Description Logics with Abstraction and Refinement

Carsten Lutz[1,2], **Lukas Schulze**[1]
[1] Department of Computer Science, Leipzig University, Germany
[2]Center for Scalable Data Analytics and Artificial Intelligence (ScaDS.AI)
{clu, lschulze}@informatik.uni-leipzig.de

Abstract

Ontologies often require knowledge representation on multiple levels of abstraction, but description logics (DLs) are not well-equipped for supporting this. We propose an extension of DLs in which abstraction levels are first-class citizens and which provides explicit operators for the abstraction and refinement of concepts and roles across multiple abstraction levels, based on conjunctive queries. We prove that reasoning in the resulting family of DLs is decidable while several seemingly harmless variations turn out to be undecidable. We also pinpoint the precise complexity of our logics and several relevant fragments.

1 Introduction

Abstraction and refinement is an important topic in many subfields of computer science such as systems verification (Dams and Grumberg 2018). The same is true for ontology design because ontologies often refer to different levels of abstraction (or equivalently, levels of granularity). To name only one example, the widely known medical ontology SNOMED CT contains the concepts Arm, Hand, Finger, Phalanx, Osteocyte, and Mitochondrion which intuitively all belong to different (increasingly finer) levels of abstraction (Stearns et al. 2001). Existing ontology languages, however, do not provide explicit support for modeling across different abstraction levels. The aim of this paper is to introduce a family of description logics (DLs) that provide such support in the form of abstraction and refinement operators.

We define the *abstraction DL* $\mathcal{ALCHI}^{\mathsf{abs}}$ as an extension of the familiar description logic $\mathcal{ALCHI}$, which may be viewed as a modest and tame core of the OWL 2 DL ontology language. In principle, however, the same extension can be applied to any other DL, both more and less expressive than $\mathcal{ALCHI}$. Abstraction levels are explicitly named and referred to in the ontology, and we provide explicit operators for the abstraction and refinement of concepts and roles. For example, the concept refinement

$$L_2{:}q_A \ \underline{\mathsf{refines}}\ L_1{:}\mathsf{Arm},$$

where q_A denotes the conjunctive query (CQ)

$$q_A = \mathsf{UArm}(x_1) \wedge \mathsf{LArm}(x_2) \wedge \mathsf{Hand}(x_3) \wedge \mathsf{joins}(x_1, x_2) \wedge \mathsf{joins}(x_2, x_3),$$

expresses that every instance of Arm on the coarser abstraction level L_1 decomposes into an ensemble of three objects on the finer level L_2 as described by q_A. Concept abstractions are dual to refinements and state that every ensemble of a certain kind on a finer level gives rise to an abstracting object on a coarser level. Semantically, there is one classical DL interpretation for each abstraction level and a (partial) refinement function that associates objects on coarser levels with ensembles on finer levels. Every object may participate in at most one ensemble and we require that abstraction levels are organized in the form of a tree. We believe that the abstraction DLs defined along these lines are useful for many application domains, examples are given in the paper. Though not limited to it, our DLs are particularly well-suited for capturing the mereological (part-whole) aspect of abstraction and refinement (Artale et al. 1996).

Base DL	cr	ca	rr	ra	Complexity
$\mathcal{ALCHI}$	X				in EXPTIME
$\mathcal{ALCHI}$	X	X	X	X	in 2EXPTIME
$\mathcal{ALC}$		X			2EXPTIME-hard
$\mathcal{ALC}$			X		2EXPTIME-hard
$\mathcal{ALC}$				X	2EXPTIME-hard

Figure 1: The complexity of satisfiability in abstraction DLs.

Our main technical contribution is to show that adding abstraction and refinement to $\mathcal{ALCHI}$ preserves decidability of the base logic, and to provide a detailed analysis of its complexity, also considering many relevant fragments. It turns out that satisfiability in $\mathcal{ALCHI}^{\mathsf{abs}}$ is 2EXPTIME-complete. Note that this is in line with the fact that CQ evaluation in $\mathcal{ALCHI}$ is 2EXPTIME-complete (Lutz 2008). For the fragments, however, such parallels to evaluation cease to hold. We use $\mathcal{ALCHI}^{\mathsf{abs}}[\text{cr}]$ to denote $\mathcal{ALCHI}^{\mathsf{abs}}$ in which only concept refinement is admitted, and likewise ca denotes concept abstraction and rr, ra denote role refinement and abstraction. We recall that $\mathcal{ALC}$ is $\mathcal{ALCHI}$ without inverse roles and role hierarchies.

We show that satisfiability in the natural fragment $\mathcal{ALCHI}^{\mathsf{abs}}[\text{cr}]$ is only EXPTIME-complete, despite the fact that it still comprises CQs. Moreover, 2EXPTIME-hardness already holds for $\mathcal{ALC}^{\mathsf{abs}}$ in contrast to the fact that CQ evaluation in $\mathcal{ALC}$ is in EXPTIME. There are actually three different sources of complexity as satisfiability is 2EXPTIME-hard already in each of the fragments $\mathcal{ALC}^{\mathsf{abs}}[\text{ca}]$, $\mathcal{ALC}^{\mathsf{abs}}[\text{rr}]$, and $\mathcal{ALC}^{\mathsf{abs}}[\text{ra}]$. In $\mathcal{ALC}^{\mathsf{abs}}[\text{ra}]$, role abstractions allow us to recover inverse roles. The same is true for $\mathcal{ALC}^{\mathsf{abs}}[\text{ca}]$ that, however, requires a more subtle reduction relying on the fact that ensembles must not overlap. Finally, $\mathcal{ALC}^{\mathsf{abs}}[\text{rr}]$ is 2EXPTIME-hard because role refinements allow us to generate objects interlinked in a complex way. See Figure 1 for a summary.

We then observe that the decidability of $\mathcal{ALCHI}^{\mathsf{abs}}$ is more fragile than it may seem on first sight and actually depends on a number of careful design decisions. In particular, we consider three natural extensions and variations and show that each of them is undecidable. The first variation is to drop the requirement that abstraction levels are organized in a tree. The second is to add the requirement that ensembles (which are tuples rather than sets) must not contain repeated elements. And the third is to drop the requirement that CQs in abstraction and refinement statements must be full, that is, to admit quantified variables in such CQs.

Proofs are in the appendix.

Related Work. A classic early article on granularity in AI is (Hobbs 1985). Granularity has later been studied in the area of foundational ontologies, focussing on a philosophically adequate modeling in first-order logic. Examples include granular partitions (Bittner and Smith 2003), the descriptions and situations framework (Gangemi and Mika 2003), and domain-specific approaches (Fonseca et al. 2002; Schulz, Boeker, and Stenzhorn 2008; Vogt 2019).

Existing approaches to representing abstraction and refinement / granularity in DLs and in OWL are rather different in spirit. Some are based on rough or fuzzy set theory (Klinov, Taylor, and Mazlack 2008; Lisi and Mencar 2018), some provide mainly a modeling discipline (Calegari and Ciucci 2010), some aim at the spatial domain (Hbeich, Roxin, and Bus 2021) or at speeding up reasoning (Glimm, Kazakov, and Tran 2017), and some take abstraction to mean the translation of queries between different data schemas (Cima et al. 2022). We also mention description logics of context (Klarman and Gutiérrez-Basulto 2016); an abstraction level can be seen as a context, but the notion of context is more general and governed by looser principles. A categorization of different forms of granularity is in (Keet 2008).

There is a close connection between DLs with abstraction as proposed in this paper and the unary negation fragment of first-order logic (UNFO). In fact, UNFO encompasses ontologies formulated in DLs such as $\mathcal{ALCI}$ and conjunctive queries. UNFO satisfiability is decidable and 2EXPTIME-complete (Segoufin and ten Cate 2013). This does, however, not imply any of the results in this paper due to the use of refinement functions in the semantics of our DLs and the fact that UNFO extended with functional relations is undecidable (Segoufin and ten Cate 2013).

2 Preliminaries

Base DLs. Fix countably infinite sets $\mathbf{C}$ and $\mathbf{R}$ of *concept names* and *role names*. A *role* is a role name or an *inverse role*, that is, an expression r^- with r a role name. If $R = r^-$ is an inverse role, then we set $R^- = r$. *$\mathcal{ALCI}$-concepts* C, D are built according to the syntax rule

$$C, D ::= A \mid \neg C \mid C \sqcap D \mid C \sqcup D \mid \exists R.C \mid \forall R.C$$

where A ranges over concept names and R over roles. We use $\top$ as an abbreviation for $A \sqcup \neg A$ with A a fixed concept name and $\bot$ for $\neg\top$. An *$\mathcal{ALC}$-concept* is an $\mathcal{ALCI}$-concept that does not use inverse roles and an *$\mathcal{EL}$-concept* is an $\mathcal{ALC}$-concepts that uses none of $\neg$, $\sqcup$, and $\forall$.

An *$\mathcal{ALCHI}$-ontology* is a finite set of *concept inclusions (CIs)* $C \sqsubseteq D$ with C and D $\mathcal{ALCI}$-concepts and *role inclusions (RIs)*, $R \sqsubseteq S$ with R, S roles. The letter $\mathcal{I}$ indicates the presence of inverse roles and $\mathcal{H}$ indicates the presence of role inclusions (also called role hierarchies), and thus it should also be clear what we mean e.g. by an $\mathcal{ALCI}$-ontology and an $\mathcal{ALCH}$-ontology. An *$\mathcal{EL}$-ontology* is a finite set of CIs $C \sqsubseteq D$ with C, D $\mathcal{EL}$-concepts.

An *interpretation* is a pair $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ with $\Delta^{\mathcal{I}}$ a non-empty set (the *domain*) and $\cdot^{\mathcal{I}}$ an *interpretation function* that maps every concept name $A \in \mathbf{C}$ to a set $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ and every role name $r \in \mathbf{R}$ to a binary relation $r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. The interpretation function is extended to compound concepts as usual, c.f. (Baader et al. 2017). An interpretation $\mathcal{I}$ *satisfies* a CI $C \sqsubseteq D$ if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ and an RI $R \sqsubseteq S$ if $R^{\mathcal{I}} \subseteq S^{\mathcal{I}}$. It is a *model* of an ontology $\mathcal{O}$ if it satisfies all CIs and RIs in it.

For any syntactic object O such as an ontology or a concept, we use $||O||$ to denote the *size* of O, that is, the number of symbols needed to write O over a suitable alphabet.

Conjunctive Queries. Let $\mathbf{V}$ be a countably infinite set of variables. A *conjunctive query (CQ)* takes the form $q(\bar{x}) = \exists \bar{y}\, \varphi(\bar{x}, \bar{y})$ with φ a conjunction of *concept atoms* $C(x)$ and *role atoms* $r(x, y)$, C a (possibly compound) concept, r a role name, and x, y variables from $\bar{x} \cup \bar{y}$. We may write $\alpha \in q$ to indicate that α is an atom in φ and $r^-(x, y) \in q$ in place of $r(y, x) \in q$. The variables in $\bar{x}$ are the *answer variables* of q. We require that every answer variable x occurs in some atom of q, but omit this atom in writing in case it is $\top(x)$. With $\mathsf{var}(q)$, we denote the set of all (answer and quantified) variables in q. If q has no answer variables then it is *Boolean*. We mostly restrict our attention to CQs q that are *full*, meaning that q has no quantified variables. A CQ q is *connected* if the undirected graph with node set $\mathsf{var}(q)$ and edge set $\{\{v, v'\} \mid r(v, v') \in q \text{ for any } r \in \mathbf{R}\}$ is. A CQ q is a *subquery* of a CQ q' if q can be obtained from q' by dropping atoms.

Let $q(\bar{x}) = \exists \bar{y}\, \varphi(\bar{x}, \bar{y})$ be a CQ and $\mathcal{I}$ an interpretation. A mapping $h : \bar{x} \cup \bar{y} \to \Delta^{\mathcal{I}}$ is a *homomorphism* from q to $\mathcal{I}$ if $C(x) \in q$ implies $h(x) \in C^{\mathcal{I}}$ and $r(x, y) \in q$ implies $(h(x), h(y)) \in r^{\mathcal{I}}$. A tuple $\bar{d} \in (\Delta^{\mathcal{I}})^{|\bar{x}|}$ is an *answer* to q on $\mathcal{I}$ if there is a homomorphism h from q to $\mathcal{I}$ with $h(\bar{x}) = \bar{d}$. We use $q(\mathcal{I})$ to denote the set of all answers to q on $\mathcal{I}$. If q is Boolean, we write $\mathcal{I} \models q$ to indicate the existence of a homomorphism from q to $\mathcal{I}$.

3 DLs with Abstraction and Refinement

We extend $\mathcal{ALCHI}$ to the DL $\mathcal{ALCHI}^{\mathsf{abs}}$ that supports abstraction and refinement. Fix a countable set $\mathbf{A}$ of *abstrac-*

tion levels. An $\mathcal{ALCHI}^{\mathsf{abs}}$-ontology is a finite set of statements of the following form:

- *labeled concept inclusions* $C \sqsubseteq_L D$,
- *labeled role inclusions* $R \sqsubseteq_L S$,
- *concept refinements* $L{:}q(\bar{x})$ $\underline{\mathsf{refines}}$ $L'{:}C$,
- *concept abstractions* $L'{:}C$ $\underline{\mathsf{abstracts}}$ $L{:}q(\bar{x})$,
- *role refinements* $L{:}q(\bar{x},\bar{y})$ $\underline{\mathsf{refines}}$ $L'{:}q_R(x,y)$,
- *role abstractions* $L'{:}R$ $\underline{\mathsf{abstracts}}$ $L{:}q(\bar{x},\bar{y})$

where L, L' range over $\mathbf{A}$, C, D over $\mathcal{ALCI}$-concepts, R, S over roles, q over full conjunctive queries, and q_R over full conjunctive queries of the form $C_1(x) \wedge R(x,y) \wedge C_2(y)$. In concept and role abstraction statements, we additionally require the CQ q to be connected. We may write $C \equiv_L D$ as shorthand for the two CIs $C \sqsubseteq_L D$ and $D \sqsubseteq_L C$. We underline abstraction and refinement operators to ensure better readability throughout the paper.

Intuitively, a concept refinement $L{:}q(\bar{x})$ $\underline{\mathsf{refines}}$ $L'{:}C$ expresses that any instance of C on abstraction level L' refines into an *ensemble* of $|\bar{x}|$ objects on abstraction level L which satisfies all properties expressed by CQ q. Conversely, a concept abstraction $L'{:}C$ $\underline{\mathsf{abstracts}}$ $L{:}q(\bar{x})$ says that any ensemble of $|\bar{x}|$ objects on abstraction level L that satisfies q abstracts into a single instance of C on abstraction level L'. Role refinements and abstractions can be understood in a similar way, where each of the two elements that participate in a role relationship refines into its own ensemble.

Note that in role refinements, we consider CQs $q_R = C_1(x) \wedge R(x,y) \wedge C_2(y)$ rather than only the role R. This is because roles are often of a general kind such as partOf or $\mathsf{interactsWith}$ and need proper context to be meaningfully refined. This context is provided by the concepts C_1, C_2.

