LTL$^{\text{bin}}$, temporal operators can be annotated by time intervals over which they are evaluated. In a query language with these operators, it would for example be possible to directly express that a query was entailed at some point during the last 5–10 time units, using the expression $\lozenge^-_{[5,10]}\phi$. It is well-known that LTL$^{\text{bin}}$ formulas can be exponentially encoded in LTL (assuming a binary encoding of the intervals) [52], which would also apply to our setting. This immediately gives complexity upper bounds due to our results, tight in the case of data complexity, but with an exponential increase in the case of combined complexity. This blow-up might be unavoidable, since propositional LTL$^{\text{bin}}$ is ExpSpace-complete. In fact, combinations of LTL$^{\text{bin}}$ operators with description logics often lead to an exponential increase in complexity [20, 53], and it would be interesting to investigate whether this also happens for the settings considered in this paper. Regarding MTL, an initial investigation of query answering with a Horn-fragment of MTL has been performed in [19].

References

[1] M. Gebser, T. Grote, R. Kaminski, P. Obermeier, O. Sabuncu and T. Schaub, Stream Reasoning with Answer Set Programming: Preliminary Report, in: *Principles of Knowledge Representation and Reasoning: Proceedings of the Thirteenth International Conference, KR 2012, Rome, Italy, June 10-14, 2012*, G. Brewka, T. Eiter and S.A. McIlraith, eds, AAAI Press, 2012. ISBN 978-1-57735-560-1. http://www.aaai.org/ocs/index.php/KR/KR12/paper/view/4504.

[2] H. Beck, M. Dao-Tran, T. Eiter and M. Fink, LARS: A Logic-Based Framework for Analyzing Reasoning over Streams, in: *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA.*, B. Bonet and S. Koenig, eds, AAAI Press, 2015, pp. 1431–1438. ISBN 978-1-57735-698-1. http://www.aaai.org/ocs/index.php/AAAI/AAAI15/paper/view/9657.

[3] H. Beck, T. Eiter and C. Folie, Ticker: A system for incremental ASP-based stream reasoning, *TPLP* **17**(5–6) (2017), 744–763. doi:10.1017/S1471068417000370. https://doi.org/10.1017/S1471068417000370.

[4] Ö.L. Özçep, R. Möller and C. Neuenstadt, A Stream-Temporal Query Language for Ontology Based Data Access, in: *KI 2014: Advances in Artificial Intelligence - 37th Annual German Conference on AI, Stuttgart, Germany, September 22-26, 2014. Proceedings*, C. Lutz and M. Thielscher, eds, Lecture Notes in Computer Science, Vol. 8736, Springer, 2014, pp. 183–194. ISBN 978-3-319-11205-3. doi:10.1007/978-3-319-11206-0_18. https://doi.org/10.1007/978-3-319-11206-0_18.

[5] F. Baader, S. Borgwardt and M. Lippmann, Temporalizing Ontology-Based Data Access, in: *Automated Deduction - CADE-24 - 24th International Conference on Automated Deduction, Lake Placid, NY, USA, June 9-14, 2013. Proceedings*, M.P. Bonacina, ed., Lecture Notes in Computer Science, Vol. 7898, Springer, 2013, pp. 330–344. ISBN 978-3-642-38573-5. doi:10.1007/978-3-642-38574-2_23. https://doi.org/10.1007/978-3-642-38574-2_23.

[6] S. Borgwardt and V. Thost, Temporal Query Answering in DL-Lite with Negation, in: *Global Conference on Artificial Intelligence, GCAI 2015, Tbilisi, Georgia, October 16-19, 2015*, G. Gottlob, G. Sutcliffe and A. Voronkov, eds, EPiC Series in Computing, Vol. 36, EasyChair, 2015, pp. 51–65. http://www.easychair.org/publications/paper/245305.

[7] S. Brandt, E.G. Kalayci, V. Ryzhikov, G. Xiao and M. Zakharyaschev, A Framework for Temporal Ontology-Based Data Access: A Proposal, in: *New Trends in Databases and Information Systems - ADBIS 2017 Short Papers and Workshops, AMSD, BigNovelTI, DAS, SW4CH, DC, Nicosia, Cyprus, September 24-27, 2017, Proceedings*, M. Kirikova, K. Nørvåg, G.A. Papadopoulos, J. Gamper, R. Wrembel, J. Darmont and S. Rizzi, eds, Communications in Computer and Information Science, Vol. 767, Springer, 2017, pp. 161–173. ISBN 978-3-319-67161-1. doi:10.1007/978-3-319-67162-8_17. https://doi.org/10.1007/978-3-319-67162-8_17.

[8] B. Motik, B. Cuenca Grau, I. Horrocks, Z. Wu, A. Fokoue and C. Lutz, OWL 2 Web Ontology Language Profiles, 2012, Available at http://www.w3.org/TR/owl2-profiles/.

[9] W. OWL Working Group, *OWL 2 Web Ontology Language: Document overview*, W3C Recommendation, 2009, Available at https://www.w3.org/TR/owl2-overview/.

[10] D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini and R. Rosati, Tractable Reasoning and Efficient Query Answering in Description Logics: The DL-Lite Family, *J. Autom. Reasoning* **39**(3) (2007), 385–429. doi:10.1007/s10817-007-9078-x. https://doi.org/10.1007/s10817-007-9078-x.

[11] C. Lutz, D. Toman and F. Wolter, Conjunctive Query Answering in the Description Logic $\mathcal{EL}$ Using a Relational Database System, in: *IJCAI 2009, Proceedings of the 21st International Joint Conference on Artificial Intelligence, Pasadena, California, USA, July 11-17, 2009*, C. Boutilier, ed., 2009, pp. 2070–2075. http://ijcai.org/Proceedings/09/Papers/341.pdf.

[12] R. Rosati, On Conjunctive Query Answering in $\mathcal{EL}$, in: *Proceedings of the 2007 International Workshop on Description Logics (DL2007), Brixen-Bressanone, near Bozen-Bolzano, Italy, 8-10 June, 2007*, D. Calvanese, E. Franconi, V. Haarslev, D. Lembo, B. Motik, A. Turhan and S. Tessaris, eds, CEUR Workshop Proceedings, Vol. 250, CEUR-WS.org, 2007. http://ceur-ws.org/Vol-250/paper_83.pdf.

[13] H. Beck, M. Dao-Tran, T. Eiter and M. Fink, Towards Ideal Semantics for Analyzing Stream Reasoning, *CoRR* **abs/1505.05365** (2015). http://arxiv.org/abs/1505.05365.

[14] A. Pnueli, The Temporal Logic of Programs, in: *18th Annual Symposium on Foundations of Computer Science, Providence, Rhode Island, USA, 31 October - 1 November 1977*, IEEE Computer Society, 1977, pp. 46–57. doi:10.1109/SFCS.1977.32. https://doi.org/10.1109/SFCS.1977.32.

[15] A. Artale, R. Kontchakov, F. Wolter and M. Zakharyaschev, Temporal Description Logic for Ontology-Based Data Access, in: *IJCAI 2013, Proceedings of the 23rd International Joint Conference on Artificial Intelligence, Beijing, China, August 3-9, 2013*, F. Rossi, ed., IJCAI/AAAI, 2013, pp. 711–717. ISBN 978-1-57735-633-2. http://www.aaai.org/ocs/index.php/IJCAI/IJCAI13/paper/view/6824.

[16] S. Klarman and T. Meyer, Querying Temporal Databases via OWL 2 QL, in: *Web Reasoning and Rule Systems - 8th International Conference, RR 2014, Athens, Greece, September 15-17, 2014. Proceedings*, R. Kontchakov and M. Mugnier, eds, Lecture Notes in Computer Science, Vol. 8741, Springer, 2014, pp. 92–107. ISBN 978-3-319-11112-4. doi:10.1007/978-3-319-11113-1_7. https://doi.org/10.1007/978-3-319-11113-1_7.

[17] A. Artale, R. Kontchakov, A. Kovtunova, V. Ryzhikov, F. Wolter and M. Zakharyaschev, First-Order Rewritability of Temporal Ontology-Mediated Queries, in: *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015*, Q. Yang and M. Wooldridge, eds, AAAI Press, 2015, pp. 2706–2712. ISBN 978-1-57735-738-4. http://ijcai.org/Abstract/15/383.

[18] S. Borgwardt, M. Lippmann and V. Thost, Temporalizing rewritable query languages over knowledge bases, *J. Web Sem.* **33** (2015), 50–70. doi:10.1016/j.websem.2014.11.007. https://doi.org/10.1016/j.websem.2014.11.007.

[19] S. Brandt, E.G. Kalayci, R. Kontchakov, V. Ryzhikov, G. Xiao and M. Zakharyaschev, Ontology-Based Data Access with a Horn Fragment of Metric Temporal Logic, in: *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA.*, S.P. Singh and S. Markovitch, eds, AAAI Press, 2017,

pp. 1070–1076. http://aaai.org/ocs/index.php/AAAI/AAAI17/paper/view/14881.

[20] F. Baader, S. Borgwardt, P. Koopmann, A. Ozaki and V. Thost, Metric Temporal Description Logics with Interval-Rigid Names, in: *Frontiers of Combining Systems - 11th International Symposium, FroCoS 2017, Brasília, Brazil, September 27-29, 2017, Proceedings*, C. Dixon and M. Finger, eds, Lecture Notes in Computer Science, Vol. 10483, Springer, 2017, pp. 60–76. ISBN 978-3-319-66166-7. doi:10.1007/978-3-319-66167-4_4. https://doi.org/10.1007/978-3-319-66167-4_4.

[21] A. Artale, R. Kontchakov, A. Kovtunova, V. Ryzhikov, F. Wolter and M. Zakharyaschev, Ontology-Mediated Query Answering over Temporal Data: A Survey (Invited Talk), in: *24th International Symposium on Temporal Representation and Reasoning, TIME 2017, October 16-18, 2017, Mons, Belgium*, S. Schewe, T. Schneider and J. Wijsen, eds, LIPIcs, Vol. 90, Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2017, pp. 1–1137. ISBN 978-3-95977-052-1. doi:10.4230/LIPIcs.TIME.2017.1. https://doi.org/10.4230/LIPIcs.TIME.2017.1.

[22] F. Baader, S. Borgwardt and M. Lippmann, Temporal query entailment in the Description Logic SHQ, *J. Web Sem.* **33** (2015), 71–93. doi:10.1016/j.websem.2014.11.008. https://doi.org/10.1016/j.websem.2014.11.008.

[23] S. Borgwardt and V. Thost, Temporal Query Answering in the Description Logic $\mathcal{EL}$, in: *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015*, Q. Yang and M. Wooldridge, eds, AAAI Press, 2015, pp. 2819–2825. ISBN 978-1-57735-738-4. http://ijcai.org/Abstract/15/399.

[24] A. Margara, J. Urbani, F. van Harmelen and H.E. Bal, Streaming the Web: Reasoning over dynamic data, *J. Web Sem.* **25** (2014), 24–44. doi:10.1016/j.websem.2014.02.001. https://doi.org/10.1016/j.websem.2014.02.001.

[25] Y. Ren and J.Z. Pan, Optimising ontology stream reasoning with truth maintenance system, in: *Proceedings of the 20th ACM Conference on Information and Knowledge Management, CIKM 2011, Glasgow, United Kingdom, October 24-28, 2011*, C. Macdonald, I. Ounis and I. Ruthven, eds, ACM, 2011, pp. 831–836. ISBN 978-1-4503-0717-8. doi:10.1145/2063576.2063696. http://doi.acm.org/10.1145/2063576.2063696.

[26] A. Mileo, A. Abdelrahman, S. Policarpio and M. Hauswirth, StreamRule: A Nonmonotonic Stream Reasoning System for the Semantic Web, in: *Web Reasoning and Rule Systems - 7th International Conference, RR 2013, Mannheim, Germany, July 27-29, 2013. Proceedings*, W. Faber and D. Lembo, eds, Lecture Notes in Computer Science, Vol. 7994, Springer, 2013, pp. 247–252. ISBN 978-3-642-39665-6. doi:10.1007/978-3-642-39666-3_23. https://doi.org/10.1007/978-3-642-39666-3_23.

[27] F. Lécué, Diagnosing Changes in An Ontology Stream: A DL Reasoning Approach, in: *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence, July 22-26, 2012, Toronto, Ontario, Canada.*, J. Hoffmann and B. Selman, eds, AAAI Press, 2012. http://www.aaai.org/ocs/index.php/AAAI/AAAI12/paper/view/4988.

[28] M. Bienvenu and C. Bourgaux, Inconsistency-Tolerant Querying of Description Logic Knowledge Bases, in: *Reasoning Web: Logical Foundation of Knowledge Graph Construction and Query Answering - 12th International Summer School 2016, Aberdeen, UK, September 5-9, 2016, Tutorial Lectures*, J.Z. Pan, D. Calvanese, T. Eiter, I. Horrocks, M. Kifer, F. Lin and Y. Zhao, eds, Lecture Notes in Computer Science, Vol. 9885, Springer, 2016, pp. 156–202. ISBN 978-3-319-49492-0. doi:10.1007/978-3-319-49493-7_5. https://doi.org/10.1007/978-3-319-49493-7_5.

[29] D. Lembo, M. Lenzerini, R. Rosati, M. Ruzzi and D.F. Savo, Inconsistency-Tolerant Semantics for Description Logics, in: *Web Reasoning and Rule Systems - Fourth International Conference, RR 2010, Bressanone/Brixen, Italy, September 22-24, 2010. Proceedings*, P. Hitzler and T. Lukasiewicz, eds, Lecture Notes in Computer Science, Vol. 6333, Springer, 2010, pp. 103–117. ISBN 978-3-642-15917-6. doi:10.1007/978-3-642-15918-3_9. https://doi.org/10.1007/978-3-642-15918-3_9.

[30] D. Lembo, M. Lenzerini, R. Rosati, M. Ruzzi and D.F. Savo, Inconsistency-tolerant query answering in ontology-based data access, *J. Web Sem.* **33** (2015), 3–29. doi:10.1016/j.websem.2015.04.002. https://doi.org/10.1016/j.websem.2015.04.002.

[31] L.E. Bertossi, *Database Repairing and Consistent Query Answering*, Synthesis Lectures on Data Management, Morgan & Claypool Publishers, 2011. doi:10.2200/S00379ED1V01Y201108DTM020. https://doi.org/10.2200/S00379ED1V01Y201108DTM020.

[32] M. Bienvenu, On the Complexity of Consistent Query Answering in the Presence of Simple Ontologies, in: *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence, July 22-26, 2012, Toronto, Ontario, Canada.*, J. Hoffmann and B. Selman, eds, AAAI Press, 2012. http://www.aaai.org/ocs/index.php/AAAI/AAAI12/paper/view/4928.

[33] M. Bienvenu and R. Rosati, Tractable Approximations of Consistent Query Answering for Robust Ontology-based Data Access, in: *IJCAI 2013, Proceedings of the 23rd International Joint Conference on Artificial Intelligence, Beijing, China, August 3-9, 2013*, F. Rossi, ed., IJCAI/AAAI, 2013, pp. 775–781. ISBN 978-1-57735-633-2. http://www.aaai.org/ocs/index.php/IJCAI/IJCAI13/paper/view/6904.

[34] D. Lembo, M. Lenzerini, R. Rosati, M. Ruzzi and D.F. Savo, Query Rewriting for Inconsistent DL-Lite Ontologies, in: *Web Reasoning and Rule Systems - 5th International Conference, RR 2011, Galway, Ireland, August 29-30, 2011. Proceedings*, S. Rudolph and C. Gutiérrez, eds, Lecture Notes in Computer Science, Vol. 6902, Springer, 2011, pp. 155–169. ISBN 978-3-642-23579-5. doi:10.1007/978-3-642-23580-1_12. https://doi.org/10.1007/978-3-642-23580-1_12.

[35] R. Rosati, On the Complexity of Dealing with Inconsistency in Description Logic Ontologies, in: *IJCAI 2011, Proceedings of the 22nd International Joint Conference on Artificial Intelligence, Barcelona, Catalonia, Spain, July 16-22, 2011*, T. Walsh, ed., IJCAI/AAAI, 2011, pp. 1057–1062. ISBN 978-1-57735-516-8. doi:10.5591/978-1-57735-516-8/IJCAI11-181. https://doi.org/10.5591/978-1-57735-516-8/IJCAI11-181.

[36] R. Rosati, M. Ruzzi, M. Graziosi and G. Masotti, Evaluation of Techniques for Inconsistency Handling in OWL 2 QL Ontologies, in: *The Semantic Web - ISWC 2012 - 11th International Semantic Web Conference, Boston, MA, USA, November 11-15, 2012, Proceedings, Part II*, P. Cudré-Mauroux, J. Heflin, E. Sirin, T. Tudorache, J. Euzenat, M. Hauswirth, J.X. Parreira, J. Hendler, G. Schreiber, A. Bernstein and

E. Blomqvist, eds, Lecture Notes in Computer Science, Vol. 7650, Springer, 2012, pp. 337–349. ISBN 978-3-642-35172-3. doi:10.1007/978-3-642-35173-0_23. https://doi.org/10.1007/978-3-642-35173-0_23.

[37] E. Tsalapati, G. Stoilos, G.B. Stamou and G. Koletsos, Efficient Query Answering over Expressive Inconsistent Description Logics, in: *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*, S. Kambhampati, ed., IJCAI/AAAI Press, 2016, pp. 1279–1285. ISBN 978-1-57735-770-4. http://www.ijcai.org/Abstract/16/185.

[38] M. Bienvenu, C. Bourgaux and F. Goasdoué, Querying Inconsistent Description Logic Knowledge Bases under Preferred Repair Semantics, in: *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, July 27 -31, 2014, Québec City, Québec, Canada.*, C.E. Brodley and P. Stone, eds, AAAI Press, 2014, pp. 996–1002. ISBN 978-1-57735-661-5. http://www.aaai.org/ocs/index.php/AAAI/AAAI14/paper/view/8231.

[39] M. Bienvenu, C. Bourgaux and F. Goasdoué, Explaining Inconsistency-Tolerant Query Answering over Description Logic Knowledge Bases, in: *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA.*, D. Schuurmans and M.P. Wellman, eds, AAAI Press, 2016, pp. 900–906. ISBN 978-1-57735-760-5. http://www.aaai.org/ocs/index.php/AAAI/AAAI16/paper/view/12025.

[40] C. Bourgaux and A.-Y. Turhan, Temporal Query Answering in DL-Lite over Inconsistent Data, in: *The Semantic Web - ISWC 2017 - 16th International Semantic Web Conference, Vienna, Austria, October 21-25, 2017, Proceedings, Part I*, C. d'Amato, M. Fernández, V.A.M. Tamma, F. Lécué, P. Cudré-Mauroux, J.F. Sequeda, C. Lange and J. Heflin, eds, Lecture Notes in Computer Science, Vol. 10587, Springer, 2017, pp. 121–137. ISBN 978-3-319-68287-7. doi:10.1007/978-3-319-68288-4_8. https://doi.org/10.1007/978-3-319-68288-4_8.

[41] G. De Giacomo, R. De Masellis and M. Montali, Reasoning on LTL on Finite Traces: Insensitivity to Infiniteness, in: *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, July 27 -31, 2014, Québec City, Québec, Canada.*, C.E. Brodley and P. Stone, eds, AAAI Press, 2014, pp. 1027–1033. ISBN 978-1-57735-661-5. http://www.aaai.org/ocs/index.php/AAAI/AAAI14/paper/view/8575.

[42] D.M. Gabbay, The Declarative Past and Imperative Future: Executable Temporal Logic for Interactive Systems, in: *Temporal Logic in Specification, Altrincham, UK, April 8-10, 1987, Proceedings*, B. Banieqbal, H. Barringer and A. Pnueli, eds, Lecture Notes in Computer Science, Vol. 398, Springer, 1987, pp. 409–448. ISBN 3-540-51803-7. doi:10.1007/3-540-51803-7_36. https://doi.org/10.1007/3-540-51803-7_36.

[43] S. Borgwardt, M. Lippmann and V. Thost, Temporal Query Answering in the Description Logic DL-Lite, in: *Frontiers of Combining Systems - 9th International Symposium, FroCoS 2013, Nancy, France, September 18-20, 2013. Proceedings*, P. Fontaine, C. Ringeissen and R.A. Schmidt, eds, Lecture Notes in Computer Science, Vol. 8152, Springer, 2013, pp. 165–180. ISBN 978-3-642-40884-7. doi:10.1007/978-3-642-40885-4_11. https://doi.org/10.1007/978-3-642-40885-4_11.

[44] C. Lutz, I. Seylan, D. Toman and F. Wolter, The Combined Approach to OBDA: Taming Role Hierarchies Using Filters, in: *The Semantic Web - ISWC 2013 - 12th International Semantic Web Conference, Sydney, NSW, Australia, October 21-25, 2013, Proceedings, Part I*, H. Alani, L. Kagal, A. Fokoue, P.T. Groth, C. Biemann, J.X. Parreira, L. Aroyo, N.F. Noy, C. Welty and K. Janowicz, eds, Lecture Notes in Computer Science, Vol. 8218, Springer, 2013, pp. 314–330. ISBN 978-3-642-41334-6. doi:10.1007/978-3-642-41335-3_20. https://doi.org/10.1007/978-3-642-41335-3_20.

