HAL
open science

Answering Counting Queries over DL-Lite Ontologies

Meghyn Bienvenu, Quentin Manière, Michaël Thomazo

To cite this version:

Meghyn Bienvenu, Quentin Manière, Michaël Thomazo. Answering Counting Queries over DL-Lite Ontologies. IJCAI 2020 - Twenty-Ninth International Joint Conference on Artificial Intelligence, Jul 2020, Yokohama, Japan. hal-02927913

HAL Id: hal-02927913

https://inria.hal.science/hal-02927913v1

Submitted on 2 Sep 2020

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.

Answering Counting Queries over DL-Lite Ontologies

Meghyn Bienvenu[1], **Quentin Manière**[1] and **Michaël Thomazo**[2]
[1]University of Bordeaux, CNRS, Bordeaux INP, LaBRI, Talence, France
[2]Inria, DI ENS, ENS, CNRS, University PSL, Paris, France
{meghyn.bienvenu, quentin.maniere}@u-bordeaux.fr, michael.thomazo@inria.fr

Abstract

Ontology-mediated query answering (OMQA) is a promising approach to data access and integration that has been actively studied in the knowledge representation and database communities for more than a decade. The vast majority of work on OMQA focuses on conjunctive queries, whereas more expressive queries that feature counting or other forms of aggregation remain largely unexplored. In this paper, we introduce a general form of counting query, relate it to previous proposals, and study the complexity of answering such queries in the presence of DL-Lite ontologies. As it follows from existing work that query answering is intractable and often of high complexity, we consider some practically relevant restrictions, for which we establish improved complexity bounds.

1 Introduction

Ontology-mediated query answering (OMQA) utilizes ontologies to provide a convenient vocabulary for query formulation and to capture domain knowledge that is exploited during the querying process to obtain more complete sets of answers [Poggi *et al.*, 2008; Bienvenu and Ortiz, 2015; Xiao *et al.*, 2018]. Much of the work on OMQA considers ontologies formulated using description logics (DLs), a family of knowledge representation languages that provide the logical foundations of the OWL web ontology language. Particular attention has been to the DL-Lite family of DLs [Calvanese *et al.*, 2007], which were developed with OMQA in mind and enjoy favorable computational properties.

The vast majority of work on OMQA supposes that user queries are given as conjunctive queries (CQs). However, there are many other kinds of database queries, beyond plain CQs, that are relevant in practice. This motivates research into the feasibility of adopting other database query languages for OMQA. While enriching CQs with either negated atoms or inequalities has been shown to lead to undecidability even in very restricted settings [Gutiérrez-Basulto *et al.*, 2015], the situation is more positive for navigational queries (like regular path queries), which can be adopted without losing decidability, sometimes even retaining tractable data complexity [Bienvenu *et al.*, 2015b].

Aggregate queries, which use numeric operators (e.g. count, sum, max) to summarize selected parts of a dataset, constitute another prominent class of database queries. Although such queries are widely used for data analysis, they have been little explored in context of OMQA. This may be partly due to the fact that it is not at all obvious how to define the semantics of such queries in the OMQA setting. A first exploration of aggregate queries in OMQA was conducted by Calvanese *et al.* (2008). They argued that the most straightforward adaptation of classical certain answer semantics to aggregate queries was unsatisfactory, as often values would differ from model to model, leading to no certain answers. For this reason, an epistemic semantics was proposed, in which variables involved in the aggregation are required to match to data constants. However, as discussed in [Kostylev and Reutter, 2015], this semantics can also give unintuitive results by ignoring ways of mapping aggregate variables to anonymous elements inferred due the ontology axioms. For instance, if no children of alex are listed in the data, then a query that asks to return the number of children will yield 0 under epistemic semantics, even if it can be inferred (e.g. due to a family tax benefit) that there must be at least 3 children. This led Kostylev and Reutter to define an alternative semantics for two kinds of counting queries (inspired by the COUNT and COUNT DISTINCT in SQL) which adopts a form of certain answer semantics but considers lower and upper bounds on the count value across different models. For the two considered logics (DL-Lite$_{\mathsf{core}}$ and DL-Lite$_{\mathcal{R}}$), only the lower bounds on the count value are non-trivial, and a complexity analysis shows that they are challenging to identify: coNP-data complexity for both logics, and Π_2^p-hard (resp. coNEXP-hard) in combined complexity for DL-Lite$_{\mathsf{core}}$ (resp. DL-Lite$_{\mathcal{R}}$). Several question were left unanswered by their work, including the exact combined complexity, the difficulty of recognizing the optimal lower bound, and the impact of allowing multiple aggregation variables.

This paper returns to the issue of handling counting queries in OMQA and makes several important contributions:

1. We propose a new notion of counting CQ that generalizes the two forms of queries from [Kostylev and Reutter, 2015] and allows arbitrarily many counting variables.
2. We show that existing complexity results for DL-Lite$_{\mathsf{core}}$ and DL-Lite$_{\mathcal{R}}$ KBs continue to hold for our more general notion of counting CQ, and provide an improved coNEXP

upper bound for the relevant case of finite-depth TBoxes.

3. We consider the impact of restricting the query structure, focusing on the class of rooted queries, in which every query variable must be connected to an answer variable or individual in the query graph. A recent result, obtained as part of a study of bag semantics for OMQA, identified a case in which rootedness leads to tractable data complexity for counting queries [Nikolaou *et al.*, 2019]. This motivates us to perform a more thorough investigation of rooted counting queries, which yields several improvements upon existing complexity bounds.
4. We prove that the problem of identifying the best certain interval is DP-complete in data complexity.

Our results close some questions that were left open by the work of Kostylev and Reutter and pave the way for further study of counting and aggregate queries in the OMQA setting.

An appendix with full proofs can be found in the long version of this paper, available on arXiv.

2 Preliminaries

We recall the basics of description logics (DLs), focusing on DL-Lite, see [Baader *et al.*, 2017] for more details.

Syntax and Semantics. A description logic vocabulary consists of a set $\mathsf{N_C}$ of *atomic concepts* (unary predicates), a set $\mathsf{N_R}$ of *atomic roles* (binary predicates), and a set $\mathsf{N_I}$ of *individual names* (constants). By *role*, we mean either an atomic role $P \in \mathsf{N_R}$ or an *inverse role* P^- (where $P \in \mathsf{N_R}$). We let $\mathsf{N_R^\pm}$ denote the set $\mathsf{N_R} \cup \{P^- \mid P \in \mathsf{N_R}\}$ of roles and use the notation R^- to mean P^- if $R = P \in \mathsf{N_R}$ and P if $R = P^-$.

A DL *knowledge base (KB)* is a pair $\mathcal{K} = (\mathcal{T}, \mathcal{A})$, consisting of an ABox $\mathcal{A}$ that contains facts about particular individuals and a TBox $\mathcal{T}$ that expresses general knowledge about the domain. Formally, an *ABox* is a finite set of *concept assertions* $A(b)$, with $A \in \mathsf{N_C}$ and $b \in \mathsf{N_I}$, and *role assertions* $P(a, b)$, with $P \in \mathsf{N_R}$ and $a, b \in \mathsf{N_I}$. We use $\mathsf{Ind}(\mathcal{A})$ to denote the set of individuals in $\mathcal{A}$. A *TBox* is a finite set of axioms, whose syntax depends on the particular DL. In DL-Lite$_{\mathsf{core}}$, axioms take the form of *concept inclusions* $B_1 \sqsubseteq (\neg)B_2$, where each B_i is either A (for $A \in \mathsf{N_C}$) or $\exists R$ (with $R \in \mathsf{N_R^\pm}$). DL-Lite$_\mathcal{R}$ TBoxes additionally allow *role inclusions* $R_1 \sqsubseteq (\neg)R_2$, where $R_1, R_2 \in \mathsf{N_R^\pm}$.

Example 1. *Our example KB talks about leading (*LeadIn*) and supporting actors (*SuppIn*) in movies:*

$$\mathcal{A}_{\mathsf{act}} = \{\mathrm{ActsIn}(\mathsf{doona}, \mathsf{cloud}), \mathrm{SuppIn}(\mathsf{berry}, \mathsf{cloud}), \mathrm{SuppIn}(\mathsf{hanks}, \mathsf{cloud}), \mathrm{SuppIn}(\mathsf{hanks}, \mathsf{catch})\}$$

$$\mathcal{T}_{\mathsf{act}} = \{\mathrm{LeadIn} \sqsubseteq \mathrm{ActsIn}, \mathrm{SuppIn} \sqsubseteq \mathrm{ActsIn}, \exists \mathrm{SuppIn}^- \sqsubseteq \exists \mathrm{LeadIn}^-\}$$

An interpretation takes the form $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$, where $\Delta^\mathcal{I}$ is a non-empty set (the *domain* of $\mathcal{I}$), and $\cdot^\mathcal{I}$ is a function that maps each $A \in \mathsf{N_C}$ to a subset $A^\mathcal{I} \subseteq \Delta^\mathcal{I}$, each $P \in \mathsf{N_R}$ to a binary relation $P^\mathcal{I} \subseteq \Delta^\mathcal{I} \times \Delta^\mathcal{I}$, and each $a \in \mathsf{N_I}$ to an element $a^\mathcal{I}$ of $\Delta^\mathcal{I}$. We make the *unique names assumption* (UNA) by requiring that $a^\mathcal{I} \neq b^\mathcal{I}$ for every $a, b \in \mathsf{N_I}$ with $a \neq b$. The function $\cdot^\mathcal{I}$ naturally extends to complex concepts and roles: $(\exists R)^\mathcal{I} = \{d \mid \exists d' : (d, d') \in R^\mathcal{I}\}$, $(P^-)^\mathcal{I} = \{(d_1, d_2) \mid (d_2, d_1) \in P^\mathcal{I}\}$, $(\neg B)^\mathcal{I} = \Delta^\mathcal{I} \setminus B^\mathcal{I}$, $(\neg R)^\mathcal{I} = \big(\Delta^\mathcal{I} \times \Delta^\mathcal{I}\big) \setminus R^\mathcal{I}$. A (concept or role) inclusion $F \sqsubseteq G$ is satisfied in $\mathcal{I}$ if $F^\mathcal{I} \subseteq G^\mathcal{I}$; assertion $A(b)$ is satisfied in $\mathcal{I}$ if $b^\mathcal{I} \in A^\mathcal{I}$; $P(a, b)$ is satisfied in $\mathcal{I}$ if $(a^\mathcal{I}, b^\mathcal{I}) \in P^\mathcal{I}$. We call $\mathcal{I}$ a *model* of $\mathcal{K}$, written $\mathcal{I} \models \mathcal{K}$, if it satisfies all inclusions and assertions in $\mathcal{K}$. A KB is *satisfiable* if has at least one model.

Queries. We recall that a *conjunctive query* (CQ) takes the form $\exists \boldsymbol{y}\, \psi(\boldsymbol{x}, \boldsymbol{y})$, where $\boldsymbol{x}$ and $\boldsymbol{y}$ are tuples of variables drawn from an infinite set of variables $\mathbf{V}$, and ψ is a conjunction of *atoms*, which can be either *concept atoms* $A(t_1)$ or *role atoms* $P(t_1, t_2)$, where $A \in \mathsf{N_C}$, $P \in \mathsf{N_R}$, and *terms* t_i are drawn from $\mathsf{N_I} \cup \boldsymbol{x} \cup \boldsymbol{y}$. Consider an interpretation $\mathcal{I}$ and CQ $q = \exists \boldsymbol{y}\, \psi(\boldsymbol{x}, \boldsymbol{y})$ with $|\boldsymbol{x}| = n$. A tuple $\boldsymbol{\alpha} \in \big(\Delta^\mathcal{I}\big)^n$ is an *answer to q in $\mathcal{I}$*, written $\mathcal{I} \models q(\boldsymbol{\alpha})$, if there is a *homomorphism of q into $\mathcal{I}$*, i.e., a function σ that maps the terms of q to elements of $\Delta^\mathcal{I}$ such that (i) $\sigma(a) = a^\mathcal{I}$ for $a \in \mathsf{N_I}$, (ii) $\sigma(t) \in A^\mathcal{I}$ for every atom $A(t)$ of q, and (iii) $(\sigma(t_1), \sigma(t_2)) \in P^\mathcal{I}$ for every atom $P(t_1, t_2)$ of q. A tuple $\boldsymbol{a} \in \mathsf{Ind}(\mathcal{A})^n$ is a *certain answer to q w.r.t. the KB $\mathcal{K}$* iff $\mathcal{I} \models q(\boldsymbol{a}^\mathcal{I})$ for every model $\mathcal{I}$ of $\mathcal{K}$.

Canonical Model. We recall the definition of the canonical model $\mathcal{C}_\mathcal{K}$ of a DL-Lite$_\mathcal{R}$ KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$. The domain of $\mathcal{C}_\mathcal{K}$ consists of $\mathsf{Ind}(\mathcal{A})$ and all words of the form $aR_1 \dots R_n$, with $a \in \mathsf{Ind}(\mathcal{A})$, $R_i \in \mathsf{N_R^\pm}$, and $n \geqslant 1$, such that:

- $\mathcal{K} \models \exists R_1(a)$ and there is no $R_1(a, b) \in \mathcal{A}$;
- for $1 \leq i < n$, $\mathcal{T} \models \exists R_i^- \sqsubseteq \exists R_{i+1}$ and $R_i^- \neq R_{i+1}$.

We interpret individuals as themselves ($a^{\mathcal{C}_\mathcal{K}} = a$) and atomic concepts and roles as follows:

$$\begin{aligned} A^{\mathcal{C}_\mathcal{K}} &= \{a \in \mathsf{Ind}(\mathcal{A}) \mid \mathcal{K} \models A(a)\} \\ &\quad \cup \{aR_1 \dots R_n \in \Delta^{\mathcal{C}_\mathcal{K}} \setminus \mathsf{Ind}(\mathcal{A}) \mid \mathcal{T} \models \exists R_n^- \sqsubseteq A\} \\ P^{\mathcal{C}_\mathcal{K}} &= \{(a, b) \mid P(a, b) \in \mathcal{A}\} \cup \\ &\quad \{(e_1, e_2) \mid e_2 = e_1 R \text{ and } \mathcal{T} \models R \sqsubseteq P\} \cup \\ &\quad \{(e_2, e_1) \mid e_2 = e_1 R \text{ and } \mathcal{T} \models R \sqsubseteq P^-\} \end{aligned}$$

The term 'canonical model' is motivated by the following well-known property of $\mathcal{C}_\mathcal{K}$ (see e.g. [Calvanese *et al.*, 2007]):

Lemma 1. *Let $\mathcal{K}$ be a satisfiable DL-Lite$_\mathcal{R}$ KB. Then $\mathcal{C}_\mathcal{K} \models \mathcal{K}$ and if $\mathcal{I} \models \mathcal{K}$, there is a homomorphism of $\mathcal{C}_\mathcal{K}$ into $\mathcal{I}$.*

A useful corollary is that the *certain answers to a CQ q w.r.t. $\mathcal{K}$* are the tuples from $\mathsf{Ind}(\mathcal{A})$ that are *answers to q in $\mathcal{C}_\mathcal{K}$*.

Note that $\mathcal{C}_\mathcal{K}$ may be infinite. The *depth* of a TBox $\mathcal{T}$ is defined as the maximal length of any word that appears in the domain of $\mathcal{C}_\mathcal{K}$ for any KB $\mathcal{K}$ whose TBox is $\mathcal{T}$. If this number is finite, we say that $\mathcal{T}$ is a *finite-depth TBox*; such TBoxes can be identified in polynomial time [Bienvenu *et al.*, 2015a].

3 Counting Queries

We now introduce our formalization of counting queries. In addition to the set $\mathbf{V}$ of (classical) variables, we assume a second infinite set of counting variables $\mathbf{V_c}$, disjoint from $\mathbf{V}$.

Definition 1. *A counting conjunctive query (CCQ) q takes the form $q(\boldsymbol{x}) = \exists \boldsymbol{y} \exists \boldsymbol{z}\, \psi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$, where $\boldsymbol{x} \cup \boldsymbol{y} \subseteq \mathbf{V}$, $\boldsymbol{z} \subseteq \mathbf{V_c}$, and ψ is a conjunction of concept and role atoms whose terms are drawn from $\mathsf{N_I} \cup \boldsymbol{x} \cup \boldsymbol{y} \cup \boldsymbol{z}$. We call $\boldsymbol{x}$ (resp. $\boldsymbol{y}$, resp. $\boldsymbol{z}$) the* answer *(resp.* existential*, resp.* counting*) variables of q.*

We first define the semantics of counting queries on a single interpretation $\mathcal{I}$, by considering those pairs $(\boldsymbol{a}, n)$ such that n is the number of possible ways to map $\boldsymbol{z}$ into $\mathcal{I}$ when $\boldsymbol{x}$ is mapped to $\boldsymbol{a}$. Such pairs are called the *answers* to q in $\mathcal{I}$.

Definition 2. *A* match *of a CCQ $q(\boldsymbol{x}) = \exists \boldsymbol{y} \exists \boldsymbol{z}\ \psi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$ in $\mathcal{I}$ is a homomorphism*[1] *from q into $\mathcal{I}$. If a match σ maps $\boldsymbol{x}$ to $\boldsymbol{a}$, then the restriction of σ to $\boldsymbol{z}$ is called a* counting match (c-match) *of $q(\boldsymbol{a})$ in $\mathcal{I}$. The set of* answers to q in $\mathcal{I}$, *denoted $q^{\mathcal{I}}$, contains all pairs $(\boldsymbol{a}, q^{\mathcal{I}}_{\boldsymbol{a}})$, where $q^{\mathcal{I}}_{\boldsymbol{a}}$ is the number of distinct c-matches of $q(\boldsymbol{a})$ in $\mathcal{I}$.*

As has been previously noted (see e.g. [Kostylev and Reutter, 2015]), the exact count values of the answers in $q^{\mathcal{I}}$ are usually too specific to hold across models. Considering *bounds* on the exact value provides more insight, while still allowing unnamed elements to be counted. This motivates the following notion of answer interval.

Definition 3. *The set $[q]^{\mathcal{I}}$ of* answer intervals for a CCQ q in $\mathcal{I}$ *contains all pairs $(\boldsymbol{a}, [m, M])$ with $\boldsymbol{a} \in \mathsf{Ind}^{|\boldsymbol{x}|}$ and m, M integers such that $m \leq q^{\mathcal{I}}_{\boldsymbol{a}} \leq M$. The set $[q]^{\mathcal{K}}$ of* certain (counting) answers to q w.r.t. KB $\mathcal{K}$ *is obtained by considering those answer intervals that hold in all models of $\mathcal{K}$: $[q]^{\mathcal{K}} = \bigcap_{\mathcal{I} \models \mathcal{K}} [q]^{\mathcal{I}}$.*

Note that $(\boldsymbol{a}, [m, M]) \in [q]^{\mathcal{K}}$ does not imply that for any $n \in [m, M]$ there exists a model $\mathcal{I}$ in which $(\boldsymbol{a}, n) \in q^{\mathcal{I}}$.

Definition 1 is a proper generalization of the two forms of counting query considered by Kostylev and Reutter. Reusing their notations, a $Cntd()$-query $q(\boldsymbol{x}, Cntd(z)) = \exists \boldsymbol{y}\ \psi(\boldsymbol{x}, \boldsymbol{y}, z)$ corresponds to the CCQ $q(\boldsymbol{x}) = \psi(\boldsymbol{x}, \boldsymbol{y}, z)$, while a $Count()$-query $q(\boldsymbol{x}, Count()) = \exists \boldsymbol{y}\ \psi(\boldsymbol{x}, \boldsymbol{y})$ corresponds to the CCQ $q(\boldsymbol{x}) = \psi(\boldsymbol{x}, \emptyset, \hat{\boldsymbol{y}})$ (with $\hat{\boldsymbol{y}}$ a tuple of variables from $\mathbf{V_c}$ in bijection with $\boldsymbol{y}$). We will use the term *exhaustive* to refer to the latter CCQs, i.e. those in which every non-answer variable is a counting variable.

Example 2. *Reconsider the KB $\mathcal{K}_{\mathsf{act}} = (\mathcal{T}_{\mathsf{act}}, \mathcal{A}_{\mathsf{act}})$. We can use CCQs to count the pairs of actors (leading role, supporting role) having acted together (q_1), return movies together with a count of their supporting actors (q_2), and count the number of actors having acted with Tom Hanks (q_3):*

$$q_1 = \exists y \exists z_1 \exists z_2\ \mathrm{LeadIn}(z_1, y) \wedge \mathrm{SuppIn}(z_2, y)$$
$$q_2(x) = \exists z\ \mathrm{SuppIn}(z, x)$$
$$q_3 = \exists y \exists z\ \mathrm{ActsIn}(\mathsf{hanks}, y) \wedge \mathrm{ActsIn}(z, y)$$

According to our semantics, we have:

- $(\emptyset, [2, +\infty]) \in [q_1]^{\mathcal{K}_{\mathsf{act}}}$, *since z_2 can be mapped to either* berry *or* hanks, *and z_1 mapped to the lead actor (which must exist due to $\mathcal{T}_{\mathsf{act}}$). As the lead actors of the two films could be the same, $(\emptyset, [3, +\infty]) \notin [q_1]^{\mathcal{K}_{\mathsf{act}}}$.*
- $(\mathsf{cloud}, [2, +\infty]) \in [q_2]^{\mathcal{K}_{\mathsf{act}}}$, *mapping z to* berry *and* hanks.
- $(\emptyset, 5) \in q_3^{\mathcal{C}_{\mathcal{K}_{\mathsf{act}}}}$, *since in $\mathcal{C}_{\mathcal{K}_{\mathsf{act}}}$, we can map z to a named actor or the two elements standing in for the lead actors.*
- $(\emptyset, [5, +\infty]) \notin [q_3]^{\mathcal{K}_{\mathsf{act}}}$, *since the lead actors could possibly be the same or one of the named actors.*

The latter two points show that the canonical model does not yield the minimal number of matches.

[1]The notion of homomorphism of a CCQ is defined in the same way as for CQs, simply treating variables from $\mathbf{V_c}$ like those in $\mathbf{V}$.

	Data	**Combined**
DL-Lite$_{\mathsf{core}}$	coNP-**c**	Π^p_2-**h**, PP-**h** & in coNEXP
DL-Lite$_{\mathcal{R}}$	coNP-**c**	coNEXP-**h** & in coN2EXP coNEXP-**c** ($\mathcal{T}$ of finite depth)

Table 1: Data and combined complexity of CCQ answering

4 General Counting CQs

We shall consider the following CCQ answering decision problem: given a KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$, CCQ q, and candidate answer $(\boldsymbol{a}, [m, M])$, decide whether $(\boldsymbol{a}, [m, M]) \in [q]^{\mathcal{K}}$.

As ontology languages, we will consider DL-Lite$_{\mathcal{R}}$ (which underlies OWL 2 QL) and its sublogic DL-Lite$_{\mathsf{core}}$. We know from [Kostylev and Reutter, 2015] that in these DLs, the least upper bound M can take one of three values (0, 1, or $+\infty$) and is easily computed. The argument[2] transfers to our more general notion of CCQ. We can therefore *restrict our attention to identifying certain answers of the form* $(\boldsymbol{a}, [m, +\infty])$.

We will consider the two usual complexity measures: *combined complexity* which is in terms of the size of the whole input ($\mathcal{T}$, $\mathcal{A}$, q, $\boldsymbol{a}$, m), and *data complexity* which is only in terms of the size of $\mathcal{A}$ and m ($\mathcal{T}$ and q are treated as fixed). We will assume that m is given in binary.

4.1 General Case

Table 1 displays complexity results for answering general CCQs over DL-Lite$_{\mathsf{core}}$ and DL-Lite$_{\mathcal{R}}$ TBoxes (we use '**-c**' and '**-h**' as abbreviations for '-complete' and '-hard').

With the exception of the PP-hardness result (discussed in Section 6.1), the lower bounds are inherited from [Kostylev and Reutter, 2015]. We will thus concentrate on the upper bounds from Table 1, which are obtained by generalizing and clarifying the constructions of Kostylev and Reutter. We give an overview of the proof both to give the flavor of the techniques involved and to enable us to discuss the necessary adaptations used to prove later results.

The proof constructs a decision procedure for the complementary problem of deciding whether $(\boldsymbol{a}, [m, +\infty]) \notin [q]^{\mathcal{K}}$. The latter holds iff there exists a *countermodel*, i.e., a model of $\mathcal{K}$ with fewer than m c-matches of $q(\boldsymbol{a})$. The main ingredient of the proof is the following theorem, which shows that it is sufficient to consider countermodels of bounded size.

Theorem 1. *For every DL-Lite$_{\mathcal{R}}$ (resp. DL-Lite$_{\mathsf{core}}$) KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ and CCQ q, if there is a model of $\mathcal{K}$ with fewer than m c-matches of $q(\boldsymbol{a})$, then there exists one of size*[3] *$O(|\mathcal{A}|^{|\mathcal{T}|^{|q|+1}})$ (resp. $O(|\mathcal{A}|^{|q|})$).*

With Theorem 1 in hand, we can easily define nondeterministic procedures that witness the complexity upper bounds from Table 1: simply guess an interpretation of polynomial / exponential / double-exponential size (depending on the case) and verify whether it is a countermodel.

The proof of Theorem 1 starts with an arbitrary countermodel $\mathcal{I}$ and modifies it in order to make it smaller, being

[2]Briefly, the upper bound is 0 if the tuple is not a certain answer; otherwise, it is either 1 if $\boldsymbol{z} = \emptyset$, else $+\infty$.

[3]As usual, $|\mathcal{T}|$ (resp. $|\mathcal{A}|$, $|q|$) denotes the size of $\mathcal{T}$ (resp. $\mathcal{A}$, q).

careful not to introduce any new c-matches of $q(\boldsymbol{a})$. We first identify a relevant subset Δ^* of the domain of $\mathcal{I}$, consisting of the interpretations of all individual names from $\mathcal{A}$ and the images of all c-matches of $q(\boldsymbol{a})$. We then define a new interpretation that intuitively preserves Δ^* and replaces the rest of $\mathcal{I}$ with parts of the canonical model, to introduce a more regular structure. Formally, we fix a homomorphism f of $\mathcal{C}_\mathcal{K}$ into $\mathcal{I}$ (see Lemma 1) and consider the following mapping $f' : \Delta^{\mathcal{C}_\mathcal{K}} \to \Delta^* \cup \Delta^{\mathcal{C}_\mathcal{K}}$ from [Kostylev and Reutter, 2015]:

$$f'(d) \quad = \quad \begin{cases} f(d) & \text{if } f(d) \in \Delta^* \\ d & \text{otherwise} \end{cases}$$

We define the *interleaving*[4] $\mathcal{I}'$ of $\mathcal{I}$ as the image of $\mathcal{C}_\mathcal{K}$ by f', i.e., with domain $f'(\Delta^{\mathcal{C}_\mathcal{K}})$ and interpretation function $f' \circ \cdot^{\mathcal{C}_\mathcal{K}}$.