Example 1. *Granularity is important in many domains. Anatomy has already been mentioned in the introduction. The concept refinement given there may be complemented by choosing q_A as in the introduction and adding the concept abstraction*

$$L_1{:}\mathsf{Arm}\ \underline{\mathsf{abstracts}}\ L_2{:}q_A.$$

We next consider bikes as a simple example for a technical domain. Let us first say how wheels refine into components: $L_2{:}q_W$ $\underline{\mathsf{refines}}$ $L_1{:}\mathsf{Wheel}$ *where*

$$q_W = \mathsf{Axle}(x_1) \wedge \mathsf{Spokes}(x_2) \wedge \mathsf{Rim}(x_3) \wedge \mathsf{Tire}(x_4) \wedge \mathsf{join}(x_2,x_1) \wedge \mathsf{join}(x_2,x_3) \wedge \mathsf{carries}(x_3,x_4).$$

We may then use the following role refinement to express how frames connect to wheels:

$$L_2{:}q_{FW}\ \underline{\mathsf{refines}}\ L_1{:}\,\mathsf{Wheel}(x) \wedge \mathsf{connTo}(x,y) \wedge \mathsf{Frame}(y)$$

where, for $\bar{x} = x_1 \cdots x_4$ and $\bar{y} = y_1 \cdots y_7$ (assuming that frames have seven components),

$$q_{FW}(\bar{x},\bar{y}) = \mathsf{Axle}(x_1) \wedge \mathsf{connTo}(x_1,y_1) \wedge \mathsf{Dropout}(y_1).$$

This expresses that if a wheel is connected to a frame, then the axle of the wheel is connected to the dropout of the frame.

Extensions $\mathcal{L}^{\mathsf{abs}}$ of other DLs $\mathcal{L}$ introduced in Section 2, such as $\mathcal{ALC}$ and $\mathcal{ALCH}$, may be defined in the expected way. We also consider various fragments of $\mathcal{ALCHI}^{\mathsf{abs}}$. With $\mathcal{ALCHI}^{\mathsf{abs}}$[cr,rr], for example, we mean the fragment of $\mathcal{ALCHI}^{\mathsf{abs}}$ that admits concept refinement and role refinement, but neither concept abstraction nor role abstraction (identified by ca and ra).

We next define the semantics of $\mathcal{ALCHI}^{\mathsf{abs}}$, based on *A-interpretations* which include one traditional DL interpretation for each abstraction level. Formally, an A-interpretation takes the form $\mathcal{I} = (\mathbf{A}_\mathcal{I}, \prec, (\mathcal{I}_L)_{L\in\mathbf{A}_\mathcal{I}}, \rho)$, where

- $\mathbf{A}_\mathcal{I} \subseteq \mathbf{A}$ is the set of relevant abstraction levels;
- $\prec\ \subseteq\ \mathbf{A}_\mathcal{I} \times \mathbf{A}_\mathcal{I}$ is such that the directed graph $(\mathbf{A}_\mathcal{I}, \{(L',L) \mid L \prec L'\})$ is a tree; intuitively, $L \prec L'$ means that L is less abstract than L' or, in other words, that the modeling granularity of L is finer than that of L';
- $(\mathcal{I}_L)_{L\in\mathbf{A}_\mathcal{I}}$ is a collection of interpretations $\mathcal{I}_L$, one for every $L \in \mathbf{A}_\mathcal{I}$, with pairwise disjoint domains; we use $L(d)$ to denote the unique $L \in \mathbf{A}_\mathcal{I}$ with $d \in \Delta^{\mathcal{I}_L}$;
- ρ is the *refinement function*, a partial function that associates pairs $(d, L) \in \Delta^\mathcal{I} \times \mathbf{A}_\mathcal{I}$ such that $L \prec L(d)$ with an *L-ensemble* $\rho(d,L)$, that is, with a non-empty tuple over $\Delta^{\mathcal{I}_L}$. We want every object to participate in only one ensemble and thus require that

(∗) for all $d \in \Delta^\mathcal{I}$, there is at most one $e \in \Delta^\mathcal{I}$ such that d occurs in $\rho(e, L(d))$.

For readability, we may write $\rho_L(d)$ in place of $\rho(d,L)$.

An A-interpretation $\mathcal{I} = (\mathbf{A}_\mathcal{I}, \prec, (\mathcal{I}_L)_{L\in\mathbf{A}_\mathcal{I}}, \rho)$ *satisfies* a

- labeled concept or role inclusion $\alpha \sqsubseteq_L \beta$ if $L \in \mathbf{A}_\mathcal{I}$ and $\alpha^{\mathcal{I}_L} \subseteq \beta^{\mathcal{I}_L}$;
- concept refinement $L{:}q(\bar{x})$ $\underline{\mathsf{refines}}$ $L'{:}C$ if $L \prec L'$ and for all $d \in C^{\mathcal{I}_{L'}}$, there is an $\bar{e} \in q(\mathcal{I}_L)$ such that $\rho_L(d) = \bar{e}$;
- concept abstraction $L'{:}C$ $\underline{\mathsf{abstracts}}$ $L{:}q(\bar{x})$ if $L \prec L'$ and for all $\bar{e} \in q(\mathcal{I}_L)$, there is a $d \in C^{\mathcal{I}_{L'}}$ s.t. $\rho_L(d) = \bar{e}$;
- role refinement $L{:}q(\bar{x},\bar{y})$ $\underline{\mathsf{refines}}$ $L'{:}q_R(x,y)$ if $L \prec L'$ and for all $(d_1,d_2) \in q_R(\mathcal{I}_{L'})$, there is an $(\bar{e}_1,\bar{e}_2) \in q(\mathcal{I}_L)$ such that $\rho_L(d_1) = \bar{e}_1$ and $\rho_L(d_2) = \bar{e}_2$;
- role abstraction $L'{:}R$ $\underline{\mathsf{abstracts}}$ $L{:}q(\bar{x},\bar{y})$ if $L \prec L'$ and for all $(\bar{e}_1,\bar{e}_2) \in q(\mathcal{I}_L)$, there is a $(d_1,d_2) \in R^{\mathcal{I}_{L'}}$ such that $\rho_L(d_1) = \bar{e}_1$ and $\rho_L(d_2) = \bar{e}_2$.

An A-interpretation is a *model* of an $\mathcal{ALCHI}^{\mathsf{abs}}$-ontology if it satisfies all inclusions, refinements, and abstractions in it.

Example 2. *We consider the domain of (robotic) actions. Assume that there is a* Fetch *action that refines into subactions:* $L_2{:}q_F$ $\underline{\mathsf{refines}}$ $L_1{:}\mathsf{Fetch}$ *where*

$$q_F = \mathsf{Locate}(x_1) \wedge \mathsf{Move}(x_2) \wedge \mathsf{Grasp}(x_3) \wedge \mathsf{precedes}(x_1,x_2) \wedge \mathsf{precedes}(x_2,x_3).$$

We might have a safe version of the fetching action and a two-handed grasping action:

$$\mathsf{SFetch} \sqsubseteq_{L_1} \mathsf{Fetch}$$
$$\mathsf{TwoHandedGrasp} \sqsubseteq_{L_1} \mathsf{Grasp}$$

A safe fetch requires a two-handed grasping subaction: $L_2{:}q_S$ $\underline{\mathsf{refines}}$ $L_1{:}\mathsf{SFetch}$ *where for $\bar{x} = x_1x_2x_3$,*

$$q_S(\bar{x}) = \mathsf{TwoHandedGrasp}(x_3).$$

We remark that abstraction statements need to be used with care since ensembles may not overlap, c.f. Condition (∗). For example, the reader may want to verify that the following CI and concept abstraction have no model:

$$\top \sqsubseteq_{L_2} \exists r.\exists r.\top \qquad L_1{:}\top \ \underline{\mathsf{abstracts}}\ L_2{:}r(x, y).$$

We are interested in the problem of *(concept) satisfiability* which means to decide, given an $\mathcal{ALCHI}^{\mathsf{abs}}$-ontology $\mathcal{O}$, an $\mathcal{ALCI}$-concept C, and an abstraction level $L \in \mathbf{A}$, whether there is a model $\mathcal{I}$ of $\mathcal{O}$ such that $C^{\mathcal{I}_L} \neq \emptyset$. We then say that C is *L-satisfiable w.r.t. $\mathcal{O}$*. As usual, the related reasoning problems of subsumption can be reduced to satisfiability in polynomial time, and vice versa (Baader et al. 2017).

4 Upper Bounds

We prove that satisfiability in $\mathcal{ALCHI}^{\mathsf{abs}}$ is decidable in 2EXPTIME. Before approaching this general case, however, we consider the fragment $\mathcal{ALCHI}^{\mathsf{abs}}[\text{cr}]$ and show that it is only EXPTIME-complete.

4.1 $\mathcal{ALCHI}^{\mathsf{abs}}[\text{cr}]$ in ExpTime

Our aim is to prove the following.

Theorem 1. *Satisfiability in $\mathcal{ALCHI}^{\mathsf{abs}}[\text{cr}]$ is EXPTIME-complete.*

The lower bound is inherited from $\mathcal{ALCHI}$ without abstraction and refinement (Baader et al. 2017). We prove the upper bound by a mosaic-based approach, that is, we decide the existence of a model $\mathcal{I}$ by trying to assemble $\mathcal{I}$ from small fragments called mosaics. Essentially, a mosaic describes a single ensemble on a single level of abstraction.

Assume that we are given as input an $\mathcal{ALCHI}^{\mathsf{abs}}[\text{cr}]$-ontology $\mathcal{O}$, an $\mathcal{ALCI}$-concept C_0, and an abstraction level L_0. We may assume w.l.o.g. that C_0 is a concept name as we can extend $\mathcal{O}$ with $A_0 \sqsubseteq_{L_0} C_0$ and test satisfiability of the fresh concept name A_0. We also assume w.l.o.g. that $\mathcal{O}$ is in *normal form*, meaning that

1. every CI has one of the forms
$$\begin{array}{lll} \top \sqsubseteq_L A & A \sqsubseteq_L \exists R.B & \exists R.B \sqsubseteq_L A \\ A_1 \sqcap A_2 \sqsubseteq_L A & A \sqsubseteq_L \neg B & \neg B \sqsubseteq_L A \end{array}$$
where A, A_1, A_2, B are concept names and R is a role;
2. in every concept refinement $L{:}q(\bar{x})$ $\underline{\mathsf{refines}}$ $L'{:}C$, C is a concept name, and so is D in all concept atoms $D(x) \in q$.

It is in fact routine to show that every $\mathcal{ALCHI}^{\mathsf{abs}}[\text{cr}]$ ontology $\mathcal{O}$ can be converted in polynomial time into an $\mathcal{ALCHI}^{\mathsf{abs}}[\text{cr}]$ ontology $\mathcal{O}'$ in normal form that is a conservative extension of $\mathcal{O}$, see e.g. (Manière 2022). We also assume that (i) $\mathcal{O}$ contains $R \sqsubseteq_L R$ for all roles R and abstraction levels L in $\mathcal{O}$, (ii) $R \sqsubseteq_L S$, $S \sqsubseteq_L T \in \mathcal{O}$ implies $R \sqsubseteq_L T \in \mathcal{O}$, and (iii) $R \sqsubseteq_L S \in \mathcal{O}$ implies $R^- \sqsubseteq_L S^- \in \mathcal{O}$. With $\prec$ we denote the smallest relation on $\mathbf{A}_\mathcal{O}$ such that $L \prec L'$ for all $L{:}q(\bar{x})$ $\underline{\mathsf{refines}}$ $L'{:}C$ in $\mathcal{O}$.

Fix a domain Δ of cardinality $||\mathcal{O}||$. A *mosaic* is a pair $M = (L, \mathcal{I})$ where $L \in \mathbf{A}_\mathcal{O}$ is the abstraction level of the mosaic and $\mathcal{I}$ is an interpretation with $\Delta^\mathcal{I} \subseteq \Delta$ such that $\mathcal{I}$ satisfies all CIs $C \sqsubseteq_L D$ in $\mathcal{O}$ and all RIs $R \sqsubseteq_L S$ in $\mathcal{O}$, with the possible exception of CIs of the form $A \sqsubseteq_L \exists r.B$. We may write L^M to denote L, and likewise for $\mathcal{I}^M$. Let $\mathcal{M}$ be a set of mosaics. We say that a mosaic $M = (L, \mathcal{I})$ is *good* in $\mathcal{M}$ if for all $d \in \Delta^\mathcal{I}$ the following hold:

1. if $A \sqsubseteq_L \exists R.B \in \mathcal{O}$, $d \in A^\mathcal{I}$, and $d \notin (\exists R.B)^\mathcal{I}$, then there is an $M' = (L, \mathcal{I}') \in \mathcal{M}$ and a $d' \in \Delta^{\mathcal{I}^{M'}}$ such that
   (a) $d' \in B^{\mathcal{I}'}$,
   (b) if $\exists S.A \sqsubseteq_L B \in \mathcal{O}$, $R \sqsubseteq_L S \in \mathcal{O}$, and $d' \in A^{\mathcal{I}'}$, then $d \in B^\mathcal{I}$;
   (c) if $\exists S.A \sqsubseteq_L B \in \mathcal{O}$, $R^- \sqsubseteq_L S \in \mathcal{O}$, and $d \in A^\mathcal{I}$, then $d' \in B^{\mathcal{I}'}$;
2. for every level $L' \in \mathbf{A}_\mathcal{O}$ such that
$$Q = \{q \mid L'{:}q(\bar{x})\ \underline{\mathsf{refines}}\ L{:}A \in \mathcal{O} \text{ and } d \in A^\mathcal{I}\} \neq \emptyset,$$
there is a mosaic $M' \in \mathcal{M}$ with $M' = (L', \mathcal{I}')$ and a tuple $\bar{e}$ over $\Delta^{\mathcal{I}'}$ such that $\bar{e} \in q(\mathcal{I}')$ for all $q \in Q$.

We now formulate the actual decision procedure. If the directed graph $(\mathbf{A}_\mathcal{O}, \{(L', L) \mid L \prec L'\})$ is not a tree, we directly return 'unsatisfiable'. Our algorithm first computes the set $\mathcal{M}_0$ of all mosaics for $\mathcal{O}$ and then repeatedly and exhaustively eliminates mosaics that are not good. Let $\mathcal{M}^*$ denote the set of mosaics at which this process stabilizes.

Lemma 1. *C_0 is L_0-satisfiable w.r.t. $\mathcal{O}$ iff $\mathcal{M}^*$ contains (i) a mosaic M with $L^M = L_0$ and $C_0^{\mathcal{I}^M} \neq \emptyset$ and (ii) a mosaic M with $L^M = L$, for every L in $\mathbf{A}_\mathcal{O}$.*

The algorithm thus returns 'satisfiable' if Conditions (i) and (ii) from Lemma 1 are satisfied and 'unsatisfiable' otherwise. It is easy to see that the algorithm runs in single exponential time.

4.2 $\mathcal{ALCHI}^{\mathsf{abs}}$ in 2ExpTime

Our aim is to prove the following.

Theorem 2. *Satisfiability in $\mathcal{ALCHI}^{\mathsf{abs}}$ is decidable in 2EXPTIME.*

A matching lower bound will be provided later on. We prove Theorem 2 by a mosaic-based approach which is, however, significantly more complex than the one used in the previous section. In particular, a mosaic now represents a 'slice' through an A-interpretation that includes multiple abstraction levels and multiple ensembles.

Assume that we are given as input an $\mathcal{ALCHI}^{\mathsf{abs}}$-ontology $\mathcal{O}$, an $\mathcal{ALCI}$-concept C_0, and an abstraction level L_0. We again assume that C_0 is a concept name. and $\mathcal{O}$ is in normal form, defined as in the previous section, but with the obvious counterparts of Point 2 for role refinements and (concept and role) abstractions. We also define the relation $\prec$ on $\mathbf{A}_\mathcal{O}$ as in the previous section, except that we now consider concept and role refinements, as well as concept and role abstractions, in the obvious way. Again, if the directed graph $(\mathbf{A}_\mathcal{O}, \{(L', L) \mid L \prec L'\})$ is not a tree, then we directly return 'unsatisfiable'.