[45] R. Kontchakov, C. Lutz, D. Toman, F. Wolter and M. Zakharyaschev, The Combined Approach to Query Answering in DL-Lite, in: *Principles of Knowledge Representation and Reasoning: Proceedings of the Twelfth International Conference, KR 2010, Toronto, Ontario, Canada, May 9-13, 2010*, F. Lin, U. Sattler and M. Truszczynski, eds, AAAI Press, 2010. http://aaai.org/ocs/index.php/KR/KR2010/paper/view/1282.

[46] G. Gottlob, NP Trees and Carnap's Modal Logic, *J. ACM* **42**(2) (1995), 421–457. doi:10.1145/201019.201031. http://doi.acm.org/10.1145/201019.201031.

[47] D. Calvanese, E.G. Kalayci, V. Ryzhikov and G. Xiao, Towards Practical OBDA with Temporal Ontologies - (Position Paper), in: *Web Reasoning and Rule Systems - 10th International Conference, RR 2016, Aberdeen, UK, September 9-11, 2016, Proceedings*, M. Ortiz and S. Schlobach, eds, Lecture Notes in Computer Science, Vol. 9898, Springer, 2016, pp. 18–24. ISBN 978-3-319-45275-3. doi:10.1007/978-3-319-45276-0_2. https://doi.org/10.1007/978-3-319-45276-0_2.

[48] V. Thost, J. Holste and Ö.L. Özçep, On Implementing Temporal Query Answering in DL-Lite (extended abstract), in: *Proceedings of the 28th International Workshop on Description Logics, Athens,Greece, June 7-10, 2015.*, D. Calvanese and B. Konev, eds, CEUR Workshop Proceedings, Vol. 1350, CEUR-WS.org, 2015. http://ceur-ws.org/Vol-1350/paper-63.pdf.

[49] J. Chomicki, D. Toman and M.H. Böhlen, Querying ATSQL databases with temporal logic, *ACM Trans. Database Syst.* **26**(2) (2001), 145–178. http://portal.acm.org/citation.cfm?id=383891.383892.

[50] D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, A. Poggi, M. Rodriguez-Muro and R. Rosati, Ontologies and Databases: The DL-Lite Approach, in: *Reasoning Web. Semantic Technologies for Information Systems, 5th International Summer School 2009, Brixen-Bressanone, Italy, August 30 - September 4, 2009, Tutorial Lectures*, S. Tessaris, E. Franconi, T. Eiter, C. Gutiérrez, S. Handschuh, M. Rousset and R.A. Schmidt, eds, Lecture Notes in Computer Science, Vol. 5689, Springer, 2009, pp. 255–356. ISBN 978-3-642-03753-5. doi:10.1007/978-3-642-03754-2_7. https://doi.org/10.1007/978-3-642-03754-2_7.

[51] R. Alur and T.A. Henzinger, A Really Temporal Logic, *J. ACM* **41**(1) (1994), 181–204. doi:10.1145/174644.174651. http://doi.acm.org/10.1145/174644.174651.

[52] C. Lutz, D. Walther and F. Wolter, Quantitative temporal logics over the reals: PSpace and below, *Inf. Comput.* **205**(1) (2007), 99–123. doi:10.1016/j.ic.2006.08.006. https://doi.org/10.1016/j.ic.2006.08.006.

[53] V. Gutiérrez-Basulto, J.C. Jung and A. Ozaki, On Metric Temporal Description Logics, in: *ECAI 2016 - 22nd European Conference on Artificial Intelligence, 29 August-2 September 2016, The Hague, The Netherlands - Including Prestigious Appli-*

cations of Artificial Intelligence (PAIS 2016), G.A. Kaminka, M. Fox, P. Bouquet, E. Hüllermeier, V. Dignum, F. Dignum and F. van Harmelen, eds, Frontiers in Artificial Intelligence and Applications, Vol. 285, IOS Press, 2016, pp. 837–845. ISBN 978-1-61499-671-2. doi:10.3233/978-1-61499-672-9-837. https://doi.org/10.3233/978-1-61499-672-9-837.

[54] C. Bourgaux and A.-Y. Turhan, Temporal Query Answering in DL-Lite over Inconsistent Data, 2017, LTCS-Report 17-06, Chair for Automata Theory, TU Dresden. See https://lat.inf.tu-dresden.de/research/reports.html.

[55] C.E. Brodley and P. Stone (eds), Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, July 27 -31, 2014, Québec City, Québec, Canada, AAAI Press, 2014. ISBN 978-1-57735-661-5. http://www.aaai.org/Library/AAAI/aaai14contents.php.

[56] F. Rossi (ed.), IJCAI 2013, Proceedings of the 23rd International Joint Conference on Artificial Intelligence, Beijing, China, August 3-9, 2013, IJCAI/AAAI, 2013. ISBN 978-1-57735-633-2. http://ijcai.org/proceedings/2013.

[57] Q. Yang and M. Wooldridge (eds), Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015, AAAI Press, 2015. ISBN 978-1-57735-738-4. http://ijcai.org/proceedings/2015.

[58] J. Hoffmann and B. Selman (eds), Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence, July 22-26, 2012, Toronto, Ontario, Canada, AAAI Press, 2012.

Appendix A. Detailed proofs

We start by defining the notions of conflicts and causes that will be used in some proofs. A *conflict* for a KB $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ is a minimal $\mathcal{T}$-inconsistent subset of $\mathcal{A}$. A *cause* for a BCQ q w.r.t. $\mathcal{K}$ is a minimal $\mathcal{T}$-consistent subset $\mathcal{C} \subseteq \mathcal{A}$ such that $\langle \mathcal{T}, \mathcal{C} \rangle \models q$. The following definitions extend these notions to the temporal setting.

Definition A.1 (Conflicts of a TKB). A *conflict* of a TKB $\mathcal{K} = \langle \mathcal{T}, (\mathcal{A}_i)_{0 \leqslant i \leqslant n} \rangle$ is a sequence of ABoxes $(\mathcal{A}'_i)_{0 \leqslant i \leqslant n}$ such that $\{(\alpha, i) \mid \alpha \in \mathcal{A}'_i, 0 \leqslant i \leqslant n\}$ is a minimal $\mathcal{T}$-inconsistent subset of $\{(\alpha, i) \mid \alpha \in \mathcal{A}_i, 0 \leqslant i \leqslant n\}$.

The conflicts of a DL-Lite$_{\mathcal{R}}$-TKB are at most binary, i.e., contain at most two timed assertions (Fact 6.11).

Definition A.2 (Causes for a BTCQ in a TKB). A *cause* for a BTCQ ϕ at time point p in $\mathcal{K} = \langle \mathcal{T}, (\mathcal{A}_i)_{0 \leqslant i \leqslant n} \rangle$ is a sequence of ABoxes $(\mathcal{C}_i)_{0 \leqslant i \leqslant n}$ such that $\{(\alpha, i) \mid \alpha \in \mathcal{C}_i, 0 \leqslant i \leqslant n\}$ is a minimal $\mathcal{T}$-consistent subset of $\{(\alpha, i) \mid \alpha \in \mathcal{A}_i, 0 \leqslant i \leqslant n\}$ such that $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, p \models \phi$.

Note that a KB (resp. TKB) is consistent iff it has no conflict, and that a BCQ (resp. BTCQ) is entailed from a KB (resp. a TKB) $\mathcal{K}$ under brave semantics iff it has some cause in $\mathcal{K}$, since such a cause can be extended to a repair that entails the query.

A.1. Proofs of complexity results

Hardness of data complexity for brave semantics in $\mathcal{EL}_\perp$.

Proposition 6.9. *BTCQ entailment from an $\mathcal{EL}_\perp$-TKB with $\mathsf{N_{RC}} \neq \emptyset$ is*

– *coNP-hard w.r.t. data complexity under AR and IAR semantics, and*
– *Σ_2^p-hard w.r.t. data complexity under brave semantics.*

Proof. The lower bounds for AR and IAR semantics follow from the atemporal case, so that we only have to provide a lower bound for brave semantics.

We show that the complement of brave TCQ entailment is Π_2^p-hard by reduction from QBF$_{2,\forall}$. Let $\Phi = \forall x_1 \ldots x_m \exists y_1 \ldots y_r \varphi$ be a QBF$_{2,\forall}$-formula, where $\varphi = \bigwedge_{i=0}^h \ell_i^0 \vee \ell_i^1 \vee \ell_i^2$ is a 3-CNF formula over the propositional variables $\{x_1, \ldots, x_m, y_1, \ldots, y_r\}$. Based on Φ, we define the TKB $\mathcal{K} = \langle \mathcal{T}, (\mathcal{A}_i)_{0 \leqslant i \leqslant 3h+2} \rangle$ and the TCQ ϕ as follows, where $\mathsf{N_{RC}} = \{\mathsf{T}\}$.

$$\mathcal{T} = \{\exists \mathsf{Pos}.\mathsf{T} \sqsubseteq \mathsf{Sat}, \exists \mathsf{Neg}.\mathsf{F} \sqsubseteq \mathsf{Sat},$$
$$\exists \mathsf{FromPos}.\mathsf{Sat} \sqsubseteq \mathsf{T}, \exists \mathsf{FromNeg}.\mathsf{Sat} \sqsubseteq \mathsf{F},$$
$$\exists \mathsf{FromY}.\mathsf{Sat} \sqsubseteq \mathsf{T}, \mathsf{T} \sqcap \mathsf{F} \sqsubseteq \perp,$$
$$\mathsf{T} \sqcap \exists \mathsf{ValY}.\mathsf{T} \sqsubseteq \perp\}$$
$$\phi = \neg \square^b (\mathsf{NotFirst}(c) \vee \mathsf{Sat}(c) \vee$$
$$\bigcirc \mathsf{Sat}(c) \vee \bigcirc\bigcirc \mathsf{Sat}(c))$$

For each clause $\ell_i^0 \vee \ell_i^1 \vee \ell_i^2$, $i \in [0, h]$, we define the following three ABoxes $\mathcal{A}_{3i+k}$ ($0 \leqslant k \leqslant 2$):

$$\mathcal{A}_{3i+0} = \mathcal{B} \cup \mathcal{B}_{3i+0}$$
$$\mathcal{A}_{3i+k} = \mathcal{B} \cup \mathcal{B}_{3i+k} \cup \{\mathsf{NotFirst}(c)\}, 1 \leqslant k \leqslant 2,$$

where

$$\mathcal{B} = \{\mathsf{T}(x_j), \mathsf{F}(x_j) \mid 1 \leqslant j \leqslant m\} \cup$$
$$\{\mathsf{ValY}(y_j, \neg y_j), \mid 1 \leqslant j \leqslant r\}$$
$$\mathcal{B}_{3i+k} = \{\mathsf{Pos}(c, x_j), \mathsf{FromPos}(x_j, c)\} \text{ if } \ell_i^k = x_j$$

$\mathcal{B}_{3i+k} = \{\mathsf{Neg}(c, x_j), \mathsf{FromNeg}(x_j, c)\}$ if $\ell_i^k = \neg x_j$

$\mathcal{B}_{3i+k} = \{\mathsf{FromY}(y_j, c)\}$ if $\ell_i^k = y_j$

$\mathcal{B}_{3i+k} = \{\mathsf{FromY}(\neg y_j, c)\}$ if $\ell_i^k = \neg y_j$.

Claim. Φ is valid iff $\mathcal{K}, 0 \not\models_{\mathrm{brave}} \phi$.

Proof of claim. The repairs of $\mathcal{K}$ correspond to the valuations of the x_j. Indeed, since T is rigid and disjoint from F, each pair of timed assertions $\{(\mathsf{T}(x_j), i_1), (\mathsf{N}(x_j), i_2)\}$ is inconsistent, so every x_j is such that a repair of $\mathcal{K}$ contains either $(\mathsf{T}(x_j), i)$ for every i, or $(\mathsf{F}(x_j), i)$ for every i. For each repair $\mathcal{A}' = (\mathcal{A}'_i)_{0 \leqslant i \leqslant 3h+2}$ of $\mathcal{K}$, we denote by $v_X^{\mathcal{A}'}$ the valuation of the x_j defined by $v_X^{\mathcal{A}'}(x_j) = \mathsf{true}$ if $\mathsf{T}(x_j) \in \mathcal{A}'_1$. Correspondingly, for every valuation v_X of the x_j, we denote by $\mathcal{A}^{v_x} = (\mathcal{A}_i^{v_x})_{0 \leqslant i \leqslant 3h+2}$ the repair of $\mathcal{K}$ defined by $\mathsf{T}(x_j) \in \mathcal{A}_i^{v_x}$ for every i if $v_X(x_j) = \mathsf{true}$.

Assume that Φ is valid and let $\mathcal{A}' = (\mathcal{A}'_i)_{0 \leqslant i \leqslant 3h+2}$ be a repair of $\mathcal{K}$. Since Φ is valid, there then exists a valuation v_Y of the y_j such that $\varphi[x_j \leftarrow v_X^{\mathcal{A}'}(x_j)]$ is satisfied by v_Y. Let $\mathcal{J} = (\mathcal{I}_i)_{i \geqslant 0}$ be a model of $\langle \mathcal{T}, (\mathcal{A}'_i)_{0 \leqslant i \leqslant 3h+2} \rangle$ that respects rigid predicates and such that for every i,

- $y_j^{\mathcal{I}_i} \in \mathsf{T}^{\mathcal{I}_i}$ iff $v_Y(y_j) = \mathsf{true}$,
- $\neg y_j^{\mathcal{I}_i} \in \mathsf{T}^{\mathcal{I}_i}$ iff $v_Y(y_j) = \mathsf{false}$, and
- if there exists some d such that $(d^{\mathcal{I}_i}, c^{\mathcal{I}_i}) \in \mathsf{FromY}^{\mathcal{I}_i}$ and $d^{\mathcal{I}_i} \in \mathsf{T}^{\mathcal{I}_i}$, then $c^{\mathcal{I}_i} \in \mathsf{Sat}^{\mathcal{I}_i}$.

One can verify that such a model always exists. First, because the role ValY connects only individuals of the type y_j and $\neg y_j$, and we only assign T to one of them, these additional constraints respect the TBox axiom $\mathsf{T} \sqcap \exists \mathsf{ValY}.\mathsf{T} \sqsubseteq \bot$. Second, the assignment of c to Sat respects $\exists \mathsf{FormY}.\mathsf{Sat} \sqsubseteq \mathsf{T}$ by construction.

It is easy to see that $\mathcal{J}, 0 \models \Box^b(\mathsf{NotFirst}(c) \vee \mathsf{Sat}(c) \vee \bigcirc \mathsf{Sat}(c) \vee \bigcirc\bigcirc \mathsf{Sat}(c))$. Indeed, at each time point $p \in [0, 3h+2]$, either $\mathsf{NotFirst}(c)$ is true, or $p = 3i$, in which case we show that $\mathsf{Sat}(c)$ is true at time point $3i+k$, where ℓ_i^k is the first literal of the clause $\ell_i^0 \vee \ell_i^1 \vee \ell_i^2$ satisfied by $v_X^{\mathcal{A}'} \cup v_Y$.

- If $\ell_i^k = x_j$, then $v_X^{\mathcal{A}'}(x_j) = \mathsf{true}$. Thus, by construction, $\mathsf{T}(x_j) \in \mathcal{A}'_{3i+k}$, and therefore $\mathcal{J}, 3i+k \models \mathsf{T}(x_j)$. Moreover, since $\mathcal{J}, 3i+k \models \mathsf{Pos}(c, x_j)$, also $\mathcal{J}, 3i+k \models \mathsf{Sat}(c)$, because $\mathcal{J}$ is a model of $\mathcal{T}$.
- If $\ell_i^k = \neg x_j$, then $v_X^{\mathcal{A}'}(x_j) = \mathsf{false}$. Thus, by construction, $\mathsf{F}(x_j) \in \mathcal{A}'_{3i+k}$, and therefore $\mathcal{J}, 3i+k \models \mathsf{F}(x_j)$. Moreover, since $\mathcal{J}, 3i+k \models \mathsf{Neg}(c, x_j)$, we obtain $\mathcal{J}, 3i+k \models \mathsf{Sat}(c)$ because $\mathcal{J}$ is a model of $\mathcal{T}$.

- If $\ell_i^k = y_j$, then $v_Y(y_j) = \mathsf{true}$. Thus $\mathcal{J}, 3i+k \models \mathsf{T}(y_j)$, and since $\mathcal{J}, 3i+k \models \mathsf{FromY}(y_j, c)$, by construction of $\mathcal{J}$, it follows that $\mathcal{J}, 3i+k \models \mathsf{Sat}(c)$.
- If $\ell_i^k = \neg y_j$, then $v_Y(y_j) = \mathsf{false}$. Thus $\mathcal{J}, 3i+k \models \mathsf{T}(\neg y_j)$, and since $\mathcal{J}, 3i+k \models \mathsf{FromY}(\neg y_j, c)$, by construction of $\mathcal{J}$, it follows that $\mathcal{J}, 3i+k \models \mathsf{Sat}(c)$.

It follows that $\mathcal{J}, 0 \not\models \phi$, so $\langle \mathcal{T}, (\mathcal{A}'_i)_{0 \leqslant i \leqslant 3h+2} \rangle, 0 \not\models \phi$. Hence, $\mathcal{K}, 0 \not\models_{\mathrm{brave}} \phi$.

In the other direction, assume that $\mathcal{K}, 0 \not\models_{\mathrm{brave}} \phi$, and let v_X be a valuation of the x_j. Since $(\mathcal{A}_i^{v_X})_{0 \leqslant i \leqslant 3h+2}$ is a repair of $\mathcal{K}$, $\langle \mathcal{T}, (\mathcal{A}_i^{v_X})_{0 \leqslant i \leqslant 3h+2} \rangle, 0 \not\models \phi$, and there exists a model $\mathcal{J} = (\mathcal{I}_i)_{i \geqslant 0}$ of $\langle \mathcal{T}, (\mathcal{A}_i^{v_X})_{0 \leqslant i \leqslant 3h+2} \rangle$ that respects rigid predicates and is such that $\mathcal{J}, 0 \not\models \phi$, i.e., $\mathcal{J}, 0 \models \Box^b(\mathsf{NotFirst}(c) \vee \mathsf{Sat}(c) \vee \bigcirc \mathsf{Sat}(c) \vee \bigcirc\bigcirc \mathsf{Sat}(c))$. Let v_Y be the (partial) valuation of the y_j defined as follows: $v_Y(y_j) = \mathsf{true}$ if there exists k such that $\mathcal{J}, k \models \mathsf{T}(y_j)$, and $v_Y(y_j) = \mathsf{false}$ if there exists k such that $\mathcal{J}, k \models \mathsf{T}(\neg y_j)$. The valuation v_Y is well-defined because $\mathcal{J}$ is a model of $\mathcal{T}$ and respects rigid predicates. Therefore, if $\mathcal{J}, k \models \mathsf{T}(y_j)$ for some k, then $\mathcal{J}, k \models \mathsf{T}(y_j)$ for every k, and $\mathcal{J}, k \not\models \mathsf{T}(\neg y_j)$. Otherwise, we would have $\mathcal{J}, k \models \mathsf{T} \sqcap \exists \mathsf{ValY}.\mathsf{T}(y_j)$, which contradicts our TBox axioms.

Since $\mathcal{J}, 0 \models \Box^b(\mathsf{NotFirst}(c) \vee \mathsf{Sat}(c) \vee \bigcirc \mathsf{Sat}(c) \vee \bigcirc\bigcirc \mathsf{Sat}(c))$, for every clause $\ell_i^0 \vee \ell_i^1 \vee \ell_i^2$, we have that $\mathcal{J}, 3i+k \models \mathsf{Sat}(c)$ for some $k \in [0, 2]$. For this $k \in [0, 2]$, we show that $v_X \cup v_Y$ satisfies ℓ_i^k.