It is not difficult to prove that the interleaving $\mathcal{I}'$ is a model of $\mathcal{K}$. Moreover, by exhibiting a homomorphism ρ from $\mathcal{I}'$ to $\mathcal{I}$, we can translate matches of $\mathcal{I}'$ into matches in $\mathcal{I}$. As the images of c-matches of $q(\boldsymbol{a})$ are contained in Δ^*, which is left unchanged in $\mathcal{I}'$, the homomorphism ρ is in fact a one-to-one mapping of c-matches of $q(\boldsymbol{a})$ in $\mathcal{I}'$ to those in $\mathcal{I}$. This shows that $\mathcal{I}'$ is also a countermodel.

The interleaving $\mathcal{I}'$ may be arbitrarily large, even infinite. To reduce its size, an equivalence relation is introduced, and elements from $\Delta^{\mathcal{I}'} \setminus \Delta^*$ that belong to the same equivalence class are merged (elements from Δ^* are retained). In the case of DL-Lite$_\mathcal{R}$, there can be double-exponentially many equivalence classes, as elements are grouped based upon the properties of their $|q|$-neighborhoods, while for DL-Lite$_{\mathsf{core}}$, we can use a more refined relation with only exponentially many classes. This means that the resulting models are either of single- or double-exponential size w.r.t. combined complexity, depending on the chosen DL.

A crucial final step is to show that the merging of elements does not introduce any new c-matches of $q(\boldsymbol{a})$, so the resulting model is still a countermodel. This part of the argument, only sketched in [Kostylev and Reutter, 2015], requires a detailed and technical analysis of the construction to ensure that this property holds for our more general class of CCQs. We show that this is indeed the case, which answers a question left open by Kostylev and Reutter about counting CQs with both existential variables and multiple counting variables.

4.2 Case of Finite-Depth TBoxes

We give an improved upper bound for finite-depth TBoxes (which arguably cover many practical ontologies [Grau *et al.*, 2013]), pinpointing the exact combined complexity.

Theorem 2. *For finite-depth DL-Lite$_\mathcal{R}$ TBoxes, CCQ answering is* coNEXP*-complete w.r.t. combined complexity.*

Proof sketch. Fix a KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$. If $\mathcal{T}$ has finite depth, then $\mathcal{C}_\mathcal{K}$ contains at most $|\mathsf{Ind}(\mathcal{A})| \times |\mathcal{T}|^{|\mathcal{T}|}$ elements, which implies that, for every model $\mathcal{I}$ of $\mathcal{K}$, the interleaving of $\mathcal{I}$ is finite and of single exponential size in $|\mathcal{K}|$. Since the interleaving of a countermodel is itself a countermodel, this shows that the smallest countermodel is of single-exponential size, from which derives the improved coNEXP upper bound. □

[4]We have slightly modified the definition of interleaving to correct a small bug in the definition from [Kostylev and Reutter, 2015].

We note that the proofs of the coNP and Π_2^p lower bounds listed in Table 1 already use finite-depth TBoxes.

5 Rooted Counting CQs

We next explore whether structural restrictions on CCQs allow us to obtain lower complexity. As the lower bounds from [Kostylev and Reutter, 2015] use disconnected counting variables, a natural idea is to consider the subclass of *rooted* queries that were introduced in [Bienvenu *et al.*, 2012] and are believed to capture a large portion of real-world CQs.

Rooted CCQs can be defined analogously to rooted CQs. The definition utilizes the notion of a Gaifman graph of a CCQ, whose vertices are the query terms, and which has an undirected edge $\{t_1, t_2\}$ iff t_1, t_2 co-occur in a role atom.

Definition 4. *A CCQ $q(\boldsymbol{x}) := \exists \boldsymbol{y} \exists \boldsymbol{z} \psi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$ is* rooted *if every connected component of the Gaifman graph of q contains at least one answer variable or individual name.*

Example queries q_2 and q_3 are rooted, while q_1 is not.

Rootedness has been shown to lower the complexity of reasoning in several settings. Most relevant to us is a recent result by Nikolaou *et al.* (2019) that rooted CQ answering under bag semantics[5] has tractable data complexity in DL-Lite$_{\mathsf{core}}$, and furthermore, the same holds for rooted versions of the $Count()$-queries of Kostylev and Reutter under suitable restrictions on the TBox. These techniques can be adapted to show tractability for arbitrary DL-Lite$_{\mathsf{core}}$TBoxes:

Theorem 3. ***(Implicit in [Nikolaou* et al.*, 2019; Cima* et al.*, 2019])*** *In DL-Lite$_{\mathsf{core}}$, exhaustive rooted CCQ answering is* TC^0*-complete*[6] *w.r.t. data complexity.*

Proof sketch. Nikolaou *et al.* prove that answering rooted CQs under bag semantics can be done via a rewriting to BCALC, whose evaluation problem is known to be in TC^0 due to [Libkin, 2001], see [Cima *et al.*, 2019] for discussion. Moreover, they further show that for a syntactically restricted class of DL-Lite$_{\mathsf{core}}$ TBoxes, it is possible to reduce exhaustive rooted CCQ answering to rooted CQ answering under bag semantics. To obtain TC^0 membership for unrestricted TBoxes, the BCALC rewriting can be adapted to set-based rather than bag interpretations. In the long version, we provide an alternative self-contained proof which directly constructs a family of TC^0 circuits. A matching lower bound has not been stated, but can be shown by a simple reduction (using an empty TBox) from the TC^0-complete problem that asks, given a binary string s and number k, whether the number of 1-bits in s exceeds k [Aehlig *et al.*, 2007]. □

The preceding result naturally leads us to ask whether rootedness also bring benefits for general CCQs. Unfortunately, we show that restricting to rooted CCQs (without exhaustiveness) does not allow us to escape existing hardness results:

[5]Bag semantics, which underly practical database systems, interprets relations using multisets rather than sets [Albert, 1991].

[6]We recall that TC^0 is a circuit complexity class defined similarly to AC^0 but additionally allowing threshhold gates. It is known that $\mathsf{AC}^0 \subsetneq \mathsf{TC}^0 \subseteq \mathsf{NC}^1 \subseteq \mathsf{LogSpace} \subseteq \mathsf{PTime}$.

Theorem 4. *In DL-Lite$_{\mathsf{core}}$, rooted CCQ answering is* coNP*-complete w.r.t. data complexity.*

Proof sketch. The proof borrows some ideas from the proofs of Lemmas 12 and 16 from [Kostylev and Reutter, 2015]. It proceeds by reduction from the well-known coNP-complete 3COL problem: given an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, return yes iff $\mathcal{G}$ has *no* 3-coloring, i.e., a mapping from $\mathcal{V}$ to $\{\mathsf{red}, \mathsf{green}, \mathsf{blue}\}$ such that adjacent vertices map to different colors (equivalently: there is no monochromatic edge).

The reduction uses atomic roles Edge and Vertex to encode the graph and HasCol to assign colors. The TBox $\mathcal{T}_{\mathsf{col}}$ has a single axiom: $\exists \mathrm{Vertex}^- \sqsubseteq \exists \mathrm{HasCol}$. The ABox $\mathcal{A}_{\mathcal{G}}$ contains an individual v for each vertex $v \in \mathcal{V}$ and an assertion $\mathrm{Edge}(\mathsf{u}, \mathsf{v})$ for each edge $\{u, v\} \in \mathcal{E}$. All vertices are connected to a special root individual a: $\mathrm{Vertex}(\mathsf{a}, \mathsf{u})$, for each $u \in \mathcal{V}$. The three colors are represented by individuals r, g and b. To ensure that the query has matches in every model, we include a 'dummy' vertex individual $\mathsf{a_v}$ and the following assertions: $\mathrm{Vertex}(\mathsf{a}, \mathsf{a_v})$, $\mathrm{Edge}(\mathsf{a_v}, \mathsf{a_v})$, $\mathrm{HasCol}(\mathsf{a_v}, \mathsf{r})$, $\mathrm{HasCol}(\mathsf{a_v}, \mathsf{g})$, and $\mathrm{HasCol}(\mathsf{a_v}, \mathsf{b})$.

The query q is the conjunction of the two subqueries:

$$q^{edge} = \exists y_c\ \exists z_1\ \exists z_2\ \mathrm{Vertex}(\mathsf{a}, z_1) \wedge \mathrm{Vertex}(\mathsf{a}, z_2) \wedge \mathrm{Edge}(z_1, z_2) \wedge \mathrm{HasCol}(z_1, y_c) \wedge \mathrm{HasCol}(z_2, y_c)$$

$$q^{col} = \exists y\ \exists z\ \mathrm{Vertex}(\mathsf{a}, y) \wedge \mathrm{HasCol}(y, z)$$

serving respectively to detect monochromatic edges and to check whether any additional colors have been introduced.

By construction, there are at least 3 c-matches for $q(\emptyset)$ in any model of the KB $\mathcal{K}_{\mathsf{col}} = (\mathcal{T}_{\mathsf{col}}, \mathcal{A}_{\mathcal{G}})$. Moreover, it can be verified that $(\emptyset, [4, +\infty])$ is a certain answer to q w.r.t. $\mathcal{K}_{\mathsf{col}}$ iff $\mathcal{G}$ is not 3-colorable. □

Theorem 5. *In DL-Lite$_{\mathcal{R}}$, rooted CCQ answering is* coNEXP*-hard w.r.t. combined complexity.*

Proof sketch. The proof adapts a reduction from the exponential grid tiling problem (Lemma 18 from [Kostylev and Reutter, 2015]), the key difference being the use of existential query variables to access (and count) the colors and bits. □

6 Exhaustive Rooted Counting CQs

We have seen in Section 5 that the rootedness restriction is not by itself sufficient to lower the complexity of CCQ answering, whereas imposing both rootedness and exhaustiveness can sometimes yield better results. This motivates us to take a closer look at the case of exhaustive rooted CCQs. The emerging complexity landscape is summarized in Table 2.

Note that exhaustive CCQs constitute a very natural form of counting query, which ask for the number of different query matches for a given answer tuple. The query q_2 from Example 2 is an exhaustive rooted CCQ.

6.1 Exhaustive Rooted CCQs in DL-Lite$_{\mathsf{core}}$

We first consider DL-Lite$_{\mathsf{core}}$ KBs and pinpoint the precise combined complexity, which had not yet been considered.

An essential ingredient is the following result that shows that it is possible to focus on query matches in the canonical model. It can be obtained by adapting a similar result about canonical bag interpretations [Nikolaou *et al.*, 2019].

	Data	**Combined**
DL-Lite$_{\mathsf{core}}$	TC^0**-c**	PP**-c**
DL-Lite$_{\mathcal{R}}$	coNP**-c**	Π_2^p**-h**, PP**-h** & in coNEXP

Table 2: Complexity results for exhaustive rooted CCQs

Theorem 6. *For every DL-Lite$_{\mathsf{core}}$ KB $\mathcal{K}$ and exhaustive rooted CCQ q, it holds that $[q]^{\mathcal{K}} = [q]^{\mathcal{C}_{\mathcal{K}}}$.*

Proof sketch. Exploiting the structure of DL-Lite$_{\mathsf{core}}$ canonical models, one can show that if σ_1, σ_2 are distinct matches of an exhaustive rooted CCQ q in $\mathcal{C}_{\mathcal{K}}$, then there exists a variable v such that $\sigma_1(v) \neq \sigma_2(v)$ and $\sigma_1(v), \sigma_2(v) \in \mathsf{Ind}(\mathcal{A})$. It follows that if we take an arbitrary model $\mathcal{I}$ of $\mathcal{K}$, and let f be a homomorphism of $\mathcal{C}_{\mathcal{K}}$ into $\mathcal{I}$, then f injectively maps query matches in $\mathcal{C}_{\mathcal{K}}$ to query matches in $\mathcal{I}$. □

We will also use the next lemma, implicit in [Bienvenu *et al.*, 2013], constraining the possible images of matches in $\mathcal{C}_{\mathcal{K}}$:

Lemma 2. *For every DL-Lite$_{\mathsf{core}}$ TBox $\mathcal{T}$ and CCQ q, we can construct in polynomial time a set of words $\Gamma_{q,\mathcal{T}}$ such that for every KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$, match σ of q in $\mathcal{C}_{\mathcal{K}}$, and variable v of q: $\sigma(v) = aw$ for some $a \in \mathsf{Ind}(\mathcal{A})$ and $w \in \Gamma_{q,\mathcal{T}}$.*

We are now ready to show that the problem is PP-complete in combined complexity, and hence in PSpace.

Theorem 7. *In DL-Lite$_{\mathsf{core}}$, exhaustive rooted CCQ answering is* PP*-complete w.r.t. combined complexity.*

Proof sketch. The class PP contains all decision problems for which there exists a non-deterministic Turing machine (TM) such that, when the input is a 'yes' instance, then at least half of the computation paths accept, while on 'no' instances, less than half of the computation paths accept.

The lower bound is obtained by a reduction from the following PP-complete problem [Bailey *et al.*, 2007]: given a propositional formula ψ in CNF and number n, decide whether ψ has at least n satisfying assignments.

We sketch the TM used to show PP membership, which takes as input a DL-Lite$_{\mathsf{core}}$ KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$, an exhaustive rooted CCQ $q(\boldsymbol{x})$, and candidate answer $(\boldsymbol{a}, [m, +\infty])$:

Phase 1. The TM constructs the set $\Gamma_{q,\mathcal{T}}$ from Lemma 2.

Phase 2. The TM guesses a mapping σ of the variables in q to elements from $\{aw \mid a \in \mathsf{Ind}(\mathcal{A}), w \in \Gamma_{q,\mathcal{T}}\}$. It then compares m with the number $C = |\Gamma_{q,\mathcal{T}}|^{|q|}$ of possible mappings and proceeds accordingly:

- if $m \geq \frac{C}{2} + 1$, the TM guesses an integer i with $0 \leq i \leq 2m - 3$ and accepts iff σ is a c-match of $q(\boldsymbol{a})$ and $i < C$;
- if $m < \frac{C}{2} + 1$, the TM guesses an integer i with $0 \leq i \leq 2C - 2m + 1$ and accepts iff σ is c-match for $q(\boldsymbol{a})$ or $i < C - 2m + 2$.

The guessed integer and comparisons ensure a suitable number of accepting paths. It can be verified that at least half of the paths are accepting iff $(\boldsymbol{a}, [m, +\infty]) \in [q]^{\mathcal{C}_{\mathcal{K}}}$. □

6.2 Exhaustive Rooted CCQs in DL-Lite$_\mathcal{R}$

We now turn to DL-Lite$_\mathcal{R}$ KBs. Our first result is negative: exhaustive rooted CCQs do not enjoy lower data complexity. This is shown by another reduction from 3COL which involves ideas from our proof of Theorem 4 and the proof of Lemma 16 from [Kostylev and Reutter, 2015].

Theorem 8. *In DL-Lite$_\mathcal{R}$, exhaustive rooted CCQ answering is* coNP*-complete w.r.t. data complexity.*

More positively, we can show an improved coNEXP upper bound in combined complexity for exhaustive rooted CCQs. We briefly sketch the proof, which involves highly non-trivial modifications to the argument used for general CCQs.

We first introduce a more refined notion of interleaving, which replaces the mapping f' by the following mapping f^*:

$$\begin{array}{rcl} f^*(a) & = & f(a) \\ f^*(\omega R) & = & \begin{cases} f(\omega R) & \text{if } f^*(\omega), f(\omega R) \in \Delta^* \\ f^*(\omega)R & \text{otherwise} \end{cases} \end{array}$$

It is possible to prove that when q is an exhaustive rooted CCQ, this modified interleaving yields a countermodel. Moreover, it has a very particular structure, essentially corresponding to the canonical model of the restriction of $f(\mathcal{C}_\mathcal{K})$ to Δ^* (viewed as an ABox). Importantly, this means that instead of guessing a whole countermodel, it suffices to guess an initial, exponential-size portion (the $|q|$-neighborhood of Δ^*), providing the basis for a coNEXP decision procedure.

Theorem 9. *In DL-Lite$_\mathcal{R}$, exhaustive rooted CCQ answering is in* coNEXP *w.r.t. combined complexity.*

7 Best Certain Answers

The definition of certain answers implies that if $(\boldsymbol{a}, [m, M]) \in [q]^\mathcal{K}$, then we also have $(\boldsymbol{a}, [m', M']) \in [q]^\mathcal{K}$ for every $m' \leq m$ and $M' \geq M$. It is naturally of interest to focus on certain answers providing the best bounds, i.e., those of the form $(\boldsymbol{a}, [\min_{\mathcal{I} \models \mathcal{K}} q^\mathcal{I}_{\boldsymbol{a}}, \max_{\mathcal{I} \models \mathcal{K}} q^\mathcal{I}_{\boldsymbol{a}}])$.

In this section, we show that the problem of identifying the best lower bound $(\min_{\mathcal{I} \models \mathcal{K}} q^\mathcal{I}_{\boldsymbol{a}})$ is DP-complete in data complexity. It is easily seen that checking whether m is such an optimal bound can be done in DP, by making a call to a coNP oracle (is $(\boldsymbol{a}, [m, +\infty]) \in [q]^\mathcal{K}$?) and an NP oracle (is $(\boldsymbol{a}, [m+1, +\infty]) \notin [q]^\mathcal{K}$?). The DP-hardness of this problem was left as an open question by Kostylev and Reutter.

Theorem 10. *The following problem is* DP*-hard in data complexity: given a DL-Lite$_{\mathsf{core}}$ KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$, rooted CCQ q, tuple $\boldsymbol{a}$, and number m, decide whether $m = \min_{\mathcal{I} \models \mathcal{K}} q^\mathcal{I}_{\boldsymbol{a}}$.*

Proof sketch. We give a reduction from the following problem (DP-complete due to [Garey *et al.*, 1976]): given *planar* graphs $\mathcal{G}_1$ and $\mathcal{G}_2$, decide if $\mathcal{G}_1 \in$ 3COL and $\mathcal{G}_2 \notin$ 3COL.

Let the TBox $\mathcal{T}_{\mathsf{col}}$ and ABoxes $\mathcal{A}_{\mathcal{G}_1}, \mathcal{A}_{\mathcal{G}_2}$ be defined as in the proof of Theorem 4. Rename the individuals to ensure $\mathsf{Ind}(\mathcal{A}_{\mathcal{G}_1}) \cap \mathsf{Ind}(\mathcal{A}_{\mathcal{G}_2}) = \emptyset$, then set $\mathcal{K} = (\mathcal{T}_{\mathsf{col}}, \mathcal{A}_{\mathcal{G}_1} \cup \mathcal{A}_{\mathcal{G}_2})$. Let q_1^{color} and q_1^{edge} (resp. q_2^{color} and q_2^{edge}) be defined as before, but using disjoint variables and the root individual from the $\mathcal{A}_{\mathcal{G}_1}$ (resp. $\mathcal{A}_{\mathcal{G}_2}$). The challenge is to make sure that we can determine the 3-colorability status of the two graphs solely by looking at the number of c-matches of the query. To be able to distinguish $\mathcal{G}_1$ from $\mathcal{G}_2$, we introduce an asymmetry by duplicating the color counter query for $\mathcal{G}_1$, i.e., create a copy q_0^{color} of q_1^{color} that uses fresh variables but the same root individual. We then take the query

$$q() := q_0^{color} \ \wedge \ q_1^{color} \ \wedge \ q_1^{edge} \ \wedge \ q_2^{color} \ \wedge \ q_2^{edge}.$$

We claim $(\boldsymbol{a}_\emptyset, [36, +\infty]) \in [q]^\mathcal{K}$ iff $\mathcal{G}_1 \in$ 3COL and $\mathcal{G}_2 \notin$ 3COL. This is proven by a case analysis, summarized here:

	$\mathcal{G}_1 \in$ 3COL	$\mathcal{G}_1 \notin$ 3COL
$\mathcal{G}_2 \in$ 3COL	27 $(= 3 \times 3 \times 3)$	48 $(= 4 \times 4 \times 3)$
$\mathcal{G}_2 \notin$ 3COL	36 $(= 3 \times 3 \times 4)$	64 $(= 4 \times 4 \times 4)$

Each of the four cells displays the least value of m such that $(\boldsymbol{a}_\emptyset, [m, +\infty]) \in [q]^\mathcal{K}$, under different assumptions on the 3-colorability of $\mathcal{G}_1$ and $\mathcal{G}_2$. To establish these values, one must first prove that every model has at least this many c-matches, and then exhibit a model that realizes the exact number. For the latter, we utilize our assumption that the graphs are planar, hence 4-colorable [Gonthier, 2008], which we use to show that the minimal number of c-matches is realized in a model that encodes proper 3- or 4-colorings of the graphs. □

The preceding reduction can be adapted to show DP-hardness also for the two kinds of CCQs from [Kostylev and Reutter, 2015], but without the rootedness restriction.

8 Conclusion & Future Work

We have revisited the issue of counting queries in OMQA and advanced our understanding of the complexity landscape, both by extending existing results to a more general notion of counting CQ and by exploring when structural restrictions on the ontology and query can lead to improved complexity.

There are several natural avenues for future study. A first challenging problem is to provide a full classification of the data complexity of ontology-mediated queries (i.e. query-ontology pairs), in order to identify further tractable cases. It would also be relevant to extend the complexity study to DLs with functional roles or quantified number restrictions, which would allow for non-trivial upper bounds on the number of matches. Tackling general CCQs for such DLs will likely require wholly different techniques from the model manipulations used in Section 4. However, a recent result by Cima *et al.* (2019) shows that the canonical model property (Theorem 6) holds also for DL-Lite$_\mathcal{F}$ (which extends DL-Lite$_{\mathsf{core}}$ with functional roles), and hence both TC^0 data complexity (Theorem 3) and our PP-completeness result (Theorem 7) for exhaustive rooted CCQs transfer to DL-Lite$_\mathcal{F}$.

Much remains to be explored for queries involving other kinds of aggregate functions (min, max, sum, average), which manipulate data values. Recent studies of bag semantics for OMQA [Nikolaou *et al.*, 2019; Cima *et al.*, 2019] and databases with incomplete information [Hernich and Kolaitis, 2017; Console *et al.*, 2017] provide important formal foundations for supporting such queries.

Acknowledgements

This work was partially supported by ANR project CQFD (ANR-18-CE23-0003).

References

[Aehlig *et al.*, 2007] Klaus Aehlig, Stephen Cook, and Phuong Nguyen. *Relativizing Small Complexity Classes and Their Theories*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007.

[Albert, 1991] Joseph Albert. Algebraic properties of bag data types. In *Proceedings of the 17th International Conference on Very Large Data Bases (VLDB)*, pages 211–219, 1991.

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

[Bailey *et al.*, 2007] Delbert D. Bailey, Víctor Dalmau, and Phokion G. Kolaitis. Phase transitions of PP-complete satisfiability problems. *Discrete Applied Mathematics*, 155(12):1627 – 1639, 2007.

[Bienvenu and Ortiz, 2015] Meghyn Bienvenu and Magdalena Ortiz. Ontology-mediated query answering with data-tractable description logics. In *Tutorial Lectures of the 11th Reasoning Web International Summer School*, pages 218–307, 2015.

[Bienvenu *et al.*, 2012] Meghyn Bienvenu, Carsten Lutz, and Frank Wolter. Query containment in description logics reconsidered. In *Proceedings of the 13th International Conference on the Principles of Knowledge Representation and Reasoning (KR)*, 2012.

[Bienvenu *et al.*, 2013] Meghyn Bienvenu, Magdalena Ortiz, Mantas Simkus, and Guohui Xiao. Tractable queries for lightweight description logics. In *Proceedings of the 23rd International Joint Conference on Artificial Intelligence (IJCAI)*, pages 768–774, 2013.

[Bienvenu *et al.*, 2015a] Meghyn Bienvenu, Stanislav Kikot, and Vladimir V. Podolskii. Tree-like queries in OWL 2 QL: Succinctness and complexity results. In *Proceedings of the 30th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pages 317–328, 2015.

[Bienvenu *et al.*, 2015b] Meghyn Bienvenu, Magdalena Ortiz, and Mantas Simkus. Regular path queries in lightweight description logics: Complexity and algorithms. *Journal of Artificial Intelligence Research (JAIR)*, 53:315–374, 2015.

[Calvanese *et al.*, 2007] Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati. Tractable reasoning and efficient query answering in description logics: The DL-Lite family. *Journal of Automated Reasoning (JAR)*, 39(3):385–429, 2007.

[Calvanese *et al.*, 2008] Diego Calvanese, Evgeny Kharlamov, Werner Nutt, and Camilo Thorne. Aggregate queries over ontologies. In *Proceedings of the 2nd International Workshop on Ontologies and Information Systems for the Semantic Web (ONISW)*, pages 97–104, 2008.

[Cima *et al.*, 2019] Gianluca Cima, Charalampos Nikolaou, Egor V. Kostylev, Mark Kaminski, Bernardo Cuenca Grau, and Ian Horrocks. Bag semantics of dl-lite with functionality axioms. In *Proceedings of the 18th International Semantic Web Conference (ISWC)*, pages 128–144, 2019.

[Console *et al.*, 2017] Marco Console, Paolo Guagliardo, and Leonid Libkin. On querying incomplete information in databases under bag semantics. In Carles Sierra, editor, *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 993–999, 2017.

[Garey *et al.*, 1976] M.R. Garey, D.S. Johnson, and L. Stockmeyer. Some simplified NP-complete graph problems. *Theoretical Computer Science*, 1(3):237 – 267, 1976.

[Gonthier, 2008] Georges Gonthier. Formal proof – The four-color theorem. *Notices of the American Mathematical Society*, 55(11):1382–1393, 2008.

[Grau *et al.*, 2013] Bernardo Cuenca Grau, Ian Horrocks, Markus Krötzsch, Clemens Kupke, Despoina Magka, Boris Motik, and Zhe Wang. Acyclicity notions for existential rules and their application to query answering in ontologies. *Journal of Artificial Intelligence Research (JAIR)*, 47:741–808, 2013.