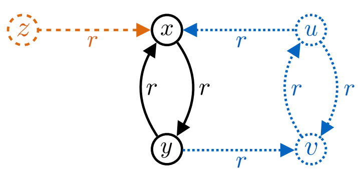
Fix a set Δ of cardinality $||\mathcal{O}||^{||\mathcal{O}||}$. A *mosaic* is a tuple
$$M = ((\mathcal{I}_L)_{L \in \mathbf{A}_\mathcal{O}}, \rho, f_{\mathsf{in}}, f_{\mathsf{out}})$$
where

- $(\mathcal{I}_L)_{L\in\mathbf{A}_\mathcal{O}}$ is a collection of interpretations and ρ is a partial function such that $(\mathbf{A}_\mathcal{O}, \prec, (\mathcal{I}_L)_{L\in\mathbf{A}_\mathcal{O}}, \rho)$ is an A-interpretation except that some interpretation domains $\Delta^{\mathcal{I}_L}$ may be empty; the length of tuples in the range of ρ may be at most $||\mathcal{O}||$;
- f_in and f_out are functions that associate every $L \in \mathbf{A}_\mathcal{O}$ with a set of pairs (q, h) where q is a CQ from an abstraction statement in $\mathcal{O}$ or a subquery thereof, and h is a partial function from $\mathsf{var}(q)$ to $\Delta^{\mathcal{I}_L}$; we call these pairs the *forbidden incoming queries* in the case of f_in and the *forbidden outgoing queries* in the case of f_out.

We may write $\mathcal{I}^M_L$ to denote $\mathcal{I}_L$, for any $L \in \mathbf{A}_\mathcal{O}$, and likewise for ρ^M, f^M_in, and f^M_out.

Every mosaic has to satisfy several additional conditions. Before we can state them, we introduce some notation. For $V \subseteq \mathsf{var}(q)$, we use $q|_V$ to denote the restriction of q to the variables in V and write $\overline{V}$ as shorthand for $\overline{V} = \mathsf{var}(q) \setminus V$. A *maximally connected component (MCC)* of q is a CQ $q|_V$ that is connected and such that V is maximal with this property. A CQ $p = E \uplus p_0$ is a *component of* q *w.r.t.* $V \subseteq \mathsf{var}(q)$ if p_0 is an MCC of $q|_{\overline{V}}$ and E is the set of all atoms from q that contain one variable from V and one variable from $\overline{V}$.

Example 3. *The following CQ has two components w.r.t. $V = \{x, y\}$, which are displayed in dashed and dotted lines:*



For example, the dotted component is defined by $p = E \uplus p_0$ with $E = r(u, x) \wedge r(y, v)$ and $p_0 = r(u, v) \wedge r(v, u)$.

With these notions at hand, let us explain the intuition of the f_in and f_out components of mosaics. Our decomposition of A-interpretations into sets of mosaics is such that every ensemble falls within a single mosaic. This means that we must avoid homomorphisms from the CQs in concept abstractions that hit multiple mosaics: such homomorphisms would hit elements from multiple ensembles while also turning the set of all elements that are hit into an ensemble; they thus generate overlapping ensembles which is forbidden. Almost the same holds for role abstractions where however the CQ takes the form $q(\bar{x}, \bar{y})$ with each of $\bar{x}$ and $\bar{y}$ describing an ensemble, and we must only avoid homomorphisms that hit multiple mosaics from the variables in $\bar{x}$, or from the variables in $\bar{y}$.

Query avoidance is implemented by the f_in and f_out components. In brief and for CQs $q(\bar{x})$ from concept abstractions, we consider any non-empty subset $V \subsetneq \mathsf{var}(q)$ of variables and homomorphism h from $q|_V$ to the current mosaic. We then have to avoid any homomorphism g from $q \setminus q|_V$ that is compatible with h and hits at least one mosaic other than the current one. The choice of V decomposes q into remaining components, which are exactly the components of q w.r.t. V defined above. We choose one such component p and put (p, h') into f_out, h' the restriction of h to the variables in p, to 'send' the information to other mosaics that this query is forbidden. The f_in component, in contrast, contains forbidden queries that we 'receive' from other mosaics.

We now formulate the additional conditions on mosaics. We require that $M = ((\mathcal{I}_L)_{L\in\mathbf{A}_\mathcal{O}}, \rho, f_\mathsf{in}, f_\mathsf{out})$ satisfies the following conditions, for all $L \in \mathbf{A}_\mathcal{O}$:

1. the A-interpretation $(\mathbf{A}_\mathcal{O}, \prec, (\mathcal{I}_L)_{L\in\mathbf{A}_\mathcal{O}}, \rho)$ satisfies all inclusions, refinements, and abstractions in $\mathcal{O}$ with the possible exception of CIs the form $A \sqsubseteq_L \exists r.B$;
2. for all concept abstractions $L'{:}A$ abstracts $L{:}q(\bar{x})$ in $\mathcal{O}$, all non-empty $V \subsetneq \mathsf{var}(q)$, and all homomorphisms h from $q|_V$ to $\mathcal{I}_L$: there is a component p of q w.r.t. V such that $(p, h|_{V\cap\mathsf{var}(p)}) \in f_\mathsf{out}(L)$
3. for all role abstractions $L'{:}R$ abstracts $L{:}q(\bar{x}, \bar{y})$ in $\mathcal{O}$, all non-empty $V \subsetneq \mathsf{var}(q)$ with $V \neq \bar{x}$ and $V \neq \bar{y}$,[1] and all homomorphisms h from $q|_V$ to $\mathcal{I}_L$: there is a component p of q w.r.t. V such that $(p, h|_{V\cap\mathsf{var}(p)}) \in f_\mathsf{out}(L)$;
4. for all $(q, h) \in f_\mathsf{in}(L)$, all $V \subseteq \mathsf{var}(q)$, and all homomorphisms g from $q|_V$ to $\mathcal{I}_L$ that extend h, there is a component p of q w.r.t. V such that $(p, g|_{V\cap\mathsf{var}(p)}) \in f_\mathsf{out}(L)$.

We next need a mechanism to interconnect mosaics. This is driven by concept names A and elements $d \in A^{\mathcal{I}_L}$ such that $A \sqsubseteq_L \exists R.B \in \mathcal{O}$ and d lacks a witness inside the mosaic. In principle, we would simply like to find a mosaic M' that has some element e on level L such that $e \in B^{\mathcal{I}^{M'}}$ and an R-edge can be put between d and e. The situation is complicated, however, by the presence of role refinements and role abstractions, which might enforce additional edges that link the two mosaics. We must also be careful to synchronize the $f_\mathsf{in}, f_\mathsf{out}$ components of the two involved mosaics across the connecting edges.

Consider mosaics $M = ((\mathcal{I}_L)_{L\in\mathbf{A}_\mathcal{O}}, \rho, f_\mathsf{in}, f_\mathsf{out})$ and $M' = ((\mathcal{I}'_L)_{L\in\mathbf{A}_\mathcal{O}}, \rho', f'_\mathsf{in}, f'_\mathsf{out})$. An M, M'*-edge* is an expression $R(d, d')$ such that R is a role, $d \in \Delta^{\mathcal{I}_L}$, and $d' \in \Delta^{\mathcal{I}'_L}$ for some $L \in \mathbf{A}_\mathcal{O}$. A set E of M, M'-edges is an *edge candidate* if the following conditions are satisfied:

1. $R(d, e) \in E$ and $L(d) = L$ implies $S(d, e) \in E$, for all $R \sqsubseteq_L S \in \mathcal{O}$;
2. if $\exists R.A \sqsubseteq_L B \in \mathcal{O}$, $R(d, d') \in E$, and $d' \in A^{\mathcal{I}'_L}$, then $d \in B^{\mathcal{I}_L}$;
3. for all $L \in \mathbf{A}_\mathcal{O}$, all $(q, h) \in f_\mathsf{out}(L)$, where $q = E_q \uplus q|_{\overline{V}}$ for $V = \mathsf{dom}(h)$, and all functions g from $\overline{V} \cap \mathsf{var}(E_q)$ to $\Delta^{\mathcal{I}'_L}$ such that $R(h(x), g(y)) \in E$ for all $R(x, y) \in E_q$, we have $(q|_{\overline{V}}, g) \in f'_\mathsf{in}(L)$;
4. for all $R(d, d') \in E$ and all $L{:}q(\bar{x}, \bar{y})$ refines $L'{:}q_R(x, y) \in \mathcal{O}$ such that $q = q|_{\bar{x}} \uplus E_q \uplus q|_{\bar{y}}$, $q_R = C_x(x) \wedge R(x, y) \wedge C_y(y)$, $d \in C_x^{\mathcal{I}_{L'}}$, and $d' \in C_y^{\mathcal{I}'_{L'}}$:
   (a) $\rho_L(d)$ and $\rho'_L(d')$ are defined;
   (b) $h : \bar{x} \mapsto \rho_L(d)$ is a homomorphism from $q|_{\bar{x}}$ to $\mathcal{I}_L$;
   (c) $h' : \bar{y} \mapsto \rho'_L(d')$ is a homomorphism from $q|_{\bar{y}}$ to $\mathcal{I}'_L$;
   (d) $\{R(h(x), h'(y)) \mid R(x, y) \in E_q\} \subseteq E$;

[1] Here we view $\bar{x}$ and $\bar{y}$ as sets.

5. for all role abstractions $L'{:}R$ $\underline{\mathsf{abstracts}}$ $L{:}q(\bar{x},\bar{y}) \in \mathcal{O}$, where $q = q|_{\bar{x}} \uplus E_q \uplus q|_{\bar{y}}$, all homomorphisms h from $q|_{\bar{x}}$ to $\mathcal{I}_L$, and all homomorphisms g from $q|_{\bar{y}}$ to $\mathcal{I}'_L$ such that $\{S(h(x),g(y)) \mid S(x,y) \in E_q\} \subseteq E$, there are $d \in \Delta^{\mathcal{I}_{L'}}$ and $d' \in \Delta^{\mathcal{I}'_{L'}}$ with $\rho_L(d) = h(\bar{x})$, $\rho'_L(d') = g(\bar{y})$, and $R(d,d') \in E$;
6. Converses of Conditions 2-5 above that go from M' to M instead of from M to M'; details are in the appendix.

Let $\mathcal{M}$ be a set of mosaics. A mosaic M is *good in* $\mathcal{M}$ if for all $A \sqsubseteq_L \exists R.B \in \mathcal{O}$ and $d \in (A \sqcap \neg\exists R.B)^{\mathcal{I}_L^M}$:

($*$) there is a mosaic $M' \in \mathcal{M}$, a $d' \in B^{\mathcal{I}_L^{M'}}$, and an edge candidate E such that $R(d,d') \in E$.

The actual algorithm is now identical to that from the previous section. We first compute the set $\mathcal{M}_0$ of all mosaics and then repeatedly and exhaustively eliminate mosaics that are not good. Let $\mathcal{M}^*$ denote the set of mosaics at which this process stabilizes.

Lemma 2. *C_0 is L_0-satisfiable w.r.t. $\mathcal{O}$ iff $\mathcal{M}^*$ contains (i) a mosaic M with $C_0^{\mathcal{I}_{L_0}^M} \neq \emptyset$ and (ii) a mosaic M with $\Delta^{\mathcal{I}_L^M} \neq \emptyset$, for every L in $\mathbf{A}_{\mathcal{O}}$.*

The algorithm thus returns 'satisfiable' if Conditions (i) and (ii) from Lemma 2 are satisfied and 'unsatisfiable' otherwise. It can be verified that the algorithm runs in double exponential time.

5 Lower Bounds

We have seen that the fragment $\mathcal{ALCHI}^{\mathsf{abs}}[\mathrm{cr}]$ of $\mathcal{ALCHI}^{\mathsf{abs}}$ which focusses on concept refinement is only EXPTIME-complete. Here we show that all other fragments that contain only a single form of abstraction/refinement are 2EXPTIME-hard, and consequently 2EXPTIME-complete. This of course also provides a matching lower bound for Theorem 2—actually three rather different lower bounds, each one exploiting a different effect. All of our lower bounds apply already when $\mathcal{ALCHI}$ is replaced with $\mathcal{ALC}$ as the underlying DL.

5.1 Role Abstraction: $\mathcal{ALC}^{\mathsf{abs}}[\mathrm{ra}]$

The 2EXPTIME-hardness of satisfiability in $\mathcal{ALCHI}^{\mathsf{abs}}$ is not entirely surprising given that we have built conjunctive queries into the logic and CQ evaluation on $\mathcal{ALCI}$ knowledge bases is known to be 2EXPTIME-hard (Lutz 2008). In fact, this is already the case for the following *simple* version of the latter problem: given an $\mathcal{ALCI}$ ontology $\mathcal{O}$, a concept name A_0, and a Boolean CQ q, decide whether $\mathcal{I} \models q$ for all models $\mathcal{I}$ of $\mathcal{O}$ with $A_0^{\mathcal{I}} \neq \emptyset$. We write $\mathcal{O}, A_0 \models q$ if this is the case.

It is easy to reduce the (complement of the) simple CQ evaluation problem to satisfiability in $\mathcal{ALCI}^{\mathsf{abs}}[\mathrm{ca}]$. Fix two abstraction levels $L \prec L'$, let $\widehat{q}$ be the CQ obtained from q by dequantifying all variables, thus making all variables answer variables, and let $\mathcal{O}'$ be the set of all concept inclusions $C \sqsubseteq_L D$ with $C \sqsubseteq D \in \mathcal{O}$ and the concept abstraction

$$L'{:}\bot \ \underline{\mathsf{abstracts}}\ L{:}\widehat{q}.$$

It is straightforward to show that A_0 is L-satisfiable w.r.t. $\mathcal{O}'$ iff $\mathcal{O}, A_0 \not\models q$.

Our aim, however, is to prove 2EXPTIME-hardness without using inverse roles. The above reduction does not help for this purpose since CQ evaluation on $\mathcal{ALC}$ knowledge bases is in EXPTIME (Lutz 2008). Also, we wish to use role abstractions in place of the concept abstraction.

As above, we reduce from the complement of simple CQ evaluation on $\mathcal{ALCI}$ ontologies. Given an input $\mathcal{O}, A_0, q$, we construct an $\mathcal{ALC}^{\mathsf{abs}}[\mathrm{ra}]$ ontology $\mathcal{O}'$ such that A_0 is L_1-satisfiable w.r.t. $\mathcal{O}'$ iff $\mathcal{O}, A_0 \not\models q$. The idea is to use multiple abstraction levels and role abstractions to simulate inverse roles. Essentially, we replace every inverse role r^- in $\mathcal{O}$ with a fresh role name $\widehat{r}$, obtaining an $\mathcal{ALC}$-ontology $\mathcal{O}_{\mathcal{ALC}}$, use three abstraction levels $L_1 \prec L_2 \prec L_3$, and craft $\mathcal{O}'$ so that in its models, the interpretation on level L_1 is a model of $\mathcal{O}_{\mathcal{ALC}}$, the interpretation on level L_2 is a copy of the one on level L_1, but additionally has an r-edge for every $\widehat{r}^-$-edge, and L_3 is an additional level used to check that there is no homomorphism from q, as in the initial reduction above. Details are in the appendix.