- If $\ell_i^k = x_j$, then $\mathcal{J}, 3i+k \models \mathsf{FromPos}(x_j, c)$, and since $\mathcal{J}, 3i+k \models \mathsf{Sat}(c)$ and $\mathcal{J}$ respects $\exists \mathsf{FromPos}.\mathsf{Sat} \sqsubseteq \mathsf{T}$, also $\mathcal{J}, 3i+k \models \mathsf{T}(x_j)$. It follows that $(\mathsf{T}(x_j), k) \in (\mathcal{A}_i^{v_x})_{0 \leqslant i \leqslant 3h+2}$ for every k (otherwise, by maximality of repairs, $(\mathsf{F}(x_j), k) \in (\mathcal{A}_i^{v_x})_{0 \leqslant i \leqslant 3h+2}$ and $\mathcal{J}$ assigns x_j to T and F at some time point, which would contradict the TBox axiom $\mathsf{T} \sqcap \mathsf{F} \sqsubseteq \bot$). Hence, $v_X(x_j) = \mathsf{true}$.
- If $\ell_i^k = \neg x_j$, then $\mathcal{J}, 3i+k \models \mathsf{FromNeg}(x_j, c)$, and since $\mathcal{J}, 3i+k \models \mathsf{Sat}(c)$ and $\mathcal{J}$ respects $\exists \mathsf{FromNeg}.\mathsf{Sat} \sqsubseteq \mathsf{F}$, also $\mathcal{J}, 3i+k \models \mathsf{F}(x_j)$. It follows that $(\mathsf{F}(x_j), k) \in (\mathcal{A}_i^{v_x})_{0 \leqslant i \leqslant 3h+2}$ for every k (otherwise $(\mathsf{T}(x_j), k) \in (\mathcal{A}_i^{v_x})_{0 \leqslant i \leqslant 3h+2}$ and $\mathcal{J}$ assigns x_j to T and F at some time point). Hence $v_X(x_j) = \mathsf{false}$.
- If $\ell_i^k = y_j$, then $\mathcal{J}, 3i+k \models \mathsf{FromY}(y_j, c)$, and since $\mathcal{J}, 3i+k \models \mathsf{Sat}(c)$, and $\mathcal{J}$ respects $\exists \mathsf{FromY}.\mathsf{Sat} \sqsubseteq \mathsf{T}$, also $\mathcal{J}, 3i+k \models \mathsf{T}(y_j)$. We obtain $v_Y(y_j) = \mathsf{true}$.
- If $\ell_i^k = \neg y_j$, then $\mathcal{J}, 3i+k \models \mathsf{FromY}(\neg y_j, c)$, and since $\mathcal{J}, 3i+k \models \mathsf{Sat}(c)$ and $\mathcal{J}$ respects $\exists \mathsf{FromY}.\mathsf{Sat} \sqsubseteq \mathsf{T}$, also $\mathcal{J}, 3i+k \models \mathsf{T}(\neg y_j)$, so that we obtain $v_Y(y_j) = \mathsf{false}$.

It follows that $\nu_X \cup \nu_Y$ satisfies every clause $\ell_i^0 \vee \ell_i^1 \vee \ell_i^2$. We have thus shown that $\varphi[x_j \leftarrow \nu_X(x_j), y_j \leftarrow \nu_Y(y_j)]$ evaluates to true, and that Φ is valid. $\qquad\square$

Canonical model property.

Theorem 7.4. *If $\mathcal{L}$ has the canonical model property for BCQ entailment, then $\mathcal{L}$ also has the canonical model property for entailment of BTCQs without negation.*

Proof. Let $\widetilde{\mathcal{I}}_{\widetilde{\mathcal{K}}}$ be the canonical model of $\widetilde{\mathcal{K}}$ and $\mathcal{J}_{\mathcal{K}} = (\mathcal{I}_i)_{i \geqslant 0} = \mathsf{temp}(\widetilde{\mathcal{I}}_{\widetilde{\mathcal{K}}})$. We show that $\mathcal{J}_{\mathcal{K}}$ is a canonical model of $\mathcal{K}$ for BTCQs without negation, that is, for every BTCQ ϕ that does not contain any negation, $\mathcal{K}, p \models \phi$ iff $\mathcal{J}_{\mathcal{K}}, p \models \phi$.

Since by Lemma 6.2, $\mathcal{J}_{\mathcal{K}}$ is a model of $\mathcal{K}$ that respects rigid predicates, if $\mathcal{K}, p \models \phi$, then also $\mathcal{J}_{\mathcal{K}}, p \models \phi$. For the other direction, we show by induction on the structure of ϕ that $\mathcal{J}_{\mathcal{K}}, p \models \phi$ implies $\mathcal{K}, p \models \phi$.

If $\phi = q$ is a BCQ, by Lemma 7.1, $\mathcal{K}, p \models q$ iff $\widetilde{\mathcal{K}} \models \widetilde{q}_p$, which is exactly the case iff $\widetilde{\mathcal{I}}_{\widetilde{\mathcal{K}}} \models \widetilde{q}_p$. By construction of $\mathcal{J}_{\mathcal{K}}$, it follows that $\mathcal{K}, p \models q$ iff $\mathcal{J}_{\mathcal{K}}, p \models q$.

Assume that for two BTCQs ϕ_1, ϕ_2 and any $p \geqslant 0$, $\mathcal{J}_{\mathcal{K}}, p \models \phi_i$ implies $\mathcal{K}, p \models \phi_i$ ($i \in \{1, 2\}$). We can show the following for BTCQs built from ϕ_1, ϕ_2.

- If $\mathcal{J}_{\mathcal{K}}, p \models \phi_1 \wedge \phi_2$, then $\mathcal{J}_{\mathcal{K}}, p \models \phi_1$ and $\mathcal{J}_{\mathcal{K}}, p \models \phi_2$. Hence by assumption, $\mathcal{K}, p \models \phi_1$ and $\mathcal{K}, p \models \phi_2$, and thus $\mathcal{K}, p \models \phi_1 \wedge \phi_2$.
- If $\mathcal{J}_{\mathcal{K}}, p \models \phi_1 \vee \phi_2$, then $\mathcal{J}_{\mathcal{K}}, p \models \phi_1$ or $\mathcal{J}_{\mathcal{K}}, p \models \phi_2$. Hence, by assumption, $\mathcal{K}, p \models \phi_1$ or $\mathcal{K}, p \models \phi_2$, and thus $\mathcal{K}, p \models \phi_1 \vee \phi_2$.
- If $\mathcal{J}_{\mathcal{K}}, p \models \bigcirc\phi_1$, then $\mathcal{J}_{\mathcal{K}}, p + 1 \models \phi_1$. Hence, by assumption, $\mathcal{K}, p + 1 \models \phi_1$, and thus $\mathcal{K}, p \models \bigcirc\phi_1$.
- We can show similarly that $\mathcal{J}_{\mathcal{K}}, p \models \bullet^b \phi_1$ implies $\mathcal{K}, p \models \bullet^b \phi_1$, that $\mathcal{J}_{\mathcal{K}}, p \models \bigcirc^-\phi_1$ implies $\mathcal{K}, p \models \bigcirc^-\phi_1$, and that $\mathcal{J}_{\mathcal{K}}, p \models \bullet^-\phi_1$ implies $\mathcal{K}, p \models \bullet^-\phi_1$.
- If $\mathcal{J}_{\mathcal{K}}, p \models \Box\phi_1$, then for every $k \geqslant p$, $\mathcal{J}_{\mathcal{K}}, k \models \phi_1$. Hence, by assumption, for every $k \geqslant p$, $\mathcal{K}, k \models \phi_1$, and thus $\mathcal{K}, p \models \Box\phi_1$.
- We can show similarly that $\mathcal{J}_{\mathcal{K}}, p \models \Box^b\phi_1$ implies $\mathcal{K}, p \models \Box^b\phi_1$, and that $\mathcal{J}_{\mathcal{K}}, p \models \Box^-\phi_1$ implies $\mathcal{K}, p \models \Box^-\phi_1$.
- If $\mathcal{J}_{\mathcal{K}}, p \models \Diamond\phi_1$, then there exists $k \geqslant p$ such that $\mathcal{J}_{\mathcal{K}}, k \models \phi_1$. Hence, by assumption, there exists $k \geqslant p$ such that $\mathcal{K}, k \models \phi_1$, and thus $\mathcal{K}, p \models \Diamond\phi_1$.
- We can show similarly that $\mathcal{J}_{\mathcal{K}}, p \models \Diamond^b\phi_1$ implies $\mathcal{K}, p \models \Diamond^b\phi_1$, and that $\mathcal{J}_{\mathcal{K}}, p \models \Diamond^-\phi_1$ implies $\mathcal{K}, p \models \Diamond^-\phi_1$.

- If $\mathcal{J}_{\mathcal{K}}, p \models \phi_1 \mathsf{U} \phi_2$, then there exists $k \geqslant p$ such that $\mathcal{J}_{\mathcal{K}}, k \models \phi_2$ and for every j such that $p \leqslant j < k$, $\mathcal{J}_{\mathcal{K}}, j \models \phi_1$. Hence, by assumption, there exists $k \geqslant p$ such that $\mathcal{K}, k \models \phi_2$ and for every j such that $p \leqslant j < k$, $\mathcal{K}, j \models \phi_1$. As a consequence, $\mathcal{K}, p \models \phi_1 \mathsf{U} \phi_2$.
- We can show similarly that $\mathcal{J}_{\mathcal{K}}, p \models \phi_1 \mathsf{U}^b \phi_2$ implies $\mathcal{K}, p \models \phi_1 \mathsf{U}^b \phi_2$, and that $\mathcal{J}_{\mathcal{K}}, p \models \phi_1 \mathsf{S} \phi_2$ implies $\mathcal{K}, p \models \phi_1 \mathsf{S} \phi_2$.

We conclude that for every BTCQ ϕ without negation, $\mathcal{J}_{\mathcal{K}}, p \models \phi$ implies $\mathcal{K}, p \models \phi$, and that $\mathcal{J}_{\mathcal{K}}$ is a canonical model of $\mathcal{K}$ for BTCQs without negation. It follows that if a DL $\mathcal{L}$ has the canonical model property for BCQ entailment, then $\mathcal{L}$ also has the canonical model property for entailment of BTCQs without negation. $\qquad\square$

Proposition 7.6. *Assume $\mathcal{L}$ has the canonical model property for BCQ entailment. Then, for every $\mathcal{L}$-TKBs $\mathcal{K}$ and $\mathcal{K}'$, if $\mathcal{K}$ and $\mathcal{K}'$ coincide for BCQ entailment, then $\mathcal{K}$ and $\mathcal{K}'$ coincide for entailment of BTCQs without negation. I.e., if for every time point p and BCQ q, $\mathcal{K}, p \models q$ iff $\mathcal{K}', p \models q$, then for every time point p and BTCQ ϕ without negation, $\mathcal{K}, p \models \phi$ iff $\mathcal{K}', p \models \phi$.*

Proof. Assume that for every time point p and BCQ q, $\mathcal{K}, p \models q$ iff $\mathcal{K}', p \models q$. We can then show by induction on the structure of ϕ that $\mathcal{K}, p \models \phi$ iff $\mathcal{K}', p \models \phi$. For $\phi = \exists\vec{y}.\psi(\vec{y})$, this holds by assumption. Assume that for two BTCQs ϕ_1, ϕ_2, $\mathcal{K}, p \models \phi_i$ iff $\mathcal{K}', p \models \phi_i$ ($i \in \{1, 2\}$). Then, since by Theorem 7.4, $\mathcal{L}$ has the canonical model property for entailment of BTCQ without negation, by applying the definitions of BTCQ satisfaction in Table 1 to the canonical models of $\mathcal{K}$ and $\mathcal{K}'$, we obtain the following about formulas composed of ϕ_1 and ϕ_2.

- $\mathcal{K}, p \models \phi_1 \wedge \phi_2$ iff $\mathcal{K}, p \models \phi_1$ and $\mathcal{K}, p \models \phi_2$, which by assumption is the case iff $\mathcal{K}', p \models \phi_1$ and $\mathcal{K}', p \models \phi_2$, i.e., iff $\mathcal{K}', p \models \phi_1 \wedge \phi_2$.
- $\mathcal{K}, p \models \phi_1 \vee \phi_2$ iff $\mathcal{K}, p \models \phi_1$ or $\mathcal{K}, p \models \phi_2$, which by assumption is the case iff $\mathcal{K}', p \models \phi_1$ or $\mathcal{K}', p \models \phi_2$, i.e., iff $\mathcal{K}', p \models \phi_1 \vee \phi_2$.
- $\mathcal{K}, p \models \bigcirc\phi_1$ iff $\mathcal{K}, p+1 \models \phi_1$, which by assumption is the case iff $\mathcal{K}', p + 1 \models \phi_1$, i.e., iff $\mathcal{K}', p \models \bigcirc\phi_1$.
- We show in the same way that $\mathcal{K}, p \models \bullet^b \phi_1$ iff $\mathcal{K}', p \models \bullet^b \phi_1$, that $\mathcal{K}, p \models \bigcirc^-\phi_1$ iff $\mathcal{K}', p \models \bigcirc^-\phi_1$, and that $\mathcal{K}, p \models \bullet^-\phi_1$ iff $\mathcal{K}', p \models \bullet^-\phi_1$.
- $\mathcal{K}, p \models \Box\phi_1$ iff for every $k, k \geqslant p, \mathcal{K}, k \models \phi_1$, which by assumption is the case iff for every $k, k \geqslant p, \mathcal{K}', k \models \phi_1$, i.e., iff $\mathcal{K}', p \models \Box\phi_1$.

- We show in the same way that $\mathcal{K}, p \models \Box^b \phi_1$ iff $\mathcal{K}', p \models \Box^b \phi_1$, and that $\mathcal{K}, p \models \Box^- \phi_1$ iff $\mathcal{K}', p \models \Box^- \phi_1$.
- $\mathcal{K}, p \models \Diamond \phi_1$ iff there exists k, $k \geqslant p$, $\mathcal{K}, k \models \phi_1$, which by assumption is the case iff there exists k, $k \geqslant p$, $\mathcal{K}', k \models \phi_1$, i.e., iff $\mathcal{K}', p \models \Diamond \phi_1$.
- We show in the same way that $\mathcal{K}, p \models \Diamond^b \phi_1$ iff $\mathcal{K}', p \models \Diamond^b \phi_1$, and that $\mathcal{K}, p \models \Diamond^- \phi_1$ iff $\mathcal{K}', p \models \Diamond^- \phi_1$.
- $\mathcal{K}, p \models \phi_1 \mathsf{U} \phi_2$ iff there exists k, $k \geqslant p$, $\mathcal{K}, k \models \phi_2$ and for every j, $p \leqslant j < k$, $\mathcal{K}, j \models \phi_1$, which by assumption is the case iff there exists k, $k \geqslant p$, $\mathcal{K}', k \models \phi_2$ and for every j, $p \leqslant j < k$, $\mathcal{K}', j \models \phi_1$, i.e., iff $\mathcal{K}', p \models \phi_1 \mathsf{U} \phi_2$.
- We show in the same way that $\mathcal{K}, p \models \phi_1 \mathsf{U}^b \phi_2$ iff $\mathcal{K}', p \models \phi_1 \mathsf{U}^b \phi_2$, and that $\mathcal{K}, p \models \phi_1 \mathsf{S} \phi_2$ iff $\mathcal{K}', p \models \phi_1 \mathsf{S} \phi_2$.

We conclude that for every BTCQ without negation ϕ and time point p, we have $\mathcal{K}, p \models \phi$ iff $\mathcal{K}', p \models \phi$. $\square$

Justification structures for brave entailment of BTCQs without negation in the case $\mathsf{N_{RC}} = \mathsf{N_{RR}} = \emptyset$. We show that if $\mathsf{N_{RC}} = \mathsf{N_{RR}} = \emptyset$ and ϕ is a BTCQ without negation, then $\mathcal{K}, p \models_{\text{brave}} \phi$ iff there exists a correct brave-justification structure J for ϕ in $\mathcal{K}$ that justifies ϕ at time point p. We prove both directions in separate lemmas.

Lemma A.3. *If* $\mathsf{N_{RC}} = \mathsf{N_{RR}} = \emptyset$ *and there is a correct brave-justification structure J for ϕ in $\mathcal{K}$ that justifies ϕ at time point p, then* $\mathcal{K}, p \models_{brave} \phi$.

Proof. In order to show $\mathcal{K}, p \models_{\text{brave}} \phi$, we determine a cause $(\mathcal{C}_i)_{0 \leqslant i \leqslant n}$ for ϕ. To do this, we first select a sequence of tuples from J as follows, where we make sure that we have exactly one tuple for every i, $0 \leqslant i \leqslant n$.

1. The tuple $(p, L_{\text{now}}^p, F_{\text{now}}^p, F_{\text{prev}}^p, F_{\text{next}}^p)$ is such that $\phi \in F_{\text{now}}^p$.
2. If the tuple $(i, L_{\text{now}}^i, F_{\text{now}}^i, F_{\text{prev}}^i, F_{\text{next}}^i)$ was selected and $0 < i \leqslant p$, then select a tuple of the form $(i-1, L_{\text{now}}^{i-1}, F_{\text{now}}^{i-1}, F_{\text{prev}}^{i-1}, F_{\text{next}}^{i-1})$, where $F_{\text{now}}^{i-1} = F_{\text{prev}}^i$ and $F_{\text{next}}^{i-1} = F_{\text{now}}^i$.
3. If the tuple $(i, L_{\text{now}}^i, F_{\text{now}}^i, F_{\text{prev}}^i, F_{\text{next}}^i)$ was selected and $p \leqslant i < n$, then select a tuple of the form $(i+1, L_{\text{now}}^{i+1}, F_{\text{now}}^{i+1}, F_{\text{prev}}^{i+1}, F_{\text{next}}^{i+1})$, where $F_{\text{now}}^{i+1} = F_{\text{next}}^i$ and $F_{\text{prev}}^{i+1} = F_{\text{now}}^i$.

Because J is correct and justifies ϕ at time point p, such a sequence can always be selected.

Based on this sequence, we construct a sequence of ABoxes $(\mathcal{C}_i)_{0 \leqslant i \leqslant n}$. For this, we take for each of the tuples $(i, L_{\text{now}}^i, F_{\text{now}}^i, F_{\text{prev}}^i, F_{\text{next}}^i)$ a cause $\mathcal{C}_i \subseteq \mathcal{A}_i$ that entails $\bigwedge_{q \in L_{\text{now}}^i} q$. Such a cause exists because $\langle \mathcal{T}, \mathcal{A}_i \rangle \models_{\text{brave}} \bigwedge_{q \in L_{\text{now}}^i} q$ by Condition 1. Since each $\mathcal{C}_i$ is consistent and we do not have rigid predicates, the TKB $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle$ is consistent.

We prove $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, p \models \phi$ by showing that $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, p \models F_{\text{now}}^p$. To do this, we consider the sets $F_{\text{now}}^{i,d} = \{ \psi \mid \psi \in F_{\text{now}}^i, \text{degree}(\psi) \leqslant d \}$ of LTL formulas, where $\text{degree}(\psi)$ is the maximal number of nested LTL operators in ψ, and prove by induction on d that for all $0 \leqslant i \leqslant n$ and for all $\psi \in F_{\text{now}}^{i,d}$, we have $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i \models \psi$, i.e., $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i \models F_{\text{now}}^{i,d}$.

For $d = 0$, $F_{\text{now}}^{i,0}$ contains only conjunctive queries of the form $\exists \vec{y} \varphi(\vec{y})$. Since for every $\psi \in BCQ(\phi)$, $F_{\text{now}}^i \models \psi$ implies $\psi \in L_{\text{now}}^i$ (Condition 4), we have that $F_{\text{now}}^{i,0} \subseteq L_{\text{now}}^i$. From this, since $\langle \mathcal{T}, \mathcal{C}_i \rangle \models \bigwedge_{q \in L_{\text{now}}^i} q$, it follows that $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i \models F_{\text{now}}^{i,0}$.

Assume that for all $0 \leqslant i \leqslant n$, $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i \models F_{\text{now}}^{i,d}$. Let $\psi \in F_{\text{now}}^{i,d+1}$ for some $0 \leqslant i \leqslant n$. If $\psi \in F_{\text{now}}^{i,d}$, then $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i \models \psi$. Otherwise, $\text{degree}(\psi) = d + 1$ and we distinguish the cases based on the syntactical form of ψ.

- $\psi = \psi_1 \wedge \psi_2$, where $\text{degree}(\psi_1) \leqslant d$, $\text{degree}(\psi_2) \leqslant d$. Since $\psi \in F_{\text{now}}^i$, then $F_{\text{now}}^i \models \psi_1$ and $F_{\text{now}}^i \models \psi_2$, and by Condition 5, $\psi_1 \in F_{\text{now}}^i$ and $\psi_2 \in F_{\text{now}}^i$. It follows that $\psi_1 \in F_{\text{now}}^{i,d}$ and $\psi_2 \in F_{\text{now}}^{i,d}$, so $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i \models \psi_1$ and $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i \models \psi_2$. Hence, $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i \models \psi_1 \wedge \psi_2$.
- $\psi = \psi_1 \vee \psi_2$, where $\text{degree}(\psi_1) \leqslant d$, $\text{degree}(\psi_2) \leqslant d$. Since $\psi \in F_{\text{now}}^i$, then by Condition 7 either $\psi_1 \in F_{\text{now}}^i$ or $\psi_2 \in F_{\text{now}}^i$. It follows that $\psi_1 \in F_{\text{now}}^{i,d}$ or $\psi_2 \in F_{\text{now}}^{i,d}$, so that $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i \models \psi_1$ or $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i \models \psi_2$. Hence, $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i \models \psi_1 \vee \psi_2$.
- $\psi = \bigcirc \psi_1$, where $\text{degree}(\psi_1) \leqslant d$. By Condition 8, either $i < n$, or $i = n$ and $\langle \emptyset, (\emptyset)_{0 \leqslant i \leqslant n} \rangle, n+1 \models \Box \psi_1$. In the latter case, note that the canonical models of $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle$ and $\langle \emptyset, (\emptyset)_{0 \leqslant i \leqslant n} \rangle$ coincide after time point n (atomic concepts and roles have empty interpretations), and since ψ_1 does not contain any past operators, $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, n + 1 \models \Box \psi_1$ is a direct consequence of $\langle \emptyset, (\emptyset)_{0 \leqslant i \leqslant n} \rangle, n + 1 \models \Box \psi_1$. Then ψ_1 is true at any time point $j > n$ and in particular, $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, n \models \bigcirc \psi_1$.
  In the former case, since $\bigcirc \psi_1 \in F_{\text{now}}^i = F_{\text{prev}}^{i+1}$, we have $\bigwedge_{q \in L_{\text{now}}^{i+1}} q \wedge \bigcirc^- (\bigwedge_{\chi \in F_{\text{prev}}^{i+1}} \chi) \wedge \bigcirc (\bigwedge_{\chi \in F_{\text{next}}^{i+1}} \chi) \models \bigcirc^- \bigcirc \psi_1 \models \psi_1$, and by Condition 6, $\psi_1 \in F_{\text{now}}^{i+1}$.