[Gutiérrez-Basulto *et al.*, 2015] Víctor Gutiérrez-Basulto, Yazmin Angélica Ibáñez-García, Roman Kontchakov, and Egor V. Kostylev. Queries with negation and inequalities over lightweight ontologies. *Journal of Web Semantics (JWS)*, 35:184–202, 2015.

[Hernich and Kolaitis, 2017] André Hernich and Phokion G. Kolaitis. Foundations of information integration under bag semantics. In *Proceedings of the 32nd Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pages 1–12, 2017.

[Kostylev and Reutter, 2015] Egor V. Kostylev and Juan L. Reutter. Complexity of answering counting aggregate queries over DL-Lite. *Journal of Web Semantics (JWS)*, 33(1):94–111, 2015.

[Libkin, 2001] Leonid Libkin. Expressive power of SQL. In *Proceedings of the 8th International Conference on Database Theory (ICDT)*, pages 1–21, 2001.

[Nikolaou *et al.*, 2019] Charalampos Nikolaou, Egor V. Kostylev, George Konstantinidis, Mark Kaminski, Bernardo Cuenca Grau, and Ian Horrocks. Foundations of ontology-based data access under bag semantics. *Artificial Intelligence (AIJ)*, 274:91 – 132, 2019.

[Poggi *et al.*, 2008] Antonella Poggi, Domenico Lembo, Diego Calvanese, Giuseppe De Giacomo, Maurizio Lenzerini, and Riccardo Rosati. Linking data to ontologies. *Journal of Data Semantics*, 10:133–173, 2008.

[Xiao *et al.*, 2018] Guohui Xiao, Diego Calvanese, Roman Kontchakov, Domenico Lembo, Antonella Poggi, Riccardo Rosati, and Michael Zakharyaschev. Ontology-based data access: A survey. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 5511–5519, 2018.

A Proofs for Section 4 (General Counting CQs)

Theorem 1. *For every DL-Lite$_\mathcal{R}$ (resp. DL-Lite$_{\mathsf{core}}$) KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ and CCQ q, if there is a model of $\mathcal{K}$ with fewer than m c-matches of $q(\boldsymbol{a})$, then there exists one of size*[7] $O(|\mathcal{A}|^{|\mathcal{T}|^{|q|+1}})$ *(resp. $O(|\mathcal{A}|^{|q|})$).*

We recall the construction of the interleaving and some of its basic properties. We start with an arbitrary countermodel $\mathcal{I}$, and consider the subdomain Δ^* consisting of individual names from $\mathcal{A}$ and the images of all c-matches of $q(\boldsymbol{a})$:

$$\Delta^* = \{a^\mathcal{I} \mid a \in \mathsf{Ind}(\mathcal{A})\} \cup \bigcup_{\sigma \text{ match for } q(\boldsymbol{a}) \text{ in } \mathcal{I}} \sigma(\boldsymbol{z}).$$

We fix a homomorphism f of $\mathcal{C}_\mathcal{K}$ into $\mathcal{I}$ (see Lemma 1) and consider the following mapping from [Kostylev and Reutter, 2015]:

$$\begin{array}{rcl} f' : \Delta^{\mathcal{C}_\mathcal{K}} & \to & \Delta^* \cup \Delta^{\mathcal{C}_\mathcal{K}} \\ d & \mapsto & \begin{cases} f(d) & \text{if } f(d) \in \Delta^* \\ d & \text{otherwise} \end{cases} \end{array}$$

The *interleaving*[8] $\mathcal{I}'$ of $\mathcal{I}$ is defined as the image of $\mathcal{C}_\mathcal{K}$ by f'. More precisely, $\mathcal{I}'$ has domain $f'(\Delta^{\mathcal{C}_\mathcal{K}}) = \{f'(d) \mid d \in \Delta^{\mathcal{C}_\mathcal{K}}\}$ and interpretation function $f' \circ \cdot^{\mathcal{C}_\mathcal{K}}$, i.e., $A^{\mathcal{I}'} = \{f'(d) \mid d \in A^{\mathcal{C}_\mathcal{K}}\}$ and $R^{\mathcal{I}'} = \{(f'(d_1), f'(d_2)) \mid (d_1, d_2) \in R^{\mathcal{C}_\mathcal{K}}\}$.

It will be useful to exhibit a homomorphism from the interleaving into the original model, which will embed matches.

Lemma 3. *The following mapping is a homomorphism from $\mathcal{I}'$ to $\mathcal{I}$.*

$$\begin{array}{rcl} \rho : \Delta^{\mathcal{I}'} & \to & \Delta^\mathcal{I} \\ f'(d) & \mapsto & f(d). \end{array}$$

Proof. We first check that the definition is well-founded, that is: $\rho(f'(d))$ does not depend on the choice of d. To this end, consider d_1, d_2 such that $f'(d_1) = f'(d_2)$. Since f' maps to $\Delta^* \cup \Delta^{\mathcal{C}_\mathcal{K}}$, we have two cases to examine:

- if $f'(d_1) = f'(d_2) \in \Delta^*$, that means $f'(d_1) = f(d_1)$ and $f'(d_2) = f(d_2)$, thus ensuring $f(d_1) = f(d_2)$.
- if $f'(d_1) = f'(d_2) \in \Delta^{\mathcal{C}_\mathcal{K}}$, that means $f'(d_1) = d_1$ and $f'(d_2) = d_2$, thus ensuring again $f(d_1) = f(d_2)$.

In both cases, we obtain $f(d_1) = f(d_2)$, so the function is well-founded.

To show that ρ is a homomorphism of $\mathcal{I}'$ into $\mathcal{I}$, we use the definition of f' and fact that f is a homomorphism of $\mathcal{C}_\mathcal{K}$ to $\mathcal{I}$. Suppose first that $f'(d) \in A^\mathcal{I}$. Then $d \in A^{\mathcal{C}_\mathcal{K}}$, and since f is a homomorphism, $\rho(f'(d)) = f(d) \in A^\mathcal{I}$. Suppose next that $(f'(d_1), f'(d_2)) \in R^\mathcal{I}$. Then there exist $(d'_1, d'_2) \in R^{\mathcal{C}_\mathcal{K}}$ such that $f'(d_1) = f'(d'_1)$ and $f'(d_2) = f'(d'_2)$. Using again the fact that f is a homomorphism, we obtain $(f(d'_1), f(d'_2)) \in R^{\mathcal{I}'}$. By the preceding paragraph, this means $(f(d_1), f(d_2)) \in R^{\mathcal{I}'}$. □

Lemma 4. *The interleaving $\mathcal{I}'$ is a model of $\mathcal{K}$.*

Proof. We check that all axioms and assertions from the KB are satisfied.

- All ABox assertions from $\mathcal{A}$ are satisfied in $\mathcal{C}_\mathcal{K}$. Since $f' = f$ when restricted to the ABox individuals, and f is a homomorphism of $\mathcal{C}_\mathcal{K}$ in $\mathcal{I}$, it follows that these ABox assertions must also be satisfied in $\mathcal{I}$.
- Since $\mathcal{I}'$ maps homomorphically into $\mathcal{I}$ (by Lemma 3), any violation of an axiom from the TBox in $\mathcal{I}'$ implies a similar violation in $\mathcal{I}$. Since $\mathcal{I}$ is a model of $\mathcal{K}$, this cannot occur. □

Theorem 11. *The interleaving $\mathcal{I}'$ is a countermodel, and every c-match σ in $\mathcal{I}'$ satisfies that $\sigma(\boldsymbol{z}) \subseteq \Delta^*$.*

Proof. Assume we have a c-match in the interleaving $\sigma : \boldsymbol{z} \to \Delta^{\mathcal{I}'}$, which has an associated match $\overline{\sigma}$ for $q(\boldsymbol{a})$. Since ρ is a homomorphism, $\rho \circ \overline{\sigma}$ is a match for $q(\boldsymbol{a})$ in the original model $\mathcal{I}$, and its restriction to $\boldsymbol{z}$, that is, $\rho \circ \sigma$, is a c-match in $\mathcal{I}$. Hence, it follows from the definition of Δ^* that $(\rho \circ \sigma)(\boldsymbol{z}) \subseteq \Delta^*$. As $\rho^{-1}(\Delta^*) = \Delta^*$, this implies $\sigma(\boldsymbol{z}) \subseteq \Delta^*$.

Moreover, since $\rho_{|\Delta^*} = id$, we in fact have $\rho \circ \sigma = \sigma$. We have thus shown that every c-match for $q(\boldsymbol{a})$ in $\mathcal{I}'$ is also a c-match for $q(\boldsymbol{a})$ in $\mathcal{I}$, which means the number of c-matches in $\mathcal{I}'$ cannot exceed the number of c-matches. As $\mathcal{I}$ was assumed to be a countermodel (i.e. having less than m c-matches), it follows that the same holds for $\mathcal{I}'$. □

In general, the interleaving has an unbounded size. To reduce the size, we will merge some domain elements, while paying attention not to introduce any new matches. To decide which elements can be merged, we will look at their local properties, which will be formalized using the following notions of chains and neighbourhoods (as in [Kostylev and Reutter, 2015]).

Definition 5 (k-chains with respect to a subdomain). *A k-chain in a model $\mathcal{M}$ with respect to a subdomain $\mathcal{D} \subseteq \Delta^\mathcal{M}$ is a sequence $(d_0, \dots d_k)$ with d_i in $\Delta^\mathcal{M}$, such that for all $0 \leq i < k$, we have (i) $d_i \notin \mathcal{D}$, and (ii) there exists a positive role R_i such that $(d_i, d_{i+1}) \in \mathrm{R}_i^\mathcal{M}$. Note that the final element d_k might belong to $\mathcal{D}$.*

[7] As usual, $|\mathcal{T}|$ (resp. $|\mathcal{A}|$, $|q|$) denotes the size of $\mathcal{T}$ (resp. $\mathcal{A}$, q).

[8] We have slightly modified the definition of interleaving to correct a small bug in the definition from [Kostylev and Reutter, 2015].

Definition 6 (n-neighbourhood with respect to a subdomain). *Consider a model $\mathcal{M}$ and an element $d \in \Delta^{\mathcal{M}}$. Its n-neighbourhood $\mathcal{N}_n(d, \mathcal{M}, \mathcal{D})$ w.r.t. a subdomain $\mathcal{D}$ is the set of elements $d' \in \Delta^{\mathcal{M}}$ such that there exists a k-chain $(d_0, \dots d_k)$ in $\mathcal{M}$ with respect to $\mathcal{D}$ such that $k \leq n$, $d_0 = d$, and $d_k = d'$.*

Recall that the definition of $\Delta^{\mathcal{I}'}$ ensures that any $d \in \Delta^{\mathcal{I}'} \setminus \Delta^*$ is actually an element of $\Delta^{\mathcal{C}_\mathcal{K}}$ and therefore we have $d = \mathsf{a}w$ for some individual name a and word w. The tree-shaped structure of $\Delta^{\mathcal{C}_\mathcal{K}}$ ensures that there exists a unique prefix $r_{n,d}$ of $\mathsf{a}w$ such that :

- $f'(r_{n,d}) \in \mathcal{N}_n(d, \mathcal{I}', \Delta^*)$;
- for any $d' \in \mathcal{N}_n(d, \mathcal{I}', \Delta^*)$, there exists a unique word $w^{d'}_{n,d}$ such that $d' = f'(r_{n,d}w^{d'}_{n,d})$.

We denote by Ω_n the set of words over the alphabet of role names occuring in the TBox $\mathcal{T}$ and with length less or equal to $2n$. Local properties around d are then captured by the following function.

$$\begin{array}{rcl} \chi_{n,d} : \Omega_n & \to & \Delta^* \cup \{\emptyset\} \\ w & \mapsto & \begin{cases} f'(r_{n,d}w) & \text{if } r_{n,d}w \in \Delta^{\mathcal{C}_\mathcal{K}} \text{ and } f'(r_{n,d}w) \in \Delta^* \\ \emptyset & \text{otherwise} \end{cases} \end{array}.$$

The next definition groups together elements having the same local properties.

Definition 7 (Equivalent elements in the interleaving). *The equivalence relation $\sim_n$ on $\Delta^{\mathcal{I}'}$ is defined as follows:*

- *for $d \in \Delta^{\mathcal{I}'} \setminus \Delta^*$, we have $d \sim_n e$ iff $w^d_{n,d} = w^e_{n,e}$, $\chi_{n,d} = \chi_{n,e}$, and $|d| = |e| \mod 2|q| + 3$,*
- *for $d \in \Delta^*$, $d \sim_n e$ iff $d = e$.*

Remark 1. *Notice that if $d \sim_n e$, then $d \sim_m e$ for any $m \leq n$. This property will be used several times without mention.*

We can now define a smaller countermodel for our CCQ q by merging elements with respect to $\sim_{|q|+1}$. We will use $\overline{d}$ for the equivalence class of d w.r.t. $\sim_{|q|+1}$, and we denote by π the canonical projection, which maps elements to their respective equivalence classes:

$$\begin{array}{rcl} \pi : \Delta^{\mathcal{I}'} & \to & \Delta^{\mathcal{I}'}/\sim_{|q|+1} \\ d & \mapsto & \overline{d} \end{array}$$

Definition 8 (Reduced interleaving). *The* reduced interleaving *$\mathcal{J}$ is the interpretation with domain $\Delta^{\mathcal{I}'}/\sim_{|q|+1}$ and interpretation of individual names, atomic concepts and roles given by $\cdot^{\mathcal{J}} := \pi \circ \cdot^{\mathcal{I}'}$.*

Once again, it follows from the definition that $\pi : \mathcal{I}' \to \mathcal{J}$ is a homomorphism and that $\mathcal{J}$ is a model of $\mathcal{K}$. Since we are considering a quotient, we will not be able to build a general homomorphism from $\mathcal{J}$ to $\mathcal{I}'$ as in Lemma 3. However, local solutions are possible. To improve the readability of the following theorem and later material, we introduce the notation $\overline{\Delta^*}$ for the set $\{\overline{\sigma} \mid \sigma \in \Delta^*\}$.

Theorem 12. *For any $d \in \Delta^{\mathcal{I}'}$, there exists a homomorphism $\rho_d : \mathcal{N}_{|q|}(\overline{d}, \mathcal{J}, \overline{\Delta^*}) \to \mathcal{N}_{|q|}(d, \mathcal{I}', \Delta^*)$ satisfying that :*

1. *if $\overline{e} \in \overline{\Delta^*}$, then $\rho_d(\overline{e}) = e$;*
2. *$\rho_d^{-1}(\Delta^*) = \overline{\Delta^*}$.*

Let us first explain how this will conclude our proof, through the following consequence.

Corollary 1. *If $\mathcal{I}$ is a countermodel, then its reduced interleaving $\mathcal{J}$ is a countermodel.*

Proof. Assume we have a match σ in $\mathcal{J}$. Consider a minimal covering of $\sigma(\boldsymbol{x} \cup \boldsymbol{y} \cup \boldsymbol{z})$ by $|q|$-neighbourhoods in $\mathcal{J}$: that is a family of neighbourhoods $(\mathcal{N}_{|q|}(\overline{d_1}, \mathcal{J}), \dots \mathcal{N}_{|q|}(\overline{d_l}, \mathcal{J}))$, with l being minimal and such that $\sigma(\boldsymbol{x} \cup \boldsymbol{y} \cup \boldsymbol{z}) \subseteq \bigcup_{k=1}^{l} \mathcal{N}_{|q|}(\overline{d_k}, \mathcal{J})$. In particular, the minimality ensures that the only possible overlapping elements between two different neighbourhoods are elements of $\overline{\Delta^*}$. Along with condition 1 from Theorem 12, this ensures that the following mapping is well defined:

$$\begin{array}{rcll} \sigma' : \boldsymbol{x} \cup \boldsymbol{y} \cup \boldsymbol{z} & \to & \mathcal{I}' & \\ v & \mapsto & \rho_{d_k}(\sigma(v)) & \text{if } \sigma(v) \in \mathcal{N}_{|q|}(\overline{d_k}, \mathcal{J}) \end{array}$$

Furthermore, as the mappings ρ_{d_k} provided by Theorem 12 are homomorphisms, it follows that σ' is a match in $\mathcal{I}'$. Hence, by Theorem 11, we have $\sigma'(\boldsymbol{z}) \subseteq \Delta^*$. However, condition 2 from Theorem 12 ensures $(\rho_{d_k})^{-1}(\Delta^*) = \overline{\Delta^*}$, hence $\sigma(\boldsymbol{z}) \subseteq \overline{\Delta^*}$. Therefore, for each $z \in \boldsymbol{z}$, we have $\sigma(z) = \{e_z\}$ with $e_z \in \Delta^*$. From condition 1, it follows that $\sigma'(z) = e_z$. In particular, the mapping $\sigma_{|\boldsymbol{z}} \mapsto \sigma'_{|\boldsymbol{z}}$ is injective, so there are at most as many counting matches in $\mathcal{J}$ than in $\mathcal{I}'$. Hence, if $\mathcal{I}'$ is a countermodel, then $\mathcal{J}$ also is.

Recalling Theorem 11, we obtain that if $\mathcal{I}$ is a countermodel, then $\mathcal{J}$ also is. $\square$

To conclude the proof of Theorem 1, notice that an equivalence class $\overline{d}$ is fully characterized by :

- $|d| \mod 2|q|+3$, that is one equivalent class among $2|q|+3$ possible classes,
- $w^d_{|q|+1,d}$, that is a word over an alphabet with at most $|\mathcal{T}|$ symbols and a length at most $|q|+1$,
- $\chi_{|q|+1,d}$, that is a function from words over an alphabet with at most $|\mathcal{T}|$ symbols and length at most $2(|q|+1)$, to a set with size at most $|\Delta^*|+1$.

Therefore, the amount of possibly different equivalence classes, that is $|\Delta^{\mathcal{J}}|$, is at most:

$$(2|q|+3) \times |\mathcal{T}|^{|q|+2} \times (|\Delta^*|+1)^{|\mathcal{T}|^{2|q|+3}}.$$

Since $|\Delta^*| \leq |\mathsf{Ind}| + n_0|q|$ (recall n_0 is the amount of c-matches in $\mathcal{I}$, and that we can assume $n_0 \leq (|\mathsf{Ind}|+|\mathcal{T}|)^{|q|}$), we have the claimed bounds for the size of $\mathcal{J}$, which proves Theorem 1.

Coming back to the proof of Theorem 12, we start by building the mappings ρ_d. To do so, we need to transform k-chains in the reduced interleaving into k-chains in the interleaving.

Definition 9. *(primary role, core of a k-chain) Given a couple (d_1, d_2) of elements in $\Delta^{\mathcal{I}'} \setminus \Delta^*$ (resp $(\overline{d_1}, \overline{d_2})$ in $\Delta^{\mathcal{J}} \setminus \overline{\Delta^*}$), if there exists a positive role R connecting these two elements, then we call the* primary role of the edge (d_1, d_2) *the role SN_{R} such that either $d_2 = d_1\mathrm{S}$ or $d_1 = d_2\mathrm{S}^-$ (resp $w^{d_2}_{|q|+1,d_2} = w^{d_1}_{|q|,d_1}\mathrm{S}$ or $w^{d_1}_{|q|+1,d_1} = w^{d_2}_{|q|,d_2}\mathrm{S}^-$). Notice that $\mathcal{T} \models \mathrm{S} \sqsubseteq \mathrm{R}$.*

The action of a role R on a word w, denoted $\mathrm{R} \diamond w$, is either w' if $w = w'\mathrm{R}^-$, or $w\mathrm{R}$ otherwise.

Given a k-chain $C = (d_0, \dots d_k)$ in $\mathcal{I}'$ (resp in $\mathcal{J}$), we consider its core*, denoted $\widetilde{C}$, being the k-sequence of role names such that $\widetilde{C}_i$ is the primary role of (d_{i-1}, d_i).*

Given the core $\widetilde{C}$ of a k-chain C, we define its action on a word w, denoted $\widetilde{C} \diamond w$, by $\widetilde{C}_k \diamond \cdots \diamond \widetilde{C}_1 \diamond w$.

Remark 2. *For any couple (d_1, d_2) of elements in $\Delta^{\mathcal{I}'} \setminus \Delta^*$, the role S is the primary role of the edge (d_1, d_2) (in $\mathcal{I}'$) iff S is the primary role of the edge $(\overline{d_1}, \overline{d_2})$ (in $\mathcal{J}$).*

Lemma 5. *For any k-chain in $\mathcal{J}$ from $\overline{d}$ to $\overline{e}$, we have $w^e_{|q|+1-k,e} = w^{\widetilde{C}\diamond d}_{|q|+1-k,\widetilde{C}\diamond d}$.*

Proof. We proceed by a straightforward induction on the length of C.

- If C is a 0-chain, that is $\overline{e} = \overline{d}$, then $\widetilde{C}$ is empty and thus $\widetilde{C} \diamond d = d$.
- Otherwise, C is a $(k+1)$-chain $(\overline{d}, \overline{d_1}, \dots \overline{d_k}, \overline{e})$, then consider the primary role S of the edge $(\overline{d_k}, \overline{e})$. If $w^{(\widetilde{C}_1 \dots C_k)\diamond d}_{|q|+1-k,(\widetilde{C}_1 \dots \widetilde{C_k})\diamond d} = w'\mathrm{S}^-$, then we have :

$$\begin{aligned} w^{\widetilde{C}\diamond d}_{|q|+1-(k+1),\widetilde{C}\diamond d} &= \mathrm{S} \diamond w^{(\widetilde{C}_1,\dots\widetilde{C}_k)\diamond d_k}_{(q+1-k,(\widetilde{C}_1,\dots\widetilde{C}_k)\diamond d_k} \\ &= \mathrm{S} \diamond w^{d_k}_{|q|+1-k,d_k} \\ &= w^e_{|q|+1-(k+1),e}. \end{aligned}$$

  Otherwise, we have :

$$\begin{aligned} w^{\widetilde{C}\diamond d}_{|q|+1-(k+1),\widetilde{C}\diamond d} &= \mathrm{S} \diamond w^{(\widetilde{C}_1,\dots\widetilde{C}_k)\diamond d_k}_{(q+1-(k+2),(\widetilde{C}_1,\dots\widetilde{C}_k)\diamond d_k} \\ &= \mathrm{S} \diamond w^{d_k}_{|q|+1-(k+2),d_k} \\ &= w^e_{|q|+1-(k+1),e}. \end{aligned}$$

□

Lemma 6. *For any $k \leq |q|$, any $d, e \in \Delta^{\mathcal{I}'} \setminus \Delta^*$, and any two k-chains C and C' from $\overline{d}$ to $\overline{e}$ in $\mathcal{J}$, we have $\widetilde{C} \diamond d = \widetilde{C'} \diamond d$.*

Proof. Since C and C' have the same endpoints $\overline{d}$ and $\overline{e}$, we have $\widetilde{C} \diamond w^d_{|q|+1,d} = \widetilde{C'} \diamond w^d_{|q|+1,d}$.

If $|w^d_{|q|+1,d}| = |q|+1$, then $\widetilde{C} \diamond d = \widetilde{C} \diamond (r_{|q+1|,d} w^d_{|q|+1,d}) = r_{|q+1|,d}(\widetilde{C} \diamond w^d_{|q|+1,d}) = r_{|q+1|,d}(\widetilde{C'} \diamond w^d_{|q|+1,d}) = \widetilde{C'} \diamond d$.

Otherwise, $|w^d_{|q|+1,d}| < |q|+1$, we have $r_{|q|+1,d} \in \Delta^*$. Therefore the action of $\widetilde{C}$ (resp $\widetilde{C'}$) cannot empty the initial word $w^d_{|q|+1,d}$, since it would lead, by Lemma 6, to an element along C (resp C') being in $\overline{\Delta^*}$. Thus, we still have $\widetilde{C} \diamond d = \widetilde{C'} \diamond d$. □

Intuitively, the latter proof means that $\widetilde{C}$ equals $\widetilde{C'}$, up to deleting "dummy" steps in both chains, that are subsequences with shape $\mathrm{S}_1 \dots \mathrm{S}_p \mathrm{S}_p^- \dots \mathrm{S}_1^-$. But since the action of such dummy steps on any word is the identity, then the action of C and C' on d are equal. Notice that in general, these dummy steps are necessary to go from $\overline{d}$ to $\overline{e}$, hence we cannot get rid of them by asking for some sort of minimality about chains.

This allows us to define an image for elements in a neighbourhood in the reduced interleaving regardless of the k-chain used to reach this element.

Lemma 7. *The following mapping is well defined :*

$$\rho_d : \mathcal{N}_{|q|}(\overline{d}, \mathcal{J}) \to \mathcal{N}_{|q|}(d, \mathcal{I}')$$
$$\overline{e} \mapsto \begin{cases} e & \text{if } \overline{e} \in \overline{\Delta^*} \\ \widetilde{C} \diamond d \text{ with } C \text{ any } k\text{-chain from } \overline{d} \text{ to } \overline{e}, \text{ with } k \leq |q| & \text{otherwise} \end{cases}$$

Furthermore, ρ_d satisfies that for any $\overline{e} \in \Delta^{\mathcal{J}}$, we have $\rho_d(\overline{e}) \sim_1 e$. In particular, it satisfies conditions 1 and 2 from Theorem 12.

Proof. It only remains to prove that the action of a k-chain always provide an actual element in $\Delta^{\mathcal{I}'}$. We proceed by induction on k, building intermediate mappings $\rho_{d,k} : \mathcal{N}_k(\overline{d}, \mathcal{J}) \to \mathcal{N}_k(d, \mathcal{I}')$. We also prove that, at each step, we have for any $\overline{e} \in \mathcal{N}_k(\overline{d}, \mathcal{J}), \rho_{d,k}(\overline{e}) \sim_{|q|+1-k} e$.

Base case: $k = 0$. We have $\mathcal{N}_0(\overline{d}, \mathcal{J}) = \{\overline{d}\}$. If $\overline{d} \in \overline{\Delta^*}$, we set $\rho_{d,0} := d$ which is well-defined. Otherwise, consider the 0-chain $(\overline{d})$, we have $\rho_{d,0}(\overline{d}) := \varepsilon \diamond d = d$, which is well defined. In both cases, we obviously have $\rho_{d,0}(\overline{d}) \sim_{|q|+1} d$.

Induction step: $k \Rightarrow k+1$. Assume the mapping $\rho_{d,k} : \mathcal{N}_k(\overline{d}, \mathcal{J}) \to \mathcal{N}_k(d, \mathcal{I}')$ is well defined for some $k < |q|$. We explain how to *extend* it to a mapping $\rho_{d,k+1} : \mathcal{N}_{k+1}(\overline{d}, \mathcal{J}) \to \mathcal{N}_{k+1}(d, \mathcal{I}')$. Consider an element $\overline{e} \in \mathcal{N}_{k+1}(\overline{d}, \mathcal{J}) \setminus \mathcal{N}_k(\overline{d}, \mathcal{J})$.