Theorem 3. *Satisfiability in $\mathcal{ALC}^{\mathsf{abs}}[\mathrm{ra}]$ is 2EXPTIME-hard.*

5.2 Concept Abstraction: $\mathcal{ALC}^{\mathsf{abs}}[\mathrm{ca}]$

When only concept abstraction is available, the simulation of inverse roles by abstraction statements presented in the previous section does not work. We are nevertheless able to craft a reduction from the simple CQ evaluation problem on $\mathcal{ALCI}$ ontologies, though in a slightly different version. What is actually proved in (Lutz 2008) is that simple CQ evaluation is 2EXPTIME-hard already in the DL $\mathcal{ALC}^{\mathsf{sym}}$, which is $\mathcal{ALC}$ with a single role name s that must be interpreted as a reflexive and symmetric relation.

We simulate $\mathcal{ALC}^{\mathsf{sym}}$-concepts $\exists s.C$ by $\mathcal{ALCI}$-concepts $\exists r^-.\exists r.C$, and likewise for $\forall s.C$. The reduction then exploits the following effect. Consider an $\mathcal{ALC}^{\mathsf{abs}}[\mathrm{ca}]$-ontology $\mathcal{O}$ that contains the CI $\top \sqsubseteq_L \forall r.\exists\widehat{r}.\top$ and the concept abstractions

$$L'{:}\top \ \underline{\mathsf{abstracts}}\ L{:}r(x,y) \quad \text{and} \quad L'{:}\top \ \underline{\mathsf{abstracts}}\ L{:}\widehat{r}(y,x).$$

Then in any model $\mathcal{I}$ of $\mathcal{O}$, $(d,e) \in r^{\mathcal{I}_L}$ implies $(e,d) \in \widehat{r}^{\mathcal{I}_L}$ and thus we can use $\widehat{r}$ as the inverse of r. This is because the first concept abstraction forces (d,e) to be an ensemble on level L, e must have a $\widehat{r}$-successor f, and the second concept abstraction forces (f,e) to be an ensemble. But since ensembles cannot overlap, this is only possible when $d = f$.

We have to be careful, though, to not produce undesired overlapping ensembles which would result in unsatisfiability. As stated, in fact, the ontology $\mathcal{O}$ does not admit models in which $\mathcal{I}_L$ contains an r-path of length two. This is why we resort to $\mathcal{ALC}^{\mathsf{sym}}$. Very briefly, the r^--part of the role composition $r^-;r$ falls inside an ensemble and is implemented based on the trick outlined above while the r-part connects two different ensembles to avoid overlapping. The details are somewhat subtle and presented in the appendix.

Theorem 4. *Satisfiability in $\mathcal{ALC}^{\mathsf{abs}}[\mathrm{ca}]$ is 2EXPTIME-hard.*

5.3 Role Refinement: $\mathcal{ALC}^{\mathsf{abs}}[\mathrm{rr}]$

While concept and role abstractions enable reductions from CQ evaluation, this does not seem to be the case for concept and role refinements. Indeed, we have seen in Section 4.1 that concept refinements do not induce 2EXPTIME-hardness. Somewhat surprisingly, role refinements behave differently and are a source of 2EXPTIME-hardness, though for rather different reasons than abstraction statements.

It is well-known that there is an exponentially space-bounded alternating Turing machine (ATM) that decides a 2EXPTIME-complete problem and on any input w makes at most $2^{|w|}$ steps (Chandra, Kozen, and Stockmeyer 1981). We define ATMs in detail in the appendix and only note here that our ATMs have a one-side infinite tape and a dedicated accepting state q_a and rejecting state q_r, no successor configuration if its state is q_a or q_r, and exactly two successor configurations otherwise.

Let $M = (Q, \Sigma, \Gamma, q_0, \Delta)$ be a concrete such ATM with $Q = Q_\exists \uplus Q_\forall \uplus \{q_a, q_r\}$. We may assume w.l.o.g that M never attempts to move left when the head is positioned on the left-most tape cell. Let $w = \sigma_1 \cdots \sigma_n \in \Sigma^*$ be an input for M. We want to construct an $\mathcal{ALC}^{\mathsf{abs}}[\mathrm{rr}]$-ontology $\mathcal{O}$ and choose a concept name S and abstraction level L_1 such that S is L_1-satisfiable w.r.t. $\mathcal{O}$ iff $w \in L(M)$. Apart from S, which indicates the starting configuration, we use the following concept names:

- A_σ, for each $\sigma \in \Gamma$, to represent tape content;
- A_q, for each $q \in Q$, to represent state and head position;
- $B_{q,\sigma,M}$ for $q \in Q$, $\sigma \in \Gamma$, $M \in \{L, R\}$, serving to choose a transition;
- $H_\leftarrow, H_\rightarrow$ indicating whether a tape cell is to the right or left of the head.

plus some auxiliary concept names whose purpose shall be obvious. We use the role name t for next tape cell and c_1, c_2 for successor configurations.

The ontology $\mathcal{O}$ uses the abstraction levels $\mathbf{A} = \{L_1, \ldots, L_n\}$ with $L_{i+1} \prec L_i$ for $1 \leq i < n$. While we are interested in L_1-satisfiability of S, the computation of M is simulated on level L_n. We start with generating an infinite computation tree on level L_1:

$$S \sqsubseteq_{L_1} \exists c_1.N \sqcap \exists c_2.N \quad N \sqsubseteq_{L_1} \exists c_1.N \sqcap \exists c_2.N.$$

In the generated tree, each configuration is represented by a single object. On levels $L_2, \ldots, L_n$, we generate similar trees where, however, configurations are represented by t-paths. The length of these paths doubles with every level and each node on a path is connected via c_1 to the corresponding node in the path that represents the first successor configuration, and likewise for c_2 and the second successor configuration. This is illustrated in Figure 2 where for simplicity we only show a first successor configuration and three abstraction levels. We use the following role refinements:

$$L_{i+1}\colon q(\bar{x}, \bar{y}) \ \underline{\text{refines}}\ L_i\colon t(x, y)$$
$$L_{i+1}\colon q_j(\bar{x}, \bar{y}) \ \underline{\text{refines}}\ L_i\colon c_i(x, y)$$

for $0 \leq i < n$ and $j \in \{1, 2\}$, and where $\bar{x} = x_1x_2$,

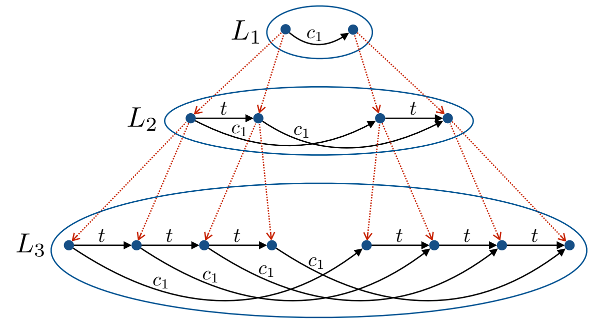


Figure 2: Dotted lines indicate refinement.

$\bar{y} = y_1y_2$ and

$$q(\bar{x}, \bar{y}) = t(x_1, x_2) \wedge t(x_2, y_1) \wedge t(y_1, y_2)$$
$$q_j(\bar{x}, \bar{y}) = t(x_1, x_2) \wedge t(y_1, y_2) \wedge c_j(x_1, y_1) \wedge c_j(x_2, y_2).$$

To make more precise what we want to achieve, let the m-*computation tree*, for $m > 0$, be the interpretation $\mathcal{I}_m$ with

$$\Delta^{\mathcal{I}_m} = \{c_0, c_1\}^* \cdot \{1, \ldots, m\}$$
$$t^{\mathcal{I}_m} = \{(wi, wj) \mid w \in \{c_0, c_1\}^*, 1 \leq i < m, j = i + 1\}$$
$$c_\ell^{\mathcal{I}_m} = \{(wj, wc_ij) \mid w \in \{c_0, c_1\}^*, 1 \leq j \leq m, i \in \{0, 1\}\}$$

for $\ell \in \{1, 2\}$. It can be shown that for any model $\mathcal{I}$ of the $\mathcal{ALC}^{\mathsf{abs}}[\mathrm{rr}]$-ontology $\mathcal{O}$ constructed so far and for all $i \in \{1, \ldots, n\}$, we must find a (homomorphic image of a) 2^i-computation tree in the interpretation $\mathcal{I}_{L_i}$. This crucially relies on the fact that ensembles cannot overlap. In Figure 2, for example, the role refinements for t and for c_1 both apply on level L_2, and for attaining the structure displayed on level L_3 it is crucial that in these applications each object on level L_2 refines into the same ensemble on level L_3.

On level L_n, we thus find a 2^n-computation tree which we use to represent the computation of M on input w. To start, the concept name S is copied down from the root of the 1-computation tree on level L_1 to that of the 2^n-computation tree on level L_n. To achieve this, we add a copy of the above role refinements for c_1, but now using the CQ

$$q_1(\bar{x}, \bar{y}) = S(x_1) \wedge c_1(x_1, y_1) \wedge c_1(x_2, y_2).$$

We next describe the initial configuration:

$$S \sqsubseteq_{L_n} A_{q_0} \sqcap A_{\sigma_1} \sqcap \forall t.A_{\sigma_2} \sqcap \cdots \sqcap \forall t^{n-1}.(A_{\sigma_n} \sqcap B_\rightarrow)$$
$$B_\rightarrow \sqsubseteq_{L_n} \forall t.(A_\square \sqcap B_\rightarrow)$$

For existential states, we consider one of the two possible successor configurations:

$$A_q \sqcap A_\sigma \sqsubseteq_{L_n} (\forall c_1.B_{q',\sigma',M'}) \sqcup (\forall c_2.B_{\bar{q},\bar{\sigma},\bar{M}})$$

for all $q \in Q_\exists$ and $\sigma \in \Gamma$ such that $\Delta(q, \sigma) = \{(q', \sigma', M'), (\bar{q}, \bar{\sigma}, \bar{M})\}$. For universal states, we use both successors:

$$A_q \sqcap A_\sigma \sqsubseteq_{L_n} (\forall c_1.B_{q',\sigma',M'}) \sqcap (\forall c_2.B_{\bar{q},\bar{\sigma},\bar{M}})$$

for all $q \in Q_\forall$ and $\sigma \in \Gamma$ such that $\Delta(q, \sigma) = \{(q', \sigma', M'), (\bar{q}, \bar{\sigma}, \bar{M})\}$. We next implement the transitions:

$$B_{q,\sigma,M} \sqsubseteq_{L_n} A_\sigma \quad \exists t.B_{q,\sigma,L} \sqsubseteq_{L_n} A_q$$
$$B_{q,\sigma,R} \sqsubseteq_{L_n} \forall t.A_q$$

for all $q \in Q$, $\sigma \in \Gamma$, and $M \in \{L, R\}$. We mark cells that are not under the head:

$$A_q \sqsubseteq_{L_n} \forall t.H_\leftarrow \qquad \exists t.A_q \sqsubseteq_{L_n} H_\rightarrow$$
$$H_\leftarrow \sqsubseteq_{L_n} \forall t.H_\leftarrow \qquad \exists t.H_\rightarrow \sqsubseteq_{L_n} H_\rightarrow$$

for all $q \in Q$. Such cells do not change:

$$(H_\leftarrow \sqcup H_\rightarrow) \sqcap A_\sigma \sqsubseteq_{L_n} \forall c_i.A_\sigma$$

for all $\sigma \in \Gamma$ and $i \in \{1, 2\}$. State, content of tape, and head position must be unique:

$$A_q \sqcap A_{q'} \sqsubseteq_{L_n} \bot \quad A_\sigma \sqcap A_{\sigma'} \sqsubseteq_{L_n} \bot$$
$$(H_\leftarrow \sqcup H_\rightarrow) \sqcap A_q \sqsubseteq_{L_n} \bot$$

for all $q, q' \in Q$ and $\sigma, \sigma' \in \Gamma$ with $q \neq q'$ and $\sigma \neq \sigma'$. Finally, all followed computation paths must be accepting:

$$A_{q_r} \sqsubseteq_{L_n} \bot.$$

This finishes the construction of $\mathcal{O}$.

Lemma 3. *S is L_1-satisfiable w.r.t. $\mathcal{O}$ iff $w \in L(M)$.*

We have thus obtained the announced result.

Theorem 5. *Satisfiability in $\mathcal{ALC}^{\mathsf{abs}}[\mathrm{rr}]$ is 2ExpTime-hard.*

6 Undecidability

One might be tempted to think that the decidability of $\mathcal{ALCHI}^{\mathsf{abs}}$ is clear given that only a finite number of abstraction levels can be mentioned in an ontology. However, achieving decidability of DLs with abstraction and refinement requires some careful design choices. In this section, we consider three seemingly harmless extensions of $\mathcal{ALCHI}^{\mathsf{abs}}$ and show that each of them results in undecidability. This is in fact already the case for $\mathcal{EL}^{\mathsf{abs}}$ where the underlying DL $\mathcal{ALCHI}$ is replaced with $\mathcal{EL}$.

6.1 Basic Observations

We make some basic observations regarding the DL $\mathcal{EL}^{\mathsf{abs}}$ and its fragments. In classical $\mathcal{EL}$, concept satisfiability is not an interesting problem because every concept is satisfiable w.r.t. every ontology. This is not the case in $\mathcal{EL}^{\mathsf{abs}}$ where we can express concept inclusions of the form $C \sqsubseteq_L \bot$ with C an $\mathcal{EL}$-concept, possibly at the expense of introducing additional abstraction levels. More precisely, let L' be the unique abstraction level with $L \prec L'$ if it exists and a fresh abstraction level otherwise. Then $C \sqsubseteq_L \bot$ can be simulated by the following CI and concept abstraction:

$$C \sqsubseteq_L \exists r_C.\exists r_C.\top$$
$$L'{:}\top \ \underline{\mathsf{abstracts}}\ L{:}\, r_C(x, y)$$

where r_C is a fresh role name. Note that this again relies on the fact that ensembles cannot overlap, and thus an r_C-path of length two in level L results in unsatisfiability. The same can be achieved by using a role abstraction in place of the concept abstraction.

To prove our undecidability results, it will be convenient to have available concept inclusions of the form $C \sqsubseteq_L \forall r.D$ with C, D $\mathcal{EL}$-concepts. Let L' be a fresh abstraction level. Then $C \sqsubseteq_L \forall r.D$ can be simulated by the following role refinement and concept abstraction:

$$L'{:}\, r(x, y) \wedge A(y) \ \underline{\mathsf{refines}}\ L{:}\, C(x) \wedge r(x, y)$$
$$L'{:}\, D \ \underline{\mathsf{abstracts}}\ L'{:}\, A(x)$$

where A is a fresh concept name. It is easy to see that the same can be achieved with a role abstraction in place of the concept abstraction.

In the following, we thus use inclusions of the forms $C \sqsubseteq_L \bot$ and $C \sqsubseteq \forall r.D$ in the context of $\mathcal{EL}^{\mathsf{abs}}$ and properly keep track of the required types of abstraction and refinement statements.

6.2 Repetition-Free Tuples

In the semantics of $\mathcal{ALCHI}^{\mathsf{abs}}$ as defined in Section 3, ensembles are tuples in which elements may occur multiple times. It would arguably be more natural to define ensembles to be repetition-free tuples. We refer to this version of the semantics as the *repetition-free* semantics.

If only concept and role refinement are admitted, then there is no difference between satisfiability under the original semantics and under the repetition-free semantics. In fact, any model of $\mathcal{O}$ under the original semantics can be converted into a model of $\mathcal{O}$ under repetition-free semantics by duplicating elements. This gives the following.