Hence, $\psi_1 \in F_{\text{now}}^{i+1,d}$, $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i+1 \models \psi_1$, and $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i \models \bigcirc \psi_1$.

- $\psi = \bigcirc^- \psi_1$, where $\text{degree}(\psi_1) \leqslant d$. This case is similar to $\bigcirc \psi_1$.

- $\psi = \bullet^b \psi_1$, where $\text{degree}(\psi_1) \leqslant d$. If $i < n$, since $\bullet^b \psi_1 \in F_{\text{now}}^i = F_{\text{prev}}^{i+1}$, we have that $\bigwedge_{q \in L_{\text{now}}^{i+1}} q \wedge \bigcirc^-(\bigwedge_{\chi \in F_{\text{prev}}^{i+1}} \chi) \wedge \bigcirc(\bigwedge_{\chi \in F_{\text{next}}^{i+1}} \chi) \models \bigcirc^- \bullet^b \psi_1 \models \psi_1$, so that by Condition 6, $\psi_1 \in F_{\text{now}}^{i+1}$. Hence $\psi_1 \in F_{\text{now}}^{i+1,d}$ and $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i+1 \models \psi_1$, which implies $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i \models \bullet^b \psi_1$. Otherwise, $i = n$, and $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, n \models \bullet^b \psi_1$ by definition of $\bullet^b$.

- $\psi = \bullet^- \psi_1$ where $\text{degree}(\psi_1) \leqslant d$. This case is similar to $\bullet^b \psi_1$.

- $\psi = \Box \psi_1$, where $\text{degree}(\psi_1) \leqslant d$. By Condition 7, $\langle \emptyset, (\emptyset)_{0 \leqslant i \leqslant n} \rangle, n+1 \models \Box \psi_1$.
  We show that $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i \models \Box \psi_1$ by descending induction on i.
  For $i = n$, note that the canonical models of $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle$ and $\langle \emptyset, (\emptyset)_{0 \leqslant i \leqslant n} \rangle$ coincide after time point n (atomic concepts and roles have empty interpretations), and since ψ_1 does not contain any past operators, $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, n+1 \models \Box \psi_1$ is a direct consequence of $\langle \emptyset, (\emptyset)_{0 \leqslant i \leqslant n} \rangle, n+1 \models \Box \psi_1$. Then, $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, n \models \Box \psi_1$ iff $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, n \models \psi_1$, that is iff $\psi_1 \in F_{\text{now}}^{n,d}$ by induction. This is the case by Condition 5.
  For $i < n$, we assume by inductive hypothesis that $\Box \psi_1 \in F_{\text{now}}^{i+1}$ implies $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i+1 \models \Box \psi_1$. Since $\Box \psi_1 \in F_{\text{now}}^i = F_{\text{prev}}^{i+1}$, we have that $\bigwedge_{q \in L_{\text{now}}^{i+1}} q \wedge \bigcirc^-(\bigwedge_{\chi \in F_{\text{prev}}^{i+1}} \chi) \wedge \bigcirc(\bigwedge_{\chi \in F_{\text{next}}^{i+1}} \chi) \models \bigcirc^- \Box \psi_1 \models \Box \psi_1$, so that by Condition 6, $\Box \psi_1 \in F_{\text{now}}^{i+1}$, and by assumption $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i+1 \models \Box \psi_1$. Moreover, since $\Box \psi_1 \in F_{\text{now}}^i$, then $F_{\text{now}}^i \models \psi_1$, and $\psi_1 \in F_{\text{now}}^i$ by Condition 5. Hence, $\psi_1 \in F_{\text{now}}^{i,d}$ and $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i \models \psi_1$.
  It follows that $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i \models \Box \psi_1$.

- $\psi = \Box^b \psi_1$ where $\text{degree}(\psi_1) \leqslant d$. We show $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i \models \Box^b \psi_1$ by descending induction on i.
  For $i = n$, if $\Box^b \psi_1 \in F_{\text{now}}^n$, then $\psi_1 \in F_{\text{now}}^n$ by Condition 8, and therefore $\psi_1 \in F_{\text{now}}^{n,d}$ and $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, n \models \psi_1$. As a consequence, we obtain that $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, n \models \Box^b \psi_1$.
  For $i < n$, we assume by inductive hypothesis that $\Box^b \psi_1 \in F_{\text{now}}^{i+1}$ implies $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i+1 \models \Box^b \psi_1$. Then, since $\Box^b \psi_1 \in F_{\text{now}}^i = F_{\text{prev}}^{i+1}$, we have that $\bigwedge_{q \in L_{\text{now}}^{i+1}} q \wedge \bigcirc^-(\bigwedge_{\chi \in F_{\text{prev}}^{i+1}} \chi) \wedge \bigcirc(\bigwedge_{\chi \in F_{\text{next}}^{i+1}} \chi) \models \bigcirc^- \Box^b \psi_1 \models \Box^b \psi_1$. Therefore, by Condition 6, $\Box^b \psi_1 \in F_{\text{now}}^{i+1}$, and by assumption, $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i+1 \models \Box^b \psi_1$. Moreover, since $\Box^b \psi_1 \in F_{\text{now}}^i$, then

$F_{\text{now}}^i \models \psi_1$, and $\psi_1 \in F_{\text{now}}^i$ by Condition 5. Hence, $\psi_1 \in F_{\text{now}}^{i,d}$ and $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i \models \psi_1$. It follows that $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i \models \Box^b \psi_1$.

- $\psi = \Box^- \psi_1$, where $\text{degree}(\psi_1) \leqslant d$. This case is similar to $\Box^b \psi_1$.

- $\psi = \Diamond \psi_1$, where $\text{degree}(\psi_1) \leqslant d$. We prove $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i \models \Diamond \psi_1$ by descending induction on i.
  For $i = n$, if $\Diamond \psi_1 \in F_{\text{now}}^n$, then $\psi_1 \in F_{\text{now}}^n$ or $\langle \emptyset, (\emptyset)_{0 \leqslant i \leqslant n} \rangle, n+1 \models \Box \psi_1$ by Condition 8.
  In the former case, $\psi_1 \in F_{\text{now}}^{n,d}$ and $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, n \models \psi_1$, which implies that $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, n \models \Diamond \psi_1$.
  In the latter case we can show as in the proof for $\bigcirc$ that $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, n+1 \models \Box \psi_1$, which implies that $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, n+1 \models \Diamond \psi_1$.
  For $i < n$, we assume by inductive hypothesis that $\Diamond \psi_1 \in F_{\text{now}}^{i+1}$ implies $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i+1 \models \Diamond \psi_1$. Since $\Diamond \psi_1 \in F_{\text{now}}^i$, by Condition 7, either (i) $\psi_1 \in F_{\text{now}}^i$, $\psi_1 \in F_{\text{now}}^{i,d}$ and $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i \models \psi_1$, and therefore $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i \models \Diamond \psi_1$, or (ii) $\Diamond \psi_1 \in F_{\text{next}}^i = F_{\text{now}}^{i+1}$, and by assumption $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i+1 \models \Diamond \psi_1$. It follows that $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i \models \Diamond \psi_1$.

- $\psi = \Diamond^b \psi_1$, where $\text{degree}(\psi_1) \leqslant d$. This case is similar as for $\Diamond \psi_1$.

- $\psi = \Diamond^- \psi_1$, where $\text{degree}(\psi_1) \leqslant d$. This case is similar to $\Diamond \psi_1$.

- $\psi = \psi_1 \mathsf{U} \psi_2$ where $\text{degree}(\psi_1) \leqslant d$, $\text{degree}(\psi_2) \leqslant d$. We show that $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i \models \psi_1 \mathsf{U} \psi_2$ by descending induction on i.
  For $i = n$, if $\psi_1 \mathsf{U} \psi_2 \in F_{\text{now}}^n$, then $\psi_2 \in F_{\text{now}}^n$, or $\langle \emptyset, (\emptyset)_{0 \leqslant i \leqslant n} \rangle, n+1 \models \Box \psi_2$ by Condition 8.
  In the former case, $\psi_2 \in F_{\text{now}}^{n,d}$, so $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, n \models \psi_2$, which implies that $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, n \models \psi_1 \mathsf{U} \psi_2$.
  In the latter case, we can show as in the proof for $\bigcirc$ that $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, n+1 \models \Box \psi_2$, which implies that $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, n+1 \models \psi_1 \mathsf{U} \psi_2$. Then $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, n \models \psi_1 \mathsf{U} \psi_2$ iff $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, n \models \psi_1$ or $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, n \models \psi_2$, that is iff $\psi_1 \in F_{\text{now}}^{n,d}$ or $\psi_1 \in F_{\text{now}}^{n,d}$ by induction. This is the case by Condition 5.
  For $i < n$, we assume by inductive hypothesis that $\psi_1 \mathsf{U} \psi_2 \in F_{\text{now}}^{i+1}$ implies $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i+1 \models \psi_1 \mathsf{U} \psi_2$. Then, since $\psi_1 \mathsf{U} \psi_2 \in F_{\text{now}}^i$, by Condition 7, either (i) $\psi_2 \in F_{\text{now}}^i$, $\psi_2 \in F_{\text{now}}^{i,d}$ and $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i \models \psi_2$, which in turn implies $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i \models \psi_1 \mathsf{U} \psi_2$, or (ii) $\psi_1 \in F_{\text{now}}^i$ and $\psi_1 \in F_{\text{now}}^{i,d}$, which implies $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i \models \psi_1$, and $\psi_1 \mathsf{U} \psi_2 \in F_{\text{next}}^i = F_{\text{now}}^{i+1}$. Therefore, by assumption we obtain $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i+1 \models \psi_1 \mathsf{U} \psi_2$ and $\langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle, i \models \psi_1 \mathsf{U} \psi_2$.

– $\psi = \psi_1 \mathsf{U}^b \psi_2$, where $\mathsf{degree}(\psi_1) \leqslant d$, $\mathsf{degree}(\psi_2) \leqslant d$. This case can be shown in the same way as for $\psi_1 \mathsf{U} \psi_2$.

– $\psi = \psi_1 \mathsf{S} \psi_2$, where $\mathsf{degree}(\psi_1) \leqslant d$, $\mathsf{degree}(\psi_2) \leqslant d$. This case is similar to $\psi_1 \mathsf{U} \psi_2$. $\qquad\square$

Lemma A.4. *If* $\mathsf{N}_{\mathsf{RC}} = \mathsf{N}_{\mathsf{RR}} = \emptyset$ *and* $\mathcal{K}, p \models_{brave} \phi$, *then there exists a brave-justification structure for* ϕ *in* $\mathcal{K}$ *that is correct and justifies* ϕ *at time point* p.

Proof. Assume $\mathcal{K}, p \models_{brave} \phi$. Then there exists a TKB $\mathcal{K}' = \langle \mathcal{T}, (\mathcal{C}_i)_{0 \leqslant i \leqslant n} \rangle$, such that $\mathcal{C}_i \subseteq \mathcal{A}_i$ and $\mathcal{K}'$ is consistent and $\mathcal{K}', p \models \phi$. Based on $\mathcal{K}'$, we construct a brave-justification structure J for ϕ in $\mathcal{K}$ that justifies ϕ at time point p. The elements of the tuples $(i, L^i_{\mathrm{now}}, F^i_{\mathrm{now}}, F^i_{\mathrm{prev}}, F^i_{\mathrm{next}})$ are selected as follows.

1. L^i_{now} is the largest subset of $BCQ(\phi)$ such that $\mathcal{K}', i \models \bigwedge_{q \in L^i_{\mathrm{now}}} q$,
2. F^i_{now} is the largest subset of $F(\phi)$ such that $\mathcal{K}', i \models F^i_{\mathrm{now}}$,
3. $F^i_{\mathrm{prev}} = F^{i-1}_{\mathrm{now}}$ for $i > 0$,
4. $F^i_{\mathrm{next}} = F^{i+1}_{\mathrm{now}}$ for $i < n$, and
5. $F^0_{\mathrm{prev}} = F^n_{\mathrm{next}} = \emptyset$.

We show that J is correct and justifies ϕ at time point p. The latter case is easy: since $\mathcal{K}', p \models \phi$, we have $\phi \in F^p_{\mathrm{now}}$, and therefore ϕ is justified by J at time point p. It remains to show that J is correct, i.e., that every tuple of J satisfies all conditions in the definition of justified tuples.

Conditions 1, 2, 3 and 4 follow straightforwardly from the construction. Condition 5 is satisfied because if $\psi \in F(\phi)$ is such that $\psi \notin F^i_{\mathrm{now}}$, then $\mathcal{K}', i \not\models \psi$ and $F^i_{\mathrm{now}} \not\models \psi$.

For Condition 6, we show that for every $\psi \in F(\phi)$ and for every $0 \leqslant i \leqslant n$, if $\bigwedge_{q \in L^i_{\mathrm{now}}} q \wedge \bigcirc^-(\bigwedge_{\chi \in F^i_{\mathrm{prev}}} \chi) \wedge \bigcirc(\bigwedge_{\chi \in F^i_{\mathrm{next}}} \chi) \models \psi$, then $\mathcal{K}', i \models \psi$, which in turn implies $\psi \in F^i_{\mathrm{now}}$. Since $\mathcal{K}'$ entails every CQ in L^i_{now} at time point i, every TCQ in F^i_{prev} at time point $i - 1$, and every TCQ in F^i_{next} at time point $i + 1$, every TCQ that corresponds to a formula entailed by L^i_{now}, $\bigcirc^-(\bigwedge_{\chi \in F^i_{\mathrm{prev}}} \chi)$ or $\bigcirc(\bigwedge_{\chi \in F^i_{\mathrm{next}}} \chi)$ is entailed from $\mathcal{K}'$ at time point i. Hence, if $\bigwedge_{q \in L^i_{\mathrm{now}}} q \wedge \bigcirc^-(\bigwedge_{\chi \in F^i_{\mathrm{prev}}} \chi) \wedge \bigcirc(\bigwedge_{\chi \in F^i_{\mathrm{next}}} \chi) \models \psi$, then $\mathcal{K}', i \models \psi$.

For Condition 7, we do a case analysis based on the structure of the elements in F_{now}, using the Proposition 9.12 and the fact that $\mathsf{N}_{\mathsf{RC}} = \mathsf{N}_{\mathsf{RR}} = \emptyset$.

– If $\mathcal{K}', i \models \psi \vee \psi'$, then $\mathcal{K}', i \models \psi$ or $\mathcal{K}', i \models \psi'$. Therefore, if $\psi \vee \psi' \in F^i_{\mathrm{now}}$, either $\psi \in F^i_{\mathrm{now}}$, or $\psi' \in F^i_{\mathrm{now}}$.

– If $\mathcal{K}', i \models \Diamond \psi$, then $\mathcal{K}', i \models \psi$ or $\mathcal{K}', i + 1 \models \Diamond \psi$. Therefore, if $\Diamond \psi \in F^i_{\mathrm{now}}$, either $\psi \in F^i_{\mathrm{now}}$ or $\Diamond \psi \in F^{i+1}_{\mathrm{now}} = F^i_{\mathrm{next}}$.

– If $\mathcal{K}', i \models \Diamond^b \psi$, then $\mathcal{K}', i \models \psi$ or $\mathcal{K}', i + 1 \models \Diamond^b \psi$. Therefore, if $\Diamond^b \psi \in F^i_{\mathrm{now}}$, either $\psi \in F^i_{\mathrm{now}}$ or $\Diamond^b \psi \in F^{i+1}_{\mathrm{now}} = F^i_{\mathrm{next}}$.

– If $\mathcal{K}', i \models \Diamond^- \psi$, then $\mathcal{K}', i \models \psi$ or $\mathcal{K}', i - 1 \models \Diamond^- \psi$. Therefore, if $\Diamond^- \psi \in F^i_{\mathrm{now}}$, either $\psi \in F^i_{\mathrm{now}}$ or $\Diamond^- \psi \in F^{i-1}_{\mathrm{now}} = F^i_{\mathrm{prev}}$.

– If $\mathcal{K}', i \models \psi \mathsf{U} \psi'$, then $\mathcal{K}', i \models \psi'$, or $\mathcal{K}', i \models \psi$ and $\mathcal{K}', i+1 \models \psi \mathsf{U} \psi'$. Therefore, if $\psi \mathsf{U} \psi' \in F^i_{\mathrm{now}}$, either $\psi' \in F^i_{\mathrm{now}}$, or $\psi \in F^i_{\mathrm{now}}$ and $\psi \mathsf{U} \psi' \in F^i_{\mathrm{next}}$.

– If $\mathcal{K}', i \models \psi \mathsf{U}^b \psi'$, then $\mathcal{K}', i \models \psi'$ or $\mathcal{K}', i \models \psi$ and $\mathcal{K}', i + 1 \models \psi \mathsf{U}^b \psi'$. Therefore, if $\psi \mathsf{U}^b \psi' \in F^i_{\mathrm{now}}$, either $\psi' \in F^i_{\mathrm{now}}$, or $\psi \in F^i_{\mathrm{now}}$ and $\psi \mathsf{U}^b \psi' \in F^i_{\mathrm{next}}$.

– If $\mathcal{K}', i \models \psi \mathsf{S} \psi'$, then $\mathcal{K}', i \models \psi'$, or $\mathcal{K}', i \models \psi$ and $\mathcal{K}', i - 1 \models \psi \mathsf{S} \psi'$. Therefore, if $\psi \mathsf{S} \psi' \in F^i_{\mathrm{now}}$, then either $\psi' \in F^i_{\mathrm{now}}$, or $\psi \in F^i_{\mathrm{now}}$ and $\psi \mathsf{S} \psi' \in F^i_{\mathrm{prev}}$.

– If ψ is of the form $\Box \varphi$ and $\psi \in F_{\mathrm{now}}$, i.e., $\mathcal{K}', i \models \Box \varphi$, then for every $j > n$, $\mathcal{K}', j \models \varphi$. Since φ contains no past operators, and without rigid predicates, no BCQ is entailed from $\mathcal{K}'$ at time point $j > n$, the only possibility is that φ is trivially entailed at any time point $j > n$. It follows that $\langle \emptyset, (\emptyset)_{0 \leqslant i \leqslant n} \rangle, n + 1 \models \Box \varphi$.

The proof of Condition 8 is as follows.

– If $\psi \in F(\phi)$ is of the form $\bullet^b \varphi$, then $\mathcal{K}', n \models \psi$ and also $\psi \in F^n_{\mathrm{now}}$.

– Assume $\psi \in F(\phi)$ is of the form $\bigcirc \varphi$ and such that $\langle \emptyset, (\emptyset)_{0 \leqslant i \leqslant n} \rangle, n + 1 \not\models \Box \varphi$. Note that, because $\langle \emptyset, (\emptyset)_{0 \leqslant i \leqslant n} \rangle, n + 1 \not\models \Box \varphi$, φ cannot be trivially entailed at any time point $j > n$, and consequently also not at $n + 1$. In the absence of rigid roles, and because ψ cannot contain past operators, we therefore have $\mathcal{K}', n + 1 \not\models \varphi$, which implies $\mathcal{K}', n \not\models \bigcirc \varphi$ and $\psi \notin F^n_{\mathrm{now}}$.

– If $\varphi \in F^n_{\mathrm{now}}$, then $\mathcal{K}', n \models \varphi$, which in turn implies $\mathcal{K}', n \models \Diamond \varphi$, $\mathcal{K}', n \models \Diamond^b \varphi$, $\mathcal{K}', n \models \Box^b \varphi$, $\mathcal{K}', n \models \varphi' \mathsf{U} \varphi$ and $\mathcal{K}', n \models \varphi' \mathsf{U}^b \varphi$. It follows that if any of those entailed TCQs are in $F(\phi)$, then they are also in F^n_{now}. For the other direction, we do a case analysis.

  * Assume $\Diamond \varphi \in F^n_{\mathrm{now}}$ and $\langle \emptyset, (\emptyset)_{0 \leqslant i \leqslant n} \rangle, n + 1 \not\models \Box \varphi$. Note that, because $\langle \emptyset, (\emptyset)_{0 \leqslant i \leqslant n} \rangle, n+1 \not\models \Box \varphi$, φ cannot be trivially entailed at any time point $j > n$. If this would be the case, due to the absence of rigid predicates, φ would also be trivially entailed at every time point $j > n$. Because $\mathcal{K}', n \models \Diamond \varphi$, we must have $\mathcal{K}', n \models \varphi$, since i) without rigid predicates, φ cannot be entailed at any time point

$j > n$, and ii) φ does not contain past operators. Therefore, $\varphi \in F^n_{\text{now}}$.