If $e \in \Delta^*$, then we set $\rho_{d,k+1}(\overline{e}) := e$, which is well-defined and satisfies $\rho_{k+1}(\overline{e}) \sim_{|q|+1-(k+1)} e$.

Otherwise, $e \notin \Delta^*$, we know that there is a $k+1$-chain $(\overline{d_0}, \dots \overline{d_{k+1}})$ linking $\overline{d}$ and $\overline{e}$. In particular, we have a role R such that $(\overline{d_k}, \overline{e}) \in \mathrm{R}^{\mathcal{J}}$. From the definition of $\mathrm{R}^{\mathcal{J}}$, we can infer the existence of $\epsilon', \epsilon \in \Delta^{\mathcal{C}_\mathcal{K}}$ such that:

$$\epsilon' \sim_{|q|+1} d_k \qquad \epsilon \sim_{|q|+1} e,$$

and either $\epsilon' = \epsilon \mathrm{S}^-$ or $\epsilon = \epsilon' \mathrm{S}$, where S denotes the primary role of the edge $(\overline{d_k}, \overline{e})$. We consider these two cases in turn:

- If $\epsilon' = \epsilon \mathrm{S}^-$, we have $\rho_{d,k}(\overline{\epsilon'}) \sim_{|q|+1-k} d_k \sim_{|q|+1-k} \epsilon'$ due to $\overline{d_k} \in \mathcal{N}_k(\overline{d}, \mathcal{J})$ and the assumption for k, which implies that $\rho_k(\overline{d_k})$ ends with S^-. The action of S on $\rho_{d,k}(d_k)$ hence provides the well-defined word obtained from $\rho_k(\overline{d_k})$ by removing its final symbol S^-. Therefore, $\rho_{k+1}(\overline{e})$ is well defined. Since the equivalence class of $\rho_{d,k}(\overline{d_k})$ for $\sim_{|q|+1-k}$ fully determines the equivalence class of its immediate neighbour $\rho_{d,k+1}(\overline{e})$ for $\sim_{|q|+1-(k+1)}$, and since we know that $\rho_{d,k}(\overline{d_k}) \sim_{|q|+1-k} d_k$ by the induction hypothesis, we obtain $\rho_{d,k+1}(\overline{e}) \sim_{|q|+1-(k+1)} e$.
- Otherwise, if $\epsilon = \epsilon' \mathrm{S}$, the action of S on $\rho_{d,k}(d_k)$ provides $\rho_{d,k}(\overline{d_k})\mathrm{S}$, which is well defined since ϵ' and $\rho_{d,k}(d_k)$ must end by the same letter, as the induction hypothesis ensures $\epsilon' \sim_1 \rho_{d,k}(d_k)$ (recall $k < |q|$). Again, since the equivalence class of $\rho_{d,k}(\overline{d_k})$ for $\sim_{|q|+1-k}$ fully determines the equivalence class of its immediate neighbour $\rho_{d,k+1}(\overline{e})$ for $\sim_{|q|+1-(k+1)}$, and since we know that $\rho_{d,k}(\overline{d_k}) \sim_{|q|+1-k} d_k$ by the induction hypothesis, we obtain $\rho_{d,k+1}(\overline{e}) \sim_{|q|+1-(k+1)} e$.

The mapping ρ_d is obtained as $\rho_{d,|q|}$. Notice the two conditions are satisfied.

□

We now need to prove ρ_d is a homomorphism. We start by proving the following lemma, which states the links of an element e_1 to elements in Δ^* fully determines such links for any other element e_2 that is 1-equivalent to e_1.

Lemma 8. *If $(\overline{e_1}, \overline{d}) \in \mathrm{R}^{\mathcal{J}}$ for some $d \in \Delta^*$, and if $e_1 \sim_1 e_2$, then $(e_2, d) \in \mathrm{R}^{\mathcal{I}'}$.*

Proof. The definition of $\mathrm{R}^{\mathcal{J}}$ provides $\epsilon_1, \delta \in \Delta^{\mathcal{C}_\mathcal{K}}$ such that :

$$\overline{f'(\epsilon_1)} = \overline{e_1} \qquad \overline{f'(\delta)} = \overline{d} \qquad (\epsilon_1, \delta) \in \mathrm{R}^{\mathcal{C}_\mathcal{K}}.$$

Notice that since $d \in \Delta^*$, we have $\overline{d} = \{d\}$ and therefore $f'(\delta) = d$.

- If $e_1 \in \Delta^*$, then $\overline{e_1} = \{e_1\}$. It follows that $f'(\epsilon_1) = e_1$ and $e_2 = e_1$. We therefore obtain $(e_2, d) = (e_1, d) = (f'(\epsilon_1), f'(d)) \in f'(\mathrm{R}^{\mathcal{C}_\mathcal{K}}) = \mathrm{R}^{\mathcal{I}'}$.
- Otherwise, $e_1 \notin \Delta^*$, which means that $f'(\epsilon_1) \notin \Delta^*$, and hence $f'(\epsilon_1) = \epsilon_1$ and $f'(\epsilon_1) \in \Delta^{\mathcal{C}_\mathcal{K}} \setminus \mathsf{Ind}(\mathcal{A})$. Then, the definition of $\mathrm{R}^{\mathcal{C}_\mathcal{K}}$ provides a role $\mathrm{S} \in \mathsf{N}_\mathsf{R}^{\pm}$ such that $\mathcal{T} \models \mathrm{S} \sqsubseteq \mathrm{R}$ and either $\epsilon_1 = \delta \mathrm{S}^-$ or $\delta = \epsilon_1 \mathrm{S}$. Let us consider these two cases in turn:

– If $\epsilon_1 = \delta \mathrm{S}^-$, then the 1-root of $f'(\epsilon_1) = \epsilon$ is $f'(\delta)$ and $w_{1,e_1}^{e_1} = \mathrm{S}^-$. We thus have: $\chi_{1,f'(\epsilon_1)}(\varepsilon) = f'(\delta) = d$ (where ε denotes the empty word). But since $f'(\epsilon_1) \sim_1 e_1 \sim_1 e_2$, we have $\chi_{1,e_2} = \chi_{1,f'(\epsilon_1)}$ and $w_{1,e_2}^{e_2} = w_{1,e_1}^{e_1}$. Combining the preceding facts, we obtain $(e_2, d) = (r_{1,e_2} w_{1,e_2}^{e_2}, \chi_{1,e_2}(\varepsilon)) = (f'(r_{1,e_2}\mathrm{S}^-), f'(r_{1,e_2})) \in f'(\mathrm{R}^{\mathcal{C}_\mathcal{K}}) = \mathrm{R}^{\mathcal{I}'}$.

– If $\delta = \epsilon_1 \mathrm{S}$, then we have $\chi_{1,f'(\epsilon_1)}(w_{1,f'(\epsilon_1)}^{f'(\epsilon_1)}\mathrm{S}) = f'(\delta) = d$. But since $f'(\epsilon_1) \sim_1 e_1 \sim_1 e_2$, we have $\chi_{1,e_2} = \chi_{1,f'(\epsilon_1)}$ and $w_{1,e_2}^{e_2} = w_{1,f'(\epsilon_1)}^{f'(\epsilon_1)}$. Hence : $(e_2, d) = (r_{1,e_2} w_{1,e_2}^{e_2}, \chi_{1,f'(\epsilon_1)}(w_{1,f'(\epsilon_1)}^{f'(\epsilon_1)}\mathrm{S})) = (r_{1,e_2} w_{1,e_2}^{e_2}, \chi_{1,e_2}(w_{1,e_2}^{e_2}\mathrm{S})) = (f'(r_{1,e_2} w_{1,e_2}^{e_2}), f'(r_{1,e_2} w_{1,e_2}^{e_2}\mathrm{S})) \in f'(\mathrm{R}^{\mathcal{C}_\mathcal{K}}) = \mathrm{R}^{\mathcal{I}'}$.

□

Proof of Theorem 12. Let $e, e' \in \Delta^{\mathcal{I}'}$, and let $\mathrm{R} \in \mathsf{N_R}$ be such that $(\overline{e}, \overline{e'}) \in \mathrm{R}^{\mathcal{J}}$.

If $\overline{e} \in \overline{\Delta^*}$, then Lemma 8 applies by setting $(e_1, e_2, d, \mathrm{R}) := (e', \rho_d(\overline{e'}), e, \mathrm{R})$ and therefore $(\rho_d(\overline{e}), \rho_d(\overline{e'}) \in \mathrm{R}^{\mathcal{I}'}$.

Otherwise, if $\overline{e'} \in \overline{\Delta^*}$, then Lemma 8 also applies, by setting $(e_1, e_2, d, \mathrm{R}) := (e, \rho_d(\overline{e}), e', \mathrm{R})$ and again $(\rho_d(\overline{e}), \rho_d(\overline{e'}) \in \mathrm{R}^{\mathcal{I}'}$.

Otherwise, $\overline{e}, \overline{e'} \notin \overline{\Delta^*}$. Notice that both cannot be in $\mathcal{N}_{|q|}(\overline{d}, \mathcal{J}) \setminus \mathcal{N}_{|q|-1}(\overline{d}, \mathcal{J})$ at the same time. Indeed, if both were, there would be a $2|q|+1$-chain from $\overline{d}$ to $\overline{d}$ (recall $\overline{e}, \overline{e'} \notin \overline{\Delta^*}$). However, the depth modulo $2|q|+3$ encoded in each equivalent class along this chain only increases or decreases by 1 at each step (since none of its element belongs to $\overline{\Delta^*}$). Hence, $|d| \mod 2|q|+3$ would equal itself up to $2|q|+1$ such 1-steps, which is impossible modulo $2|q|+3$. Therefore, we can assume that $e \in \mathcal{N}_{|q|-1}(\overline{d}, \mathcal{J})$. We have a k-chain $C_{\overline{d} \to \overline{e}}$ from $\overline{d}$ to $\overline{e}$, with $k < |q|$. Complete it by R into a $k+1$ chain $C_{\overline{d} \to \overline{e'}}$ to reach $\overline{e'}$. Since $k+1 \leq |q|$, we have by definition $\rho_d(e') = \mathrm{S} \diamond \rho_d(\overline{e})$, with S the primary role of the edge (e, e'). In both cases, $\rho_d(\overline{e'})$ ending by S^- or not, it ensures that $(\rho_d(e), \rho_d(e') \in \mathrm{S}^{\mathcal{I}'} \subseteq \mathrm{R}^{\mathcal{I}'}$.

The preservation of positive concepts follows. Indeed, if we have an element $\overline{e}$ and a concept name A such that $\overline{e} \in \mathrm{A}^{\mathcal{J}}$, then, from the definition of $\mathrm{A}^{\mathcal{J}}$, either $\overline{e}$ is the interpretation of an individual name e and $\mathrm{B}(\mathsf{e}) \in \mathcal{A}$ for some concept B such that $\mathcal{T} \models \mathrm{B} \sqsubseteq \mathrm{A}$, or there exists another element $\overline{e'}$ connected to $\overline{e}$ by a positive role S such that $\mathcal{T} \models \exists \mathrm{S}^- \sqsubseteq \mathrm{A}$.

In the first case, we have in particular $\overline{e} \in \overline{\Delta^*}$, thus $\rho_d(\overline{e}) = e = \mathsf{e}^{\mathcal{I}'} \in \mathrm{A}^{\mathcal{I}'}$ since $\mathcal{I}'$ is a model.

In the second case, since ρ_d preserves positive roles, we have $(\rho_d(e'), \rho_d(e)) \in \mathrm{S}^{\mathcal{I}'}$, and therefore $\rho_d(e) \in \mathrm{A}^{\mathcal{I}'}$ since $\mathcal{I}'$ is a model.

□

B Proofs for Section 5 (Rooted Counting CQs)

Theorem 3. ***(Implicit in [Nikolaou* et al.*, 2019; Cima* et al.*, 2019])*** *In DL-Lite$_{\mathsf{core}}$, exhaustive rooted CCQ answering is* TC^0*-complete*[9] *w.r.t. data complexity.*

Proof. We start with the TC^0 hardness. The reduction from the NUMONES problem works as follows: given an instance (s, k), we create an ABox $\mathcal{A}_s := \{\mathrm{R}(\mathsf{a}, \mathsf{s}_k) \mid s_k \in s \wedge s_k = 1\}$, along with the empty TBox $\mathcal{T} = \emptyset$ and exhaustive rooted CCQ $q := \exists z\, \mathrm{R}(\mathsf{a}, z)$. It is clear that $(\emptyset, [k, +\infty]) \in [q]^{(\mathcal{T}, \mathcal{A}_s)} \iff (s, k) \in$ NUMONES. It can be verified that this simple reduction can be implemented by AC^0 circuits (so constitutes an AC^0-reduction, as required).

As explained in the body of the paper, TC^0 membership follows from results in [Nikolaou *et al.*, 2019]. While that paper only formally states membership in LogSpace, a follow-up paper on bag semantics [Cima *et al.*, 2019] states TC^0 membership for DL-Lite$_{\mathcal{F}}$ (which properly contains DL-Lite$_{\mathsf{core}}$), by making use of prior complexity results for bag relational algebra. We believe it is nevertheless instructive to have a direct proof and therefore describe what follows how to construct a family of TC^0 circuits to decide our problem.

We need a family of circuits in order to be able to handle ABoxes of different sizes. More precisely, we will create one circuit for each possible number ℓ of individual names. We can assume w.l.o.g. that the same set of individuals, denoted Ind_ℓ, is used for all of the ABoxes having ℓ individuals. Let us now explain how to represent an input $(\mathcal{A}^*, \boldsymbol{a}^*, m^*)$ to the circuit that handles ℓ-individual ABoxes.

- Each atomic role P appearing in $\mathcal{T}$ and/or q is represented by input gates $\bigcirc_{\mathrm{P}(\mathsf{a},\mathsf{b}) \in \mathcal{A}?}$ for $\mathsf{a}, \mathsf{b} \in \mathsf{Ind}_\ell$. The gate $\bigcirc_{\mathrm{P}(\mathsf{a},\mathsf{b}) \in \mathcal{A}?}$ is set to 1 iff $\mathrm{P}(\mathsf{a}, \mathsf{b}) \in \mathcal{A}^*$.
- Each atomic concept A appearing in $\mathcal{T}$ and/or q is represented by input gates $\bigcirc_{\mathrm{A}(\mathsf{a}) \in \mathcal{A}?}$ for $\mathsf{a} \in \mathsf{Ind}_\ell$. The gate $\bigcirc_{\mathrm{A}(\mathsf{a}) \in \mathcal{A}?}$ is set to 1 iff $\mathrm{A}(\mathsf{a}) \in \mathcal{A}^*$.
- The tuple $\boldsymbol{a}^*$ is represented by input gates $\bigcirc_{\boldsymbol{a}_k = \mathsf{a}}$ for $1 \leq k \leq |\boldsymbol{x}|$ and $\mathsf{a} \in \mathsf{Ind}_\ell$. The gate is set to 1 iff $\boldsymbol{a}^*_k = \mathsf{a}$.

[9] We recall that TC^0 is a circuit complexity class defined similarly to AC^0 but additionally allowing threshhold gates. It is known that $\mathsf{AC}^0 \subsetneq \mathsf{TC}^0 \subseteq \mathsf{NC}^1 \subseteq \mathsf{LogSpace} \subseteq \mathsf{PTime}$.

- The integer m^* is represented in binary by input gates $\bigcirc_{b_k=1}$ for each $0 \leq k < \log_2(|\mathsf{Ind}(\mathcal{A}^*)| + |\mathcal{T}|)^{|q|})$. The gate $\bigcirc_{b_k=1}$ is set to 1 iff the kth bit of m^* is 1 (with 0th-bit being the least significant bit).

Regarding the last point, we use the observation from [Kostylev and Reutter, 2015] that if $(\boldsymbol{a}^*, [m^*, +\infty]) \in [q]^{(\mathcal{T},\mathcal{A}^*)}$, then m^* cannot exceed $(|\mathsf{Ind}(\mathcal{A}^*)| + |\mathcal{T}|)^{|q|} = (|\mathsf{Ind}_\ell| + |\mathcal{T}|)^{|q|}$. This is a direct consequence of the fact that every satisfiable DL-Lite$_\mathcal{R}$ KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ has a model with at most $|\mathsf{Ind}(\mathcal{A})| + |\mathcal{T}|$ elements.

We now describe the other parts of the circuit. We introduce, for each relevant positive concept C (i.e., atomic concept or existential concept $\exists R$ that uses concept and role names from $\mathcal{T}$ and/or q) and each individual name $\mathsf{a} \in \mathsf{Ind}_\ell$, a disjunctive gate $\bigvee_{\mathcal{K}\models \mathrm{C}(\mathsf{a})?}$ taking as inputs:

- $\bigcirc_{\mathrm{A}(\mathsf{a})\in\mathcal{A}?}$ for each atomic concept A such that $\mathcal{T} \models \mathrm{A} \sqsubseteq \mathrm{C}$.
- $\bigcirc_{\mathrm{P}(\mathsf{a},\mathsf{b})\in\mathcal{A}?}$ for all $\mathsf{b} \in \mathsf{Ind}(\mathcal{A})$ such that $\mathcal{T} \models \exists \mathrm{P} \sqsubseteq \mathrm{C}$.
- $\bigcirc_{\mathrm{P}(\mathsf{b},\mathsf{a})\in\mathcal{A}?}$ for all $\mathsf{b} \in \mathsf{Ind}(\mathcal{A})$ such that $\mathcal{T} \models \exists \mathrm{P}^- \sqsubseteq \mathrm{C}$.

The preceding gates determine the ABox part of the canonical model. We next need to decide the existence of each element of the form $\mathsf{a}w$, where $a \in \mathsf{Ind}(\mathcal{A})$ and $w \in \Gamma_{q,\mathcal{T}} \setminus \varepsilon$ (by Lemma 2, these are the only anonymous elements that can occur in a match for q). For each such $\mathsf{a}w$, we denote by R_w the first role name of w and introduce a conjunctive gate $\bigwedge_{\mathsf{a}w\in\Delta^{\mathcal{C}_\mathcal{K}}?}$ which takes as input:

- The negation $\neg_{\forall \mathsf{b}\in\mathsf{Ind}(\mathcal{A})\ \neg\mathrm{R}(\mathsf{a},\mathsf{b})?}$ of a disjunctive gate $\bigvee_{\exists \mathsf{b}\in\mathsf{Ind}(\mathcal{A})\ \mathrm{R}(\mathsf{a},\mathsf{b})?}$ taking as inputs the gates:
  - $\bigcirc_{\mathrm{P}(\mathsf{a},\mathsf{b})}$ for each $\mathsf{b} \in \mathsf{Ind}_\ell$, if $\mathrm{R} = \mathrm{P} \in \mathsf{N}_\mathsf{R}$.
  - $\bigcirc_{\mathrm{P}(\mathsf{b},\mathsf{a})}$ for each $\mathsf{b} \in \mathsf{Ind}_\ell$ if $\mathrm{R} = \mathrm{P}^-$ with $\mathrm{P} \in \mathsf{N}_\mathsf{R}$.

  which verifies that there is not already a R_w-successor to a.
- The gate $\bigvee_{\mathcal{K}\models\exists \mathrm{R}_w(\mathsf{a})?}$ that checks that a witnessing R_w-successor is needed.

The circuit next determines for each mapping $\sigma : \boldsymbol{x} \cup \boldsymbol{z} \mapsto \{\mathsf{a}w | \mathsf{a} \in \mathsf{Ind}_\ell, w \in \Gamma_{q,\mathcal{T}}\}$, whether σ is a match for $q(\boldsymbol{a}^*)$. Notice that, regardless of the input ABox, we can restrict to a set of relevant mappings by keeping only those which map the answer variables $\boldsymbol{x}$ to individuals from Ind_ℓ and which map variables v_1, v_2 occurring in a role atom $\mathrm{R}(v_1, v_2)$ from q onto either:

- a pair of individual names, or
- a pair w_1, w_2 such that $w_2 = w_1\mathrm{R}$ or $w_1 = \mathrm{R}^- w_2$.

Similarly, we can restrict the set of relevant mappings by keeping only those which map variable v occuring in a concept atom $\mathrm{A}(v)$ from q onto either an individual name, or an element $aw\mathrm{R}$, where $\mathcal{K} \models \exists \mathrm{R}^- \sqsubseteq \mathrm{A}$. Clearly, any mapping σ that does not respect these conditions cannot be a match, due to the definition of $\mathrm{R}^{\mathcal{C}_\mathcal{K}}$. This restriction simplifies the process of checking if a mapping is a match for $q(\boldsymbol{a}^*)$: we are only left with verifying the existence of the anonymous elements in its image, as well as the validity of the atoms mapped onto the ABox part of the canonical model.

For each relevant mapping σ, we introduce a conjunctive gate $\bigwedge_{\sigma \text{ match }?}$ taking as inputs all gates:

- $\bigcirc_{\boldsymbol{a}_k=\sigma(x_k)?}$ for each $1 \leq k \leq |\boldsymbol{x}|$ (to check $\boldsymbol{x}$ is mapped on $\boldsymbol{a}^*$).
- $\bigwedge_{\sigma(z)\in\Delta^{\mathcal{C}_\mathcal{K}}?}$ for each $z \in \boldsymbol{z}$ such that $\sigma(z) \notin \mathsf{Ind}_\ell$ (to check for existence of $\sigma(z)$ under input $\mathcal{A}^*$).
- $\bigcirc_{\mathrm{R}(\sigma(v_1),\sigma(v_2))\in\mathcal{A}?}$ for each $v_1, v_2 \in \boldsymbol{x} \cup \boldsymbol{z}$ such that $\mathrm{R}(v_1, v_2) \in q$ and $\sigma(v_1), \sigma(v_2) \in \mathsf{Ind}(\mathcal{A})$ (to check the validity of the mapping for pairs of variables mapped on individual names).
- $\bigvee_{\mathcal{K}\models\mathrm{A}(\sigma(v))?}$ for each $v \in \boldsymbol{x} \cup \boldsymbol{z}$ such that $\mathrm{A}(v) \in q$ and $\sigma(v) \in \mathsf{Ind}(\mathcal{A})$ (to check the validity of the mapping for variables mapped on individual names).

We will next use threshold gates in order to compute the total number of matches. Introduce, for each $k = 0, \ldots, (\mathsf{Ind}_\ell \times \Gamma_{q,\mathcal{T}})^{|q|}$, a threshold gate $\top_k{}_{q_{\boldsymbol{a}}^{\mathcal{C}_\mathcal{K}} \geq k?}$ taking as input every $\bigwedge_{\sigma \text{ match }?}$. The gate $\top_k{}_{q_{\boldsymbol{a}}^{\mathcal{C}_\mathcal{K}} \geq k?}$ returns 1 iff at least k of its inputs are 1. By construction, the latter holds iff there are at least k matches for $q(\boldsymbol{a}^*)$.

In parallel, we introduce a conjunctive gate $\bigwedge_{m=k?}$ for each $k = 0, \ldots, (\mathsf{Ind}_\ell \times \Gamma_{q,\mathcal{T}})^{|q|}$ taking as inputs:

- the input gates $\bigcirc_{b_j=1?}$ such that the jth bit of the binary encoding of k is 1
- the negation of each input gate $\bigcirc_{b_j=1?}$ such that the jth bit of the binary encoding of k is 0

The gate $\bigwedge_{m=k?}$ returns 1 iff $m^* = k$.

We combine the preceding two types of gates to compare m^* and the computed number of matches. For each $k = 0, \ldots, (\mathsf{Ind}_\ell \times \Gamma_{q,\mathcal{T}})^{|q|}$, we introduce a conjunctive gate $\bigwedge_{q_{\boldsymbol{a}}^{c_\mathcal{K}} \geq m?}$ taking as input $\top_{k\,q_{\boldsymbol{a}}^{c_\mathcal{K}} \geq k?}$ and $\bigwedge_{m=k?}$.

Finally, our output gate is a disjunctive gate $\bigvee_{\mathsf{output}}$ taking as inputs all gates $\bigwedge_{q_{\boldsymbol{a}}^{c_\mathcal{K}} \geq m?}$. By construction, this gate outputs 1 iff there are at least m^* matches of $q(\boldsymbol{a}^*)$ in the canonical model of the considered KB.

The depth of the circuit is 7, and is hence constant, showing membership in TC^0. □

Example 3. *Let $\mathcal{T}$ be the following DL-Lite$_{\mathsf{core}}$ TBox*

$$\mathcal{T} = \{\mathrm{D} \sqsubseteq \exists \mathrm{R}, \exists \mathrm{R}^- \sqsubseteq \exists \mathrm{R}\}$$

and q be the exhaustive rooted CCQ given by

$$q(x) = \exists z\ \mathrm{R}(x, z)$$

Observe that even if the second axiom from $\mathcal{T}$ suggests the need to consider suffixes $\mathrm{R} \ldots \mathrm{R}$ of arbitrary length, we only have $\Gamma_{q,\mathcal{T}} = \{\varepsilon, \mathrm{R}\}$.

We propose to illustrate the construction of the circuit designed for 2-individual ABoxes, with individual names a *and* b*. We thus require $\lceil \log_2((2+2)^1) \rceil + 1 = 3$ input gates representing the input integer, and we have 6 relevant matches given by:*

- (σ_1) $x \mapsto \mathsf{a}$ $z \mapsto \mathsf{a}$
- (σ_2) $x \mapsto \mathsf{a}$ $z \mapsto \mathsf{b}$
- (σ_3) $x \mapsto \mathsf{a}$ $z \mapsto \mathsf{aR}$
- (σ_4) $x \mapsto \mathsf{b}$ $z \mapsto \mathsf{b}$
- (σ_5) $x \mapsto \mathsf{b}$ $z \mapsto \mathsf{a}$
- (σ_6) $x \mapsto \mathsf{b}$ $z \mapsto \mathsf{bR}$

The corresponding circuit is depicted in Figure 1.