Proposition 1. *For every $\mathcal{ALCI}$-concept C, abstraction level L, and $\mathcal{ALCHI}^{\mathsf{abs}}[\mathrm{cr}, \mathrm{rr}]$-ontology $\mathcal{O}$: C is L-satisfiable w.r.t. $\mathcal{O}$ iff C is L-satisfiable w.r.t. $\mathcal{O}$ under the repetition-free semantics.*

The situation changes once we admit abstraction.

Theorem 6. *Under the repetition-free semantics, satisfiability is undecidable in $\mathcal{ALC}^{\mathsf{abs}}[\mathrm{ca}]$, $\mathcal{ALC}^{\mathsf{abs}}[\mathrm{ra}]$, $\mathcal{EL}^{\mathsf{abs}}[\mathrm{rr}, \mathrm{ca}]$, and $\mathcal{EL}^{\mathsf{abs}}[\mathrm{rr}, \mathrm{ra}]$.*

In the following, we prove undecidability for satisfiability in $\mathcal{ALC}^{\mathsf{abs}}[\mathrm{ca}]$. The result for $\mathcal{ALC}^{\mathsf{abs}}[\mathrm{ra}]$ is a minor variation and the results for $\mathcal{EL}^{\mathsf{abs}}$ are obtained by applying the observations from Section 6.1.

We reduce the complement of the halting problem for deterministic Turing machines (DTMs) on the empty tape. Assume that we are given a DTM $M = (Q, \Sigma, \Gamma, q_0, \delta)$. As in Section 5.3, we assume that M has a one-side infinite tape and never attempts to move left when the head is on the left end of the tape. We also assume that there is a dedicated halting state $q_h \in Q$.

We want to construct an $\mathcal{ALC}^{\mathsf{abs}}[\mathrm{ca}]$-ontology $\mathcal{O}$ and choose a concept name S and abstraction level L such that S is L-satisfiable w.r.t. $\mathcal{O}$ iff M does not halt on the empty tape. We use essentially the same concept and role names as in Section 5.3, except that only a single role name c is used for transitions to the (unique) next configuration. Computations are represented in the form of a grid as shown on the left-hand side of Figure 3, where the concept names X_i must be disregarded as they belong to a different reduction (and so do the queries). We use two abstraction levels L and L' with $L \prec L'$. The computation of M is represented on level L.

We first generate an infinite binary tree in which every node has one t-successor and one c-successor:

$$\top \sqsubseteq_L \exists t.\top \sqcap \exists c.\top$$

To create the desired grid structure, it remains to enforce that grid cells close, that is, the t-successor of a c-successor of

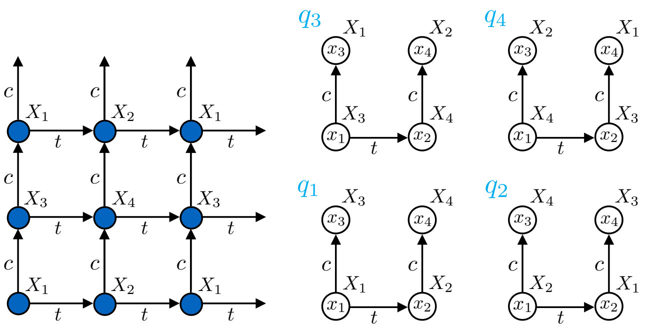


Figure 3: Grid structure and queries for the DAG Semantics.

any node coincides with the c-successor of the t-successor of that node. We add the concept abstraction

$L' \colon \bot$ <u>abstracts</u> $L\colon q(\bar{x})$ where
$q(\bar{x}) = c(x_1, x_2) \wedge t(x_1, x_3) \wedge c(x_3, x_4) \wedge t(x_2, x_4')$.

The idea is that any non-closing grid cell admits a repetition-free answer to q on $\mathcal{I}_L$, thus resulting in unsatisfiability. If all grid cells close, there will still be answers, but all of them are repetitive. The above abstraction alone, however, does not suffice to implement this idea. It still admits, for instance, a non-closing grid cell in which the two left elements have been identified. We thus need to rule out such unintended identifications and add the concept abstraction $L' \colon \bot$ <u>abstracts</u> $L\colon q$ for the following six CQs q:

$$\begin{array}{ll} t(x_1,x_2) \wedge c(x_1,x_2) & t(x_1,x_2) \wedge c(x_2,x_1) \\ t(x_1,x_2) \wedge c(x_1,x_3) \wedge t(x_3,x_2) & c(x_1,x_1) \\ t(x_1,x_2) \wedge c(x_1,x_3) \wedge c(x_2,x_3) & t(x_1,x_1) \end{array}$$

The rest of the reduction is now very similar to that given in Section 5.3, details are in the appendix.

6.3 DAG Semantics

Our semantics requires abstraction levels to be organized in a tree. While this is very natural, admitting a DAG structure might also be useful. This choice, which we refer to as the *DAG semantics*, leads to undecidability.

Theorem 7. *Under the DAG semantics, satisfiability is undecidable in* $\mathcal{ALC}^{\mathsf{abs}}[\text{ca,cr}]$ *and* $\mathcal{EL}^{\mathsf{abs}}[\text{ca, cr, rr}]$.

The result is again proved by a reduction from (the complement of) the halting problems for DTMs. In fact, the reduction differs from that in Section 6.2 only in how the grid is constructed and thus we focus on that part. We present the reduction for $\mathcal{ALC}^{\mathsf{abs}}[\text{ca, cr}]$.

Assume that we are given a DTM $M = (Q, \Sigma, \Gamma, q_0, \delta)$. We want to construct an ontology $\mathcal{O}$ and choose a concept name S and abstraction level L such that S is L-satisfiable w.r.t. $\mathcal{O}$ iff M does not halt on the empty tape. We use abstraction levels L, L_1, L_2, L_3, L_4 with $L \prec L_i$ for all $i \in \{1, \dots, 4\}$. The computation of M is simulated on level L. We start with generating an infinite t-path with outgoing infinite c-paths from every node:

$$\begin{array}{ll} S \sqsubseteq_L \exists t.A_t & A_t \sqsubseteq_L \exists t.A_t \\ A_t \sqsubseteq_L \exists c.A_c & A_c \sqsubseteq_L \exists c.A_c. \end{array}$$

In principle, we would like to add the missing t-links using the following concept abstraction and refinement:

$L_1 \colon U_1$ <u>abstracts</u> $L\colon q$
$L\colon q \wedge t(x_3, x_4)$ <u>refines</u> $L_1 \colon U_1$ where
$q = c(x_1, x_3) \wedge t(x_1, x_2) \wedge c(x_2, x_4)$.

This would then even show undecidability under the original semantics, but it does not work because it creates overlapping ensembles and thus simply results in unsatisfiability. We thus use the four abstraction levels $L_1, \dots, L_4$ in place of only L_1. This results in different kinds of ensembles on level L, one for each level L_i, and an L_i-ensemble can overlap with an L_j-ensemble if $i \neq j$. We label the grid with concept names $X_1, \dots, X_4$ as shown in Figure 3, using CIs

$$\begin{array}{ll} X_1 \sqsubseteq_L \forall c.X_3 \sqcap \forall t.X_2 & X_2 \sqsubseteq_L \forall c.X_4 \sqcap \forall t.X_1 \\ X_3 \sqsubseteq_L \forall c.X_1 \qquad X_4 \sqsubseteq_L \forall c.X_2 & S \sqsubseteq X_1. \end{array}$$

We define four variations $q_1, \dots, q_4$ of the above CQ q, as shown on the right-hand side of Figure 3, and use the following concept abstraction and refinement, for $i \in \{1, \dots, 4\}$:

$L_i \colon U_i$ <u>abstracts</u> $L\colon q_i$
$L\colon q_i \wedge t(x_3, x_4)$ <u>refines</u> $L_i \colon U_i$.

It can be verified that this eliminates overlapping ensembles and indeed generates a grid.

6.4 Quantified Variables

The final variation that we consider is syntactic rather than semantic: we admit quantified variables in CQs in abstraction and refinement statements.

Theorem 8. *In the extension with quantified variables, satisfiability is undecidable in* $\mathcal{ALC}^{\mathsf{abs}}[\text{ca,cr}]$ *and* $\mathcal{EL}^{\mathsf{abs}}[\text{ca, cr, rr}]$.

We use a DTM reduction that follows the same lines as the previous reduction and only explain how to generate a grid. We again start with an infinite t-path with outgoing infinite c-paths from every node. In the previous reduction, the main issue when adding the missing t-links was that a naive implementation creates overlapping ensembles. It is here that quantified variables help since they allow us to speak about elements without forcing them to be part of an ensemble. We use the following concept abstraction and refinement:

$L_1 \colon U_1$ <u>abstracts</u> $L\colon q$
$L\colon q \wedge t(x_3, x_4)$ <u>refines</u> $L_1 \colon U_1$ where
$q = \exists x_1 \exists x_2 \, c(x_1, x_3) \wedge t(x_1, x_2) \wedge c(x_2, x_4)$.

7 Conclusion

We have introduced DLs that support multiple levels of abstraction and include operators based on CQs for relating these levels. As future work, it would be interesting to analyse the complexity of abstraction DLs based on Horn DLs such as $\mathcal{EL}$, $\mathcal{ELI}$ and Horn-$\mathcal{ALCI}$. It would also be interesting to design an ABox formalism suitable for abstraction DLs, and to use such DLs for ontology-mediated querying. Finally, our work leaves open some decidability questions such as for $\mathcal{ALC}^{\mathsf{abs}}[\text{cr}]$ and $\mathcal{EL}^{\mathsf{abs}}[\text{cr, ca}]$ under the DAG semantics and with quantified variables.

Acknowledgments

The research reported in this paper has been supported by the German Research Foundation DFG, as part of Collaborative Research Center (Sonderforschungsbereich) 1320 Project-ID 329551904 “EASE - Everyday Activity Science and Engineering”, University of Bremen (http://www.ease-crc.org/). The research was conducted in subproject “P02 - Ontologies with Abstraction”.

This work is partly supported by BMBF (Federal Ministry of Education and Research) in DAAD project 57616814 (SECAI, School of Embedded Composite AI) as part of the program Konrad Zuse Schools of Excellence in Artificial Intelligence.

References

Artale, A.; Franconi, E.; Guarino, N.; and Pazzi, L. 1996. Part-whole relations in object-centered systems: An overview. *Data & Knowledge Engineering* 20(3):347–383.

Baader, F.; Horrocks, I.; Lutz, C.; and Sattler, U. 2017. *An Introduction to Description Logic*. Cambridge University Press.

Bittner, T., and Smith, B. 2003. A theory of granular partitions. In *Foundations of Geographic Information Science*. Taylor & Francis. 117–149.

Calegari, S., and Ciucci, D. 2010. Granular computing applied to ontologies. *Int. J. Approx. Reason.* 51(4):391–409.

Chandra, A. K.; Kozen, D. C.; and Stockmeyer, L. J. 1981. Alternation. *Journal of the ACM (JACM)* 28(1):114–133.

Cima, G.; Console, M.; Lenzerini, M.; and Poggi, A. 2022. Monotone abstractions in ontology-based data management. In *Proc. of AAAI*, 5556–5563. AAAI Press.

Dams, D., and Grumberg, O. 2018. Abstraction and abstraction refinement. In *Handbook of Model Checking*. Springer. 385–419.

Fonseca, F. T.; Egenhofer, M. J.; Jr., C. A. D.; and Câmara, G. 2002. Semantic granularity in ontology-driven geographic information systems. *Ann. Math. Artif. Intell.* 36(1-2):121–151.

Gangemi, A., and Mika, P. 2003. Understanding the semantic web through descriptions and situations. In *Proc. of OTM*, volume 2888 of *LNCS*, 689–706. Springer.

Glimm, B.; Kazakov, Y.; and Tran, T. 2017. Ontology materialization by abstraction refinement in horn SHOIF. In *Proc. of AAAI*, 1114–1120. AAAI Press.

Hbeich, E.; Roxin, A.; and Bus, N. 2021. Connecting granular and topological relations through description logics (short paper). In *Proc. of the GeoLD Workshop*, volume 2977 of *CEUR Workshop Proceedings*, 97–104. CEUR-WS.org.

Hobbs, J. R. 1985. Granularity. In *Proc. of IJCAI*, 432–435. Morgan Kaufmann.

Keet, C. M. 2008. *A Formal Theory of Granularity*. Ph.D. Dissertation, KRDB Research Centre, Faculty of Computer Science, Free University of Bozen-Bolzano.

Klarman, S., and Gutiérrez-Basulto, V. 2016. Description logics of context. *J. Log. Comput.* 26(3):817–854.

Klinov, P.; Taylor, J. M.; and Mazlack, L. J. 2008. Interval rough mereology and description logic: An approach to formal treatment of imprecision in the semantic web ontologies. *Web Intell. Agent Syst.* 6(2):157–174.

Lisi, F. A., and Mencar, C. 2018. A granular computing method for OWL ontologies. *Fundam. Informaticae* 159(1-2):147–174.

Lutz, C. 2008. The complexity of conjunctive query answering in expressive description logics. In *Proc. of IJCAR*, volume 5195 of *LNCS*, 179–193. Springer.

Manière, Q. 2022. *Counting queries in ontology-based data access*. Ph.D. Dissertation, Université de Bordeaux.

Schulz, S.; Boeker, M.; and Stenzhorn, H. 2008. How granularity issues concern biomedical ontology integration. In *Proc. of MIE*, volume 136 of *Studies in Health Technology and Informatics*, 863–868. IOS Press.

Segoufin, L., and ten Cate, B. 2013. Unary negation. *Log. Methods Comput. Sci.* 9(3).

Stearns, M. Q.; Price, C.; Spackman, K. A.; and Wang, A. Y. 2001. SNOMED clinical terms: overview of the development process and project status. In *Proc. of AMIA*.

Vogt, L. 2019. Levels and building blocks - toward a domain granularity framework for the life sciences. *J. Biomed. Semant.* 10(1):4:1–4:29.

A Preliminaries

We introduce some additional preliminaries that are used in the detailed proofs provided in the appendix.

A.1 Conservative Extensions

A *signature* is a set of concept and role names that in this context are uniformly referred to as *symbols*. The set of symbols that occur in an ontology $\mathcal{O}$ is denoted by $\mathsf{sig}(\mathcal{O})$. Note that this does not include abstraction levels.

Given two ontologies $\mathcal{O}_1$ and $\mathcal{O}_2$, we say that $\mathcal{O}_2$ is a *conservative extension* of $\mathcal{O}_1$ if

1. $\mathsf{sig}(\mathcal{O}_1) \subseteq \mathsf{sig}(\mathcal{O}_2)$,
2. every model of $\mathcal{O}_2$ is a model of $\mathcal{O}_1$, and
3. for every model $\mathcal{I}_1$ of $\mathcal{O}_1$ there exists a model $\mathcal{I}_2$ of $\mathcal{O}_2$ such that

   $A^{\mathcal{I}_1} = A^{\mathcal{I}_2}$ for all concept names $A \in \mathsf{sig}(\mathcal{O}_1)$ and
   $r^{\mathcal{I}_1} = r^{\mathcal{I}_2}$ for all role names $r \in \mathsf{sig}(\mathcal{O}_1)$.