* If $\Diamond^b \varphi \in F^n_{\text{now}}$, then $\mathcal{K}', n \models \Diamond^b \varphi$, which in turn implies $\mathcal{K}', n \models \varphi$ and $\varphi \in F^n_{\text{now}}$.

* If $\Box^b \varphi \in F^n_{\text{now}}$, then $\mathcal{K}', n \models \Box^b \varphi$, which in turn implies $\mathcal{K}', n \models \varphi$ and $\varphi \in F^n_{\text{now}}$.

* Assume $\varphi' \mathsf{U} \varphi \in F^n_{\text{now}}$ and $\langle \emptyset, (\emptyset)_{0 \leqslant i \leqslant n} \rangle, n+1 \not\models \Box \varphi$. Similar to the case for $\Diamond$, then $\mathcal{K}', n \models \varphi' \mathsf{U} \varphi$ and $\mathcal{K}', n \models \varphi$, because i) φ cannot be entailed at time point $j > n$ in the absence of rigid predicates, and ii) φ does not contain past operators. Consequently, $\varphi \in F^n_{\text{now}}$.

* If $\varphi' \mathsf{U}^b \varphi \in F^n_{\text{now}}$, also $\mathcal{K}', n \models \varphi' \mathsf{U}^b \varphi$, which in turn implies $\mathcal{K}', n \models \varphi$ and $\varphi \in F^n_{\text{now}}$.

Condition 9 can be shown similarly as Condition 8. We have shown that every tuple in J is justified, and consequently that J is correct and justifies ϕ at p. $\square$

A.2. Proofs for Section 9

The following properties of $chase_{\text{rig}}(\mathcal{K})$ will be useful for the proofs of Subsection 9.1.

Proposition A.5. $chase_{\text{rig}}(\mathcal{K})$ satisfies the following properties.

(P1) $x^{i_1}_{aP_1} \in \Gamma_N$ implies $P_1(a, x^{i_1}_{aP_1}) \in chase^{\mathcal{K}}_{\text{rig}}(\mathcal{A}_{i_1})$.

(P2) $x^{i_1 \dots i_l}_{aP_1 \dots P_l} \in \Gamma_N, l > 1$ implies that $chase^{\mathcal{K}}_{\text{rig}}(\mathcal{A}_{i_l})$ contains $P_l(x^{i_1 \dots i_{l-1}}_{aP_1 \dots P_{l-1}}, x^{i_1 \dots i_l}_{aP_1 \dots P_l})$.

(P3) $chase^{\mathcal{K}}_{\text{rig}}(\mathcal{A}_i) \models B(x^{i_1 \dots i_l}_{aP_1 \dots P_l})$ implies that $\mathcal{T} \models \exists P^-_l \sqsubseteq B$.

(P4) $x^{i_1 \dots i_l}_{aP_1 \dots P_l} \in \Gamma_N, l > 1$ implies $\mathcal{T} \models \exists P^-_{l-1} \sqsubseteq \exists P_l$.

(P5) $chase^{\mathcal{K}}_{\text{rig}}(\mathcal{A}_i) \models B(a), a \in \mathsf{N}^{\mathcal{K}}_\mathsf{I}$ implies $\langle \mathcal{T}, \mathcal{A}_i \rangle \models B(a)$ or there exists $B' := A|\exists R|\exists R^-$ with $A \in \mathsf{N}_{\mathsf{RC}}, R \in \mathsf{N}_{\mathsf{RR}}$ such that $\mathcal{T} \models B' \sqsubseteq B$ and there exists j such that $\langle \mathcal{T}, \mathcal{A}_j \rangle \models B'(a)$.

(P6) $chase^{\mathcal{K}}_{\text{rig}}(\mathcal{A}_i) \models B(x^{i_1 \dots i_l}_{aP_1 \dots P_l})$ implies $i = i_l$ or there exists $B' := A|\exists R|\exists R^-$ with $A \in \mathsf{N}_{\mathsf{RC}}, R \in \mathsf{N}_{\mathsf{RR}}$ such that $\mathcal{T} \models B' \sqsubseteq B$ and $chase^{\mathcal{K}}_{\text{rig}}(\mathcal{A}_{i_l}) \models B'(x^{i_1 \dots i_l}_{aP_1 \dots P_l})$.

(P7) $P(a, b) \in chase^{\mathcal{K}}_{\text{rig}}(\mathcal{A}_i), a, b \in \mathsf{N}^{\mathcal{K}}_\mathsf{I}$ implies $\langle \mathcal{T}, \mathcal{A}_i \rangle \models P(a, b)$ or there exists $P' := R|R^-$ with $R \in \mathsf{N}_{\mathsf{RR}}$ such that $\mathcal{T} \models P' \sqsubseteq P$ and there exists j such that $\langle \mathcal{T}, \mathcal{A}_j \rangle \models P'(a, b)$.

(P8) $P(a, x^{i_1}_{aP_1}) \in chase^{\mathcal{K}}_{\text{rig}}(\mathcal{A}_i), a \in \mathsf{N}^{\mathcal{K}}_\mathsf{I}, i_1 = i$ implies $\mathcal{T} \models P_1 \sqsubseteq P$ and $\langle \mathcal{T}, \mathcal{A}_i \rangle \models \exists x. P_1(a, x)$ or there exists $B := A|\exists R|\exists R^-$ with $A \in \mathsf{N}_{\mathsf{RC}}, R \in \mathsf{N}_{\mathsf{RR}}$ such that $\mathcal{T} \models B \sqsubseteq \exists P_1$ and there exists j such that $\langle \mathcal{T}, \mathcal{A}_j \rangle \models B(a)$.

(P9) $P(a, x^{i_1}_{aP_1}) \in chase^{\mathcal{K}}_{\text{rig}}(\mathcal{A}_i), a \in \mathsf{N}^{\mathcal{K}}_\mathsf{I}, i_1 \neq i$ implies there exists $P' := R|R^-$ with $R \in \mathsf{N}_{\mathsf{RR}}$ such that $\mathcal{T} \models P_1 \sqsubseteq P' \sqsubseteq P$.

(P10) $P(x, y) \in chase^{\mathcal{K}}_{\text{rig}}(\mathcal{A}_i), x, y \in \Gamma_N$ implies $x = x^{i_1 \dots i_l}_{aP_1 \dots P_l}, y = x^{i_1 \dots i_l i_{l+1}}_{aP_1 \dots P_l P_{l+1}}$ and $\mathcal{T} \models P_{l+1} \sqsubseteq P$ or $x = x^{i_1 \dots i_l i_{l+1}}_{aP_1 \dots P_l P_{l+1}}, y = x^{i_1 \dots i_l}_{aP_1 \dots P_l}$ and $\mathcal{T} \models P_{l+1} \sqsubseteq P^-$.

(P11) $P(x^{i_1 \dots i_l}_{aP_1 \dots P_l}, x^{i_1 \dots i_l i_{l+1}}_{aP_1 \dots P_l P_{l+1}}) \in chase^{\mathcal{K}}_{\text{rig}}(\mathcal{A}_i), i_{l+1} \neq i$ implies there exists $P' := R|R^-$ with $R \in \mathsf{N}_{\mathsf{RR}}$ such that $\mathcal{T} \models P_{l+1} \sqsubseteq P' \sqsubseteq P$ and $P'(x^{i_1 \dots i_l}_{aP_1 \dots P_l}, x^{i_1 \dots i_l i_{l+1}}_{aP_1 \dots P_l P_{l+1}}) \in chase^{\mathcal{K}}_{\text{rig}}(\mathcal{A}_{i_{l+1}})$.

(P12) $P_l(x^{i_1 \dots i_{l-1}}_{aP_1 \dots P_{l-1}}, x^{i_1 \dots i_l}_{aP_1 \dots P_l}) \in chase^{\mathcal{K}}_{\text{rig}}(\mathcal{A}_{i_l})$ implies $\exists j, \langle \mathcal{T}, \mathcal{A}_j \rangle \models \exists xy. P_{l-1}(x, y)$.

Proof. We refer to [54] for a detailed proof of these properties. $\square$

Lemma 9.3. *If $\mathcal{K}$ is consistent, then $\mathcal{J}_{\mathcal{K}}$ is a model of $\mathcal{K}$ that respects rigid predicates.*

Proof. We first show that $\mathcal{J}_{\mathcal{K}}$ is a model of $\mathcal{K}$, i.e., that for every $i \in [0, n]$, $\mathcal{I}_i \models \mathcal{A}_i$ and for every $i \geqslant 0$, $\mathcal{I}_i \models \mathcal{T}$. It is easy to see that for every $i \in [0, n]$, $\mathcal{I}_i \models \mathcal{A}_i$, because $\mathcal{A}_i \subseteq chase^{\mathcal{K}}_{\text{rig}}(\mathcal{A}_i)$. We can show that $\mathcal{I}_i$ satisfies every PI in $\mathcal{T}$ with similar arguments as those used in [50]. We only consider the case where $i \leqslant n + 1$ explicitly. For the case where $i > n + 1$, we assume $\mathcal{A}_i$ to be replaced by $\mathcal{A}_{n+1}$ in what follows. If a PI $\alpha \in \mathcal{T}_p$ is not satisfied, there is an assertion $\beta \in chase^{\mathcal{K}}_{\text{rig}}(\mathcal{A}_i)$ such that α is applicable to β in $chase^{\mathcal{K}}_{\text{rig}}(\mathcal{A}_i)$. This is impossible given that every PI applicable to β in $\mathcal{S}^j_i$ at step j of the construction of the rigid chase becomes not applicable to β in $\mathcal{S}^k_i$ for some $k \geqslant j$. Indeed, because each PI can only be applied once to a given assertion, there are only finitely many assertions before β, and only finitely many PIs are applied to the assertions that precede β. Finally, we show that because $\mathcal{K}$ is consistent, $\mathcal{I}_i$ satisfies every negative inclusion of $\mathcal{T}$. Indeed, if a negative inclusion would not be satisfied, this would imply the existence of a conflict $\mathcal{B}$ in $chase^{\mathcal{K}}_{\text{rig}}(\mathcal{A}_i)$. If $\mathcal{B} = \{\alpha\}$, the timed assertion $(\alpha', j) \in (\mathcal{A}_i)_{0 \leqslant i \leqslant n}$ from which α has been

derived by applying PIs from $\mathcal{T}_p$ is clearly inconsistent. Otherwise $\mathcal{B} = \{\alpha, \beta\}$ with α derived from (α', j) and β derived from (β', k). If $j = k$, $\{(\alpha', j), (\beta', k)\}$ is clearly inconsistent. If $j \neq k$, since α and β belong to $chase^{\mathcal{K}}_{\text{rig}}(\mathcal{A}_i)$, if $j \neq i$ (resp. $k \neq i$), there exists $\alpha'' \in chase^{\mathcal{K}}_{\text{rig}}(\mathcal{A}_i)$ rigid such that α derives from α'', which derives from α' (resp. $\beta'' \in chase^{\mathcal{K}}_{\text{rig}}(\mathcal{A}_i)$ rigid such that β derives from β'', which derives from β'). Therefore, and because no sequence of interpretations that respects rigid predicates can be a model of $\{(\alpha', j), (\beta', k)\}$ and $\mathcal{T}$, $\{(\alpha', j), (\beta', k)\}$ is inconsistent.

Moreover, the model $\mathcal{J}_{\mathcal{K}}$ respects rigid predicates, because if an assertion β of $chase^{\mathcal{K}}_{\text{rig}}(\mathcal{A}_i)$ is rigid, either $\beta \in \mathcal{A}_i$ and by construction $\beta \in \mathcal{S}^0_k = \mathcal{A}'_k$ for every k, or β has been derived at some step j by applying some PI to an assertion of $\mathcal{S}^j$ and $\beta \in \mathcal{S}^{j+1}_k$ for every k, so that in both cases $\beta \in chase^{\mathcal{K}}_{\text{rig}}(\mathcal{A}_k)$ for every k. $\square$

Lemma 9.4. *If $\mathcal{K}$ is consistent, then for every BTCQ without negation ϕ such that $\mathsf{N}^\phi_\mathsf{I} \subseteq \mathsf{N}^{\mathcal{K}}_\mathsf{I}$, $\mathcal{K}, p \models \phi$ iff $\mathcal{J}_{\mathcal{K}}, p \models \phi$.*

Proof. Since $\mathcal{J}_{\mathcal{K}} = (\mathcal{I}_i)_{i \geqslant 0}$ with $\mathcal{I}_i = \langle \Delta, \cdot^{\mathcal{I}_i} \rangle$ is a model of $\mathcal{K}$ that respects rigid predicates, the first direction is clear, and we only need to show that $\mathcal{J}_{\mathcal{K}}, p \models \phi$ implies $\mathcal{K}, p \models \phi$. Let $\mathcal{J} = (\mathcal{I}'_i)_{i \geqslant 0}$ with $\mathcal{I}'_i = \langle \Delta', \cdot^{\mathcal{I}'_i} \rangle$ be a model of $\mathcal{K}$ that respects rigid predicates. We show by structural induction on ϕ that if $\mathcal{J}_{\mathcal{K}}, p \models \phi$, then $\mathcal{J}, p \models \phi$.

If ϕ is a BCQ $\exists \vec{y}.\psi(\vec{y})$, we show that if there exists a homomorphism π of $\exists \vec{y}.\psi(\vec{y})$ into $\mathcal{I}_p$, then $\mathcal{I}'_p \models \exists \vec{y}.\psi(\vec{y})$. We define a mapping h from Δ into Δ', where we assume w.l.o.g. that Δ and Δ' are disjoint.

1. For every $a \in \mathsf{N}^{\mathcal{K}}_\mathsf{I}$, set $h(a^{\mathcal{I}_p}) = a^{\mathcal{I}'_p}$.
2. For every $x^{i_1}_{aP_1} \in \Gamma_N$, set $h(x^{i_1 \mathcal{I}_p}_{aP_1}) = y$, where $(a^{\mathcal{I}'_p}, y) \in P^{\mathcal{I}'_{i_1}}_1$. If there are several such y, choose one of them randomly.
3. For every $x^{i_1 \ldots i_l}_{aP_1 \ldots P_l} \in \Gamma_N$ with $l > 1$, set $h(x^{i_1 \ldots i_l \mathcal{I}_p}_{aP_1 \ldots P_l}) = y$, where $(h(x^{i_1 \ldots i_{l-1} \mathcal{I}_p}_{aP_1 \ldots P_{l-1}}), y) \in P^{\mathcal{I}'_{i_l}}_l$. If there are several such y, choose one of them randomly.

We first show that h is well-defined, i.e., that in the two latter cases, there always exists a y as required. We show this by induction on l. For $l = 1$, because $x^{i_1}_{aP_1} \in \Gamma_N$, by (P1) $P_1(a, x^{i_1}_{aP_1}) \in chase^{\mathcal{K}}_{\text{rig}}(\mathcal{A}_{i_1})$. Therefore, by (P8), either (i) $\langle \mathcal{T}, \mathcal{A}_{i_1} \rangle \models \exists x.P_1(a, x)$, and since $\mathcal{I}'_{i_1}$ is a model of $\langle \mathcal{T}, \mathcal{A}_{i_1} \rangle$, there exists $(a^{\mathcal{I}'_p}, y) \in P^{\mathcal{I}'_{i_1}}_1$, or (ii) there exists $B := A | \exists R | \exists R^-$ with $A \in \mathsf{N}_{\mathsf{RC}}$,

$R \in \mathsf{N}_{\mathsf{RR}}$, such that $\mathcal{T} \models B \sqsubseteq \exists P_1$, and there exists j such that $\langle \mathcal{T}, \mathcal{A}_j \rangle \models B(a)$. In the latter case, since $\mathcal{J}$ is a model of $\mathcal{K}$ that respects rigid predicates, $\mathcal{I}'_{i_1} \models B(a)$. Since $\mathcal{I}'_{i_1}$ is a model of $\mathcal{T}$, there exists $(a^{\mathcal{I}'_p}, y) \in P^{\mathcal{I}'_{i_1}}_1$. Then, for $l > 1$, since $x^{i_1 \ldots i_l}_{aP_1 \ldots P_l} \in \Gamma_N$, by (P4), $\mathcal{T} \models \exists P^-_{l-1} \sqsubseteq \exists P_l$. Since by induction we have $(x, h(x^{i_1 \ldots i_{l-1} \mathcal{I}_p}_{aP_1 \ldots P_{l-1}})) \in P^{\mathcal{I}'_{i_l}}_{l-1}$, it follows that there exists $(h(x^{i_1 \ldots i_{l-1} \mathcal{I}_p}_{aP_1 \ldots P_{l-1}}), y) \in P^{\mathcal{I}'_{i_l}}_l$.

Next, we show that h is a homomorphism of $\mathcal{I}_p$ into $\mathcal{I}'_p$, which then implies that $h \circ \pi$ is a homomorphism of $\exists \vec{y}.\psi(\vec{y})$ into $\mathcal{I}'_p$. We only consider the case $p \leqslant n$ explicitly, and assume $\mathcal{A}_p$ to be replaced with $\mathcal{A}_{n+1}$ for the case $p > n$.

For every $a \in \mathsf{N}^{\mathcal{K}}_\mathsf{I}$ and concept A, if $a^{\mathcal{I}_p} \in A^{\mathcal{I}_p}$, i.e., $A(a) \in chase^{\mathcal{K}}_{\text{rig}}(\mathcal{A}_p)$, then by (P5), either (i) $\langle \mathcal{T}, \mathcal{A}_p \rangle \models A(a)$, and since $\mathcal{I}'_p$ is a model of $\langle \mathcal{T}, \mathcal{A}_p \rangle$, also $h(a^{\mathcal{I}_p}) = a^{\mathcal{I}'_p} \in A^{\mathcal{I}'_p}$, or (ii) there exists a concept $B = C | \exists R | \exists R^-$ with $C \in \mathsf{N}_{\mathsf{RC}}$, $R \in \mathsf{N}_{\mathsf{RR}}$, such that $\mathcal{T} \models B \sqsubseteq A$ and there exists j such that $\langle \mathcal{T}, \mathcal{A}_j \rangle \models B(a)$. In the latter case, since $\mathcal{J}$ is a model of $\mathcal{K}$ that respects rigid predicates, $\mathcal{I}'_p \models B(a)$. Since $\mathcal{I}'_p$ is a model of $\mathcal{T}$, it follows that $\mathcal{I}'_p \models A(a)$, so $h(a^{\mathcal{I}_p}) = a^{\mathcal{I}'_p} \in A^{\mathcal{I}'_p}$. For every pair $a, b \in \mathsf{N}^{\mathcal{K}}_\mathsf{I}$ and role P, if $(a^{\mathcal{I}_p}, b^{\mathcal{I}_p}) \in P^{\mathcal{I}_p}$, by (P7), similar arguments can be used to prove that $(h(a^{\mathcal{I}_p}), h(b^{\mathcal{I}_p})) = (a^{\mathcal{I}'_p}, b^{\mathcal{I}'_p}) \in P^{\mathcal{I}'_p}$.

For every $x^{i_1 \ldots i_l}_{aP_1 \ldots P_l} \in \Gamma_N$, such that $x^{i_1 \ldots i_l \mathcal{I}_p}_{aP_1 \ldots P_l} \in A^{\mathcal{I}_p}$, i.e., $A(x^{i_1 \ldots i_l}_{aP_1 \ldots P_l}) \in chase^{\mathcal{K}}_{\text{rig}}(\mathcal{A}_p)$, by (P6) we are in one of the following cases.

1. $i_l = p$. By (P3), $\mathcal{T} \models \exists P^-_l \sqsubseteq A$ and by construction of h, $h(x^{i_1 \ldots p \mathcal{I}_p}_{aP_1 \ldots P_l}) = y$ with $(h(x^{i_1 \ldots i_{l-1} \mathcal{I}_p}_{aP_1 \ldots P_{l-1}}), y) \in P^{\mathcal{I}'_p}_l$. Note that if $l = 1$, then $x^{i_1 \ldots i_{l-1}}_{aP_1 \ldots P_{l-1}} = a$. Since $\mathcal{I}'_p$ is a model of $\mathcal{T}$, it follows that $y \in A^{\mathcal{I}'_p}$.
2. There exists $B := C | \exists R | \exists R^-$ with $C \in \mathsf{N}_{\mathsf{RC}}, R \in \mathsf{N}_{\mathsf{RR}}$ such that $\mathcal{T} \models B \sqsubseteq A$ and $chase^{\mathcal{K}}_{\text{rig}}(\mathcal{A}_{i_l}) \models B(x^{i_1 \ldots i_l}_{aP_1 \ldots P_l})$. As in Case 1, by (P3) and definition of h we have that $h(x^{i_1 \ldots i_l \mathcal{I}_p}_{aP_1 \ldots P_l}) = y \in B^{\mathcal{I}'_{i_l}}$. Since B is rigid, $y \in B^{\mathcal{I}'_p}$. Since $\mathcal{I}'_p$ is a model of $\mathcal{T}$, it follows that $y \in A^{\mathcal{I}'_p}$.