Theorem 4. *In DL-Lite$_{\mathsf{core}}$, rooted CCQ answering is* coNP*-complete w.r.t. data complexity.*

Proof. We briefly recall the reduction sketched in the body of the paper. Starting from an instance $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ of the decision problem 3COL, we consider the ABox $\mathcal{A}_\mathcal{G}$ given by:

$$\begin{aligned}\mathcal{A}_\mathcal{G} =& \{\mathrm{Vertex}(\mathsf{a}, \mathsf{u}) \mid u \in \mathcal{V}\} \cup \{\mathrm{Edge}(\mathsf{u}_1, \mathsf{u}_2) \mid (u_1, u_2) \in \mathcal{E}\} \\ & \cup \{\mathrm{Vertex}(\mathsf{a}, \mathsf{a_v}), \mathrm{Edge}(\mathsf{a_v}, \mathsf{a_v}), \mathrm{HasCol}(\mathsf{a_v}, \mathsf{r}), \mathrm{HasCol}(\mathsf{a_v}, \mathsf{g}), \mathrm{HasCol}(\mathsf{a_v}, \mathsf{b})\}\end{aligned}$$

and the TBox $\mathcal{T} := \{\exists \mathrm{Vertex}^- \sqsubseteq \exists \mathrm{HasCol}\}$, and we denote by $\mathcal{K}_\mathcal{G} = (\mathcal{T}, \mathcal{A}_\mathcal{G})$ the resulting KB. A part of the canonical model of $\mathcal{K}_\mathcal{G}$ is depicted in Figure 2.

We consider the two following rooted CCQs:

$$\begin{aligned} q^{edge} &= \exists y_c\ \exists z_1\ \exists z_2\ \mathrm{Vertex}(\mathsf{a}, z_1) \wedge \mathrm{Vertex}(\mathsf{a}, z_2) \wedge \mathrm{Edge}(z_1, z_2) \wedge \mathrm{HasCol}(z_1, y_c) \wedge \mathrm{HasCol}(z_2, y_c) \\ q^{col} &= \exists y\ \exists z\ \mathrm{Vertex}(\mathsf{a}, y) \wedge \mathrm{HasCol}(y, z) \end{aligned}$$

We let q be the query obtained by taking the conjunction of these two queries and keeping all of the variables existentially quantified. The query q is displayed in Figure 3. The three counting variables (z_1, z_2, z) are indicated by large gray dots.
It is not hard to see that $(\boldsymbol{a}_\emptyset, [3, +\infty]) \in [q]^{\mathcal{K}_\mathcal{G}}$. Indeed, there are at least 9 matches of q in any model $\mathcal{I}$ of $\mathcal{K}$, given by:

$$z_1, z_2, y \mapsto \mathsf{a_v} \qquad y_c \mapsto \mathsf{r} \mid \mathsf{g} \mid \mathsf{b} \qquad z \mapsto \mathsf{r} \mid \mathsf{g} \mid \mathsf{b}$$

These 9 matches give rise to 3 c-matches for q, corresponding to the three ways of mapping counting variable z. To complete the proof, we establish the following claim.

Claim. $(\boldsymbol{a}_\emptyset, [4, +\infty]) \in [q]^{\mathcal{K}_\mathcal{G}} \iff \mathcal{G} \notin$ 3COL.

($\Rightarrow$) Assume $(\boldsymbol{a}_\emptyset, [4, +\infty]) \in [q]^{\mathcal{K}_\mathcal{G}}$, and take some possible coloring $\tau : \mathcal{V} \to \{\mathsf{r}, \mathsf{g}, \mathsf{b}\}$ of the graph $\mathcal{G}$. Let $\mathcal{I}_\tau^\mathcal{G}$ be the model of $\mathcal{K}_\mathcal{G}$ whose domain is $\mathsf{Ind}(\mathcal{A}_\mathcal{G})$ and which interprets roles Vertex and Edge exactly following the ABox, and which interprets HasCol according to τ:

$$\mathrm{HasCol}^{\mathcal{I}_\tau^\mathcal{G}} = \{(\mathsf{a_v}, \mathsf{r}), (\mathsf{a_v}, \mathsf{g}), (\mathsf{a_v}, \mathsf{b})\} \cup \{(\mathsf{v}, \tau(v)) \mid v \in \mathcal{V}\}$$

Intuitively, $\mathcal{I}_\tau$ is obtained from the canonical model by replacing the element vHasCol with $\tau(v)$.

By hypothesis, there is a fourth c-match σ for q in $\mathcal{I}_\tau^\mathcal{G}$. It is easily verified that the additional match can only result from the atom $\mathrm{Edge}(z_1, z_2)$ being mapped onto an edge $\mathrm{Edge}(\mathsf{u}_1, \mathsf{u}_2)$ that is different from $\mathrm{Edge}(\mathsf{a_v}, \mathsf{a_v})$. From the definition of $\mathcal{I}_\tau^\mathcal{G}$, this implies that the edge (u_1, u_2) of $\mathcal{G}$ is monochromatic, both vertices sharing the color $\sigma(y_c)$. Thus, τ is not a 3-coloring. As this construction holds for any possible coloring τ, we obtain $\mathcal{G} \notin$ 3COL.

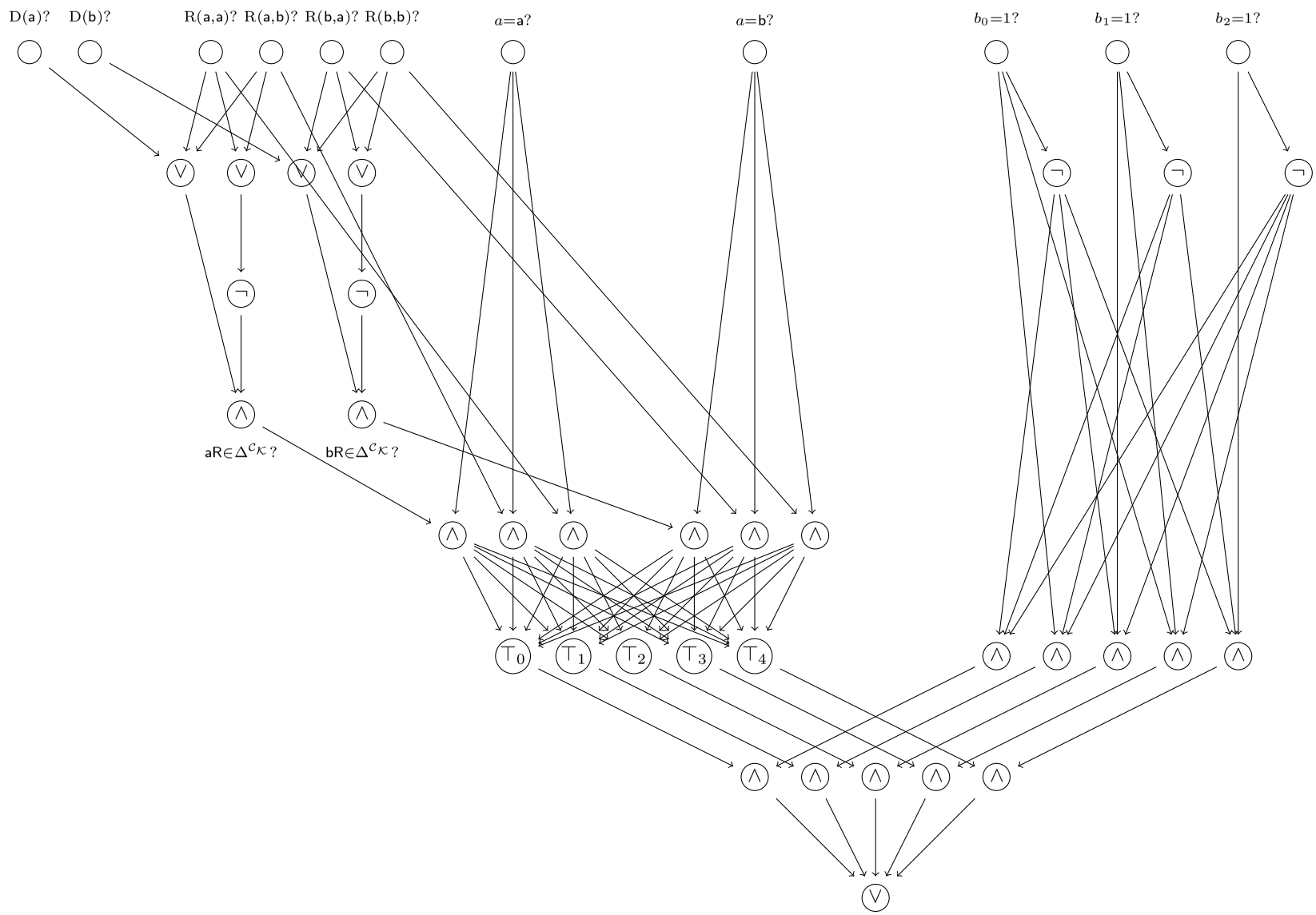

Figure 1: The TC^0 circuit built for 2-individual ABoxes w.r.t. $\mathcal{T}$ and q.

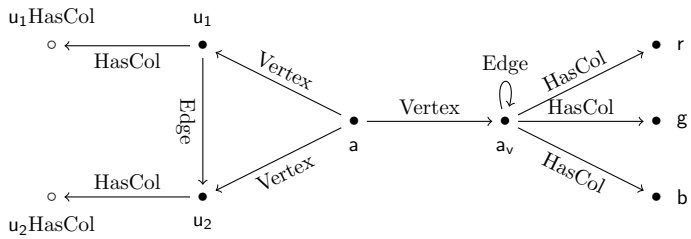

Figure 2: A part of $\mathcal{C}_{\mathcal{K}_\mathcal{G}}$ with $(u_1, u_2) \in \mathcal{E}$.

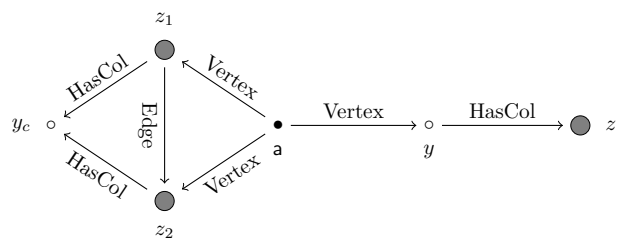

Figure 3: The rooted CCQ q, which is the conjunction of q^{edge} (left part) and q^{col} (right part).

($\Leftarrow$) Assume $\mathcal{G} \notin$ 3COL, and take some model $\mathcal{I}$ of $\mathcal{K}_\mathcal{G}$. By Lemma 1, there is a homomorphism $f : \mathcal{C}_{\mathcal{K}_\mathcal{G}} \to \mathcal{I}$ (which preserves individual names). Define $\tau : \mathcal{V} \to \Delta^\mathcal{I}$ as follows: $\tau(u) = f(\text{uHasCol})$. There are two cases to consider:

- If there exists $u \in \mathcal{V}$ such that $\tau(u) \notin \{\mathsf{r}, \mathsf{g}, \mathsf{b}\}$, then this provides a match of q in $\mathcal{I}$ given by $z \mapsto \tau(u)$ and $y \mapsto \mathsf{u}^\mathcal{I}$, whose restriction to the counting variables is a new c-match.
- Else, since $\mathcal{G} \notin$ 3COL, there exists an edge $(u_1, u_2) \in \mathcal{E}$ such that $\tau(u_1) = \tau(u_2)$. It provides a new match given by:

$$z \mapsto \mathsf{r} \quad y \mapsto \mathsf{a_v} \quad z_1 \mapsto \mathsf{u_1} \quad z_2 \mapsto \mathsf{u_2} \quad y_c \mapsto \tau(u_1)\ (= \tau(u_2))$$

In both cases, there is a fourth c-match for q. We thus obtain $(\boldsymbol{a}_\emptyset, [4, +\infty]) \in [q]^{\mathcal{K}_\mathcal{G}}$. □

Theorem 5. *In DL-Lite$_\mathcal{R}$, rooted CCQ answering is* coNEXP*-hard w.r.t. combined complexity.*

Proof. The proof is by reduction from the exponential grid tiling problem (EXPTIL). We recall that an instance of this problem consists of a set $\mathcal{C}$ of colors, two relations $\mathcal{H}, \mathcal{V} \subseteq \mathcal{C} \times \mathcal{C}$ that give the horizontal and vertical tiling conditions, and a number n. The task is to decide whether there exists a valid $(\mathcal{H}, \mathcal{V})$-tiling of an $2^n \times 2^n$ grid, i.e., a mapping $\tau : \{0, \dots, 2^n - 1\} \times \{0, \dots, 2^n - 1\} \mapsto \mathcal{C}$ such that $(\tau(i, j), \tau(i + 1, j)) \in \mathcal{H}$ for every $0 \leq i < 2^n - 1$ and $(\tau(i, j), \tau(i, j + 1)) \in \mathcal{V}$ for every $0 \leq j < 2^n - 1$. In what follows, we consider an instance $(n, \mathcal{C}, \mathcal{H}, \mathcal{V})$ be an instance of the EXPTIL problem.

To be able to test for the existence of a tiling of a $2^n \times 2^n$ grid, we must start by ensuring we can find such a grid in each model. Furthermore, we will need to detect horizontal and vertical adjacency in this grid, it is thus appropriate to use horizontal/vertical coordinates. To ensure a polynomial reduction, we need to use a binary encoding of these coordinates. We start from a root a and an initial element b and use TBox axioms to build two witnesses to represent the two possible values for the nth bit of the horizontal coordinates:

$$\text{Roots}(\mathsf{a}, \mathsf{b}) \qquad \exists \text{Roots}^- \sqsubseteq \exists \text{H}_0^n \qquad \exists \text{Roots}^- \sqsubseteq \exists \text{H}_1^n$$

We use further axioms to generate all possible horizontal coordinates, and we proceed similarly with the vertical coordinates, until we generate all possible pairs of coordinates. Concretely, we include the following axioms:

$$\exists (\text{H}_b^i)^- \sqsubseteq \exists \text{H}_{b'}^{i-1} \qquad \exists (\text{H}_b^1)^- \sqsubseteq \exists \text{V}_{b'}^n \qquad \exists (\text{V}_b^i)^- \sqsubseteq \exists \text{V}_{b'}^{i-1} \qquad \text{for all } b, b' \in \{0, 1\}, 1 < i \leq n$$

The preceding axioms will generate a binary tree of height $2n$ in the canonical model, whose leaves represent all possible grid positions. We use the following two axioms assign a color to each of the points representing a grid position:

$$\exists (\text{V}_0^1)^- \sqsubseteq \exists \text{HasCol} \qquad \exists (\text{V}_1^1)^- \sqsubseteq \exists \text{HasCol}$$

To help us compare positions, we will include the following TBox axioms, for all $b \in \{0, 1\}$ and $1 \leq i \leq n$:

$$\exists (\text{H}_b^i)^- \sqsubseteq \exists \text{HasBit}_b \qquad \exists (\text{V}_b^i)^- \sqsubseteq \exists \text{HasBit}_b$$

We will also introduce a general role (HV) to more compactly navigate the tree:

$$\mathrm{H}_b^i \sqsubseteq \mathrm{HV} \qquad \mathrm{V}_b^i \sqsubseteq \mathrm{HV} \qquad (b \in \{0,1\}, 1 \leq i \leq n)$$

This completes our description of the TBox. We will finish our description of the ABox later in the proof, but it will be useful to know that it will contain an ABox individual c for every color $c \in \mathcal{C}$ and two ABox individuals $(\mathsf{one}, \mathsf{zero})$ to represent bits.

Let us now define the query q. To keep track of the colors used in a candidate tiling, we will use the following subquery:

$$q^{col} = \exists y_0^{col} \ldots \exists y_{2n}^{col} \exists z^{col}\ \mathrm{Roots}(\mathsf{a}, y_0^{col}) \wedge \bigwedge_{i=0}^{2n-1} \mathrm{HV}(y_i^{col}, y_{i+1}^{col}) \wedge\ \mathrm{HasCol}(y_{2n}^{col}, z^{col})$$

Observe that z^{col} is the only counting variable. We also need to be able to detect if other bits than the intended ones $(\mathsf{one}, \mathsf{zero})$ are being used to satisfy the axioms $\mathrm{H}_b^- \sqsubseteq \exists \mathrm{HasBit}_b$ and $\mathrm{V}_b^- \sqsubseteq \exists \mathrm{HasBit}_b$. For this purpose, we will introduce the two following subqueries:

$$q^0 = \exists y_0^0 \ldots \exists y_{2n}^0 \exists z^0\ \mathrm{Roots}(\mathsf{a}, y_0^0) \wedge \bigwedge_{i=0}^{2n-1} \mathrm{HV}(y_i^0, y_{i+1}^0) \wedge\ \mathrm{HasBit}_0(y_{2n}^0, z^0)$$

$$q^1 = \exists y_0^1 \ldots \exists y_{2n}^1 \exists z^1\ \mathrm{Roots}(\mathsf{a}, y_0^1) \wedge \bigwedge_{i=0}^{2n-1} \mathrm{HV}(y_i^1, y_{i+1}^1) \wedge\ \mathrm{HasBit}_1(y_{2n}^1, z^1)$$

We note that each of the preceding queries has a single counting variable (z^0 or z^1). The axioms for HV together with the construction of the ABox will ensure that every element used as a bit (i.e., in the second argument of HasBit) gives rise to a c-match for one of these two queries.

We next discuss the parts of the query that are used to check the tiling conditions. To detect adjacency, we remark that two grid positions $(h_1, v_1), (h_2, v_2) \in \{0, \ldots, 2^n - 1\} \times \{0, \ldots, 2^n - 1\}$ are vertically adjacent iff:

- $h_1 = h_2$, so the binary encodings of h_1 and h_2 are the same;
- $v_2 = v_1 + 1$, so the binary encodings of v_2 and v_1 are the same until, at some point, v_2 ends with $1 \cdot 0^k$ while v_1 ends with $0 \cdot 1^k$.

To detect a violation of the vertical tiling condition (i.e. two vertically adjacent tiles with colors c and c' such that $(c, c') \notin \mathcal{V}$), we need n queries, one for each possible position where the bit from the vertical coordinates differ. For each $1 \leq k \leq n$, we create a subquery $q^{\mathcal{V},(c,c'),k}$ defined as follows. Note that the variables in $q^{\mathcal{V},(c,c'),k}$ all have the superscript $\cdot^{\mathcal{V},(c,c'),k}$, which means they do not occur in any other subquery, but these superscripts are omitted in the definition for the sake of readability.

$$\begin{aligned} q^{\mathcal{V},(c,c'),k} = {} & \exists z \exists y_{l,1} \ldots \exists y_{l,2n} \exists y_{r,1} \ldots \exists y_{r,2n} \exists y_{s,1} \ldots \exists y_{s,n+k} \\ & \mathrm{Roots}(\mathsf{a}, z) \wedge \mathrm{HV}(z, y_{l,1}) \wedge \mathrm{HV}(z, y_{r,1}) \wedge \left(\bigwedge_{i=1}^{2n-1} \mathrm{HV}(y_{l,i}, y_{l,i+1}) \wedge \mathrm{HV}(y_{r,i}, y_{r,i+1}) \right) \\ & \wedge \mathrm{HasCol}(y_{l,2n}, \mathsf{c}) \wedge \mathrm{HasCol}(y_{r,2n}, \mathsf{c}') \wedge \left(\bigwedge_{i=1}^{n+k-1} \mathrm{HasBit}(y_{l,i}, y_{s,i}) \wedge \mathrm{HasBit}(y_{r,i}, y_{s,i}) \right) \\ & \wedge \mathrm{HasBit}(y_{l,n+k}, \mathsf{zero}) \wedge \mathrm{HasBit}(y_{r,n+k}, \mathsf{one}) \wedge \left(\bigwedge_{i=n+k+1}^{2n} \mathrm{HasBit}(y_{l,i}, \mathsf{one}) \wedge \mathrm{HasBit}(y_{r,i}, \mathsf{zero}) \right) \end{aligned}$$

Note that z is the only counting variable of $q^{\mathcal{V},(c,c'),k}$. We can similarly define a set of subqueries $q^{\mathcal{H},(c,c'),k}$ $(1 \leq k \leq n)$ that detect violations of the horizontal tiling conditions.

Finally, we let q be the conjunction of the all of the preceding subqueries. It is displayed in Figure 4. The set of counting variables of q is the union of the counting variables of its subqueries. We observe that q is rooted, as it has a single connected component which contains the individual a.

We can now define the ABox, which introduces individuals for the intended colors and bits and a further individual d that serves to ensure that all parts of the query can be matched:

$$\begin{aligned} \mathcal{A} = {} & \{\mathrm{Roots}(\mathsf{a}, \mathsf{b}), \mathrm{Roots}(\mathsf{a}, \mathsf{d}), \mathrm{HV}(\mathsf{b}, \mathsf{b}), \mathrm{HasBit}_0(\mathsf{d}, \mathsf{zero}), \mathrm{HasBit}_1(\mathsf{d}, \mathsf{one})\} \\ & \cup \{\mathrm{H}_b^k(\mathsf{d}, \mathsf{d}) \mid b \in \{0,1\}, k = 1, \ldots n\} \cup \{\mathrm{V}_b^k(\mathsf{d}, \mathsf{d}) \mid b \in \{0,1\}, k = 1, \ldots n\} \\ & \cup \{\mathrm{HasCol}(\mathsf{d}, \mathsf{c}) \mid c \in \mathcal{C}\}. \end{aligned}$$

Let $p = |\mathcal{C}|$, and let $\mathcal{K}$ be the KB with the preceding TBox and ABox. A part of the canonical model $\mathcal{C}_\mathcal{K}$ is displayed in Figure 5. To complete the proof, it suffices to establish the following claim:

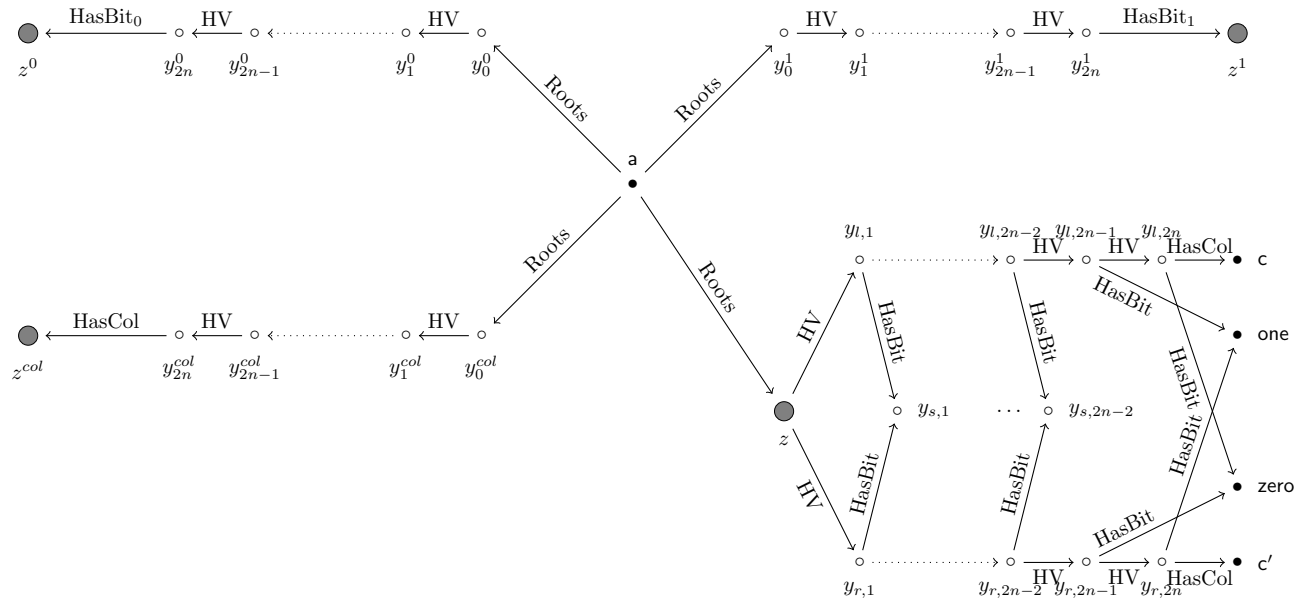

Figure 4: A part of the rooted query q, being the conjunction of q^0 (above left), q^1 (above right), q^{col} (below left), several $q^{\mathcal{V},(c,c'),k}$ (below right, only one is depicted with omitted superscripts), and several $q^{\mathcal{H},(c,c'),k}$ (none is depicted).

Claim $(\emptyset, [p+1, +\infty]) \in [q]^{\mathcal{K}} \iff (n, \mathcal{C}, \mathcal{H}, \mathcal{V}) \notin \mathsf{EXPTIL}$.

The proof of this claim is similar in spirit to the proof of Theorem 4. First observe that there are always at least p c-matches given by mapping the counting variables as follows:

$$z^{col} \mapsto \mathsf{c}_1 \mid \ldots \mid \mathsf{c}_p \qquad z^0 \mapsto \mathsf{zero} \qquad z^1 \mapsto \mathsf{one} \qquad z^{\mathcal{H},(c,c'),k}, z^{\mathcal{V},(c,c'),k} \mapsto \mathsf{d}$$

and mapping all of the existential variables to d.