A.2 Alternating Turing Machines

We briefly recall the definition of alternating Turing machines (ATMs). An ATM M is a tuple $M = (Q, \Sigma, \Gamma, q_0, \Delta)$ where

- $Q = Q_\exists \uplus Q_\forall \uplus \{q_a, q_r\}$ is the set of *states* and consists of *existentital states* in $Q_\exists$, *universal states* in $Q_\forall$, an *accepting state* q_a, and a *rejecting state* q_r;
- Σ is the *input alphabet*;
- Γ is the *work alphabet* that contains the *blank symbol* $\square$ and satisfies $\Sigma \subseteq \Gamma$;
- $q_0 \in Q_\exists \cup Q_\forall$ is the *starting state*; and
- $\Delta \subseteq Q \setminus \{q_a, q_r\} \times \Gamma \times Q \times \Gamma \times \{L, R\}$ is the *transition relation*.

We use $\Delta(q, \sigma)$ to denote

$$\{(q', \sigma', M) \mid (q, \sigma, q', \sigma', M) \in \Delta\}$$

and assume w.l.o.g. $|\Delta(q, \sigma)| = 2$ for all such sets.

A *configuration* of an ATM is a word wqw' with $w, w' \in \Sigma^*$ and $q \in Q$. This has the intended meaning that the one-sided tape contains ww' followed by only blanks, the ATM is in state q, and the head is on the symbol just after w. The *successor configurations* of a configuration are defined as usual in terms of the transition relation Δ. A *halting configuration* is of the form wqw' with $q \in \{q_a, q_r\}$.

A *computation* of an ATM M on a word w is a (finite or infinite) sequence of configurations $K_0, K_1, \dots$ such that $K_0 = q_0 w$ and K_{i+1} is a successor configuration of K_i for all $i \geq 0$. The *acceptance* of a configuration $K = wqw'$ depends on q: if $q = q_a$ then K is accepting; if $q = q_r$ then K is rejecting; if $q \in Q_\exists$, then K is accepting iff at least one successor configuration is accepting; if $q \in Q_\forall$, then K is accepting iff both successor configuration are accepting. An ATM M with starting state q_0 *accepts* the word w if the *initial configuration* $q_0 w$ is accepting. We use $L(M)$ to denote the language M is accepting.

B Proofs for Section 4.1

We prove the correctness of the algorithm, stated as Lemma 1 in the main body of the paper. We also analyze the running time of the algorithm.

We repeat Lemma 1 here for the reader's convenience.

Lemma 1. *C_0 is L_0-satisfiable w.r.t. $\mathcal{O}$ iff $\mathcal{M}^*$ contains (i) a mosaic M with $L^M = L_0$ and $C_0^{\mathcal{I}^M} \neq \emptyset$ and (ii) a mosaic M with $L^M = L$, for every L in $\mathbf{A}_\mathcal{O}$.*

We split the proof of Lemma 1 into a soundness part ("if" direction) and a completeness part ("only if" direction).

B.1 Soundness

Assume that our algorithm returns 'satisfiable'. This implies that there is a set $\mathcal{M}$ of mosaics in which all mosaics are good, that contains a mosaic M_0 with $L^{M_0} = L_0$ and $C_0^{\mathcal{I}^M_{L_0}} \neq \emptyset$, and for each $L \in \mathbf{A}_\mathcal{O}$ it contains a mosaic of level L. We have to show that we can construct a model $\mathcal{I}$ of $\mathcal{O}$ such that ${C_0}^{\mathcal{I}_{L_0}} \neq \emptyset$.

Note that throughout the following construction, when working with multiple mosaics we generally consider their interpretation domains to be disjoint; this can be easily achieved by renaming. In the following, we construct a sequence $\mathcal{I}^0, \mathcal{I}^1, \dots$ of A-interpretations and obtain the desired model $\mathcal{I}$ in the limit. For bookkeeping purposes, along with $\mathcal{I}^0, \mathcal{I}^1, \dots$ we construct a mapping M that assigns a mosaic $M(d)$ to every domain element $d \in \bigcup_{i \geq 0} \Delta^{\mathcal{I}^i}$.

We start with defining $\mathcal{I}^0 = (\prec, (\mathcal{I}^0_L)_{L \in \mathbf{A}_\mathcal{O}}, \rho^0)$ by setting

$$\mathcal{I}^0_L = \biguplus_{\substack{M \in \mathcal{M} \\ L^M = L}} \mathcal{I}^M \qquad \text{for all } L \in \mathbf{A}_\mathcal{O};$$
$$\rho^0 = \emptyset;$$

and set $M(d) = M$ if $d \in \Delta^{\mathcal{I}^M}$, for all $d \in \Delta^{\mathcal{I}^0}$ and $M \in \mathcal{M}^*$.

To construct $\mathcal{I}^{i+1} = (\prec, (\mathcal{I}^{i+1}_L)_{L \in \mathbf{A}_\mathcal{O}}, \rho^{i+1})$ from $\mathcal{I}^i$ we start with $\mathcal{I}^{i+1} = \mathcal{I}^i$. Then for every $L \in \mathbf{A}_\mathcal{O}$:

1. consider every $A \sqsubseteq_L \exists R.B \in \mathcal{O}$ and $d \in A^{\mathcal{I}^i_L}$ with $d \notin (\exists R.B)^{\mathcal{I}^i_L}$. Condition 1 of goodness of mosaics implies that there is a mosaic $M' \in \mathcal{M}$ with $L^{M'} = L$ and a $d' \in \Delta^{\mathcal{I}^{M'}}$ that satisfy the subconditions $1(a)$ to (c) w.r.t. $M(d)$. We do the following:
   - (disjointly) add $\mathcal{I}^{M'}$ to $\mathcal{I}^{i+1}_L$ and
   - add (d, d') to $S^{\mathcal{I}^{i+1}_L}$ for all $R \sqsubseteq_L S \in \mathcal{O}$.

   Also set $M(d) = M'$ for all $d \in \Delta^{\mathcal{I}^{M'}}$.
2. consider every concept refinement $L'{:}q(\bar{x})$ <u>refines</u> $L{:}A$ in $\mathcal{O}$ such that $d \in A^{\mathcal{I}^i_L}$. Specifically let $Q = \{q \mid L'{:}q(\bar{x})$ <u>refines</u> $L{:}A \in \mathcal{O}$ and $d \in A^{\mathcal{I}}\}$. If $Q \neq \emptyset$, then by Condition 2 of goodness of mosaics there is a mosaic $M' \in \mathcal{M}$ with $L^{M'} = L$ and tuple $\bar{e}$ over $\Delta^{\mathcal{I}^{M'}}$ such that $\bar{e} \in q(\mathcal{I}^{M'})$ for all $q \in Q$. We do the following:
   - (disjointly) add $\mathcal{I}^{M'}$ to $\mathcal{I}^{i+1}_{L'}$,

- (disjointly) add $\bar{e}$ to $\rho_{L'}^{i+1}(d)$.

Also set $M(d) = M'$ for all $d \in \Delta^{\mathcal{I}^{M'}}$.

As announced, we take $\mathcal{I}$ to be the limit of the constructed sequence of $\mathcal{I}^0, \mathcal{I}^1, \ldots$.

What remains to show is that $\mathcal{I}$ is a model of $\mathcal{O}$ and that C_0 is satisfied on L_0. We do this step by step starting with the basic condition that $\mathcal{I}$ is an A-interpretation. Subsequently, we will show that it satisfies all CIs and RIs in $\mathcal{O}$ and lastly that it satisfies all concept refinements in $\mathcal{O}$ and also that C_0 is satisfied on L_0.

Lemma 4. *$\mathcal{I} = (\prec, (\mathcal{I}_L)_{L \in \mathbf{A}_\mathcal{O}}, \rho)$ is an A-interpretation.*

Proof. We go through the three conditions of A-interpretations.

- our relation "$\prec$" is such that the directed graph $(\mathbf{A}_\mathcal{I}, \{(L', L) \mid L \prec L'\})$ is a tree, since our algorithm did not abort;
- by definition, $\Delta^{\mathcal{I}_L^0}$ is non-empty for all $L \in \mathbf{A}_\mathcal{O}$; thus the same holds for $\Delta^{\mathcal{I}_L}$;
- whenever we added a tuple to the range of ρ in the construction of $\mathcal{I}$, all the elements were part of a freshly added mosaic. Thus there is always at most one $d \in \Delta^\mathcal{I}$ such that e occurs in $\rho_{L(e)}(d)$.

❏

Lemma 5. *$\mathcal{I}$ satisfies all CIs and RIs in $\mathcal{O}$.*

Proof. First, consider any role inclusion $R \sqsubseteq_L S \in \mathcal{O}$. For an edge $(d, e) \in R^{\mathcal{I}_L}$ there are two cases. If (d, e) was part of a mosaic then the definition of mosaics and our construction imply $(d, e) \in S^{\mathcal{I}_L}$. Otherwise (d, e) was added to $R^{\mathcal{I}_L}$ in Step 1 of the construction of $\mathcal{I}$ which immediately implies $(d, e) \in S^{\mathcal{I}_L}$ since this is part of Step 1.

Now we go through each form a CI in $\mathcal{O}$ can have.

- $\top \sqsubseteq_L A, A_1 \sqcap A_2 \sqsubseteq_L A, A \sqsubseteq_L \neg B, \neg B \sqsubseteq_L \neg A$, are satisfied because of the definition of mosaics and the construction of $\mathcal{I}$.
- If $A \sqsubseteq_L \exists R.B \in \mathcal{O}$ and $d \in A^{\mathcal{I}_L}$ then Step 1 in the construction of $\mathcal{I}$ implies that there is a $d' \in B^{\mathcal{I}_L}$ with $(d, d') \in R^{\mathcal{I}_L}$.
- If $\exists R.B \sqsubseteq_L A \in \mathcal{O}$ and $(d, d') \in R^{\mathcal{I}_L}$ with $d' \in B^{\mathcal{I}_L}$, then there are two cases. First, consider the case that $M(d) = M(d')$. Then the definition of mosaics and our construction imply $d \in A^{\mathcal{I}_L}$. If $M(d) \neq M(d')$ then (d, d') must have been added to $R^{\mathcal{I}_L}$ in Step 1 of constructing $\mathcal{I}$ which in turn implies $d \in A^{\mathcal{I}_L}$.

❏

The following lemma establishes soundness of the algorithm.

Lemma 6. *$\mathcal{I}$ is a model of $\mathcal{O}$ with ${C_0}^{\mathcal{I}_{L_0}} \neq \emptyset$.*

Proof. By Lemma 4, $\mathcal{I}$ is an A-interpretation and by Lemma 5, it satisfies all CIs and RIs in $\mathcal{O}$. Since $M_0 \in \mathcal{M}$, our construction of $\mathcal{I}^0$ implies that ${C_0}^{\mathcal{I}^0_{L_0}} \neq \emptyset$ and by the construction of $\mathcal{I}$ this holds true for $\mathcal{I}$ as well. It only remains to show that all concept refinements are satisfied but this follows directly from Step 2 in the construction of $\mathcal{I}$ since it ensures that all concept refinements are satisfied in $\mathcal{I}^i$ for each i and thus also in $\mathcal{I}$. ❏

B.2 Completeness

Assume that there is a model $\mathcal{I} = (\prec, (\mathcal{I}_L)_{L \in \mathbf{A}_\mathcal{O}}, \rho)$ of $\mathcal{O}$ with ${C_0}^{\mathcal{I}_{L_0}} \neq \emptyset$. We have to show that our algorithm returns 'satisfiable'.

Let $V \subseteq \Delta^{\mathcal{I}_L}$ be a set of domain elements for some $L \in \mathbf{A}_\mathcal{O}$. If $|V| \leq ||\mathcal{O}||$, we use $M_V = (L, \mathcal{J})$ to denote the mosaic *of V in $\mathcal{I}_L$* where $\mathcal{J}$ is defined as follows:

$$\begin{aligned} \Delta^\mathcal{J} &= V \\ A^\mathcal{J} &= A^{\mathcal{I}_L} \cap V \qquad \text{for all } A \text{ in } \mathsf{sig}(\mathcal{O}) \cap \mathbf{C} \\ r^\mathcal{J} &= \{(d, e) \mid d, e \in V \text{ and } (d, e) \in r^{\mathcal{I}_L}\} \\ &\qquad\qquad \text{for all } r \text{ in } \mathsf{sig}(\mathcal{O}) \cap \mathbf{R} \end{aligned}$$

It is easy to verify that M_V is a mosaic since $\mathcal{I}$ is a model of $\mathcal{O}$ and $|\Delta^\mathcal{J}| \leq ||\mathcal{O}||$. Technically we have to rename elements which can be done by defining a simple injective function that maps $\Delta^\mathcal{J}$ to Δ. If we renamed an element $d \in \Delta^\mathcal{I}$ to $e \in \Delta^{\mathcal{I}^{M_V}}$, then we use $e^\downarrow$ to denote d.

The algorithm computes the final set $\mathcal{M}^*$ of mosaics by starting with the set $\mathcal{M}_0$ of all mosaics for $\mathcal{O}$ and then repeatedly and exhaustively eliminating mosaics that are not good. Let us assume the computation did n eliminations of mosaics that were not good. We then use $\mathcal{M}_0, \mathcal{M}_1, \ldots, \mathcal{M}_n$ to denote the sets of the computation, implying $\mathcal{M}_n = \mathcal{M}^*$.

We now want to show that all the M_Vs are contained in $\mathcal{M}^*$ since that will trivially imply that our algorithm returns 'satisfiable'.

Lemma 7. *For all $i \leq n$:*

$$\{M_V \mid \exists L \in \mathbf{A}_\mathcal{O} \text{ s.t. } V \subseteq \Delta^{\mathcal{I}_L} \text{ with } |V| \leq ||\mathcal{O}||\} \subseteq \mathcal{M}_i.$$

Proof. The proof is by induction on i. For $i = 0$, our assumption follows from the fact that all the M_Vs are mosaics.

For $i > 0$, let us consider a mosaic $M_V \in \mathcal{M}_i$ for any $L \in \mathbf{A}_\mathcal{O}$ and $V \subseteq \Delta^{\mathcal{I}_L}$ with $|V| \leq ||\mathcal{O}||$. Let $A \sqsubseteq_L \exists R.B \in \mathcal{O}$, $d \in A^{\mathcal{I}^{M_V}}$, and $d \notin (\exists R.B)^{\mathcal{I}^{M_V}}$. We have to show that there are a mosaic $M' \in \mathcal{M}_i$ with $L^{M'} = L$ and a domain element $d' \in \Delta^{\mathcal{I}^{M'}}$ that satisfy Condition 1 of goodness of mosaics. From the construction of M_V we know that $d^\downarrow \in (\exists R.B)^{\mathcal{I}_L}$ and $\mathcal{I}$ being a model of $\mathcal{O}$ implies that there is an $e \in A^{\mathcal{I}_L}$ with $(d^\downarrow, e) \in R^{\mathcal{I}_L}$. By the IH $M_{\{e\}} \in \mathcal{M}_i$, and thus Condition 1 of goodness is satisfied.

For Condition 2 of goodness, let $d \in A^{\mathcal{I}^{M_V}}$ and $Q = \{q \mid L'{:}q(\bar{x}) \ \underline{\text{refines}}\ L{:}A \in \mathcal{O}\} \neq \emptyset$. The construction of M_V implies that $d^\downarrow \in A^{\mathcal{I}_L}$ and since $\mathcal{I}$ is a model there is a tuple $\bar{e} \in \rho_{L'}(d)$ such that $\bar{e} \in q(\mathcal{I}_L)$ for all $q \in Q$. Now we can apply the IH to get $M_{\{e \mid e \text{ in } \bar{e}\}} \in \mathcal{M}_i$ and thus Condition 2 of goodness is satisfied as well. ❏

Now since $\mathcal{I}$ is a model there is at least one element $d \in \Delta^{\mathcal{I}_L}$ for all $L \in \mathbf{A}_\mathcal{O}$, and also an element $d_0 \in {C_0}^{\mathcal{I}_{L_0}}$. We can now apply Lemma 7 to imply that $M_{\{d\}} \in \mathcal{M}^*$ for all these mentioned domain elements and thus the conditions for our algorithm to return 'satisfiable' are met.