For every pair $x, y \in \Gamma_N$ and role P such that $(x^{\mathcal{I}_p}, y^{\mathcal{I}_p}) \in P^{\mathcal{I}_p}$, by (P10) $x = x^{i_1 \ldots i_l}_{aP_1 \ldots P_l}$, $y = x^{i_1 \ldots i_l i_{l+1}}_{aP_1 \ldots P_l P_{l+1}}$ and $\mathcal{T} \models P_{l+1} \sqsubseteq P$, or $x = x^{i_1 \ldots i_l i_{l+1}}_{aP_1 \ldots P_l P_{l+1}}$, $y = x^{i_1 \ldots i_l}_{aP_1 \ldots P_l}$ and $\mathcal{T} \models P_{l+1} \sqsubseteq P^-$. We can assume w.l.o.g. that we are in the first case. Otherwise we

consider $(y^{\mathcal{I}_p}, x^{\mathcal{I}_p}) \in P^{-\mathcal{I}_p}$. If $i_{l+1} = p$, by definition of h, $(h(x^{\mathcal{I}_p}), h(y^{\mathcal{I}_p})) \in P_{l+1}^{\mathcal{I}_p'}$, and since $\mathcal{I}_p'$ is a model of $\mathcal{T}$, $(h(x^{\mathcal{I}_p}), h(y^{\mathcal{I}_p})) \in P^{\mathcal{I}_p'}$. Otherwise, by (P11), there exists $P' := R|R^-$ with $R \in \mathsf{N}_{\mathsf{RR}}$ such that $\mathcal{T} \models P_{l+1} \sqsubseteq P' \sqsubseteq P$ and $P'(x,y) \in \mathit{chase}_{\mathrm{rig}}^{\mathcal{K}}(\mathcal{A}_{i_{l+1}})$. With the same arguments as in the first case, we show that $(h(x^{\mathcal{I}_p}), h(y^{\mathcal{I}_p})) \in P'^{\mathcal{I}_{i_{l+1}}'}$, and since P' is rigid $(h(x^{\mathcal{I}_p}), h(y^{\mathcal{I}_p})) \in P'^{\mathcal{I}_p'}$. Since $\mathcal{I}_p'$ is a model of $\mathcal{T}$, it follows that $(h(x^{\mathcal{I}_p}), h(y^{\mathcal{I}_p})) \in P^{\mathcal{I}_p'}$.

Finally, if $a \in \mathsf{N}_{\mathsf{I}}^{\mathcal{K}}$ and $x \in \Gamma_N$, then $(a^{\mathcal{I}_p}, x^{\mathcal{I}_p}) \in P^{\mathcal{I}_p}$ only if $x = x_{aP_1}^{i_1}$. If $i_1 = p$, by definition of h, $(h(a^{\mathcal{I}_p}), h(x^{\mathcal{I}_p})) \in P_1^{\mathcal{I}_p'}$. Since by (P8), $\mathcal{T} \models P_1 \sqsubseteq P$ and $\mathcal{I}_p'$ is a model of $\mathcal{T}$, it follows that $(h(a^{\mathcal{I}_p}), h(x^{\mathcal{I}_p})) \in P^{\mathcal{I}_p'}$. If $i_1 \neq p$, by (P9), there exists a rigid role P' such that $\mathcal{T} \models P_1 \sqsubseteq P' \sqsubseteq P$, and since by definition of h, $(h(a^{\mathcal{I}_p}), h(x^{\mathcal{I}_p})) \in P_1^{\mathcal{I}_{i_1}'}$, then $(h(a^{\mathcal{I}_p}), h(x^{\mathcal{I}_p})) \in P'^{\mathcal{I}_{i_1}'}$. Since $\mathcal{J}$ respects rigid predicates, it follows that $(h(a^{\mathcal{I}_p}), h(x^{\mathcal{I}_p})) \in P'^{\mathcal{I}_p'}$ and $(h(a^{\mathcal{I}_p}), h(x^{\mathcal{I}_p})) \in P^{\mathcal{I}_p'}$.

We have thus shown that $\mathcal{J}_{\mathcal{K}}, p \models \exists \vec{y}.\psi(\vec{y})$ implies $\mathcal{J}, p \models \exists \vec{y}.\psi(\vec{y})$.

Now for the inductive step, assume that for two BTCQs ϕ_1, ϕ_2 such that $\mathsf{N}_{\mathsf{I}}^{\phi_1} \subseteq \mathsf{N}_{\mathsf{I}}^{\mathcal{K}}$ and $\mathsf{N}_{\mathsf{I}}^{\phi_2} \subseteq \mathsf{N}_{\mathsf{I}}^{\mathcal{K}}$, we have that $\mathcal{J}_{\mathcal{K}}, p \models \phi_i$ implies $\mathcal{J}, p \models \phi_i$ ($i \in \{1,2\}$). We show that then, for every BTCQ ϕ that we can construct in one step from ϕ_1 and ϕ_2, $\mathcal{J}_{\mathcal{K}}, p \models \phi$ also implies $\mathcal{J}, p \models \phi$. We distinguish the cases based on ϕ.

- If $\mathcal{J}_{\mathcal{K}}, p \models \phi_1 \wedge \phi_2$, then $\mathcal{J}_{\mathcal{K}}, p \models \phi_1$ and $\mathcal{J}_{\mathcal{K}}, p \models \phi_2$, and therefore by assumption, $\mathcal{J}, p \models \phi_1$ and $\mathcal{J}, p \models \phi_2$. Hence, $\mathcal{J}, p \models \phi_1 \wedge \phi_2$.
- If $\mathcal{J}_{\mathcal{K}}, p \models \phi_1 \vee \phi_2$, then $\mathcal{J}_{\mathcal{K}}, p \models \phi_1$ or $\mathcal{J}_{\mathcal{K}}, p \models \phi_2$, and therefore by assumption, $\mathcal{J}, p \models \phi_1$ or $\mathcal{J}, p \models \phi_2$. Hence, $\mathcal{J}, p \models \phi_1 \vee \phi_2$.
- If $\mathcal{J}_{\mathcal{K}}, p \models \bigcirc \phi_1$, then $\mathcal{J}_{\mathcal{K}}, p+1 \models \phi_1$, and therefore by assumption, $\mathcal{J}, p+1 \models \phi_1$. Hence, $\mathcal{J}, p \models \bigcirc \phi_1$.
- In the same way as in the last case, we can show that $\mathcal{J}_{\mathcal{K}}, p \models \bullet^b \phi_1$ implies $\mathcal{J}, p \models \bullet^b \phi_1$, that $\mathcal{J}_{\mathcal{K}}, p \models \bigcirc^- \phi_1$ implies $\mathcal{J}, p \models \bigcirc^- \phi_1$, and that $\mathcal{J}_{\mathcal{K}}, p \models \bullet^- \phi_1$ implies $\mathcal{J}, p \models \bullet^- \phi_1$.
- If $\mathcal{J}_{\mathcal{K}}, p \models \Box \phi_1$, then for every $k \geqslant p$, $\mathcal{J}_{\mathcal{K}}, k \models \phi_1$, and therefore, by assumption, for every $k \geqslant p$, $\mathcal{J}, k \models \phi_1$. Hence, $\mathcal{J}, p \models \Box \phi_1$.
- In the same way as in the last case, we can show that $\mathcal{J}_{\mathcal{K}}, p \models \Box^b \phi_1$ implies $\mathcal{J}, p \models \Box^b \phi_1$ and that $\mathcal{J}_{\mathcal{K}}, p \models \Box^- \phi_1$ implies $\mathcal{J}, p \models \Box^- \phi_1$.
- If $\mathcal{J}_{\mathcal{K}}, p \models \Diamond \phi_1$, then there exists $k \geqslant p$, $\mathcal{J}_{\mathcal{K}}, k \models \phi_1$, and therefore by assumption $\mathcal{J}, k \models \phi_1$. Hence, $\mathcal{J}, p \models \Diamond \phi_1$.
- In the same way as in the last case, we can show that $\mathcal{J}_{\mathcal{K}}, p \models \Diamond^b \phi_1$ implies $\mathcal{J}, p \models \Diamond^b \phi_1$, and that $\mathcal{J}_{\mathcal{K}}, p \models \Diamond^- \phi_1$ implies $\mathcal{J}, p \models \Diamond^- \phi_1$.
- If $\mathcal{J}_{\mathcal{K}}, p \models \phi_1 \mathsf{U} \phi_2$, then there exists $k \geqslant p$ such that $\mathcal{J}_{\mathcal{K}}, k \models \phi_2$ and for every $j, p \leqslant j < k$, $\mathcal{J}_{\mathcal{K}}, j \models \phi_1$. Therefore, by assumption $\mathcal{J}, k \models \phi_2$, and for every $j, p \leqslant j < k$, $\mathcal{J}, j \models \phi_1$. Hence, $\mathcal{J}, p \models \phi_1 \mathsf{U} \phi_2$.
- We can show in the same way as in the last case that $\mathcal{J}_{\mathcal{K}}, p \models \phi_1 \mathsf{U}^b \phi_2$ implies $\mathcal{J}, p \models \phi_1 \mathsf{U}^b \phi_2$, and that $\mathcal{J}_{\mathcal{K}}, p \models \phi_1 \mathsf{S} \phi_2$ implies $\mathcal{J}, p \models \phi_1 \mathsf{S} \phi_2$.

We conclude by induction that for every BTCQ ϕ without negation such that $\mathsf{N}_{\mathsf{I}}^{\phi} \subseteq \mathsf{N}_{\mathsf{I}}^{\mathcal{K}}$, $\mathcal{J}_{\mathcal{K}}, p \models \phi$ implies $\mathcal{J}, p \models \phi$. It follows that $\mathcal{J}_{\mathcal{K}}, p \models \phi$ implies $\mathcal{K}, p \models \phi$.

We have thus shown that for every BTCQ ϕ without negation such that $\mathsf{N}_{\mathsf{I}}^{\phi} \subseteq \mathsf{N}_{\mathsf{I}}^{\mathcal{K}}$, we have $\mathcal{K}, p \models \phi$ iff $\mathcal{J}_{\mathcal{K}}, p \models \phi$. □

Lemma 9.7. *Let $q = \exists \vec{y}.\psi(\vec{y})$ be such that $\mathsf{N}_{\mathsf{I}}^q \subseteq \mathsf{N}_{\mathsf{I}}^{\mathcal{K}}$. Then, for every $p \in [0,n]$, if $\mathcal{K}_{\mathcal{R}}, p \models q$ then $\mathcal{K}, p \models q$.*

Proof. Assume that $\mathcal{K}_{\mathcal{R}}, p \models \exists \vec{y}.\psi(\vec{y})$. By Proposition 3.6, since $\mathsf{N}_{\mathsf{RC}} = \mathsf{N}_{\mathsf{RR}} = \emptyset$, $\langle \mathcal{T}, (\mathcal{A}_p \cup \mathcal{R}) \rangle \models \exists \vec{y}.\psi(\vec{y})$. Let $\mathcal{I}_p^{\mathcal{R}} = \langle \Delta^{\mathcal{I}_p^{\mathcal{R}}}, \cdot^{\mathcal{I}_p^{\mathcal{R}}} \rangle$ be the canonical model of $\langle \mathcal{T}, (\mathcal{A}_p \cup \mathcal{R}) \rangle$, and let $\mathcal{J}_{\mathcal{K}} = (\mathcal{I}_i)_{i \geqslant 0}$ be the canonical model of $\mathcal{K}$, where $\mathcal{I}_i = \langle \Delta, \cdot^{\mathcal{I}_i} \rangle$. There exists a homomorphism π of $\exists \vec{y}.\psi(\vec{y})$ into $\mathcal{I}_p^{\mathcal{R}}$. We first define a mapping σ from $\{ x^{\mathcal{I}_p^{\mathcal{R}}} \mid x \in \mathsf{N}_{\mathsf{I}}^{\mathcal{K}}$ or occurs in $\mathcal{R} \}$ into $\{ x^{\mathcal{I}_p} \mid x \in \mathsf{N}_{\mathsf{I}}^{\mathcal{K}} \cup \Gamma_N, x$ occurs in $\mathit{chase}_{\mathrm{rig}}^{\mathcal{K}}(\mathcal{A}_p) \}$, where we assume Δ and $\Delta^{\mathcal{I}_p^{\mathcal{R}}}$ to be disjoint, by

- $\sigma(a^{\mathcal{I}_p^{\mathcal{R}}}) = a^{\mathcal{I}_p}$ for $a \in \mathsf{N}_{\mathsf{I}}^{\mathcal{K}}$,
- $\sigma(x_{aP}^{\mathcal{I}_p^{\mathcal{R}}}) = x^{\mathcal{I}_p}$ such that $P(a,x) \in \mathit{chase}_{\mathrm{rig}}^{\mathcal{K}}(\mathcal{A}_p)$,
- $\sigma(x_P^{\mathcal{I}_p^{\mathcal{R}}}) = x^{\mathcal{I}_p}$ such that there exists $P(y,x) \in \bigcup_{i=0}^n \mathit{chase}_{\mathrm{rig}}^{\mathcal{K}}(\mathcal{A}_i)$, and
- $\sigma(x_{PP'}^{\mathcal{I}_p^{\mathcal{R}}}) = x^{\mathcal{I}_p}$ such that $P'(y,x) \in \mathit{chase}_{\mathrm{rig}}^{\mathcal{K}}(\mathcal{A}_p)$ with $\sigma(x_P^{\mathcal{I}_p^{\mathcal{R}}}) = y^{\mathcal{I}_p}$.

Claim 1. σ is well-defined.

Proof of claim. If x_{aP} occurs in $\mathcal{R}$, there exists i such that $\langle \mathcal{T}, \mathcal{A}_i \rangle \models \exists x.P(a,x)$. Since $\mathcal{I}_i$ is a model of $\langle \mathcal{T}, \mathcal{A}_i \rangle$, it follows that there is some $P(a,x) \in \mathit{chase}_{\mathrm{rig}}^{\mathcal{K}}(\mathcal{A}_i)$. Moreover, since P is rigid, $P(a,x) \in \mathit{chase}_{\mathrm{rig}}^{\mathcal{K}}(\mathcal{A}_p)$.

If x_P occurs in $\mathcal{R}$, there exists i such that $\langle \mathcal{T}, \mathcal{A}_i \rangle \models \exists xy.P(x,y)$. Since $\mathcal{I}_i$ is a model of $\langle \mathcal{T}, \mathcal{A}_i \rangle$, it follows that there exist $x, y \in \mathsf{N}_{\mathsf{I}}^{\mathcal{K}} \cup \Gamma_N$ such that $P(y,x) \in \mathit{chase}_{\mathrm{rig}}^{\mathcal{K}}(\mathcal{A}_i)$. Moreover, x occurs in $\mathit{chase}_{\mathrm{rig}}^{\mathcal{K}}(\mathcal{A}_p)$ because there exists $B := A|\exists R|\exists R^-$ with $A \in \mathsf{N}_{\mathsf{RC}}$

and $R \in \mathsf{N_{RR}}$ such that $\mathcal{T} \models \exists P^- \sqsubseteq B$, and therefore there is a rigid assertion $\beta \models B(x)$ such that $\beta \in chase_{\mathrm{rig}}^{\mathcal{K}}(\mathcal{A}_p)$.

If $x_{PP'}$ occurs in $\mathcal{R}$, then x_P also occurs in $\mathcal{R}$. It follows that there exist i and $y \in \mathsf{N_I^{\mathcal{K}}} \cup \Gamma_N$ such that $P(y, \sigma(x_P^{\mathcal{I}_p^{\mathcal{R}}})) \in chase_{\mathrm{rig}}^{\mathcal{K}}(\mathcal{A}_i)$. Moreover, by construction of $\mathcal{R}$, P' is rigid and such that $\mathcal{T} \models \exists P^- \sqsubseteq \exists P'$. Since $\mathcal{I}_i$ is a model of $\mathcal{T}$, there then exists $x \in \mathsf{N_I^{\mathcal{K}}} \cup \Gamma_N$ such that $P'(\sigma(x_P^{\mathcal{I}_p^{\mathcal{R}}}), x) \in chase_{\mathrm{rig}}^{\mathcal{K}}(\mathcal{A}_i)$. Hence, $P'(\sigma(x_P^{\mathcal{I}_p^{\mathcal{R}}}), x) \in chase_{\mathrm{rig}}^{\mathcal{K}}(\mathcal{A}_p)$. ∎

Claim 2. σ is a partial homomorphism of $\mathcal{I}_p^{\mathcal{R}}$ into $\mathcal{I}_p$.

Proof of claim. For every $a \in \mathsf{N_I^{\mathcal{K}}}$ and concept A, if $a^{\mathcal{I}_p^{\mathcal{R}}} \in A^{\mathcal{I}_p^{\mathcal{R}}}$, since $\mathcal{I}_p^{\mathcal{R}}$ is the canonical model of $\langle \mathcal{T}, (\mathcal{A}_p \cup \mathcal{R}) \rangle$, then $\langle \mathcal{T}, (\mathcal{A}_p \cup \mathcal{R}) \rangle \models A(a)$. Let $\{\alpha\}$ be a cause for $A(a)$. If $\alpha \in \mathcal{A}_p$, then $\alpha \in chase_{\mathrm{rig}}^{\mathcal{K}}(\mathcal{A}_p)$. In this case, since $\mathcal{I}_p$ is a model of $\mathcal{T}$ and $\langle \mathcal{T}, \alpha \rangle \models A(a)$, then $\sigma(a^{\mathcal{I}_p^{\mathcal{R}}}) = a^{\mathcal{I}_p} \in A^{\mathcal{I}_p}$. Otherwise, $\alpha \in \mathcal{R}$, and α is either of the form $A'(a)$ with $A' \in \mathsf{N_{RC}}$, or of the form $P(a, b)$ or $P(a, x_{aP})$, where P is rigid. In the first two cases, there exists i such that $\langle \mathcal{T}, \mathcal{A}_i \rangle \models \alpha$. Therefore, since $\mathcal{I}_i$ is a model of $\langle \mathcal{T}, \mathcal{A}_i \rangle$, $\alpha \in chase_{\mathrm{rig}}^{\mathcal{K}}(\mathcal{A}_i)$. Since α is rigid, $\alpha \in chase_{\mathrm{rig}}^{\mathcal{K}}(\mathcal{A}_p)$, and therefore, since $\mathcal{I}_p$ is a model of $\mathcal{T}$ and $\langle \mathcal{T}, \alpha \rangle \models A(a)$, we obtain that $\sigma(a^{\mathcal{I}_p^{\mathcal{R}}}) = a^{\mathcal{I}_p} \in A^{\mathcal{I}_p}$. In the last case, if $\alpha = P(a, x_{aP})$, there exists i such that $\langle \mathcal{T}, \mathcal{A}_i \rangle \models \exists x.P(a, x)$. Since $\mathcal{I}_i$ is a model of $\langle \mathcal{T}, \mathcal{A}_i \rangle$, there is some $P(a, x) \in chase_{\mathrm{rig}}^{\mathcal{K}}(\mathcal{A}_i)$. Since P is rigid, $P(a, x) \in chase_{\mathrm{rig}}^{\mathcal{K}}(\mathcal{A}_p)$, and since $\mathcal{I}_p$ is a model of $\mathcal{T}$ and $\langle \mathcal{T}, P(a, x) \rangle \models A(a)$, we obtain $\sigma(a^{\mathcal{I}_p^{\mathcal{R}}}) = a^{\mathcal{I}_p} \in A^{\mathcal{I}_p}$.

For every pair $a, b \in \mathsf{N_I^{\mathcal{K}}}$ and role P, if $(a^{\mathcal{I}_p^{\mathcal{R}}}, b^{\mathcal{I}_p^{\mathcal{R}}}) \in P^{\mathcal{I}_p^{\mathcal{R}}}$, we can use similar arguments to show that $(\sigma(a^{\mathcal{I}_p^{\mathcal{R}}}), \sigma(b^{\mathcal{I}_p^{\mathcal{R}}})) = (a^{\mathcal{I}_p}, b^{\mathcal{I}_p}) \in P^{\mathcal{I}_p}$.

For every x_{aP} that occurs in $\mathcal{R}$ and $A \in \mathsf{N_C}$, if $x_{aP}^{\mathcal{I}_p^{\mathcal{R}}} \in A^{\mathcal{I}_p^{\mathcal{R}}}$, since $\mathcal{I}_p^{\mathcal{R}}$ is the canonical model of $\langle \mathcal{T}, (\mathcal{A}_p \cup \mathcal{R}) \rangle$, then $\langle \mathcal{T}, (\mathcal{A}_p \cup \mathcal{R}) \rangle \models A(x_{aP})$. Let $\{\alpha\}$ be a cause for $A(x_{aP})$. By construction, the only assertion of $\mathcal{A}_p \cup \mathcal{R}$ that involves x_{aP} is $P(a, x_{aP})$. Therefore, $\alpha = P(a, x_{aP})$ and $\langle \mathcal{T}, P(a, x_{aP}) \rangle \models A(x_{aP})$. Since $\sigma(x_{aP}^{\mathcal{I}_p^{\mathcal{R}}}) = x^{\mathcal{I}_p}$ is such that $P(a, x) \in chase_{\mathrm{rig}}^{\mathcal{K}}(\mathcal{A}_p)$, and $\mathcal{I}_p$ is a model of $\mathcal{T}$, then $\sigma(x_{aP}^{\mathcal{I}_p^{\mathcal{R}}}) \in A^{\mathcal{I}_p}$.