($\Rightarrow$) Assume $(\emptyset, [p+1, +\infty]) \in [q]^{\mathcal{K}}$, and take some potential tiling $\tau : \{0, \ldots 2^n - 1\} \times \{0, \ldots 2^n - 1\} \to \{\mathsf{c} \mid c \in \mathcal{C}\}$. Let $\mathcal{I}_\tau$ be the model of $\mathcal{K}$ that is obtained from $\mathcal{C}_{\mathcal{K}}$ as follows:

- $\Delta^{\mathcal{I}_\tau}$ contains all elements from $\Delta^{\mathcal{C}_{\mathcal{K}}}$ except those anonymous elements whose last symbol is HasCol, HasBit_0, or HasBit_1 (i.e. witnesses for axioms involving $\exists\mathrm{HasCol}$, $\exists\mathrm{HasBit}_0$, or $\exists\mathrm{HasBit}_1$);
- the roles HasCol, HasBit_0, HasBit_1 are interpreted as follows:

$$\begin{aligned}
\mathrm{HasBit}_0^{\mathcal{I}_\tau} :=& \{(\mathsf{d}, \mathsf{zero})\} \cup \{(\mathsf{b}\mathrm{H}^n_{h_n} \ldots \mathrm{H}^k_{h_k} \mathrm{H}^{k-1}_0, \mathsf{zero}) \mid h_n, \ldots h_k \in \{0,1\}, k = 1, \ldots n+1\} \\
&\cup \{(\mathsf{b}\mathrm{H}^n_{h_n} \ldots \mathrm{H}^1_{h_1} \mathrm{V}^n_{v_n} \ldots \mathrm{V}^k_{v_k} \mathrm{V}^{k-1}_0, \mathsf{zero}) \mid h_n, \ldots h_1, v_n, \ldots v_k \in \{0,1\}, k = 0, \ldots n+1\} \\
\mathrm{HasBit}_1^{\mathcal{I}_\tau} :=& \{(\mathsf{d}, \mathsf{one})\} \cup \{(\mathsf{b}\mathrm{H}^n_{h_n} \ldots \mathrm{H}^k_{h_k} \mathrm{H}^{k-1}_1, \mathsf{one}) \mid h_n, \ldots h_k \in \{0,1\}, k = 1, \ldots n+1\} \\
&\cup \{(\mathsf{b}\mathrm{H}^n_{h_n} \ldots \mathrm{H}^1_{h_1} \mathrm{V}^n_{v_n} \ldots \mathrm{V}^k_{v_k} \mathrm{V}^{k-1}_1, \mathsf{one}) \mid h_n, \ldots h_1, v_n, \ldots v_k \in \{0,1\}, k = 0, \ldots n+1\} \\
\mathrm{HasCol}^{\mathcal{I}_\tau} :=& \{(\mathsf{d}, \mathsf{c}_k) \mid k = 1, \ldots p\} \\
&\cup \{(\mathsf{b}\mathrm{H}^n_{h_n} \ldots \mathrm{H}^1_{h_1} \mathrm{V}^n_{v_n} \ldots \mathrm{V}^1_{v_1}, \tau(h_n \ldots h_1, v_n \ldots v_1) \mid h_n, \ldots h_1, v_n, \ldots v_1 \in \{0,1\}\}
\end{aligned}$$

  where by a slight abuse of notation, we use $\tau(h_n \ldots h_1, v_n \ldots v_1)$ to mean $\tau(h, v)$, with h and v the numbers whose binary encodings are $h_n \ldots h_1$ and $v_n \ldots v_1$ respectively;
- the remaining roles are interpreted exactly as in $\mathcal{C}_{\mathcal{K}}$.

The model $\mathcal{I}_\tau$ is displayed in Figure 6. By hypothesis, there is an additional c-match σ for q in $\mathcal{I}_\tau$. It is easily verified that the additional match can only result from an atom $\mathrm{Roots}(\mathsf{a}, z^{\mathcal{D},(c,c'),k})$, with $\mathcal{D} \in \{\mathcal{H}, \mathcal{V}\}$, $(c, c') \in (\mathcal{C} \times \mathcal{C}) \setminus \mathcal{D}$ and $k \in \{1, \ldots n\}$, being mapped onto $\mathrm{Edge}(\mathsf{a}, \mathsf{b})$. From the definition of $\mathcal{I}_\tau$, this implies that there are two horizontally (or vertically) adjacent tiles, which positions are given by the elements $\sigma(y^{\mathcal{D},(c,c'),k}_{l,2n})$ and $\sigma(y^{\mathcal{D},(c,c'),k}_{r,2n})$, whose respective colors c and c' violate $\mathcal{D}$. Thus, τ is not an $(\mathcal{H}, \mathcal{V})$-tiling. As this construction holds for any possible tiling τ, we can infer that $(n, \mathcal{C}, \mathcal{H}, \mathcal{V}) \notin \mathsf{EXPTIL}$.

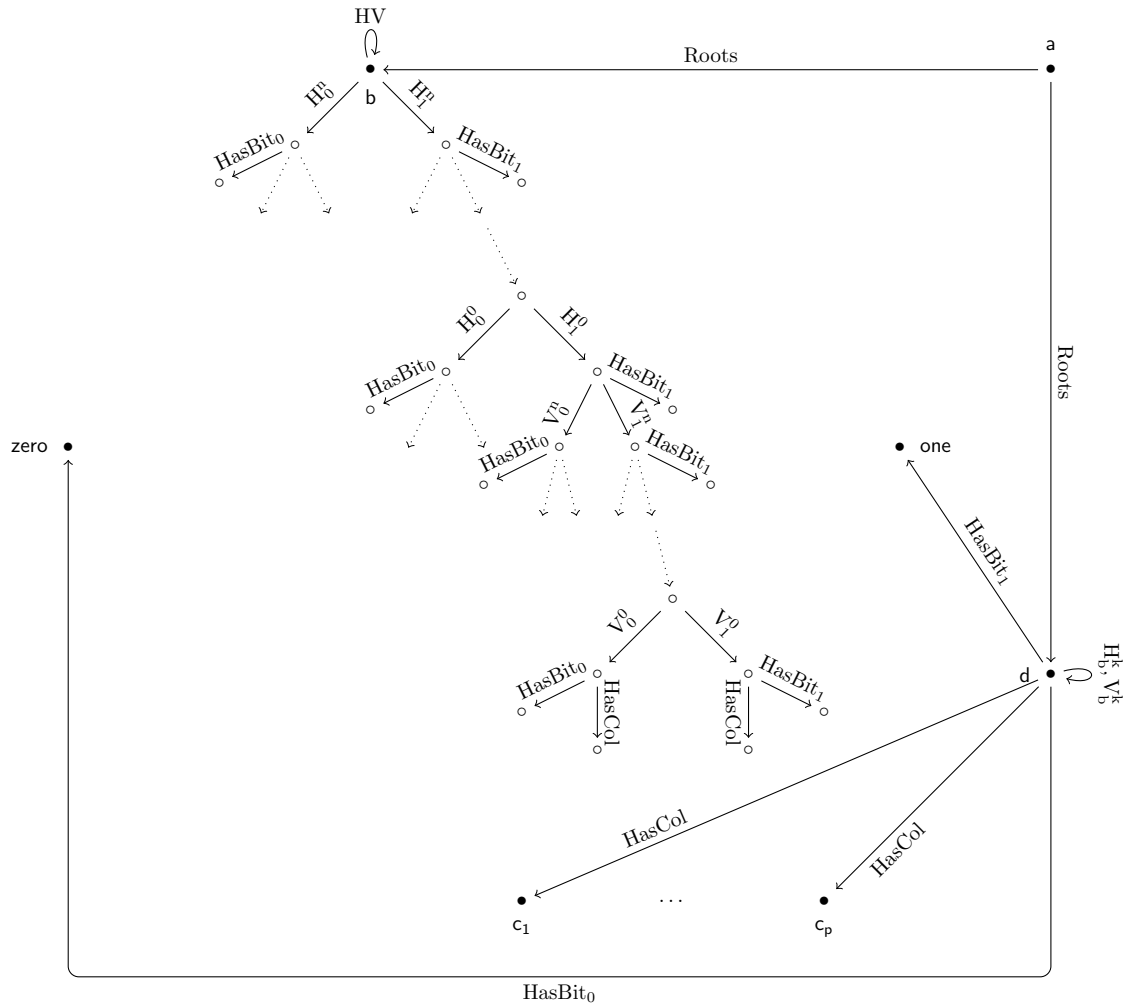

Figure 5: A part of the canonical model $\mathcal{C}_\mathcal{K}$.

($\Leftarrow$) Assume $(n, \mathcal{C}, \mathcal{H}, \mathcal{V}) \notin$ EXPTIL, and take some model $\mathcal{I}$ of $\mathcal{K}$. By Lemma 1, there is a homomorphism $f : \mathcal{C}_\mathcal{K} \to \mathcal{I}$. Define $\tau : \{0, \dots, 2^n - 1\} \times \{0, \dots n\} \to \Delta^\mathcal{I}$ as follows: $\tau(h_n \dots h_1, v_n \dots v_1) := f(\mathsf{b}\mathrm{H}^n_{h_n} \dots \mathrm{H}^1_{h_1} \mathrm{V}^n_{v_n} \dots \mathrm{V}^1_{v_1} \mathrm{HasCol})$ (again slightly abusing notation by applying working with binary encodings of numbers). There are three cases to consider:

- If there exists $(h_n \dots h_1, v_n \dots v_1)$ such that $\tau(h_n \dots h_1, v_n \dots v_1) \notin \{\mathsf{c} \mid c \in \mathcal{C}\}$, then this provides a match of q in $\mathcal{I}$ in which the subquery q^{col} is mapped as follows:

$$z^{col} \mapsto \tau(h_n \dots h_1, v_n \dots v_1) \qquad y^{col}_0 \mapsto \mathsf{b} \qquad y^{col}_1 \mapsto f(\mathsf{b}\mathrm{H}^n_{h_n}) \qquad \dots \qquad y^{col}_{2n} \mapsto f(\mathsf{b}\mathrm{H}^n_{h_n} \dots \mathrm{H}^1_{h_1} \mathrm{V}^n_{v_n} \dots \mathrm{V}^1_{v_1}),$$

  and whose restriction to the counting variables provides a new c-match.

- Otherwise, suppose there exists an element that is in the range of HasBit_0 that is not zero, or that is in the range of HasBit_1 but not equal to one, then this also provides a new c-match of q, in which either z^0 or z^1 is mapped to this element. Note that this kind of 'error' may occur at any level of the tree of positions. This is why we included the ABox assertion $\mathrm{HV}(\mathsf{b}, \mathsf{b})$, which makes it possible for us to loop as long as needed in order to obtain a sufficiently long chain of HV to satisfy the query q^0 or q^1.

- Else, since $(n, \mathcal{C}, \mathcal{H}, \mathcal{V}) \notin$ EXPTIL, there exist two adjacent positions with coordinates $(h_n \dots h_1, v_n \dots v_1)$ and $(h'_n \dots h'_1, v'_n \dots v'_1)$ such that $(\tau(h_n \dots h_1, v_n \dots v_1), \tau(h'_n \dots h'_1, v'_n \dots v'_1)) \in (\mathcal{C} \times \mathcal{C}) \setminus \mathcal{D}$, for $\mathcal{D}$ either $\mathcal{H}$ or $\mathcal{V}$. Letting k be the bit from which the encoding of the non-$\mathcal{D}$ coordinate differs, we obtain a new c-match for q, in which the subquery $q^{\mathcal{D}, (\tau(h_n \dots h_1, v_n \dots v_1), \tau(h'_n \dots h'_1, v'_n \dots v'_1)), k)}$ is satisfied by mapping $z^{\mathcal{D}, (\tau(h_n \dots h_1, v_n \dots v_1), \tau(h'_n \dots h'_1, v'_n \dots v'_1)), k)}$ to b.

In every case, there is an additional c-match for q. We thus obtain $(\boldsymbol{a}_\emptyset, [p+1, +\infty]) \in [q]^\mathcal{K}$. $\square$

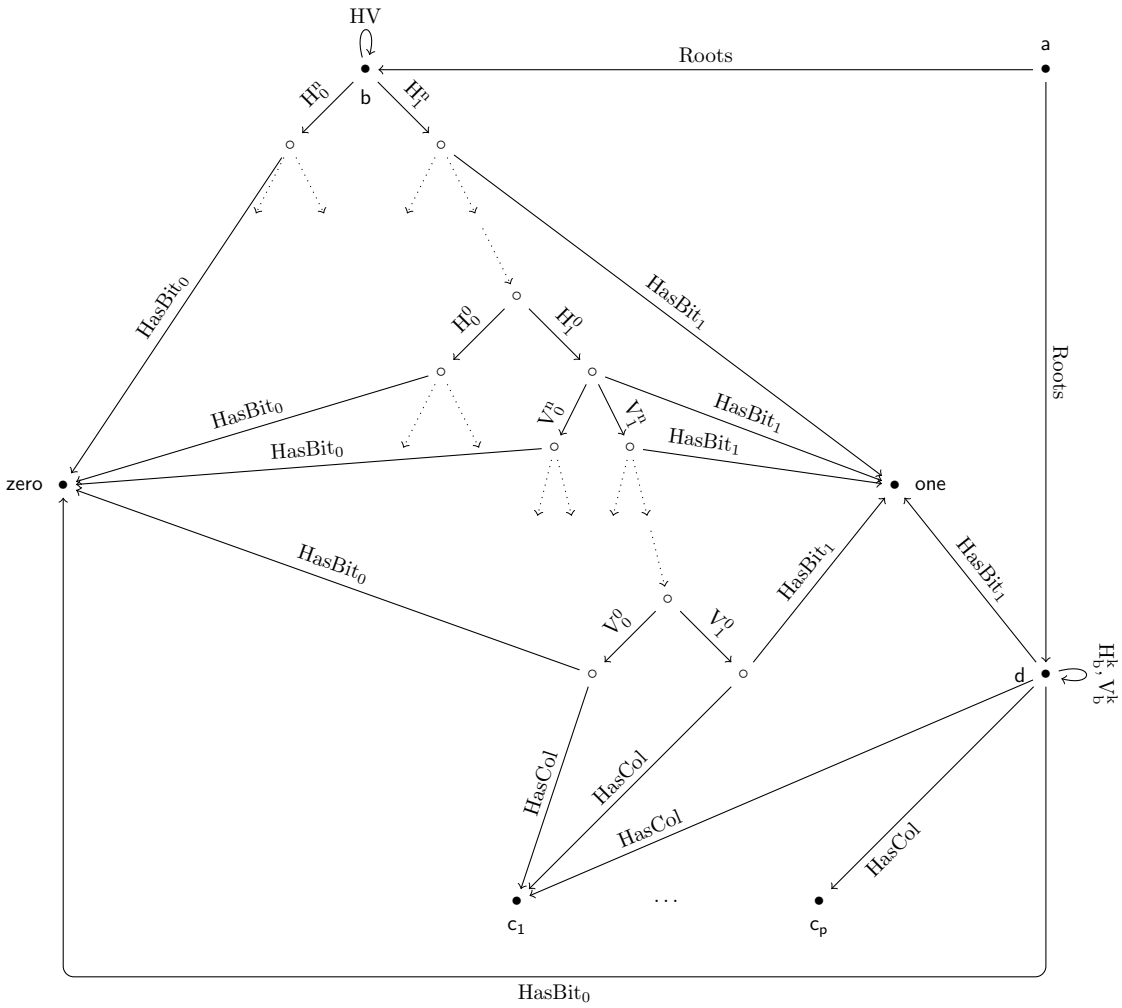

Figure 6: A part of $\mathcal{I}_\tau$ with $\tau(2^n-1, 2^n-2) = \tau(2^n-1, 2^n-1) = c_1$.

C Proofs for Section 6 (Exhaustive Rooted Counting CQs)

We start by recalling an important property of the canonical model construction for DL-Lite$_{\mathsf{core}}$ KBs.

Lemma 9. *For any role $R \in \mathsf{N}_\mathsf{R}^\pm$ and anonymous element d_1 in the canonical model $\mathcal{C}_\mathcal{K}$ of a DL-Lite$_{\mathsf{core}}$ KB $\mathcal{K}$, there is at most one element $d_2 \in \mathcal{C}_\mathcal{K}$ such that $(d_1, d_2) \in R^{\mathcal{C}_\mathcal{K}}$.*

Proof. We provide a proof for the sake of completeness. From the definition of $R^{\mathcal{C}_\mathcal{K}}$, if d_1 is an anonymous domain element and $(d_1, d_2) \in R^{\mathcal{C}_\mathcal{K}}$, then either:

- $d_1 = d_2\mathrm{S}^-$ for some role S such that $\mathcal{T} \models \mathrm{S} \sqsubseteq \mathrm{R}$, or
- $d_2 = d_1\mathrm{S}$ for some role S such that $\mathcal{T} \models \mathrm{S} \sqsubseteq \mathrm{R}$.

In both cases, since $\mathcal{T}$ is a DL-Lite$_{\mathsf{core}}$ TBox, the condition on S holds only if $\mathrm{S} = \mathrm{R}$. Moreover, we observe that if the first case holds, i.e., $d_1 = d_2\mathrm{R}^-$, then the definition of $\Delta^{\mathcal{C}_\mathcal{K}}$ prevents the creation of an element $d_1\mathrm{R}$. It follows that only one of the preceding cases can hold, and so there can be at most one d_2 with $(d_1, d_2) \in R^{\mathcal{C}_\mathcal{K}}$. □

Theorem 6. *For every DL-Lite$_{\mathsf{core}}$ KB $\mathcal{K}$ and exhaustive rooted CCQ q, it holds that $[q]^\mathcal{K} = [q]^{\mathcal{C}_\mathcal{K}}$.*

Proof. The general argument was given in the sketch. All that remains to show is that if σ_1, σ_2 are distinct matches of a exhaustive rooted CCQ q in $\mathcal{C}_\mathcal{K}$, then there exists a variable v such that $\sigma_1(v) \neq \sigma_2(v)$ and $\sigma_1(v), \sigma_2(v) \in \mathsf{Ind}(\mathcal{A})$.

Suppose for a contradiction that this is not the case. There there are distinct matches σ_1, σ_2 of q in $\mathcal{C}_\mathcal{K}$ such that for all variables v such that $\sigma_1(v) \neq \sigma_2(v)$, either $\sigma_1(v) \notin \mathsf{Ind}(\mathcal{A})$ or $\sigma_2(v) \notin \mathsf{Ind}(\mathcal{A})$. As q is exhaustive rooted, every variable v is connected to either an answer variable or individual in the Gaifman graph. Let $d(v)$ denote the length of the shortest path from

v to an answer variable of individual. Note that $d(v) = 0$ iff v is an answer variable. Since σ_1 and σ_2 are distinct, there exists a variable v such that $\sigma_1(v) \neq \sigma_2(v)$. Choose such a variable v^* with minimal d-value, i.e., if $d(u) < d(v^*)$, then $\sigma_1(u) = \sigma_2(u)$. By assumption, either $\sigma_1(v^*) \notin \mathsf{Ind}(\mathcal{A})$ or $\sigma_2(v^*) \notin \mathsf{Ind}(\mathcal{A})$. We'll suppose the former (the other case is treated analogously). Note that v^* cannot be an answer variable (else we would have $\sigma_1(v^*) \in \mathsf{Ind}(\mathcal{A})$). It follows that $d(v^*) > 0$, and so we can find another variable u^* and role name $R \in \mathsf{N}_\mathsf{R}^\pm$, with $d(u^*) = d(v^*) - 1$ and either $R(u^*, v^*) \in q$ or $R^-(v^*, u^*) \in q$ (recall that if $R = P^-$, then $R^- = P$). As σ_1 and σ_2 are matches of q in $\mathcal{C}_\mathcal{K}$, we therefore have $(\sigma_1(u^*), \sigma_1(v^*)) \in R^{\mathcal{C}_\mathcal{K}}$ and $(\sigma_2(u^*), \sigma_2(v^*)) \in R^{\mathcal{C}_\mathcal{K}}$. Moreover, since $d(u^*) < d(v^*)$, we have $\sigma_1(u^*) = \sigma_2(u^*)$. There are two cases to consider:

- Case 1: $\sigma_1(u^*) = \sigma_2(u^*) = c \in \mathsf{Ind}(\mathcal{A})$. From the proof of Lemma 9, we know that $\sigma_1(v^*) = cR$. The fact that $cR \in \Delta^{\mathcal{C}_\mathcal{K}}$ implies that there is no individual b such that $(c, b) \in R^{\mathcal{C}_\mathcal{K}}$. Thus, we must have $\sigma_2(v^*) = cR$, which yields $\sigma_1(v^*) = \sigma_2(v^*)$, contradicting our earlier assumption.
- Case 2: $\sigma_1(u^*) = \sigma_2(u^*) \notin \mathsf{Ind}(\mathcal{A})$. By Lemma 9, there is a unique element e such that $(\sigma_1(u^*), e) \in R^{\mathcal{C}_\mathcal{K}}$. We thus obtain $\sigma_1(v^*) = e = \sigma_2(v^*)$, a contradiction.

As both cases lead to a contradiction, it must therefore be the case that the statement holds. □

Theorem 7. *In DL-Lite$_{\mathsf{core}}$, exhaustive rooted CCQ answering is* PP*-complete w.r.t. combined complexity.*

We start by completing the argument for the PP upper bound.

Proof. Recall the algorithm described in the proof sketch.

Phase 1 The TM deterministically constructs the set $\Gamma_{q,\mathcal{T}}$ of words from Lemma 2.

Phase 2 The TM guesses a mapping σ of the variables in q to elements from $\{aw \mid a \in \mathsf{Ind}(\mathcal{A}), w \in \Gamma_{q,\mathcal{T}}\}$. It then compares m with the number $C = |\Gamma_{q,\mathcal{T}}|^{|q|}$ of possible mappings and proceeds accordingly:

- if $m \geq \frac{C}{2} + 1$, the TM guesses an integer i with $0 \leq i \leq 2m - 3$ and accepts iff σ is a c-match of $q(\boldsymbol{a})$ and $i < C$;
- if $m < \frac{C}{2} + 1$, the TM guesses an integer i with $0 \leq i \leq 2C - 2m + 1$ and accepts iff σ is c-match for $q(\boldsymbol{a})$ or $i < C - 2m + 2$.

Due to Theorem 6 and Lemma 2, an input is a 'yes' instance iff $q_{\boldsymbol{a}}^{\mathcal{C}_\mathcal{K}} \geq m$ (recall that $q_{\boldsymbol{a}}^{\mathcal{C}_\mathcal{K}}$ denotes the exact number of c-matches for $q(\boldsymbol{a})$ in $\mathcal{C}_\mathcal{K}$). To finish the proof of PP membership, we need to examine the number of accepting computation paths for the described TM and show that when $q_{\boldsymbol{a}}^{\mathcal{C}_\mathcal{K}} \geq m$, at least half of the computation paths accept, and when $q_{\boldsymbol{a}}^{\mathcal{C}_\mathcal{K}} < m$, less than half of the computation paths accept. Let us consider the two cases from Phase 2:

- If $m \geqslant \frac{C}{2} + 1$, then the number of accepting computation paths is $q_{\boldsymbol{a}}^{\mathcal{C}_\mathcal{K}} \times C$, corresponding to cases where the TM guesses a mapping that is a c-match, then guess a number $0 \leq i < C$. The total number of computation paths is $C \times (2m - 2)$, corresponding to a guess of one of the C mappings, then the guess of an integer $0 \leq i \leq 2m - 3$.
- If $m < \frac{C}{2} + 1$, then the number of accepting computation paths is

$$q_{\boldsymbol{a}}^{\mathcal{C}_\mathcal{K}} \times (2C - 2m + 2) + (C - q_{\boldsymbol{a}}^{\mathcal{C}_\mathcal{K}}) \times (C - 2m + 2) = C(C - 2m + q_{\boldsymbol{a}}^{\mathcal{C}_\mathcal{K}} + 2),$$

  corresponding to the sum of the number of cases where we guess a c-match followed by an integer $0 \leq i \leq 2C - 2m + 1$ and the number of cases where we guess a mapping that is not a c-match followed by an integer i with $0 \leq i < C - 2m + 2$. The total number of computation paths is $C \times (2C - 2m + 2)$ (guess one of the C mappings, then guess an integer $0 \leq i \leq 2C - 2m + 1$).

In both cases, it is easily verified that:

$$q_{\boldsymbol{a}}^{\mathcal{C}_\mathcal{K}} \geq m \iff \frac{\#\text{accepting computation paths}}{\#\text{possible computation paths}} > \frac{1}{2}.$$

(Note that in the first case, we always have $m \geq 2$, so the value $2m - 2$ in the denominator is positive, while in the second case, $C \geq 1$ implies that the value $(2C - 2m + 2)$ in the denominator is positive.) □

We next give the proof of PP-hardness.

Proof. We recall that the lower bound is by reduction from the following PP-complete problem: given a propositional formula ψ in CNF and number n, decide whether ψ has at least n satisfying assignments.

Consider an instance of this problem, given by the formula $\psi := \exists \boldsymbol{u} \bigwedge_{k=1}^{l} \xi_k$ (with ξ_k is a 3-clause) and number N. We consider the KB $\mathcal{K}_\psi = (\emptyset, \mathcal{A}_\psi)$, which has an empty TBox, and whose ABox $\mathcal{A}_\psi$ contains the following assertions:

- $\mathrm{Clause_k}(\mathsf{a}, \xi_\mathsf{k}^\mathsf{p})$ for each clause ξ_k and each $p \in \{1, ...7\}$, with each $\xi_\mathsf{k}^\mathsf{p}$ representing one of the 7 satisfying assignments for the clause ξ_k;
- $\mathrm{Asn}_1(\xi_\mathsf{k}^\mathsf{p}, \xi_k^p(\omega_k^1))$, $\mathrm{Asn}_2(\xi_\mathsf{k}^\mathsf{p}, \xi_k^p(\omega_k^2))$ and $\mathrm{Asn}_3(\xi_\mathsf{k}^\mathsf{p}, \xi_k^p(\omega_k^3))$ for each $p = 1, ...7$ and each clause ξ_k, where $\xi_k^p(\omega_k^i)$ is the truth value (true or false) assigned by ξ_k^p to the ith variable occurring in the kth clause.

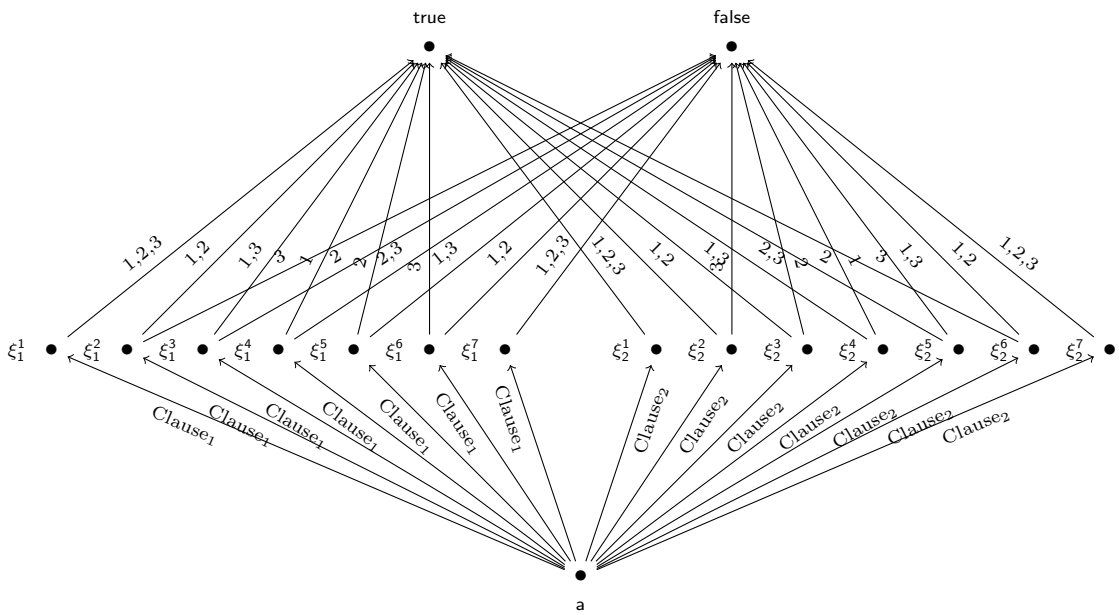

Figure 7: The canonical model $\mathcal{C}_{\mathcal{K}_\psi}$ with $\psi = (u_1 \vee \neg u_2 \vee \neg u_3) \wedge (\neg u_1 \vee u_3 \vee u_4)$

It may be helpful to refer to Figure 7, which depicts the canonical model of $\mathcal{C}_{\mathcal{K}_\psi}$ for an example formula ψ.