B.3 Running Time

Since the domain of a mosaic has a maximum of $||\mathcal{O}||$ elements, there are at most $2^{\mathsf{poly}(||\mathcal{O}||)}$ different mosaics. Consequently the length of the constructed sequence $\mathcal{M}_0, \mathcal{M}_1, \ldots$ is also bounded by $2^{\mathsf{poly}(||\mathcal{O}||)}$. Checking goodness of mosaics requires us to compute answers to CQs. Doing this in a brute force way results in checking at most $||\mathcal{O}||^{||\mathcal{O}||} \in 2^{\mathsf{poly}(||\mathcal{O}||)}$ candidates for homomorphisms from a CQ into a mosaic (for a fixed answer). In summary, the running time of our algorithm is at most $2^{\mathsf{poly}(||\mathcal{O}||)}$.

C Proofs for Section 4.2

In the main part of the paper, we have omitted details regarding the converses of Conditions 2-5 of edge candidates. We start with giving a complete list of conditions, including the converses.

Let $M = ((\mathcal{I}_L)_{L\in\mathbf{A}_\mathcal{O}}, \rho, f_{\mathsf{in}}, f_{\mathsf{out}})$ and $M' = ((\mathcal{I}'_L)_{L\in\mathbf{A}_\mathcal{O}}, \rho', f'_{\mathsf{in}}, f'_{\mathsf{out}})$. A set E of M, M'-edges is an *edge candidate* if it satisfies the following conditions:

1. $R(d,e) \in E$ and $L(d) = L$ implies $S(d,e) \in E$, for all $R \sqsubseteq_L S \in \mathcal{O}$;
2. if $\exists R.A \sqsubseteq_L B \in \mathcal{O}$, $R(d,d') \in E$, and $d' \in A^{\mathcal{I}'_L}$, then $d \in B^{\mathcal{I}_L}$;

2′. if $\exists R^-.A \sqsubseteq_L B \in \mathcal{O}$, $R(d,d') \in E$, and $d \in A^{\mathcal{I}_L}$, then $d' \in B^{\mathcal{I}_L}$;

3. for all $L \in \mathbf{A}_\mathcal{O}$, all $(q,h) \in f_{\mathsf{out}}(L)$, where $q = E_q \uplus q|_{\overline{V}}$ for $V = \mathsf{dom}(h)$, and all functions g from $\overline{V} \cap \mathsf{var}(E_q)$ to $\Delta^{\mathcal{I}'_L}$ such that $R(h(x), g(y)) \in E$ for all $R(x,y) \in E_q$, we have $(q|_{\overline{V}}, g) \in f'_{\mathsf{in}}(L)$;

3′. for all $L \in \mathbf{A}_\mathcal{O}$, all $(q,h) \in f'_{\mathsf{out}}(L)$, where $q = E_q \uplus q|_{\overline{V}}$ for $V = \mathsf{dom}(h)$, and all functions g from $\overline{V} \cap \mathsf{var}(E_q)$ to $\Delta^{\mathcal{I}_L}$ such that $R(h(x), g(y)) \in E$ for all $R(x,y) \in E_q$, we have $(q|_{\overline{V}}, g) \in f_{\mathsf{in}}(L)$;

4. for all $R(d,d') \in E$ and all role refinements $L{:}q(\bar{x},\bar{y})$ <u>refines</u> $L'{:}q_R(x,y) \in \mathcal{O}$, such that $q = q|_{\bar{x}} \uplus E_q \uplus q|_{\bar{y}}$, $q_R = C_x(x) \wedge R(x,y) \wedge C_y(y)$, $C_x \in t(d)$, and $C_y \in t'(d')$:
   (a) $\rho_L(d)$ and $\rho'_L(d')$ are defined;
   (b) $h : \bar{x} \mapsto \rho_L(d)$ is a homomorphism from $q|_{\bar{x}}$ to $\mathcal{I}_L$;
   (c) $h' : \bar{y} \mapsto \rho'_L(d')$ is a homomorphism from $q|_{\bar{y}}$ to $\mathcal{I}'_L$;
   (d) $\{R(h(x), h'(y)) \mid R(x,y) \in E_q\} \subseteq E$;

4′. for all $R(d',d) \in E$ and role refinements $L{:}q(\bar{x},\bar{y})$ <u>refines</u> $L'{:}q_R(x,y) \in \mathcal{O}$, such that $q = q|_{\bar{x}} \uplus E_q \uplus q|_{\bar{y}}$, $q_R = C_x(x) \wedge R(x,y) \wedge C_y(y)$, $C_x \in t'(d')$, and $C_y \in t(d)$:
   (a) $\rho_L(d)$ and $\rho'_L(d')$ are defined;
   (b) $h : \bar{x} \mapsto \rho'_L(d')$ is a homomorphism from $q|_{\bar{x}}$ to $\mathcal{I}'_L$;
   (c) $h' : \bar{y} \mapsto \rho_L(d)$ is a homomorphism from $q|_{\bar{y}}$ to $\mathcal{I}_L$;
   (d) $\{R(h(x), h'(y)) \mid R(x,y) \in E_q\} \subseteq E$;

5. for all role abstractions $L'{:}R$ <u>abstracts</u> $L{:}q(\bar{x},\bar{y}) \in \mathcal{O}$, where $q = q|_{\bar{x}} \uplus E_q \uplus q|_{\bar{y}}$, all homomorphisms h from $q|_{\bar{x}}$ to $\mathcal{I}_L$, and all homomorphisms g from $q|_{\bar{y}}$ to $\mathcal{I}'_L$ such that $\{S(h(x), g(y)) \mid S(x,y) \in E_q\} \subseteq E$, there are $d \in \Delta^{\mathcal{I}_{L'}}$ and $d' \in \Delta^{\mathcal{I}'_{L'}}$ with $\rho_L(d) = h(\bar{x})$, $\rho'_L(d') = g(\bar{y})$, and $R(d,d') \in E$;

5′. for all role abstractions $L'{:}R$ <u>abstracts</u> $L{:}q(\bar{x},\bar{y}) \in \mathcal{O}$, where $q = q|_{\bar{x}} \uplus E_q \uplus q|_{\bar{y}}$, all homomorphisms h from $q|_{\bar{x}}$ to $\mathcal{I}'_L$, and all homomorphisms g from $q|_{\bar{y}}$ to $\mathcal{I}_L$ such that $\{S(h(x), g(y)) \mid S(x,y) \in E_q\} \subseteq E$, there are $d \in \Delta^{\mathcal{I}_{L'}}$ and $d' \in \Delta^{\mathcal{I}'_{L'}}$ with $\rho_L(d) = g(\bar{y})$, $\rho'_L(d') = h(\bar{x})$, and $R(d',d) \in E$.

We now prove the correctness of the algorithm, stated as Lemma 2 in the main body of the paper. We also analyze the running time of the algorithm.

We repeat Lemma 2 here for the reader's convenience.

Lemma 2. *C_0 is L_0-satisfiable w.r.t. $\mathcal{O}$ iff $\mathcal{M}^*$ contains (i) a mosaic M with $C_0^{\mathcal{I}^M_{L_0}} \neq \emptyset$ and (ii) a mosaic M with $\Delta^{\mathcal{I}^M_L} \neq \emptyset$, for every L in $\mathbf{A}_\mathcal{O}$.*

We split the proof of Lemma 2 into a soundness part ("if" direction) and a completeness part ("only if" direction).

C.1 Soundness

Assume that our algorithm returns 'satisfiable' and thus satisfies Condition (i) and (ii) of Lemma 2. We have to show that we can construct a model $\mathcal{I}$ of $\mathcal{O}$ such that $C_0{}^{\mathcal{I}_{L_0}} \neq \emptyset$.

Note that throughout the following construction, when working with multiple mosaics we generally consider their interpretation domains to be disjoint; this can be easily achieved by renaming. In the following, we construct a sequence $\mathcal{I}^0, \mathcal{I}^1, \ldots$ of A-interpretations and obtain the desired model $\mathcal{I}$ in the limit. For bookkeeping purposes, along with $\mathcal{I}^0, \mathcal{I}^1, \ldots$ we construct a mapping M that assigns a mosaic $M(d)$ to every domain element $d \in \bigcup_{i\geq 0} \Delta^{\mathcal{I}^i}$.

We start with defining $\mathcal{I}^0 = (\prec, (\mathcal{I}^0_L)_{L\in\mathbf{A}_\mathcal{O}}, \rho^0)$ by setting

$$\mathcal{I}^0_L = \biguplus_{M\in\mathcal{M}^*} \mathcal{I}^M_L \text{ for all } L \in \mathbf{A}_\mathcal{O};$$
$$\rho^0 = \biguplus_{M\in\mathcal{M}^*} \rho^M.$$

and set $M(d) = M$ if $d \in \Delta^{\mathcal{I}^M}$, for all $d \in \Delta^{\mathcal{I}^0}$ and $M \in \mathcal{M}^*$.

To construct $\mathcal{I}^{i+1} = (\prec, (\mathcal{I}^{i+1}_L)_{L\in\mathbf{A}_\mathcal{O}}, \rho^{i+1})$ from $\mathcal{I}^i$ we start with $\mathcal{I}^{i+1} = \mathcal{I}^i$. Then consider every $L \in \mathbf{A}_\mathcal{O}$, $A \sqsubseteq_L \exists R.B \in \mathcal{O}$ and $d \in A^{\mathcal{I}^i_L}$ with $d \notin (\exists R.B)^{\mathcal{I}^i_L}$. By the goodness condition of mosaics, there is a mosaic $M' \in \mathcal{M}$, a $d' \in B^{\mathcal{I}^{M'}_L}$ and an M, M' edge candidate E such that $R(d,d') \in E$.

For all $L \in \mathbf{A}_\mathcal{O}$, do the following:

- (disjointly) add $\mathcal{I}^{M'}_L$ to $\mathcal{I}^{i+1}_L$,
- (disjointly) add $\rho^{M'}$ to ρ^{i+1}, and
- add $\{(e,e') \mid R(e,e') \in E \text{ and } L(e) = L(e') = L\}$ to $R^{\mathcal{I}^{i+1}_L}$.

Also set $M(d) = M'$ for all $d \in \Delta^{\mathcal{I}^{M'}_L}$.

If an element $d \in \Delta^{\mathcal{I}^i}$ for any $i \geq 0$ was introduced as a copy of $e \in \Delta^{\mathcal{I}^M}$ for some $M \in \mathcal{M}$, then we use $d^\uparrow$ to denote e. As announced, we take $\mathcal{I}$ to be the limit of the constructed sequence of $\mathcal{I}^0, \mathcal{I}^1, \ldots$.

What remains to show is that $\mathcal{I}$ is a model of $\mathcal{O}$ and that C_0 is satisfied on L_0. We do this step by step starting with the basic condition that $\mathcal{I}$ is an A-interpretation. After that we will show that it satisfies all CIs and RIs in $\mathcal{O}$ and lastly that it satisfies all abstraction and refinement statements in $\mathcal{O}$ and that C_0 is satisfied on L_0.

Lemma 8. *$\mathcal{I} = (\prec, (\mathcal{I}_L)_{L \in \mathbf{A}_\mathcal{O}}, \rho)$ is an A-interpretation.*

Proof. We go through the three conditions of A-interpretations.

- our relation "$\prec$" is such that the directed graph $(\mathbf{A}_\mathcal{I}, \{(L', L) \mid L \prec L'\})$ is a tree, since our algorithm did not abort;
- by definition, $\Delta^{\mathcal{I}_L^0}$ is non-empty for all $L \in \mathbf{A}_\mathcal{O}$; thus the same holds for $\Delta^{\mathcal{I}_L}$;
- all the ρ^M are already refinement functions and we only add them disjointly to $\mathcal{I}$ in every step of the construction. Thus every object participates in at most one ensemble.

❏

Lemma 9. *$\mathcal{I}$ satisfies all CIs and RIs in $\mathcal{O}$.*

Proof. First, consider any role inclusion $R \sqsubseteq_L S \in \mathcal{O}$. For an edge $(d, e) \in R^{\mathcal{I}_L}$ there are two cases. The edge might have been part of a mosaic. Then Condition 1 of mosaics and our construction imply $(d, e) \in S^{\mathcal{I}_L}$. Otherwise (d, e) was added to $R^{\mathcal{I}_L}$ in Step 1 of the construction of $\mathcal{I}$ and then Condition 1 of edge candidates implies $(d, e) \in S^{\mathcal{I}_L}$.

Now we go through each form a CI in $\mathcal{O}$ can have.

- $\top \sqsubseteq_L A, A_1 \sqcap A_2 \sqsubseteq_L A, A \sqsubseteq_L \neg B, \neg B \sqsubseteq_L \neg A$, are satisfied because of Condition 1 of mosaics and the construction of $\mathcal{I}$.
- If $A \sqsubseteq_L \exists R.B \in \mathcal{O}$ and $d \in A^{\mathcal{I}_L}$ then Step 1 in the construction of $\mathcal{I}$ implies that there is a $d' \in B^{\mathcal{I}_L}$ with $(d, d') \in R^{\mathcal{I}_L}$.
- If $\exists R.B \sqsubseteq_L A \in \mathcal{O}$ and $(d, d') \in R^{\mathcal{I}_L}$ with $d' \in B^{\mathcal{I}_L}$, then there are two cases. First, consider the case that $M(d) = M(d')$. Then Condition 1 of mosaics and our construction imply $d \in A^{\mathcal{I}_L}$. If $M(d) \neq M(d')$ then (d, d') must have been added to $R^{\mathcal{I}_L}$ in Step 1 of constructing $\mathcal{I}$ which implies $d \in A^{\mathcal{I}_L}$ because of Conditions 2 and 2' of edge candidates.

❏

Before showing that $\mathcal{I}$ also satisfies all abstractions and refinements in $\mathcal{O}$ we show two intermediary lemmas. Intuitively they say that the L-ensembles that are the result of matching abstraction CQs are never part of multiple mosaics.

Lemma 10. *For all concept abstractions $L'{:}A$ $\underline{\mathsf{abstracts}}$ $L{:}q(\bar{x})$ in $\mathcal{O}$ and all homomorphisms h from q to $\mathcal{I}_L$, $M(d) = M(d')$ for all $d, d' \in \mathsf{ran}(h)$.*

Proof. Proof by contradiction. Assume there is a homomorphsim h from q to $\mathcal{I}_L$ such that for some elements $d, d' \in \mathsf{ran}(h)$, we have $M(d) \neq M(d')$. Let $M(d) = M_0$ and $V_0 = \{x \mid x \in \mathsf{var}(q) \text{ and } M(h(x)) = M_0\}$ be the set of variables in q that are matched into M_0. Note that by our assumption V_0 is neither empty nor equal to $\mathsf{var}(q)$. Thus Condition 2 of mosaics applies, which states that there is a component p_0 of q w.r.t. V_0 with $p_0 = E_{p_0} \uplus p_0|_{\overline{V_0} \cap \mathsf{var}(p_0)}$ and $(p_0, h|_{V_0 \cap \mathsf{var}(p_0)}) \in f_{\mathsf{out}}^{M_0}(L)$.