For every $a \in \mathsf{N_I^{\mathcal{K}}}$, $x \notin \mathsf{N_I^{\mathcal{K}}}$ that occurs in $\mathcal{R}$ and role P, if $(a^{\mathcal{I}_p^{\mathcal{R}}}, x^{\mathcal{I}_p^{\mathcal{R}}}) \in P^{\mathcal{I}_p^{\mathcal{R}}}$, since $\mathcal{I}_p^{\mathcal{R}}$ is the canonical model of $\langle \mathcal{T}, (\mathcal{A}_p \cup \mathcal{R}) \rangle$, then $\langle \mathcal{T}, (\mathcal{A}_p \cup \mathcal{R}) \rangle \models P(a, x)$. Let $\{\alpha\}$ be a cause for $P(a, x)$. By construc-

tion of $\mathcal{R}$, $x = x_{aP_1}$ and $\alpha = P_1(a, x_{aP_1})$, and by definition of σ, $(\sigma(a^{\mathcal{I}_p^{\mathcal{R}}}), \sigma(x_{aP_1P}^{\mathcal{I}_p^{\mathcal{R}}})) \in P_1^{\mathcal{I}_p}$. Since $\langle \mathcal{T}, P_1(a, x) \rangle \models P(a, x)$ and $\mathcal{I}_p$ is a model of $\mathcal{T}$, it follows that $(\sigma(a^{\mathcal{I}_p^{\mathcal{R}}}), \sigma(x_{aP_1P}^{\mathcal{I}_p^{\mathcal{R}}})) \in P^{\mathcal{I}_p}$.

For every x_{P_1} that occurs in $\mathcal{R}$ and $A \in \mathsf{N_C}$, if $x_{P_1}^{\mathcal{I}_p^{\mathcal{R}}} \in A^{\mathcal{I}_p^{\mathcal{R}}}$, since $\mathcal{I}_p^{\mathcal{R}}$ is the canonical model of $\langle \mathcal{T}, (\mathcal{A}_p \cup \mathcal{R}) \rangle$, then $\langle \mathcal{T}, (\mathcal{A}_p \cup \mathcal{R}) \rangle \models A(x_{P_1})$. Let $\{\alpha\}$ be a cause for $A(x_{P_1})$. By construction, either $\alpha = A'(x_{P_1})$ with $A' \in \mathsf{N_{RC}}$ and $\mathcal{T} \models \exists P_1^- \sqsubseteq A'$, or $\alpha = P_2(x_{P_1}, x_{P_1P_2})$ with P_2 rigid and $\mathcal{T} \models \exists P_1^- \sqsubseteq \exists P_2$. Since $\sigma(x_{P_1}^{\mathcal{I}_p^{\mathcal{R}}}) = x^{\mathcal{I}_p}$ is such that there exists i such that $P_1(y, x) \in chase_{\mathrm{rig}}^{\mathcal{K}}(\mathcal{A}_i)$ and $\mathcal{I}_i$ is a model of $\mathcal{T}$, then $A'(x) \in chase_{\mathrm{rig}}^{\mathcal{K}}(\mathcal{A}_i)$ (resp. there is some $P_2(x, z) \in chase_{\mathrm{rig}}^{\mathcal{K}}(\mathcal{A}_i)$). Therefore $A'(x) \in chase_{\mathrm{rig}}^{\mathcal{K}}(\mathcal{A}_p)$ (resp. there is some $P_2(x, z) \in chase_{\mathrm{rig}}^{\mathcal{K}}(\mathcal{A}_p)$). Because $\mathcal{I}_p$ is a model of $\mathcal{T}$, it follows that $\sigma(x_{P_1}^{\mathcal{I}_p^{\mathcal{R}}}) \in A^{\mathcal{I}_p}$.

For every $x_{P_1P_2}$ that occurs in $\mathcal{R}$ and $A \in \mathsf{N_C}$, if $x_{P_1P_2}^{\mathcal{I}_p^{\mathcal{R}}} \in A^{\mathcal{I}_p^{\mathcal{R}}}$, since $\mathcal{I}_p^{\mathcal{R}}$ is the canonical model of $\langle \mathcal{T}, (\mathcal{A}_p \cup \mathcal{R}) \rangle$, then $\langle \mathcal{T}, (\mathcal{A}_p \cup \mathcal{R}) \rangle \models A(x_{P_1P_2})$. Let $\{\alpha\}$ be a cause for $A(x_{P_1P_2})$. By construction, $\alpha = P_2(x_{P_1}, x_{P_1P_2})$, P_2 is rigid, and $\mathcal{T} \models \exists P_1^- \sqsubseteq \exists P_2$. Since $\sigma(x_{P_1P_2}^{\mathcal{I}_p^{\mathcal{R}}}) = x^{\mathcal{I}_p}$ is such that there exists $P_2(y, x) \in chase_{\mathrm{rig}}^{\mathcal{K}}(\mathcal{A}_p)$ (with $y^{\mathcal{I}_p^{\mathcal{R}}} = \sigma(x_{P_1}^{\mathcal{I}_p^{\mathcal{R}}})$) and $\mathcal{I}_p$ is a model of $\mathcal{T}$, then $\sigma(x_{P_1P_2}^{\mathcal{I}_p^{\mathcal{R}}}) \in A^{\mathcal{I}_p}$.

Finally, for every $x, y \notin \mathsf{N_I^{\mathcal{K}}}$ that occur in $\mathcal{R}$ and for every role P, if $(x^{\mathcal{I}_p^{\mathcal{R}}}, y^{\mathcal{I}_p^{\mathcal{R}}}) \in P^{\mathcal{I}_p^{\mathcal{R}}}$, since $\mathcal{I}_p^{\mathcal{R}}$ is the canonical model of $\langle \mathcal{T}, (\mathcal{A}_p \cup \mathcal{R}) \rangle$, then $\langle \mathcal{T}, (\mathcal{A}_p \cup \mathcal{R}) \rangle \models P(x, y)$. Let $\{\alpha\}$ be a cause for $P(x, y)$. By construction, $x = x_{P_1}$, $y = x_{P_1P_2}$, $\alpha = P_2(x_{P_1}, x_{P_1P_2})$, and P_2 is rigid, and therefore, as previously, $(\sigma(x_{P_1}^{\mathcal{I}_p^{\mathcal{R}}}), \sigma(x_{P_1P_2}^{\mathcal{I}_p^{\mathcal{R}}})) \in P^{\mathcal{I}_p}$. ∎

Claim 3. σ can be extended to a homomorphism σ' of $\mathcal{I}_p^{\mathcal{R}}$ into $\mathcal{I}_p$.

Proof of claim. Since $\mathcal{I}_p^{\mathcal{R}}$ is the canonical model of $\langle \mathcal{T}, (\mathcal{A}_p \cup \mathcal{R}) \rangle$, $\mathcal{I}_p$ is a model of $\mathcal{T}$, and σ preserves the concept or role memberships, we can extend σ to a homomorphism σ' of $\mathcal{I}_p^{\mathcal{R}}$ into $\mathcal{I}_p$ by mapping the anonymous part of $\mathcal{I}_p^{\mathcal{R}}$ rooted in $x^{\mathcal{I}_p^{\mathcal{R}}} \in \{x^{\mathcal{I}_p^{\mathcal{R}}} \mid x \in \mathsf{N_I^{\mathcal{K}}} \text{ or occurs in } \mathcal{R}\}$ to the part of $\mathcal{I}_p$ rooted in $\sigma(x^{\mathcal{I}_p^{\mathcal{R}}})$. ∎

From Claim 3, it follows that $\sigma' \circ \pi$ is a homomorphism of $\exists \vec{y}.\psi(\vec{y})$ into $\mathcal{I}_p$. We have thus shown that $\mathcal{K}_{\mathcal{R}}, p \models \exists \vec{y}.\psi(\vec{y})$ implies $\mathcal{I}_p \models \exists \vec{y}.\psi(\vec{y})$, i.e.,

$\mathcal{J}_\mathcal{K}, p \models \exists \vec{y}.\psi(\vec{y})$. Hence, if $\mathcal{K}_\mathcal{R}, p \models \exists \vec{y}.\psi(\vec{y})$, then $\mathcal{K}, p \models \exists \vec{y}.\psi(\vec{y})$. $\qquad\square$

Lemma 9.8. *Let $q = \exists \vec{y}.\psi(\vec{y})$ be such that $\mathsf{N}_\mathsf{I}^q \subseteq \mathsf{N}_\mathsf{I}^\mathcal{K}$. For every $p \in [0, n]$, if $\mathcal{K}, p \models q$ then $\mathcal{K}_\mathcal{R}, p \models q$.*

Proof. Assume that $\mathcal{K}, p \models \exists \vec{y}.\psi(\vec{y})$. Let $\mathcal{J}_\mathcal{K} = (\mathcal{I}_i)_{i \geqslant 0}$ be the canonical model of $\mathcal{K}$, where $\mathcal{I}_i = \langle \Delta, \cdot^{\mathcal{I}_i} \rangle$. Then, $\mathcal{I}_p \models \exists \vec{y}.\psi(\vec{y})$, and there exists a homomorphism π of $\exists \vec{y}.\psi(\vec{y})$ into $\mathcal{I}_p$. Let $\mathcal{I}_p^\mathcal{R} = \langle \Delta^{\mathcal{I}_p^\mathcal{R}}, \cdot^{\mathcal{I}_p^\mathcal{R}} \rangle$ be a model of $\langle \mathcal{T}, (\mathcal{A}_i \cup \mathcal{R}) \rangle$. We define a mapping $h_p^\mathcal{R}$ from $\{ x^{\mathcal{I}_p} \mid x \in \mathsf{N}_\mathsf{I}^\mathcal{K} \cup \Gamma_N, x \text{ occurs in } chase_{\mathrm{rig}}^\mathcal{K}(\mathcal{A}_p) \}$ into $\Delta^{\mathcal{I}_p^\mathcal{R}}$, where we again assume that Δ and $\Delta^{\mathcal{I}_p^\mathcal{R}}$ are disjoint.

- For every $a \in \mathsf{N}_\mathsf{I}^\mathcal{K}$, we set $h_p^\mathcal{R}(a^{\mathcal{I}_p}) = a^{\mathcal{I}_p^\mathcal{R}}$.
- For every $x_{aP_1}^{i_1}$, where $i_1 \neq p$ and P_1 is rigid, we set $h_p^\mathcal{R}(x_{aP_1}^{i_1 \mathcal{I}_p}) = x_{aP_1}^{\mathcal{I}_p^\mathcal{R}}$.
- For every $x_{aP_1...P_l}^{i_1...i_l}$ with $l > 1$, such that for every $j \in [1, l]$, $i_j \neq p$, P_l is rigid, and P_{l-1} is not rigid, we set $h_p^\mathcal{R}(x_{aP_1...P_l}^{i_1...i_l \mathcal{I}_p}) = x_{P_{l-1} P_l}^{\mathcal{I}_p^\mathcal{R}}$.
- For every $x_{aP_1...P_l}^{i_1...i_l}$ with $l > 1$, such that for every $j \in [1, l]$, $i_j \neq p$, and P_l and P_{l-1} are rigid, we set $h_p^\mathcal{R}(x_{aP_1...P_l}^{i_1...i_l \mathcal{I}_p}) = y$, where $(h_p^\mathcal{R}(x_{aP_1...P_{l-1}}^{i_1...i_{l-1} \mathcal{I}_p}), y) \in P_l^{\mathcal{I}_p^\mathcal{R}}$. If there are several such y, we choose one of them randomly.
- For every $x_{aP_1...P_l}^{i_1...i_l}$ such that for every $k \in [1, l]$, $i_j \neq p$, and non-rigid role P_l, we set $h_p^\mathcal{R}(x_{aP_1...P_l}^{i_1...i_l \mathcal{I}_p}) = x_{P_l}^{\mathcal{I}_p^\mathcal{R}}$.
- For every $x_{aP_1...P_l}^{i_1...i_l}$ such that for some $j \in [1, l]$, $i_j = p$, we set $h_p^\mathcal{R}(x_{aP_1...P_l}^{i_1...i_l \mathcal{I}_p}) = y$, where we have $(h_p^\mathcal{R}(x_{aP_1...P_{l-1}}^{i_1...i_{l-1} \mathcal{I}_p}), y) \in P_l^{\mathcal{I}_p^\mathcal{R}}$. If there are several such y, we choose one of them randomly.

Claim 1. $h_p^\mathcal{R}$ is well-defined.

Proof of claim. We distinguish the cases based on the argument of $h_p^\mathcal{R}$.

- Case $x_{aP_1}^{i_1}$ with $i_1 \neq p$ and P_1 is rigid, $h_p^\mathcal{R}(x_{aP_1}^{i_1 \mathcal{I}_p}) = x_{aP_1}^{\mathcal{I}_p^\mathcal{R}}$. Since $x_{aP_1}^{i_1} \in \Gamma_N$, by (P1) and (P8), $\exists x.P_1(a, x)$ is entailed by some $\langle \mathcal{T}, \mathcal{A}_j \rangle$. Therefore, x_{aP_1} appears in $\mathcal{R}$.
- Case $x_{aP_1...P_l}^{i_1...i_l}$ with $l > 1$, such that for every $j \in [1, l]$, $i_j \neq p$, P_l is rigid and P_{l-1} is not rigid, $h_p^\mathcal{R}(x_{aP_1...P_l}^{i_1...i_l \mathcal{I}_p}) = x_{P_{l-1} P_l}^{\mathcal{I}_p^\mathcal{R}}$.

Since $x_{aP_1...P_l}^{i_1...i_l} \in \Gamma_N$, by (P4), $\mathcal{T} \models \exists P_{l-1}^- \sqsubseteq \exists P_l$, and by (P2) and (P12), there is some j such that $\langle \mathcal{T}, \mathcal{A}_j \rangle \models \exists xy.P_{l-1}(x, y)$. Moreover, P_l is rigid and P_{l-1} is not rigid, and therefore $x_{P_{l-1} P_l}$ appears in $\mathcal{R}$.
- Case $x_{aP_1...P_l}^{i_1...i_l}$ with $l > 1$, such that every $j \in [1, l]$, $i_j \neq p$, and P_l and P_{l-1} are rigid, $h_p^\mathcal{R}(x_{aP_1...P_l}^{i_1...i_l \mathcal{I}_p}) = y$, where $(h_p^\mathcal{R}(x_{aP_1...P_{l-1}}^{i_1...i_{l-1} \mathcal{I}_p}), y) \in P_l^{\mathcal{I}_p^\mathcal{R}}$.
We show by induction on the length $length = l - r$ of the sequence of rigid roles $P_r...P_{l-1}$ that there is always such a $(h_p^\mathcal{R}(x_{aP_1...P_{l-1}}^{i_1...i_{l-1} \mathcal{I}_p}), y) \in P_l^{\mathcal{I}_p^\mathcal{R}}$.
- If $length = 1$, we are in one of the following cases.
(i) $r > 1$ and $h_p^\mathcal{R}(x_{aP_1...P_{l-1}}^{i_1...i_{l-1} \mathcal{I}_p}) = x_{P_{l-2} P_{l-1}}^{\mathcal{I}_p^\mathcal{R}}$. Then $(x_{P_{l-2}}^{\mathcal{I}_p^\mathcal{R}}, x_{P_{l-2} P_{l-1}}^{\mathcal{I}_p^\mathcal{R}}) \in P_{l-1}^{\mathcal{I}_p^\mathcal{R}}$, because $\mathcal{I}_p^\mathcal{R}$ is a model of $\mathcal{R}$. Since $x_{aP_1...P_l}^{i_1...i_l} \in \Gamma_N$, by (P4), $\mathcal{T} \models \exists P_{l-1}^- \sqsubseteq \exists P_l$. Therefore, since $\mathcal{I}_p^\mathcal{R}$ is a model of $\mathcal{T}$, there is some $(h_p^\mathcal{R}(x_{aP_1...P_{l-1}}^{i_1...i_{l-1} \mathcal{I}_p}), y) \in P_l^{\mathcal{I}_p^\mathcal{R}}$.
(ii) $r = 1$ and $h_p^\mathcal{R}(x_{aP_1...P_{l-1}}^{i_1...i_{l-1} \mathcal{I}_p}) = h_p^\mathcal{R}(x_{aP_1}^{i_1 \mathcal{I}_p}) = x_{aP_1}^{\mathcal{I}_p^\mathcal{R}}$ is such that $(a^{\mathcal{I}_p^\mathcal{R}}, x_{aP_1}^{\mathcal{I}_p^\mathcal{R}}) \in P_1^{\mathcal{I}_p^\mathcal{R}}$ because $P_1(a, x_{aP_1}) \in \mathcal{R}$. Since $x_{aP_1 P_2}^{i_1 i_2} \in \Gamma_N$, $\mathcal{T} \models \exists P_1^- \sqsubseteq \exists P_2$ by (P4). Therefore, since $\mathcal{I}_p^\mathcal{R}$ is a model of $\mathcal{T}$, there exists $(x_{aP_1}^{\mathcal{I}_p^\mathcal{R}}, y) \in P_2^{\mathcal{I}_p^\mathcal{R}}$.
- For $length > 1$, $\mathcal{T} \models \exists P_{l-1}^- \sqsubseteq \exists P_l$ by (P4). It follows that, since by inductive hypothesis there exists $(x, h_p^\mathcal{R}(x_{aP_1...P_{l-1}}^{i_1...i_{l-1} \mathcal{I}_p})) \in P_{l-1}^{\mathcal{I}_p^\mathcal{R}}$, there then exists $(h_p^\mathcal{R}(x_{aP_1...P_{l-1}}^{i_1...i_{l-1} \mathcal{I}_p}), y) \in P_l^{\mathcal{I}_p^\mathcal{R}}$.
- Case $x_{aP_1...P_l}^{i_1...i_l}$ such that for every $j \in [1, l]$, $i_j \neq p$, and role P_l that is not rigid, $h_p^\mathcal{R}(x_{aP_1...P_l}^{i_1...i_l \mathcal{I}_p}) = x_{P_l}^{\mathcal{I}_p^\mathcal{R}}$.
Since $\mathcal{T}$ does not contain any role inclusion of the form $P' \sqsubseteq P$ with $P' := R_1|R_1^-$, $R_1 \in \mathsf{N}_\mathsf{R} \backslash \mathsf{N}_{\mathsf{RR}}$ and $P := R_2|R_2^-$, $R_2 \in \mathsf{N}_{\mathsf{RR}}$, and P_l is not rigid, there is no P such that $P_l \sqsubseteq P$ and P is rigid. Therefore, since $i_l \neq p$, there is no P such that $P(x_{aP_1...P_{l-1}}^{i_1...i_{l-1}}, x_{aP_1...P_l}^{i_1...i_l}) \in chase_{\mathrm{rig}}^\mathcal{K}(\mathcal{A}_p)$. We obtain that $x_{aP_1...P_l}^{i_1...i_l}$ occurs in $chase_{\mathrm{rig}}^\mathcal{K}(\mathcal{A}_p)$ only if there is $B := A|\exists R|\exists R^-$ with $A \in \mathsf{N}_{\mathsf{RC}}$, $R \in \mathsf{N}_{\mathsf{RR}}$ such that $chase_{\mathrm{rig}}^\mathcal{K}(\mathcal{A}_p) \models B(x_{aP_1...P_l}^{i_1...i_l})$. By (P3), $\mathcal{T} \models \exists P_l^- \sqsubseteq B$, and by (P2) and (P12), there is some j such that $\langle \mathcal{T}, \mathcal{A}_j \rangle \models \exists xy.P_{l-1}(x, y)$. It follows that x_{P_l} appears in $\mathcal{R}$.
- Case $x_{aP_1...P_l}^{i_1...i_l}$ such that for some $j \in [1, l]$, $i_j = p$, $h_p^\mathcal{R}(x_{aP_1...P_l}^{i_1...i_l \mathcal{I}_p}) = y$, where $(h_p^\mathcal{R}(x_{aP_1...P_{l-1}}^{i_1...i_{l-1} \mathcal{I}_p}), y) \in P_l^{\mathcal{I}_p^\mathcal{R}}$.
We show by induction on the length $length = l - r$ of the chain of roles that links $x_{aP_1...P_l}^{i_1...i_l}$ to the first individual $x_{aP_1...P_r}^{i_1...i_r}$ such that $i_r = p$ that there is always

such $(h_p^{\mathcal{R}}(x_{aP_1...P_{l-1}}^{i_1...i_{l-1}\mathcal{I}_p}), y) \in P_l^{\mathcal{I}_p^{\mathcal{R}}}$.