As for the query, we consider the following exhaustive rooted CCQ (depicted in Figure 8):

$$q_\psi := \exists z_{\xi_1} \ldots \exists z_{\xi_l}\ \exists z_{u_1} \ldots \exists z_{u_n} \bigwedge_{k=1}^{l} \left(\mathrm{Clause_k}(\mathsf{a}, z_{\xi_k}) \wedge \bigwedge_{i=1}^{3} \left(\mathrm{Asn_i}(z_{\xi_k}, z_{\omega_k^i}) \right) \right)$$

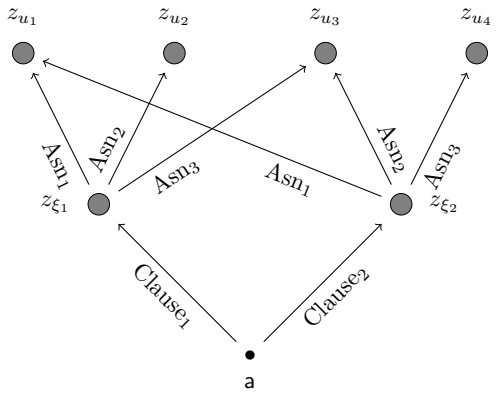

Figure 8: The query q_ψ with $\psi = (u_1 \vee \neg u_2 \vee \neg u_3) \wedge (\neg u_1 \vee u_3 \vee u_4)$

To complete the proof, we establish the following claim.

Claim. $(\emptyset, [N, +\infty]) \in [q_\psi]^{\mathcal{K}_\psi} \iff \psi$ has at least N satisfiying assignments

($\Rightarrow$) Assume $(\emptyset, [N, +\infty]) \in [q_\psi]^{\mathcal{K}_\psi}$. This implies in particular that there are N c-matches for q_ψ in $\mathcal{C}_{\mathcal{K}_\psi}$. Since the TBox is empty, the domain of $\mathcal{C}_{\mathcal{K}_\psi}$ is $\mathsf{Ind}(\mathcal{A}_\psi)$, and $\mathcal{C}_{\mathcal{K}_\psi}$ makes true precisely the assertions in $\mathcal{A}_\psi$. By examining q_ψ and $\mathcal{A}_\psi$, we see that each of the matches of q_ψ in $\mathcal{C}_{\mathcal{K}_\psi}$ maps each of the variables z_{u_i} to either true or false. We can therefore associate with each match σ the following truth assignment for the variables $u_1, \ldots, u_n$: $\tau_\sigma(u_i) = \sigma(z_{u_i})$. By further examining the definition of the individuals $\xi_\mathsf{k}^\mathsf{p}$ and the roles $\mathrm{Asn}_1, \mathrm{Asn}_2, \mathrm{Asn}_3$, it is easy to verify that each τ_σ is a satisfying assignment for ψ. Moreover, since we know we have N such assignments, it only remains to show that each match σ yields a distinct assignment τ_σ. To see why this is the case, observe that once we know the images of all of the variables z_{u_i}, there is a unique way of mapping the variables z_{ξ_p}. It follows that ψ has at least N satisfying assignments.

($\Leftarrow$) Assume ψ has at least N satisfying assignments. Therefore, we have $\tau_1, \dots \tau_N$ distinct assignments for $u_1, \dots u_n$ satisfying ψ. This ensures that, if we define $\sigma_{\tau_m}(z_{u_i}) = \tau_m(u_i)$, we can always extend the mapping $\sigma_{\tau_m}(z_{u_i})$ into a match for q_ψ, yielding N distinct matches. Note that this holds in any model since we only need the 'ABox part' of the model, hence $(\emptyset, [N, +\infty]) \in [q_\psi]^{\mathcal{K}_\psi}$.

□

Theorem 8. *In DL-Lite$_\mathcal{R}$, exhaustive rooted CCQ answering is* coNP*-complete w.r.t. data complexity.*

Proof. The main idea is the same as in proof of Theorem 4. However, due to the lack of existential variables, we can no longer 'reach' the colors without taking into account the paths leading to them. To address this difficulty, we translate into our context an idea from [Kostylev and Reutter, 2015], which takes advantage of role inclusions.

Starting from an instance $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ of the decision problem 3COL, we consider the ABox $\mathcal{A}_\mathcal{G}$ given by:

$$\begin{aligned}\mathcal{A}_\mathcal{G} =& \{\mathrm{Vertex}(\mathsf{a}, \mathsf{u}) \mid u \in \mathcal{V}\} \cup \{\mathrm{Edge}(\mathsf{u}_1, \mathsf{u}_2) \mid (u_1, u_2) \in \mathcal{E}\} \\ &\cup \{\mathrm{Vertex}(\mathsf{a}, \mathsf{a_v}), \mathrm{Edge}(\mathsf{a_v}, \mathsf{a_v}), \mathrm{HasCol}(\mathsf{a_v}, \mathsf{r})\} \\ &\cup \{\mathrm{Colors}(\mathsf{u}, \mathsf{r}) \mid u \in \mathcal{V}\} \cup \{\mathrm{Colors}(\mathsf{u}, \mathsf{g}) \mid u \in \mathcal{V}\} \cup \{\mathrm{Colors}(\mathsf{u}, \mathsf{b}) \mid u \in \mathcal{V}\}\end{aligned}$$

and the TBox $\mathcal{T} := \{\exists \mathrm{Vertex}^- \sqsubseteq \exists \mathrm{HasCol}, \mathrm{HasCol} \sqsubseteq \mathrm{Colors}\}$, and we denote by $\mathcal{K}_\mathcal{G} = (\mathcal{T}, \mathcal{A}_\mathcal{G})$ the resulting KB. A part of the canonical model of $\mathcal{K}_\mathcal{G}$ is depicted in Figure 9. As in the proof of Theorem 4, we use $\exists \mathrm{Vertex}^- \sqsubseteq \exists \mathrm{HasCol}$ to assign colors to vertices, and the more general role Colors will be used to detect colors.

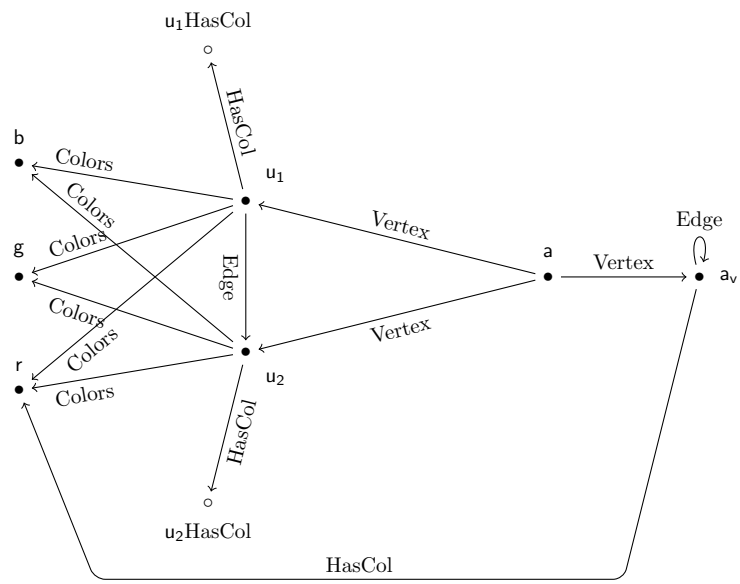

Figure 9: A part of $\mathcal{C}_{\mathcal{K}_\mathcal{G}}$ with $(u_1, u_2) \in \mathcal{E}$.

We consider the two following exhaustive rooted CCQs:

$$\begin{aligned} q^{edge} &= \exists z_c\, \exists z_1\, \exists z_2\, \mathrm{Vertex}(\mathsf{a}, z_1) \wedge \mathrm{Vertex}(\mathsf{a}, z_2) \wedge \mathrm{Edge}(z_1, z_2) \wedge \mathrm{HasCol}(z_1, z_c) \wedge \mathrm{HasCol}(z_2, z_c) \\ q^{col} &= \exists z_v\, \exists z\, \mathrm{Vertex}(\mathsf{a}, z_v) \wedge \mathrm{Colors}(z_v, z) \end{aligned}$$

and let q be the query obtained by taking the conjunction of these two queries and keeping all of the variables existentially quantified. The query q is displayed in Figure 10. Observe that while it is similar to the query from the proof of Theorem 4 (see Figure 3), the two existential variables in that query (y_c, y) have been replaced with counting variables (z_c, z_v), and one of the HasCol atom has been changed to a Colors atom.

It is not hard to see that $(\boldsymbol{a}_\emptyset, [3|\mathcal{V}| + 1, +\infty]) \in [q]^{\mathcal{K}_\mathcal{G}}$. Indeed, there are at least $3|\mathcal{V}|$ matches of q in any model $\mathcal{I}$ of $\mathcal{K}$, obtained as follows:

$$z_1, z_2 \mapsto \mathsf{a_v} \qquad z_c \mapsto \mathsf{r} \qquad z_v \mapsto \mathsf{u}\ (u \in \mathcal{V}) \qquad z \mapsto \mathsf{r} \mid \mathsf{g} \mid \mathsf{b}$$

and one additional match given by:

$$z_1, z_2, z_v \mapsto \mathsf{a_v} \qquad z_c, z \mapsto \mathsf{r}$$

To complete the proof, we establish the following claim.

Claim. $(\emptyset, [3|\mathcal{V}| + 2, +\infty]) \in [q]^{\mathcal{K}_\mathcal{G}} \iff \mathcal{G} \notin \mathsf{3COL}$.

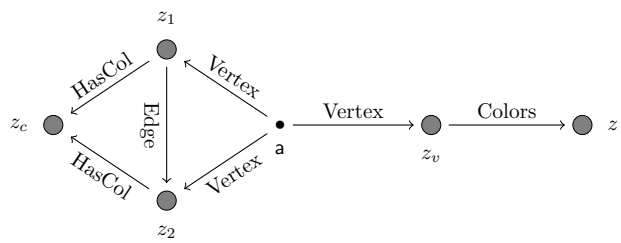

Figure 10: The exhaustive rooted CCQ q, which is the conjunction of q^{edge} (left part) and q^{col} (right part).

($\Rightarrow$) This direction is proven in the same manner as the claim in the proof of Theorem 4. We assume $(\boldsymbol{a}_\emptyset, [3|\mathcal{V}|+2, +\infty]) \in [q]^{\mathcal{K}_\mathcal{G}}$ and take a possible coloring $\tau : \mathcal{V} \to \{\mathsf{r}, \mathsf{g}, \mathsf{b}\}$. We then use τ to build a model $\mathcal{I}_\tau$ of $\mathcal{K}_\mathcal{G}$ and use the existence of an additional match σ to show that τ contains a monochromatic edge (hence $\mathcal{G} \notin \mathsf{3COL}$).

($\Leftarrow$) Assume $\mathcal{G} \notin \mathsf{3COL}$, and take some model $\mathcal{I}$ of $\mathcal{K}_\mathcal{G}$. By Lemma 1, there is a homomorphism $f : \mathcal{C}_{\mathcal{K}_\mathcal{G}} \to \mathcal{I}$. Define $\tau : \mathcal{V} \to \Delta^\mathcal{I}$ as follows: $\tau(u) = f(\mathrm{uHasCol})$. Note that τ is well defined, as the inclusion $\exists \mathrm{Vertex}^- \sqsubseteq \exists \mathrm{HasCol}$ ensures that there is an element uHasCol in $\mathcal{C}_{\mathcal{K}_\mathcal{G}}$. There are two cases to consider:

- If there exists $u \in \mathcal{V}$ such that $\tau(u) \notin \{\mathsf{r}, \mathsf{g}, \mathsf{b}\}$, then the axiom $\mathrm{HasCol} \sqsubseteq \mathrm{Colors}$ ensures $(\mathsf{u}^\mathcal{I}, \tau(u)) \in \mathrm{Colors}^\mathcal{I}$, which provides an additional match of q^{color} in $\mathcal{I}$ with $z \mapsto \tau(u)$ and $z_v \mapsto \mathsf{u}^\mathcal{I}$.
- Else, since $\mathcal{G} \notin \mathsf{3COL}$, there exists an edge $(u_1, u_2) \in \mathcal{E}$ such that $\tau(u_1) = \tau(u_2)$. It yields a new match given by:

$$z \mapsto \mathsf{r} \quad z_v \mapsto \mathsf{a_v} \quad z_1 \mapsto \mathsf{u_1} \quad z_2 \mapsto \mathsf{u_2} \quad z_c \mapsto \tau(u_1)\ (= \tau(u_2))$$

In both cases, there is an additional c-match for q. We thus obtain $(\boldsymbol{a}_\emptyset, [3|\mathcal{V}|+2, +\infty]) \in [q]^{\mathcal{K}_\mathcal{G}}$. □

Theorem 9. *In DL-Lite$_\mathcal{R}$, exhaustive rooted CCQ answering is in* coNEXP *w.r.t. combined complexity.*

Proof. We may assume w.l.o.g. that the initial homomorphism $f : \mathcal{C}_\mathcal{K} \to \mathcal{I}$ is chosen to respect the following property (★): if $w_1\mathrm{R}, w_2\mathrm{R} \in \Delta^{\mathcal{C}_\mathcal{K}}$ and $f(w_1) = f(w_2)$, then $f(w_1\mathrm{R}) = f(w_2\mathrm{R})$. Such a homomorphism can easily be built starting from an arbitrary homomorphism g, by choosing a 'main branch' whenever a choice is possible and copying from it. Formally, given a breadth-first ordering $\preceq$ of elements in $\Delta^{\mathcal{C}_\mathcal{K}}$, we start by setting $f := g$. Then, we explore the elements according to $\preceq$ and at each step, say at element w_1, we explore all elements w_2 such that $w_1 \preceq w_2$. If ever $f(w_1) = f(w_2)$, we redefine $f(w_2w) := f(w_1w)$ for every word w such that $w_1w, w_2w \in \Delta^{\mathcal{C}_\mathcal{K}}$. Since $\preceq$ is breadth-first, the image $f(w_1)$ will no longer be redefined after step w_1, which ensures the resulting homomorphism f is well-defined.

Recall that we introduced in the body of the paper a more refined notion of interleaving, which replaces the mapping f' by the following mapping f^*:

$$\begin{array}{rcl} f^* : \Delta^{\mathcal{C}_\mathcal{K}} & \to & \Delta^* \cup \Delta^{\mathcal{C}_\mathcal{K}} \\ a & \mapsto & f(a) \\ wR & \mapsto & \begin{cases} f(wR) & \text{if } f^*(w), f(wR) \in \Delta^* \\ f^*(w)R & \text{otherwise} \end{cases} \end{array}$$

We let $\mathcal{I}^*$ be the interpretation obtained by applying f^* to $\mathcal{C}_\mathcal{K}$. It is helpful to observe that $\mathcal{I}^*$ essentially coincides with the canonical model $\mathcal{C}^{\mathcal{K}^*}$ of the KB $\mathcal{K}^*$ whose TBox is $\mathcal{T}$ and whose ABox $\mathcal{A}^*$ consists of the facts from $\Delta^* \cap f^*(\mathcal{C}^\mathcal{K})$ (treating such elements as ABox individuals). More explicitly, $\mathcal{A}^*$ contains the concept assertion $\mathrm{A}(\mathsf{t})$ for each atomic concept $A \in \mathsf{N_C}$ and domain element $t \in \Delta^* \cap f^*(\mathcal{C}^\mathcal{K})$ such that $t \in f^*(\mathrm{A}^{\mathcal{C}_\mathcal{K}})$, and the role assertion $\mathrm{R}(\mathsf{t_1}, \mathsf{t_2})$ for each atomic role $\mathrm{R} \in \mathsf{N_R}$ and domain elements $t_1, t_2 \in \Delta^* \cap f^*(\mathcal{C}^\mathcal{K})$ such that $(t_1, t_2) \in f^*(\mathrm{R}^{\mathcal{C}_\mathcal{K}})$.

This alternative way of viewing $\mathcal{I}^*$, together with our assumption (★), makes clear that the following mapping is a homomorphism from $\mathcal{I}^*$ to $\mathcal{I}$:

$$\begin{array}{rcl} \rho^* : \Delta^{\mathcal{I}^*} & \to & \Delta^\mathcal{I} \\ f^*(d) & \mapsto & f(d). \end{array}$$

Indeed, (★) ensures the choice of d doesn't affect the image $f(d)$, thus ρ^* is well-defined. Formally, we proceed by induction on elements of $\mathcal{I}^*$. If $f^*(d) \in \Delta^*$, then by the definition of f^*, we must have either $d \in \mathsf{Ind}$ with $f^*(d) = f(d)$, or $d = w\mathrm{R}$ with $f^*(w) \in \Delta^*$, $f(d) \in \Delta^*$ and $f^*(d) = f(d)$. In both cases, $\rho^*(f^*(d)) = f(d) = f^*(d)$, which is independent from the choice of d. Otherwise, suppose ρ^* is well-defined for ω, and consider some $f^*(d) = \omega\mathrm{R} \notin \Delta^*$ and d' such that $f^*(d') = f^*(d) = \omega\mathrm{R}$. By the definition of f^*, we must have $d = w\mathrm{R}$ with $f^*(w) = \omega$, and same for d', that is, $d' = w'\mathrm{R}$ with $f^*(w') = \omega$. By our inductive assumption, we have $\rho^*(\omega) = f(w) = f(w')$. Property (★) now ensures $f(w\mathrm{R}) = f(w'\mathrm{R})$, that is $\rho^*(f^*(d)) = \rho^*(f^*(d'))$.

The mapping ρ^* being a homomorphism then follows from the definition of concept and role interpretations in $\mathcal{I}^*$. In particular, this means that $\mathcal{I}^*$ is a model of $\mathcal{K}$. Compared to the homomorphism ρ used to connect the interleaving $\mathcal{I}'$ with the countermodel $\mathcal{I}$, we have lost the property that $\rho^{-1}(\Delta^*) = \Delta^*$. Therefore, proving that $\mathcal{I}^*$ is a countermodel requires a different argument that exploits the exhaustive rooted assumption on the query.

Consider a match $\sigma : \boldsymbol{x} \cup \boldsymbol{z} \to \Delta^{\mathcal{I}^*}$ of $q(\boldsymbol{a})$ in $\mathcal{I}^*$. Let us first suppose that there is a counting variable $z \in \boldsymbol{z}$ such that $\sigma(z) \notin \Delta^*$, in which case we must have $\sigma(z) = \mathtt{t}w$ for some $\mathtt{t} \in \Delta^* \cap f^*(\mathcal{C}^{\mathcal{K}})$ and some non-empty word w. Since q is exhaustive rooted, all intermediate elements $\mathtt{t}w'$ with w' a prefix of w, must be reached by some other counting variables. In particular, one of these counting variables, say z_0, must map onto $\mathtt{t}w_0$, with w_0 the first symbol of w. From the definition of f^*, we also have a word $w_\mathtt{t}$ such that $f^*(w_\mathtt{t}) = f(w_\mathtt{t}) = \mathtt{t}$. However, via the homomorphism ρ^*, we can transform σ into a match $\rho^* \circ \sigma : \boldsymbol{x} \cup \boldsymbol{z} \to \Delta^{\mathcal{I}}$ in the original countermodel $\mathcal{I}$. In particular, we will have $\rho^*(\sigma(z_0)) = \rho^*(\mathtt{t}w_0) = \rho^*(f^*(w_\mathtt{t})w_0) = \rho^*(f^*(w_\mathtt{t}w_0)) = f(w_\mathtt{t}w_0)$. Thus, $f(w_\mathtt{t}w_0)$ belongs to the image of the match $\rho^* \circ \sigma$ in $\mathcal{I}$. From the definition of Δ^*, we can thus infer that $f(w_\mathtt{t}w_0) \in \Delta^*$. But since $f^*(w_\mathtt{t}) = \mathtt{t} \in \Delta^*$ and $f(w_\mathtt{t}w_0) \in \Delta^*$, we have $f^*(w_\mathtt{t}w_0) = f(w_\mathtt{t}w_0)$, and therefore the element $\mathtt{t}w_0$ is not introduced by f^* (it would contradict the property (★)), which contradicts z_0 mapping onto this element. Therefore, this situation, that is, the existence of a match in $\mathcal{I}^*$ with a counting variable mapping outside Δ^*, does not occur. Hence, we have $\sigma(\boldsymbol{x} \cup \boldsymbol{z}) \subseteq \Delta^*$. Then since $\rho^*_{|\Delta^*} = \mathsf{id}$, we have $\rho^* \circ \sigma = \sigma$, which shows that the mapping $\sigma \mapsto \rho^* \circ \sigma$ is injective. This means that $\mathcal{I}$ contains at least as many c-matches as $\mathcal{I}^*$, and since $\mathcal{I}$ is a countermodel, $\mathcal{I}^*$ must also be a countermodel.

As observed earlier, the obtained countermodel $\mathcal{I}^*$ has a particular structure: it can be seen as the canonical model of an ABox $\mathcal{A}^*$ whose size is polynomially bounded by the size of Δ^*, itself being single exponential in the size of the input. The modified interleaving thus allows us to improve the algorithm used in the general case. Indeed, it is now sufficient for a Turing machine to (i) guess an ABox of single-exponential size in $|\mathcal{K}|$ and $|q|$, and (ii) check that the canonical model of the guessed ABox and original TBox contains fewer c-matches for $q(\boldsymbol{a})$ than the integer provided as input. Importantly, due to our assumption that q is exhaustive rooted, matches cannot reach elements in the canonical model that have depth greater than $|q|$. There are thus only single-exponentially many domain elements that may appear in the image of a match, and so it is possible to enumerate and count all matches in single-exponential time w.r.t. $|\mathcal{K}|$ and $|q|$. □

D Proofs for Section 7 (Best Certain Answers)

Theorem 10. *The following problem is* DP*-hard in data complexity: given a DL-Lite*$_{\mathsf{core}}$ *KB* $\mathcal{K} = (\mathcal{T}, \mathcal{A})$*, rooted CCQ* q*, tuple* $\boldsymbol{a}$*, and number* m*, decide whether* $m = \min_{\mathcal{I} \models \mathcal{K}} q^{\mathcal{I}}_{\boldsymbol{a}}$*.*

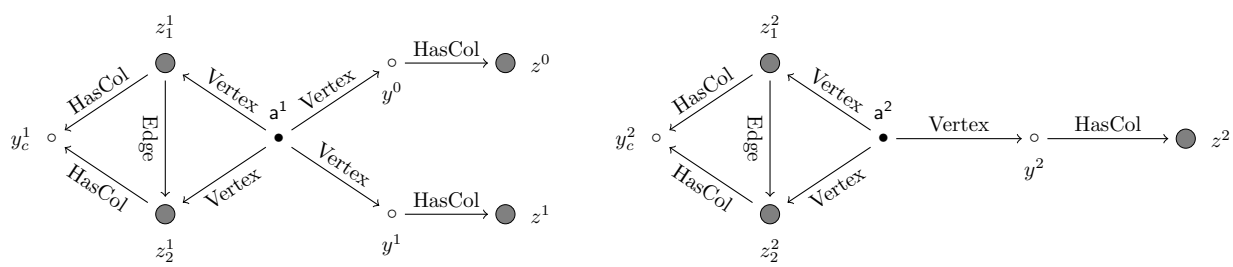

Figure 11: The rooted CCQ q, which is the conjunction of q_1, q_0 (left part) and q_2 (right part).

Proof. We provide more details on the case analysis mentioned in the body of the paper. In what follows, $\mathcal{I}$ will denote an arbitrary model of $\mathcal{K} = (\mathcal{T}_{\mathsf{col}}, \mathcal{A}_{\mathcal{G}_1} \cup \mathcal{A}_{\mathcal{G}_2})$. We first remark that every model contains the c-matches given by:

$$z^0, z^1 \mapsto \mathsf{r}^1 \mid \mathsf{g}^1 \mid \mathsf{b}^1 \quad z_1^1, z_2^1 \mapsto \mathsf{a_v}^1 \qquad z^2 \mapsto \mathsf{r}^2 \mid \mathsf{g}^2 \mid \mathsf{b}^2 \quad z_1^2, z_2^2 \mapsto \mathsf{a_v}^2$$

Hence, $q^{\mathcal{I}}_{\emptyset} \geq 3 \times 3 \times 1 \times 3 \times 1 = 27$.

In what follows, we will use $\mathcal{I}^{\mathcal{G}}_{\tau}$ to denote a minimal model of $\mathcal{K}_{\mathcal{G}}$ complying with a given coloring τ of a graph $\mathcal{G}$, constructed as in the proof of Theorem 4. We observe that if τ_1 and τ_2 are respectively colorings for the graphs $\mathcal{G}_1$ and $\mathcal{G}_2$, then the interpretation $\mathcal{I}^{\mathcal{G}_1}_{\tau_1} \cup \mathcal{I}^{\mathcal{G}_2}_{\tau_2}$ which is the disjoint union of $\mathcal{I}^{\mathcal{G}_1}_{\tau_1}$ and $\mathcal{I}^{\mathcal{G}_2}_{\tau_2}$ is a model of the considered KB $\mathcal{K}$. We use such models to establish the minimum number of c-matches in the four different cases:

- $\mathcal{G}_1, \mathcal{G}_2 \in$ 3COL: We have already seen that every model of $\mathcal{K}$ contains at least 27 c-matches. Let τ_1 (resp. τ_2) be a 3-coloring for $\mathcal{G}_1$ (resp. $\mathcal{G}_2$). Then the model $\mathcal{I}^{\mathcal{G}_1}_{\tau_1} \cup \mathcal{I}^{\mathcal{G}_2}_{\tau_2}$ has exactly 27 c-matches.
- $\mathcal{G}_1 \in$ 3COL, $\mathcal{G}_2 \notin$ 3COL: As $\mathcal{G}_2$ is not 3-colorable, the part of $\mathcal{I}$ describing $\mathcal{G}_2$ must either introduce a fourth color, providing a new value for z^2 (hence at least $3 \times 3 \times 1 \times 4 \times 1 = 36$ c-matches), or contain a monochromatic edge, providing another possible value for (z_1^2, z_2^2) (hence at least $3^2 \times 1 \times 3 \times 2 = 54$ c-matches). Therefore, every model contains at least 36 c-matches for q. To show we cannot ensure more than 36 c-matches, let τ_1 (resp. τ_2) be a 3-coloring (resp. 4-coloring) for $\mathcal{G}_1$ (resp $\mathcal{G}_2$). Then $\mathcal{I}^{\mathcal{G}_1}_{\tau_1} \cup \mathcal{I}^{\mathcal{G}_2}_{\tau_2}$ has exactly 36 c-matches.