– If *length* $= 0$, then $i_l = p$ and there is no $j < l$ such that $i_j = p$. We are thus in one of the following cases. Either (i) $h_p^{\mathcal{R}}(x_{aP_1...P_{l-1}}^{i_1...i_{l-1}\mathcal{I}_p}) = a^{\mathcal{I}_p^{\mathcal{R}}}$, (ii) $h_p^{\mathcal{R}}(x_{aP_1...P_{l-1}}^{i_1...i_{l-1}\mathcal{I}_p}) = x_{aP_1}^{\mathcal{I}_p^{\mathcal{R}}}$, (iii) $h_p^{\mathcal{R}}(x_{aP_1...P_{l-1}}^{i_1...i_{l-1}\mathcal{I}_p}) = x_{P_{l-2}P_{l-1}}^{\mathcal{I}_p^{\mathcal{R}}}$, (iv) $h_p^{\mathcal{R}}(x_{aP_1...P_{l-1}}^{i_1...i_{l-1}\mathcal{I}_p})$ is such that $(h_p^{\mathcal{R}}(x_{aP_1...P_{l-2}}^{i_1...i_{l-2}\mathcal{I}_p}), h_p^{\mathcal{R}}(x_{aP_1...P_{l-1}}^{i_1...i_{l-1}\mathcal{I}_p})) \in P_{l-1}^{\mathcal{I}_p^{\mathcal{R}}}$, or (v) $h_p^{\mathcal{R}}(x_{aP_1...P_{l-1}}^{i_1...i_{l-1}\mathcal{I}_p}) = x_{P_{l-1}}^{\mathcal{I}_p^{\mathcal{R}}}$.

(i) If $h_p^{\mathcal{R}}(x_{aP_1...P_{l-1}}^{i_1...i_{l-1}\mathcal{I}_p}) = a^{\mathcal{I}_p^{\mathcal{R}}}$, by definition of $h_p^{\mathcal{R}}$, $x_{aP_1...P_{l-1}}^{i_1...i_{l-1}} = a$, and therefore $x_{aP_1...P_l}^{i_1...p} = x_{aP_1}^p$. Since $x_{aP_1}^p \in \Gamma_N$, by (P1), $P_1(a, x_{aP_1}^p) \in chase_{\text{rig}}^{\mathcal{K}}(\mathcal{A}_p)$. By (P8), either (a) $\langle \mathcal{T}, \mathcal{A}_p \rangle \models \exists x.P_1(a, x)$, and there is some $(a^{\mathcal{I}_p^{\mathcal{R}}}, y) \in P_1^{\mathcal{I}_p^{\mathcal{R}}}$ because $\mathcal{I}_p^{\mathcal{R}}$ is a model of $\langle \mathcal{T}, \mathcal{A}_p \rangle$, or (b) there exists $B := A|\exists R|\exists R^-$ with $A \in \mathsf{N_{RC}}, R \in \mathsf{N_{RR}}$, such that $\mathcal{T} \models B \sqsubseteq \exists P_1$ and there exists j such that $\langle \mathcal{T}, \mathcal{A}_j \rangle \models B(a)$. In the latter case, $\mathcal{R} \models B(a)$ by construction of $\mathcal{R}$, and since $\mathcal{I}_p^{\mathcal{R}}$ is a model of $\mathcal{R}$, we obtain $\mathcal{I}_p^{\mathcal{R}} \models B(a)$. Since $\mathcal{I}_p^{\mathcal{R}}$ is a model of $\mathcal{T}$, there exists $(a^{\mathcal{I}_p^{\mathcal{R}}}, y) \in P_1^{\mathcal{I}_p^{\mathcal{R}}}$.

(ii) If $h_p^{\mathcal{R}}(x_{aP_1...P_{l-1}}^{i_1...i_{l-1}\mathcal{I}_p}) = x_{aP_1}^{\mathcal{I}_p^{\mathcal{R}}}$, by definition of $h_p^{\mathcal{R}}$, $x_{aP_1...P_{l-1}}^{i_1...i_{l-1}} = x_{aP_1}^{i_1}$ and P_1 is rigid. By (P1), $P_1(a, x_{aP_1}^{i_1}) \in chase_{\text{rig}}^{\mathcal{K}}(\mathcal{A}_{i_1})$, and therefore by (P8), either (a) $\langle \mathcal{T}, \mathcal{A}_{i_1} \rangle \models \exists x.P_1(a, x)$ and $P_1(a, x_{aP_1}) \in \mathcal{R}$, since P_1 is rigid, or (b) there exists $B := A|\exists R|\exists R^-$ with $A \in \mathsf{N_{RC}}, R \in \mathsf{N_{RR}}$, such that $\mathcal{T} \models B \sqsubseteq \exists P_1$, and there exists j such that $\langle \mathcal{T}, \mathcal{A}_j \rangle \models B(a)$. In the latter case, $\langle \mathcal{T}, \mathcal{A}_j \rangle \models \exists x.P_1(a, x)$, and therefore $P_1(a, x_{aP_1}) \in \mathcal{R}$. In both cases, $(a^{\mathcal{I}_p^{\mathcal{R}}}, x_{aP_1}^{\mathcal{I}_p^{\mathcal{R}}}) \in P_1^{\mathcal{I}_p^{\mathcal{R}}}$ since $\mathcal{I}_p^{\mathcal{R}}$ is a model of $\mathcal{R}$. Moreover, since $x_{aP_1...P_l}^{i_1...i_l} = x_{aP_1P_2}^{i_1p} \in \Gamma_N$, by (P4), $\mathcal{T} \models \exists P_1^- \sqsubseteq \exists P_2$. Therefore, since $\mathcal{I}_p^{\mathcal{R}}$ is a model of $\mathcal{T}$, there exists $(x_{aP_1}^{\mathcal{I}_p^{\mathcal{R}}}, y) \in P_2^{\mathcal{I}_p^{\mathcal{R}}}$.

(iii) If $h_p^{\mathcal{R}}(x_{aP_1...P_{l-1}}^{i_1...i_{l-1}\mathcal{I}_p}) = x_{P_{l-2}P_{l-1}}^{\mathcal{I}_p^{\mathcal{R}}}$, by definition of $\mathcal{R}$, since $x_{P_{l-2}P_{l-1}}$ appears in $\mathcal{R}$, $P_{l-1}(x_{P_{l-2}}, x_{P_{l-2}P_{l-1}}) \in \mathcal{R}$, and therefore $(x_{P_{l-2}}^{\mathcal{I}_p^{\mathcal{R}}}, x_{P_{l-2}P_{l-1}}^{\mathcal{I}_p^{\mathcal{R}}}) \in P_{l-1}^{\mathcal{I}_p^{\mathcal{R}}}$. Since $x_{aP_1...P_l}^{i_1...i_l} \in \Gamma_N$, by (P4), $\mathcal{T} \models \exists P_{l-1}^- \sqsubseteq \exists P_l$, and there exists $(x_{P_{l-2}P_{l-1}}^{\mathcal{I}_p^{\mathcal{R}}}, y) \in P_l^{\mathcal{I}_p^{\mathcal{R}}}$.

(iv) If $(h_p^{\mathcal{R}}(x_{aP_1...P_{l-2}}^{i_1...i_{l-2}\mathcal{I}_p}), h_p^{\mathcal{R}}(x_{aP_1...P_{l-1}}^{i_1...i_{l-1}\mathcal{I}_p})) \in P_{l-1}^{\mathcal{I}_p^{\mathcal{R}}}$, since $x_{aP_1...P_l}^{i_1...i_l} \in \Gamma_N$, by (P4), $\mathcal{T} \models \exists P_{l-1}^- \sqsubseteq \exists P_l$, and there exists $(h_p^{\mathcal{R}}(x_{aP_1...P_{l-1}}^{i_1...i_{l-1}\mathcal{I}_p}), y) \in P_l^{\mathcal{I}_p^{\mathcal{R}}}$.

(v) If $h_p^{\mathcal{R}}(x_{aP_1...P_{l-1}}^{i_1...i_{l-1}\mathcal{I}_p}) = x_{P_{l-1}}^{\mathcal{I}_p^{\mathcal{R}}}$, by (P2) and since $i_l = p$, we obtain $P_l(x_{aP_1...P_{l-1}}^{i_1...i_{l-1}}, x_{aP_1...P_l}^{i_1...p}) \in chase_{\text{rig}}^{\mathcal{K}}(\mathcal{A}_p)$. By (P6), since $chase_{\text{rig}}^{\mathcal{K}}(\mathcal{A}_p) \models \exists P_l(x_{aP_1...P_{l-1}}^{i_1...i_{l-1}})$ and $i_{l-1} \neq p$, there exists $B := A|\exists R|\exists R^-$ with $A \in \mathsf{N_{RC}}, R \in \mathsf{N_{RR}}$, such that $\mathcal{T} \models B \sqsubseteq \exists P_l$ and $chase_{\text{rig}}^{\mathcal{K}}(\mathcal{A}_{i_{l-1}}) \models B(x_{aP_1...P_{l-1}}^{i_1...i_{l-1}})$. By (P3), $\mathcal{T} \models \exists P_{l-1}^- \sqsubseteq B$, and $\mathcal{R} \models B(x_{P_{l-1}})$ (since $x_{P_{l-1}}$ occurs in $\mathcal{R}$ and B is rigid). We obtain that $\langle \mathcal{T}, \mathcal{R} \rangle \models \exists x.P_l(x_{P_{l-1}}, x)$. Since $\mathcal{I}_p^{\mathcal{R}}$ is a model of $\langle \mathcal{T}, \mathcal{R} \rangle$, there exists $(x_{P_{l-1}}^{\mathcal{I}_p^{\mathcal{R}}}, y) \in P_l^{\mathcal{I}_p^{\mathcal{R}}}$.

– For *length* > 0, since $x_{aP_1...P_l}^{i_1...i_l} \in \Gamma_N$, by (P4), $\mathcal{T} \models \exists P_{l-1}^- \sqsubseteq \exists P_l$. Since by the inductive hypothesis, there exists $(x, h_p^{\mathcal{R}}(x_{aP_1...P_{l-1}}^{i_1...i_{l-1}\mathcal{I}_p})) \in P_{l-1}^{\mathcal{I}_p^{\mathcal{R}}}$, there then exists $(h_p^{\mathcal{R}}(x_{aP_1...P_{l-1}}^{i_1...i_{l-1}\mathcal{I}_p}), y) \in P_l^{\mathcal{I}_p^{\mathcal{R}}}$. ∎

Claim 2. $h_p^{\mathcal{R}}$ is a homomorphism of $\mathcal{I}_p$ into $\mathcal{I}_p^{\mathcal{R}}$.

Proof of claim For every $a \in \mathsf{N_I^{\mathcal{K}}}$ and concept A, if $a^{\mathcal{I}_p} \in A^{\mathcal{I}_p}$, i.e., $A(a) \in chase_{\text{rig}}^{\mathcal{K}}(\mathcal{A}_p)$, then by (P5), either (i) $\langle \mathcal{T}, \mathcal{A}_p \rangle \models A(a)$, and since $\mathcal{I}_p^{\mathcal{R}}$ is a model of $\langle \mathcal{T}, \mathcal{A}_p \rangle$, then $h_p^{\mathcal{R}}(a^{\mathcal{I}_p}) = a^{\mathcal{I}_p^{\mathcal{R}}} \in A^{\mathcal{I}_p^{\mathcal{R}}}$, or (ii) there exists $B := C|\exists R|\exists R^-$ with $C \in \mathsf{N_{RC}}, R \in \mathsf{N_{RR}}$, such that $\mathcal{T} \models B \sqsubseteq A$, and there exists j such that $\langle \mathcal{T}, \mathcal{A}_j \rangle \models B(a)$. In the latter case $\mathcal{R} \models B(a)$. Therefore, since $\mathcal{I}_p^{\mathcal{R}}$ is a model of $\mathcal{R}$, $\mathcal{I}_p^{\mathcal{R}} \models B(a)$, and $\mathcal{I}_p^{\mathcal{R}} \models A(a)$ because $\mathcal{I}_p^{\mathcal{R}}$ is a model of $\mathcal{T}$. It follows that $h_p^{\mathcal{R}}(a^{\mathcal{I}_p}) = a^{\mathcal{I}_p^{\mathcal{R}}} \in A^{\mathcal{I}_p^{\mathcal{R}}}$. For every pair $a, b \in \mathsf{N_I^{\mathcal{K}}}$ and role P, if $(a^{\mathcal{I}_p}, b^{\mathcal{I}_p}) \in P^{\mathcal{I}_p}$, by (P7), similar arguments can be used to prove that $(h_p^{\mathcal{R}}(a^{\mathcal{I}_p}), h_p^{\mathcal{R}}(b^{\mathcal{I}_p})) = (a^{\mathcal{I}_p^{\mathcal{R}}}, b^{\mathcal{I}_p^{\mathcal{R}}}) \in P^{\mathcal{I}_p^{\mathcal{R}}}$.

For every $x_{aP_1...P_l}^{i_1...i_l} \in \Gamma_N$, such that $x_{aP_1...P_l}^{i_1...i_l\mathcal{I}_p} \in A^{\mathcal{I}_p}$, by (P3), $\mathcal{T} \models \exists P_l^- \sqsubseteq A$, and by construction of $h_p^{\mathcal{R}}$, $h_p^{\mathcal{R}}(x_{aP_1...P_l}^{i_1...i_l}) = y$ is such that either (i) there exists $(x, y) \in P_l^{\mathcal{I}_p^{\mathcal{R}}}$, and since $\mathcal{I}_p^{\mathcal{R}}$ is a model of $\mathcal{T}$, we have $y \in A^{\mathcal{I}_p^{\mathcal{R}}}$, or (ii) $y = x_{P_l}^{\mathcal{I}_p^{\mathcal{R}}}$, P_l is not rigid, and for every $j \in [1, l]$, $i_j \neq p$. In the latter case, by (P6), there exists $B := C|\exists R|\exists R^-$ with $C \in \mathsf{N_{RC}}, R \in \mathsf{N_{RR}}$, such that $\mathcal{T} \models B \sqsubseteq A$ and $chase_{\text{rig}}^{\mathcal{K}}(\mathcal{A}_{i_l}) \models B(x_{aP_1...P_l}^{i_1...i_l})$. By (P3), $\mathcal{T} \models \exists P_l^- \sqsubseteq B$. Therefore, by construction of $\mathcal{R}$, $\mathcal{R} \models B(x_{P_l})$ and $\langle \mathcal{T}, \mathcal{R} \rangle \models A(x_{P_l})$. It follows that $y \in A^{\mathcal{I}_p^{\mathcal{R}}}$.

For every pair $x, y \in \Gamma_N$ and role P such that $(x^{\mathcal{I}_p}, y^{\mathcal{I}_p}) \in P^{\mathcal{I}_p}$, by (P10), either (i) $x = x_{aP_1...P_l}^{i_1...i_l}$, $y = x_{aP_1...P_lP_{l+1}}^{i_1...i_li_{l+1}}$ and $\mathcal{T} \models P_{l+1} \sqsubseteq P$, or (ii) $x = x_{aP_1...P_lP_{l+1}}^{i_1...i_li_{l+1}}$, $y = x_{aP_1...P_l}^{i_1...i_l}$ and $\mathcal{T} \models P_{l+1} \sqsubseteq P^-$. We

can assume w.l.o.g. that we are in the first case. Otherwise, we consider $(y^{\mathcal{I}_p}, x^{\mathcal{I}_p}) \in P^{-\mathcal{I}_p}$. If $i_{l+1} = p$, by definition of $h_p^{\mathcal{R}}$, we have $(h_p^{\mathcal{R}}(x^{\mathcal{I}_p}), h_p^{\mathcal{R}}(y^{\mathcal{I}_p})) \in P_{l+1}^{\mathcal{I}_p^{\mathcal{R}}}$. Otherwise, by (P11), there exists $P' := R|R^-$ with $R \in \mathsf{N_{RR}}$ such that $\mathcal{T} \models P_{l+1} \sqsubseteq P' \sqsubseteq P$ and $P'(x, y) \in chase_{\mathrm{rig}}^{\mathcal{K}}(\mathcal{A}_{i_{l+1}})$. In this case, there are two possibilities.

(i) If P_l is not rigid, given that $\mathcal{T} \models P_{l+1} \sqsubseteq P'$ and P' is rigid, P_{l+1} is rigid by our hypothesis on the TBox. It follows that $h_p^{\mathcal{R}}(y^{\mathcal{I}_p}) = x_{P_l P_{l+1}}^{\mathcal{I}_p^{\mathcal{R}}}$. If there is no $i_j = p$, then $h_p^{\mathcal{R}}(x^{\mathcal{I}_p}) = x_{P_l}^{\mathcal{I}_p^{\mathcal{R}}}$. Therefore, since $P_{l+1}(x_{P_1}, x_{P_l P_{l+1}}) \in \mathcal{R}$, then $(h_p^{\mathcal{R}}(x^{\mathcal{I}_p}), h_p^{\mathcal{R}}(y^{\mathcal{I}_p})) \in P_{l+1}^{\mathcal{I}_p^{\mathcal{R}}}$. Otherwise, there exists $i_j = p$, and we obtain $(h_p^{\mathcal{R}}(x^{\mathcal{I}_p}), h_p^{\mathcal{R}}(y^{\mathcal{I}_p})) \in P_{l+1}^{\mathcal{I}_p^{\mathcal{R}}}$ by definition of $h_p^{\mathcal{R}}$.

(ii) If P_l is rigid, then $h_p^{\mathcal{R}}(y^{\mathcal{I}_p})$ is such that $(h_p^{\mathcal{R}}(x^{\mathcal{I}_p^{\mathcal{R}}}), h_p^{\mathcal{R}}(y^{\mathcal{I}_p^{\mathcal{R}}})) \in P_{l+1}^{\mathcal{I}_p^{\mathcal{R}}}$.

Since in all cases, $(h_p^{\mathcal{R}}(x^{\mathcal{I}_p}), h_p^{\mathcal{R}}(y^{\mathcal{I}_p})) \in P_{l+1}^{\mathcal{I}_p^{\mathcal{R}}}$ and $\mathcal{I}_p^{\mathcal{R}}$ is a model of $\mathcal{T}$, we obtain that $P^{\mathcal{I}_p^{\mathcal{R}}}$ contains $(h_p^{\mathcal{R}}(x^{\mathcal{I}_p}), h_p^{\mathcal{R}}(y^{\mathcal{I}_p}))$.

Finally, if $a \in \mathsf{N_I^{\mathcal{K}}}$ and $x \in \Gamma_N$, we have that $(a^{\mathcal{I}_p}, x^{\mathcal{I}_p}) \in P^{\mathcal{I}_p}$ only if $x = x_{aP_1}^{i_1}$. If $i_1 = p$, by definition of $h_p^{\mathcal{R}}$, $(h_p^{\mathcal{R}}(a^{\mathcal{I}_p}), h_p^{\mathcal{R}}(x^{\mathcal{I}_p})) \in P_1^{\mathcal{I}_p^{\mathcal{R}}}$, and since by (P8), $\mathcal{T} \models P_1 \sqsubseteq P$, we obtain that $(h_p^{\mathcal{R}}(a^{\mathcal{I}_p}), h_p^{\mathcal{R}}(x^{\mathcal{I}_p})) \in P^{\mathcal{I}_p^{\mathcal{R}}}$. If $i_1 \neq p$, by (P9), there exists a rigid role P' such that $\mathcal{T} \models P_1 \sqsubseteq P' \sqsubseteq P$, and by our restriction on the TBox, P_1 must be rigid. By (P1) and (P8), there is some j such that $\langle \mathcal{T}, \mathcal{A}_j \rangle \models \exists x.P_1(a, x)$. Therefore, $P_1(a, x_{aP_1}) \in \mathcal{R}$ and $(h_p^{\mathcal{R}}(a^{\mathcal{I}_p}), h_p^{\mathcal{R}}(x^{\mathcal{I}_p})) = (a^{\mathcal{I}_p^{\mathcal{R}}}, x_{aP_1}^{\mathcal{I}_p^{\mathcal{R}}}) \in P_1^{\mathcal{I}_p^{\mathcal{R}}}$. We obtain that $(h_p^{\mathcal{R}}(a^{\mathcal{I}_p}), h_p^{\mathcal{R}}(x^{\mathcal{I}_p})) \in P^{\mathcal{I}_p^{\mathcal{R}}}$. ∎

It follows from Claim 2 that $h_p^{\mathcal{R}} \circ \pi$ is a homomorphism of $\exists \vec{y}.\psi(\vec{y})$ into $\mathcal{I}_p^{\mathcal{R}}$. Therefore, we have shown that $\mathcal{I}_p \models \exists \vec{y}.\psi(\vec{y})$ implies $\mathcal{K}_{\mathcal{R}}, p \models \exists \vec{y}.\psi(\vec{y})$. This means that $\mathcal{K}, p \models \exists \vec{y}.\psi(\vec{y})$ implies $\mathcal{K}_{\mathcal{R}}, p \models \exists \vec{y}.\psi(\vec{y})$. □