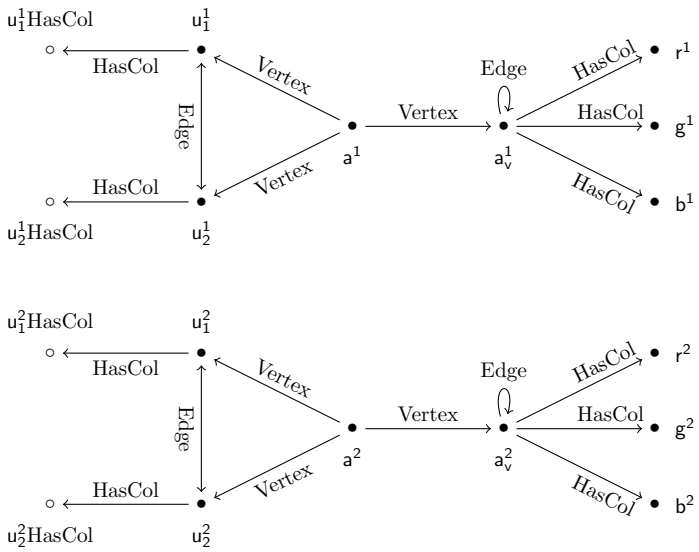

Figure 12: A part of $\mathcal{C}_{\mathcal{K}_{(\mathcal{G}_1,\mathcal{G}_2)}}$ with $(u_1^1, u_2^1) \in \mathcal{E}_1$ and $(u_1^2, u_2^2) \in \mathcal{E}_2$.

- $\mathcal{G}_1 \notin$ 3COL, $\mathcal{G}_2 \in$ 3COL: The part of $\mathcal{I}$ describing $\mathcal{G}_1$ must introduce either a fourth color, providing a new value for z^0 and z^1 (hence at least $4 \times 4 \times 1 \times 3 \times 1 = 48$ c-matches), or contain a monochromatic edge, providing another possible value for (z_1^1, z_2^1) (hence at least $3 \times 3 \times 2 \times 3 \times 1 = 54$ c-matches). It follows that every model contains at least 48 c-matches. To show this is the best value that can be attained, let τ_1 (resp. τ_2) be a 4-coloring (resp. 3-coloring) for $\mathcal{G}_1$ (resp. $\mathcal{G}_2$). Then $\mathcal{I}_{\tau_1}^{\mathcal{G}_1} \cup \mathcal{I}_{\tau_2}^{\mathcal{G}_2}$ has exactly 48 c-matches.
- $\mathcal{G}_1, \mathcal{G}_2 \notin$ 3COL: For each of the two graphs, $\mathcal{I}$ must introduce either a fourth color or a monochromatic edge. There are four cases to consider:

	Fourth color in $\mathcal{G}_1$'s part	Monochromatic edge in $\mathcal{G}_1$'s part
Fourth color in $\mathcal{G}_2$'s part	$4^2 \times 1 \times 4 \times 1 = 64$	$3^2 \times 2 \times 4 \times 1 = 72$
Monochromatic edge in $\mathcal{G}_2$'s part	$4^2 \times 1 \times 3 \times 2 = 96$	$3^2 \times 2 \times 3 \times 2 = 108$

  We therefore see that every model contains at least 64 c-matches of q. To realize the minimal number, we let τ_1 (resp. τ_2) be a 4-coloring (resp. 4-coloring) for $\mathcal{G}_1$ (resp. $\mathcal{G}_2$) and observe that $\mathcal{I}_{\tau_1}^{\mathcal{G}_1} \cup \mathcal{I}_{\tau_2}^{\mathcal{G}_2}$ has exactly 64 c-matches.

This completes the case analysis, the rest of the argument is contained in the proof sketch. □

DP-hardness for Count queries from [Kostylev and Reutter, 2015]

Proof. We recall that the Count queries from [Kostylev and Reutter, 2015] are obtaining by requiring all of the non-answer variables to be counting variables. The queries from the preceding reduction do not satisfy this restriction, as they use existential variables, but we can modify the reduction in order to make it work for such queries.

In the modified reduction, each vertex is described in the ABox with a specific concept, either Vertex_1 or Vertex_2 depending on which graph it appears in. The TBox contains the following axioms:

$$\{\mathrm{Vertex}_1 \sqsubseteq \exists \mathrm{HasCol}_1, \mathrm{Vertex}_2 \sqsubseteq \exists \mathrm{HasCol}_2, \exists \mathrm{HasCol}_1^- \sqsubseteq \mathrm{Color}_1, \exists \mathrm{HasCol}_2^- \sqsubseteq \mathrm{Color}_2\}.$$

The subqueries q_i^{edge} and q_i^{col} are then modified as follows for $i \in \{1, 2\}$:

$$\begin{aligned} q_i^{edge} =& \exists z_c^i\ \exists z_1^i\ \exists z_2^i\ \mathrm{Edge}(z_1^i, z_2^i) \wedge \mathrm{HasCol_i}(z_1^i, z_c^i) \wedge \mathrm{HasCol_i}(z_2^i, z_c^i) \\ q_i^{col} =& \exists z^i\ \mathrm{Color_i}(z^i) \end{aligned}$$

and q_0^{col} is redefined as: $\exists z^0\ \mathrm{Color}_1(z^0)$.

It is easily verified that after these modifications, the query q now corresponds to a Count query as defined in [Kostylev and Reutter, 2015]. The query q is displayed in Figure 13, and the slightly ajusted canonical model $\mathcal{C}_{\mathcal{K}_{(\mathcal{G}_1,\mathcal{G}_2)}}$ is displayed in Figure 14.

We can then redo the argument in the same manner as before, and the case analysis will give rise to precisely the same numbers of c-matches.

□

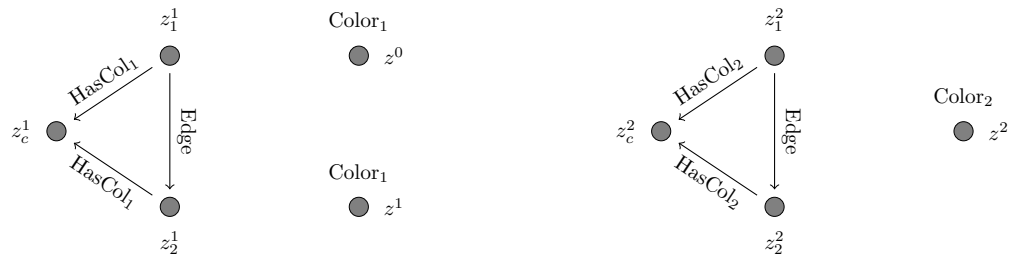

Figure 13: The Count CQ q, which is the conjunction of $q_1^{edge}, q_1^{col}, q_0^{col}$ (left part) and q_2^{edge}, q_2^{col} (right part).

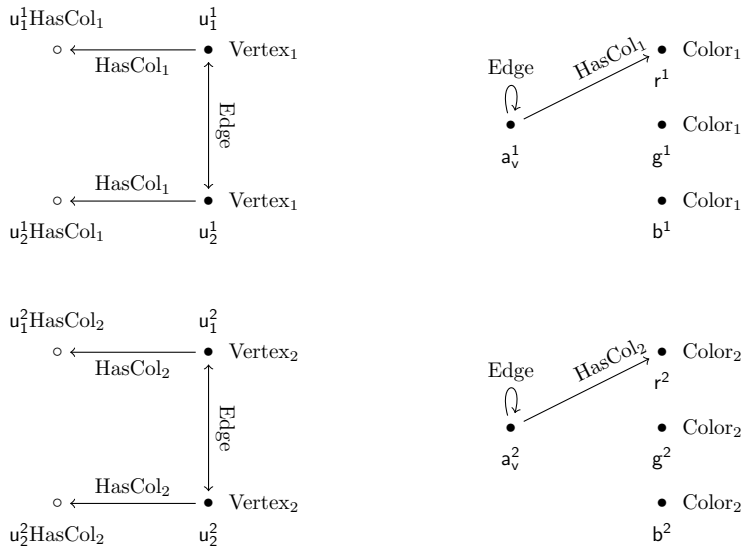

Figure 14: A part of $\mathcal{C}_{\mathcal{K}_{(\mathcal{G}_1,\mathcal{G}_2)}}$ with $(u_1^1, u_2^1) \in \mathcal{E}_1$ and $(u_1^2, u_2^2) \in \mathcal{E}_2$.

DP-hardness for Cntd queries from [Kostylev and Reutter, 2015]

Proof. We recall that the Cntd queries from [Kostylev and Reutter, 2015] correspond to CCQs with exactly one counting variable. As in the previous reductions, we aim to force additional matches whenever an input graph is not 3-colorable, and the challenge is to track of the amount of colors used to color the two graphs.

Having only a single counting variable forces us to count colors used for $\mathcal{G}_1$ in exactly the same as we count those used for $\mathcal{G}_2$. In particular, the asymmetry we introduced in the query must now be introduced into the ABox. This is done by considering a copy of our first graph. However, this is not enough as two different graphs could use the same additional color, making it impossible to detect with our single counting variable that both graphs are using more than three colors. Therefore, we will provide a set of basic colors *for each graph* and additionally check whether a graph uses a color that is intended for another graph. Concretely, we achieve this by connecting vertices from different graphs using a new role Diff, and by adding a new subquery that will generate new c-matches whenever two vertices connected by Diff use the same color.

Let us now give a more formal description of the construction. As mentioned earlier, we will introduce a copy $\mathcal{G}_0 = (\mathcal{V}_0, \mathcal{E}_0)$ of the graph $\mathcal{G}_1$. Without loss of generality, we can assume that $\mathcal{V}_0 \cap \mathcal{V}_1 \cap \mathcal{V}_2 = \emptyset$. As ABox individuals, we will use:

- an individual name u for each $u \in \mathcal{V}_0 \cup \mathcal{V}_1 \cup \mathcal{V}_2$, to represent our graphs;
- individuals $\mathsf{r}_0, \mathsf{g}_0, \mathsf{b}_0$ (resp. $\mathsf{r}_1, \mathsf{g}_1, \mathsf{b}_1$ and $\mathsf{r}_2, \mathsf{g}_2, \mathsf{b}_2$), intended to color $\mathcal{G}_0$ (resp. $\mathcal{G}_1$ and $\mathcal{G}_2$);
- auxiliary individuals for vertices (a_0, a_1, a_2, c, d, e) and auxiliary individuals for colors (r, g, b).

We then consider the following ABox:

$$\begin{aligned}\mathcal{A}_{(\mathcal{G}_1,\mathcal{G}_2)} = &\{\mathrm{Vertex}(\mathsf{u}) \mid u \in \mathcal{V}_0 \cup \mathcal{V}_1 \cup \mathcal{V}_2\}\\ &\cup \{\mathrm{Edge}(\mathsf{u}_1, \mathsf{u}_2) \mid (u_1, u_2) \in \mathcal{E}_0 \cup \mathcal{E}_1 \cup \mathcal{E}_2\}\\ &\cup \{\mathrm{Edge}(\mathsf{a}_0, \mathsf{a}_0), \mathrm{Edge}(\mathsf{a}_1, \mathsf{a}_1), \mathrm{Edge}(\mathsf{a}_2, \mathsf{a}_2), \mathrm{Edge}(\mathsf{c}, \mathsf{c}), \mathrm{Edge}(\mathsf{d}, \mathsf{d})\}\\ &\cup \{\mathrm{Diff}(\mathsf{u}_1, \mathsf{u}_2) \mid u_1 \in \mathcal{V}_i, u_2 \in \mathcal{V}_j, i \neq j\}\end{aligned}$$

$$
\begin{aligned}
&\cup \{\mathrm{Diff}(\mathsf{u}, \mathsf{a}_i) \mid u \in \mathcal{V}_j, i, j \in \{0,1,2\}, i \neq j\} \\
&\cup \{\mathrm{Diff}(\mathsf{a}_0, \mathsf{a}_0), \mathrm{Diff}(\mathsf{a}_1, \mathsf{a}_1), \mathrm{Diff}(\mathsf{a}_2, \mathsf{a}_2), \mathrm{Diff}(\mathsf{c}, \mathsf{c}), \mathrm{Diff}(\mathsf{e}, \mathsf{e})\} \\
&\cup \{\mathrm{Aux_e}(\mathsf{a}_0, \mathsf{a}_0), \mathrm{Aux_e}(\mathsf{a}_1, \mathsf{a}_1), \mathrm{Aux_e}(\mathsf{a}_2, \mathsf{a}_2), \mathrm{Aux_e}(\mathsf{d}, \mathsf{d})\} \\
&\cup \{\mathrm{Aux_e}(\mathsf{e}, \mathsf{u}) \mid u \in \mathcal{V}_0 \cup \mathcal{V}_1 \cup \mathcal{V}_2\} \\
&\cup \{\mathrm{Aux_e}(\mathsf{u}, \mathsf{c}) \mid u \in \mathcal{V}_0 \cup \mathcal{V}_1 \cup \mathcal{V}_2\} \\
&\cup \{\mathrm{Aux_d}(\mathsf{a}_0, \mathsf{a}_0), \mathrm{Aux_d}(\mathsf{a}_1, \mathsf{a}_1), \mathrm{Aux_d}(\mathsf{a}_2, \mathsf{a}_2), \mathrm{Aux_d}(\mathsf{e}, \mathsf{e})\} \\
&\cup \{\mathrm{Aux_d}(\mathsf{d}, \mathsf{u}) \mid u \in \mathcal{V}_0 \cup \mathcal{V}_1 \cup \mathcal{V}_2\} \\
&\cup \{\mathrm{Aux_d}(\mathsf{u}, \mathsf{c}) \mid u \in \mathcal{V}_0 \cup \mathcal{V}_1 \cup \mathcal{V}_2\} \\
&\cup \{\mathrm{HasCol}(\mathsf{a}_i, t) \mid t \in \{\mathsf{r}_i, \mathsf{g}_i, \mathsf{b}_i\}, i \in \{0,1,2\}\} \\
&\cup \{\mathrm{HasCol}(\mathsf{c}, \mathsf{r}), \mathrm{HasCol}(\mathsf{d}, \mathsf{r}), \mathrm{HasCol}(\mathsf{d}, \mathsf{g}), \mathrm{HasCol}(\mathsf{d}, \mathsf{b}), \mathrm{HasCol}(\mathsf{e}, \mathsf{r}), \mathrm{HasCol}(\mathsf{e}, \mathsf{g}), \mathrm{HasCol}(\mathsf{e}, \mathsf{b})\}.
\end{aligned}
$$

and the TBox $\mathcal{T} := \{\mathrm{Vertex} \sqsubseteq \exists \mathrm{HasCol}\}$. We denote by $\mathcal{K}_\mathcal{G} = (\mathcal{T}, \mathcal{A})$ the resulting KB. A part of the canonical model of $\mathcal{K}$ is depicted in Figure **??**.

We consider the three following subqueries:

$$
\begin{aligned}
q^{diff}(y) &= \exists y_1^d\, \exists y_2^d\, \exists y_c^d\ \mathrm{Aux_d}(y, y_1^d) \wedge \mathrm{Diff}(y_1^d, y_2^d) \wedge \mathrm{HasCol}(y_1^d, y_c^d) \wedge \mathrm{HasCol}(y_2^d, y_c^d) \\
q^{edge}(y) &= \exists y_1^e\, \exists y_2^e\, \exists y_c^e\ \mathrm{Aux_e}(y, y_1^e) \wedge \mathrm{Edge}(y_1^e, y_2^e) \wedge \mathrm{HasCol}(y_1^e, y_c^e) \wedge \mathrm{HasCol}(y_2^e, y_c^e) \\
q^{col}(y) &= \exists z\ \mathrm{HasCol}(y, z)
\end{aligned}
$$

and let $q = \exists y\ q^{diff}(y) \wedge q^{edge} \wedge q^{col}$ be the complete CCQ, which corresponds to a Cntd query class as there is only one counting variable z. The query q is displayed in Figure 15.

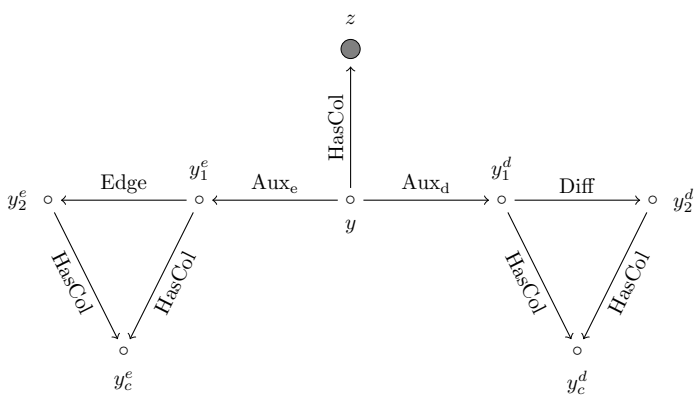

Figure 15: The Cntd CQ q, which is the conjunction of q^{edge} (left part), q^{diff} (right part) and q^{col} (upper part).

Claim: $(\boldsymbol{a}_\emptyset, [10, +\infty]) \in [q]^\mathcal{K}$ iff $\mathcal{G}_1 \in \mathsf{3COL}$ and $\mathcal{G}_2 \notin \mathsf{3COL}$.

We prove this claim using the following case analysis:

	$\mathcal{G}_1 \in \mathsf{3COL}$	$\mathcal{G}_1 \notin \mathsf{3COL}$
$\mathcal{G}_2 \in \mathsf{3COL}$	$9\ (= 3+3+3)$	$11\ (= 4+4+3)$
$\mathcal{G}_2 \notin \mathsf{3COL}$	$10\ (= 3+3+4)$	$12\ (= 4+4+4)$

To obtain the values in the preceding table, consider an arbitrary model $\mathcal{I}$ of $\mathcal{K}$, along with a homomorphism $f : \mathcal{C}_\mathcal{K} \to \mathcal{I}$. First observe that there are always 9 c-matches, which are obtained from the matches given by:

$$z \mapsto \mathsf{r}_i \mid \mathsf{g}_i \mid \mathsf{b}_i \qquad y, y_1^e, y_2^e, y_1^d, y_2^d \mapsto \mathsf{a}_i \qquad y_c^e, y_c^d \mapsto \mathsf{r}_i \qquad (i \in \{0,1,2\})$$

Hence $q_\emptyset^\mathcal{I} \geq 3+3+3 = 9$.

Furthermore, let us define $\tau_\mathcal{I} : \mathcal{V}_0 \cup \mathcal{V}_1 \cup \mathcal{V}_2 \to \Delta^\mathcal{I}$ as follows: $\tau_\mathcal{I}(u) = f(\mathrm{uHasCol})$. We'll use the notation $\tau_\mathcal{I}(\mathcal{V}_i)$ to refer to the set $\{\tau_\mathcal{I}(u) \mid u \in \mathcal{V}_i\}$. Notice that, if $\tau_\mathcal{I}(\mathcal{V}_i) \cap \tau_\mathcal{I}(\mathcal{V}_j) \neq \emptyset$ with $i \neq j$, that is, we have $u \in \mathcal{G}_i, v \in \mathcal{G}_j$ with $i \neq j$ and $\tau_\mathcal{I}(u) = \tau_\mathcal{I}(v)$, then we have 3 additional c-matches corresponding to the matches given by:

$$z, y_c^e \mapsto \mathsf{r} \mid \mathsf{g} \mid \mathsf{b} \qquad y, y_1^e, y_2^e \mapsto \mathsf{d} \qquad y_1^d \mapsto \mathsf{u} \qquad y_2^d \mapsto \mathsf{v} \qquad y_c^d \mapsto \tau_\mathcal{I}(u)$$

Therefore, in such a model $\mathcal{I}$, we have $q_{\emptyset}^{\mathcal{I}} \geq 9 + 3 = 12$, and thus sufficiently many c-matches w.r.t. the numbers in the table. We will therefore assume in the following that $\tau_{\mathcal{I}}(\mathcal{V}_i) \cap \tau_{\mathcal{I}}(\mathcal{V}_j) = \emptyset$ for $i \neq j$ (assumption (i)).

The same applies in the case where $\tau_{\mathcal{I}}(\mathcal{V}_i) \cap \{\mathsf{r}_j, \mathsf{g}_j, \mathsf{b}_j\} \neq \emptyset$ for $i \neq j$, as one can exhibit the same three additional c-matches by replacing the individual v by a_j in the latter definition of matches. Therefore, we can also assume in what follows that $\tau_{\mathcal{I}}(\mathcal{V}_i) \cap \{\mathsf{r}_j, \mathsf{g}_j, \mathsf{b}_j\} = \emptyset$ for all $i \neq j$ (assumption (ii)).

Finally, notice that if $\tau_{\mathcal{I}}$ introduces a monochromatic edge, i.e. an edge $(u, v) \in \mathcal{E}_0 \cup \mathcal{E}_1 \cup \mathcal{E}_2$ such that $\tau_{\mathcal{I}}(u) = \tau_{\mathcal{I}}(v)$, we again have 3 additional c-matches obtained from the matches given by:

$$z, y_c^d \mapsto \mathsf{r} \mid \mathsf{g} \mid \mathsf{b} \qquad y, y_1^d, y_2^d \mapsto \mathsf{e} \qquad y_1^e \mapsto \mathsf{u} \qquad y_2^e \mapsto \mathsf{v} \qquad y_c^e \mapsto \tau_{\mathcal{I}}(u)$$

Therefore, we can also restrict our attention to models without monochromatic edges (assumption (iii)). Any model that satisfies properties (i), (ii) and (iii) will be called *non-trivial*.

We now proceed to consider the four cases. In each case, the minimal amount of c-matches is obtained by exhibiting a model built from colorings for each graph that use a minimal amount of colors. The only important difference w.r.t preceding reductions is that when more than one graph utilizes a fourth color, we need to use distinct fourth colors for each graph. We now complete the proof by showing that every non-trivial model has at least the number of c-matches as listed in the table.

- $\mathcal{G}_1, \mathcal{G}_2 \in \mathsf{3COL}$: We have already seen that every model contains at least 9 c-matches.
- $\mathcal{G}_1 \notin \mathsf{3COL}, \mathcal{G}_2 \in \mathsf{3COL}$: Since $\mathcal{G}_0$ and $\mathcal{G}_1$ are not 3-colorable, any non-trivial model $\mathcal{I}$ must satisfy $\tau_{\mathcal{I}}(\mathcal{V}_0) \geq 4$ and $\tau_{\mathcal{I}}(\mathcal{V}_1) \geq 4$, due to assumption (iii). In particular, we have a vertex $u_0 \in \mathcal{V}_0$ (resp. $u_1 \in \mathcal{V}_1$) such that $\tau_{\mathcal{I}}(u_0) \notin \{\mathsf{r}_0, \mathsf{g}_0, \mathsf{b}_0\}$ (resp. $\tau_{\mathcal{I}}(u_1) \notin \{\mathsf{r}_1, \mathsf{g}_1, \mathsf{b}_1\}$). This yields the following matches:

  $$z \mapsto \tau_{\mathcal{I}}(u_i) \qquad y \mapsto \mathsf{u_i} \qquad y_1^e, y_2^e, y_1^d, y_2^d \mapsto \mathsf{c} \qquad y_c^e, y_c^d \mapsto \mathsf{r} \qquad (i \in \{0, 1\})$$

  which give rise to two new c-matches because of assumptions (i) (ensuring the two colors $\tau_{\mathcal{I}}(u_0)$ and $\tau_{\mathcal{I}}(u_1)$ are different) and (ii) (ensuring $\tau_{\mathcal{I}}(u_0)$ and $\tau_{\mathcal{I}}(u_1)$ are different from the colors in the 9 basic c-matches). Hence, every non-trivial model contains at least 11 c-matches.
- $\mathcal{G}_1 \in \mathsf{3COL}, \mathcal{G}_2 \notin \mathsf{3COL}$: Since $\mathcal{G}_2$ is not in $\mathsf{3COL}$, any non-trivial model $\mathcal{I}$ must satisfy $\tau_{\mathcal{I}}(\mathcal{V}_2) \geq 4$ because of assumption (iii). In particular, we have a vertex $u_2 \in \mathcal{V}_2$ such that $\tau_{\mathcal{I}}(u_2) \notin \{\mathsf{r}_2, \mathsf{g}_2, \mathsf{b}_2\}$. This provides a new match given by:

  $$z \mapsto \tau_{\mathcal{I}}(u_2) \qquad y \mapsto \mathsf{u_2} \qquad y_1^e, y_2^e, y_1^d, y_2^d \mapsto \mathsf{c} \qquad y_c^e, y_c^d \mapsto \mathsf{r}$$

  which gives rise to a new c-match because of the assumption (ii) (which ensures $\tau_{\mathcal{I}}(u_2)$ is different from the colors in the 9 basic c-matches). Hence, every non-trivial model contains at least 10 c-matches.
- $\mathcal{G}_1, \mathcal{G}_2 \notin \mathsf{3COL}$: We can proceed similarly to the two previous cases to exhibit $u_0 \in \mathcal{V}_0, u_1 \in \mathcal{V}_1, u_2 \in \mathcal{V}_2$ that are assigned new colors, providing three new matches given by:

  $$z \mapsto \tau_{\mathcal{I}}(u_i) \qquad y \mapsto \mathsf{u_i} \qquad y_1^e, y_2^e, y_1^d, y_2^d \mapsto \mathsf{c} \qquad y_c^e, y_c^d \mapsto \mathsf{r} \qquad (i \in \{0, 1, 2\})$$

  which give rise to three new c-matches because of assumptions (i) (the colors $\tau_{\mathcal{I}}(u_0), \tau_{\mathcal{I}}(u_1), \tau_{\mathcal{I}}(u_2)$ are all different) and (ii) (they are also different from the colors in the 9 basic c-matches). Hence, we have that every non-trivial model contains at least 12 matches.

□