Let $V_1 = \overline{V_0} \cap \mathsf{var}(E_{p_0})$. Then V_1 is non-empty and $M(h(x)) \neq M_0$ for all $x \in V_1$, by our assumption. Furthermore, $M(h(x)) = M(h(y))$ for all $x, y \in V_1$, by the definition of components. Let $M_1 = M(h(x))$ denote this mosaic that is the same for all $x \in V_1$. The construction of $\mathcal{I}$ implies that in our algorithm there was an M_0, M_1 edge candidate E with $R(h(x), h(y)) \in E$ for all $R(x, y) \in E_{p_0}$. Condition 3 of edge candidates then yields $(p_0|_{\overline{V_0} \cap \mathsf{var}(p_0)}, h|_{V_1}) \in f_{\mathsf{in}}^{M_1}(L)$. For the rest of the proof let $p_1 = p_0|_{\overline{V_0} \cap \mathsf{var}(p_0)}$.

Now we can apply Condition 4 of mosaics on $(p_1, h|_{V_1})$. Let $V_1' = \{x \mid x \in \mathsf{var}(p_1) \text{ and } M(h(x)) = M_1\}$. There are now two cases. If $V_1' = \mathsf{var}(p_1)$, then we immediately contradict Condition 4 of mosaics because w.r.t. $\mathsf{var}(p_0)$ there can be no component of p_1. If $V_1' \subsetneq \mathsf{var}(p_1)$, then by Condition 4 of mosaics there is a component p_1' with $p_1' = E_{p_1'} \uplus p_1'|_{\overline{V_1} \cap \mathsf{var}(p_1')}$ of p_1 w.r.t. V_1' such that $(p_1', h|_{V_1' \cap \mathsf{var}(p_1')}) \in f_{\mathsf{out}}^{M_1}(L)$.

The next step would be to choose V_2 as $V_2 = \overline{V_1} \cap \mathsf{var}(E_{p_1'})$ and repeat the back and forth between f_{out} and f_{in} of mosaics we just showed. The only difference is that from M_2 and onwards we also need Condition 3' of edge candidates, since for example p_1' might match back into M_0 via the edges in E (which means M_2 would be $M_2 = M_0$).

The number of iterations is, however, restricted by some $i \leq ||\mathcal{O}||$, since the size of the query is bound by $||\mathcal{O}||$ and consequently at some point we get a forbidden ingoing query $(p_i, h|_{V_i}) \in f_{\mathsf{in}}^{M_i}(L)$ such that h matches all variables of p_i into $\mathcal{I}_L^{M_i}$. This contradicts Condition 4 of mosaics and thus every case leads to a contradiction of our assumption. ❏

Lemma 11. *For all role abstractions $L'{:}R$ $\underline{\mathsf{abstracts}}$ $L{:}q(\bar{x}, \bar{y})$ in $\mathcal{O}$ and all homomorphisms h from q to $\mathcal{I}_L$,*

1. *$M(d) = M(d')$ for all $d, d' \in \mathsf{ran}(h|_{\bar{x}})$ and*
2. *$M(e) = M(e')$ for all $e, e' \in \mathsf{ran}(h|_{\bar{y}})$.*

Proof. Point 1 and Point 2 can be proven analogously to Lemma 10 using Condition 3 of mosaics instead of Condition 2. ❏

Now we are ready to prove the following lemma which establishes the soundness of the algorithm. The main thing left to prove is that all abstractions and refinements of $\mathcal{O}$ are satisfied.

Lemma 12. *$\mathcal{I}$ is a model of $\mathcal{O}$ with ${C_0}^{\mathcal{I}_{L_0}} \neq \emptyset$.*

Proof. We just proved in Lemma 8 that $\mathcal{I}$ is an A-Intepretation and in Lemma 9 that all CIs and RIs are satisfied. Since by Condition (i) of acceptance for our algorithm there is a mosaic $M \in \mathcal{M}^*$ with $C_0^{\mathcal{I}^M_{L_0}}$, our construction of $\mathcal{I}^0$ implies that ${C_0}^{\mathcal{I}^0_{L_0}} \neq \emptyset$ and by the construction of $\mathcal{I}$ this holds true for $\mathcal{I}$ as well.

Condition 1 of mosaics implies that each mosaic satisfies all concept refinements and since the satisfiability of concept refinements is independent of any roles we add, our construction of $\mathcal{I}$ naturally implies that $\mathcal{I}$ also satisfies all concept refinements.

Let $(d_1, d_2) \in q_R(\mathcal{I}_L)$ for some role refinement $L'{:}q(\bar{x}, \bar{y})$ refines $L{:}q_R(x, y)$ in $\mathcal{O}$ with $q = q|_{\bar{x}} \uplus E_q \uplus q|_{\bar{y}}$. There are two cases. The $d_1^\uparrow$ and $d_2^\uparrow$ can be inside the same mosaic, that is $M(d_1) = M(d_2) = M$. Here Condition 1 of mosaics yields that the role refinement is satisfied in M and thus also in $\mathcal{I}$ by our construction of $\mathcal{I}$.

In the second case $M(d_1) = M$, $M(d_2) = M'$ and $M \neq M'$. The definition of $\mathcal{I}$ then implies that there is an edge candidate E between M and M' with $R(d_1^\uparrow, d_2^\uparrow) \in E$ or $R^-(d_2^\uparrow, d_1^\uparrow) \in E$. Let us assume $R(d_1^\uparrow, d_2^\uparrow) \in E$ as the other case is analogous.

By Condition 4a to 4d of edge candidates, there are mappings $h : \bar{x} \mapsto \rho_L^M(d_1^\uparrow)$ and $h' : \bar{y} \mapsto \rho_L^{M'}(d_2^\uparrow)$ such that $\bar{x}\bar{y} \mapsto \rho_L^M(d_1^\uparrow)\rho_L^{M'}(d_2^\uparrow)$ is a homomorphism from q to $\mathcal{I}_\cup$ where $\mathcal{I}_\cup$ is the union of the interpretations of the two mosaics and the edges from the edge candidate defined as

$$\begin{aligned}
\Delta^{\mathcal{I}_\cup} &= \Delta^{\mathcal{I}_L^M} \uplus \Delta^{\mathcal{I}_L^{M'}};\\
A^{\mathcal{I}_\cup} &= A^{\mathcal{I}_L^M} \uplus A^{\mathcal{I}_L^{M'}};\\
R^{\mathcal{I}_\cup} &= R^{\mathcal{I}_L^M} \uplus R^{\mathcal{I}_L^{M'}} \uplus \{(h(x), (h'(y)) \mid R(x, y) \in E_q\};
\end{aligned}$$

where A ranges over concept names and R over roles.

Since $\mathcal{I}_\cup$ is part of $\mathcal{I}_L$ and both ρ^M and $\rho^{M'}$ are part of ρ we have $(\rho_{L'}(d_1), \rho_{L'}(d_2)) \in q(\mathcal{I}_{L'})$, as required. If earlier in the proof we would have had $R^-(d_2^\uparrow, d_1^\uparrow) \in E$ instead of $R(d_1^\uparrow, d_2^\uparrow) \in E$ then we use Condition 4' of edge candidates instead of Condition 4.

Let $h : \bar{x} \mapsto \bar{e}$ be a homomorphism from q to $\mathcal{I}_L$ for some concept abstraction $L'{:}A$ abstracts $L{:}q(\bar{x})$ in $\mathcal{O}$. Lemma 10 tells us that h maps all variables in q to the domain elements of a single mosaic M. Now Condition 1 of mosaics implies that the concept abstraction is satisfied in M and thus also in $\mathcal{I}$ by our construction of $\mathcal{I}$

Let $h : \bar{x}\bar{y} \mapsto \bar{e}_1\bar{e}_2$ be a homomorphism from q to $\mathcal{I}_L$ for some role abstraction $L'{:}R$ abstracts $L{:}q(\bar{x}, \bar{y})$ in $\mathcal{O}$. We use $h^\uparrow$ to denote replacing every element $d \in \mathsf{ran}(h)$ with $d^\uparrow$. Lemma 11 tells us that $h^\uparrow|_{\bar{x}}$ maps all variables in $q|_{\bar{x}}$ to the domain elements of a single mosaic M and $h^\uparrow|_{\bar{y}}$ maps all variables in $q_{\bar{y}}$ to the domain elements of a single mosaic M'. We consider two cases.

If $M = M'$ then Condition 1 of mosaics implies that the role abstraction is satisfied in M and thus also in $\mathcal{I}$ by our construction of $\mathcal{I}$.

If $M \neq M'$ then by the construction of $\mathcal{I}$, there is an edge candidate E between M and M' such that $h^\uparrow$ is a homomorphism from q to $\mathcal{I}_\cup$ where we construct $\mathcal{I}_\cup$ as in the role refinement case. Thus, either Condition 5 or Condition 5' of edge candidates must apply. If Condition 5 applies, then there are $d \in \Delta^{\mathcal{I}_L^M}$ and $d' \in \Delta^{\mathcal{I}_L^{M'}}$ with $\rho_L^M(d) = \bar{e}_1^\uparrow$, $\rho_L^{M'}(d') = \bar{e}_2^\uparrow$, and $R(d, d') \in E$. Again it is easy to verify that our construction of $\mathcal{I}$ then satisfies $L'{:}R$ abstracts $L{:}q(\bar{x}, \bar{y})$ since we copied all the components of the mosaics and edges in the edge candidate. If Condition 5' of edge candidates applies the proof works analogously. ❑

C.2 Completeness

Assume that there is a model $\mathcal{I} = (\prec, (\mathcal{I}_L)_{L \in \mathbf{A}_\mathcal{O}}, \rho)$ of $\mathcal{O}$ with a domain element $d_0 \in {C_0}^{\mathcal{I}_{L_0}}$. We have to show that our algorithm returns 'satisfiable'.

We assume w.l.o.g. that $\mathcal{I}$ does not contain any *unnecessarily large ensemble*, that is, an L-ensemble $\bar{e}$ for any $L \in \mathbf{A}_\mathcal{O}$ with $||\bar{e}|| > ||\mathcal{O}||$. We can assume this since any query in $\mathcal{O}$ is naturally of size at most $||\mathcal{O}||$.

For brevity we define $\mu(d) = \{e \mid e \text{ element in } \rho_L(d) \text{ for any } L \in \mathbf{A}_\mathcal{O}\}$ for all $d \in \Delta^\mathcal{I}$. For all $d \in \Delta^\mathcal{I}$ we define $\mu_1(d) = \{d\}$ and exhaustively compute

$$\mu_{i+1}(d) = \mu_i(d) \cup \bigcup_{e \in \mu_i(d)} \{e' \mid e \in \mu_i(e')\} \cup \bigcup_{e \in \mu_i(d)} \mu_i(e)$$

We use μ^* to denote the function at which this process stabilizes. Thus $\mu^*(d)$ contains all elements that can be reached by iterated refinements via ρ_L and abstractions via the inverse of ρ_L. Intuitively, these are organized in a tree whose elements are the members of $\mu^*(d)$ and with an edge (e, e') if e' occurs in $\rho_L(e)$. Also note that because there are no unnecessarily large ensembles, $|\mu^*(d)| \leq ||\mathcal{O}||^{||\mathcal{O}||}$.

For any $d \in \Delta^\mathcal{I}$, we define a mosaic M_d of d *in* $\mathcal{I}$. The first four components of M_d are as follows, for all $L \in \mathbf{A}_\mathcal{O}$:

$$\begin{aligned}
\Delta^{\mathcal{I}_L^{M_d}} &= \{e \mid e \in \Delta^{\mathcal{I}_L} \cap \mu^*(d)\};\\
A^{\mathcal{I}_L^{M_d}} &= \{e \mid e \in A^{\mathcal{I}_L} \cap \mu^*(d)\};\\
r^{\mathcal{I}_L^{M_d}} &= \{(e_1, e_2) \mid e_1, e_2 \in \Delta^{\mathcal{I}_L} \cap \mu^*(d) \text{ and}\\
&\qquad (e_1, e_2) \in r^{\mathcal{I}_L}\};\\
\rho_L^{M_d}(e) &= \bar{e} \text{ with } \rho_L(e) = \bar{e} \qquad \text{for all } e \in \Delta^{\mathcal{I}^{M_d}};
\end{aligned}$$

where A ranges of concept names and r over role names in $\mathsf{sig}(\mathcal{O})$. What remains is to define $f_{\mathsf{in}}^{M_d}$ and $f_{\mathsf{out}}^{M_d}$.

For all concept abstractions $L'{:}A$ abstracts $L{:}q(\bar{x})$ in $\mathcal{O}$, all non-empty $V \subsetneq \mathsf{var}(q)$, all homomorphisms h from $q|_V$ to $\mathcal{I}_L$, and all components p of q w.r.t. V such that $p = E_p \uplus p|_{\overline{V}}$ consider the following cases:

1. if $\mathsf{ran}(h) \subseteq \Delta^{\mathcal{I}_L^{M_d}}$ and h can be extended to a homomorphism from q to $\mathcal{I}_L$, then add $(p, h|_{V \cap \mathsf{var}(p)})$ to $f_{\mathsf{out}}^{M_d}(L)$;
2. if $\mathsf{ran}(h) \subseteq \Delta^{\mathcal{I}_L^{M_d}}$ and h cannot be extended to a homomorphism from $q|_V \cup p$ to $\mathcal{I}_L$, then add $(p, h|_{V \cap \mathsf{var}(p)})$ to $f_{\mathsf{out}}^{M_d}(L)$;
3. for all homomorphisms g from $q|_V \cup E_p$ to $\mathcal{I}_L$ that extend h but cannot be extended to a homomorphism from $q|_V \cup p$ to $\mathcal{I}_L$, if $\mathsf{ran}(g|_{\overline{V} \cap \mathsf{var}(E_p)}) \subseteq \Delta^{\mathcal{I}_L^{M_d}}$, then add $(p|_{\overline{V}}, g|_{\overline{V} \cap \mathsf{var}(E_p)})$ to $f_{\mathsf{in}}^{M_d}(L)$.

For all role abstractions $L'{:}R$ abstracts $L{:}q(\bar{x},\bar{y})$ in $\mathcal{O}$, all non-empty $V \subsetneq \mathsf{var}(q)$ with $V \neq \mathsf{set}(\bar{x})$ and $V \neq \mathsf{set}(\bar{y})$, all homomorphisms h from $q|_V$ to $\mathcal{I}_L$, and all components p of q w.r.t. V such that $p = E_p \uplus p|_{\overline{V}}$ consider the following cases:

4. if $\mathsf{ran}(h) \subseteq \Delta^{\mathcal{I}_L^{M_d}}$ and h can be extended to a homomorphism from q to $\mathcal{I}_L$, then add $(p, h|_{V\cap\mathsf{var}(p)})$ to $f_{\mathsf{out}}^{M_d}(L)$;
5. if $\mathsf{ran}(h) \subseteq \Delta^{\mathcal{I}_L^{M_d}}$ and h cannot be extended to a homomorphism from $q|_V \cup p$ to $\mathcal{I}_L$, then add $(p, h|_{V\cap\mathsf{var}(p)})$ to $f_{\mathsf{out}}^{M_d}(L)$;
6. for all homomorphisms g from $q|_V \cup E_p$ to $\mathcal{I}_L$ that extend h but cannot be extended to a homomorphism from $q|_V \cup p$ to $\mathcal{I}_L$, if $\mathsf{ran}(g|_{\overline{V}\cap\mathsf{var}(E_p)}) \subseteq \Delta^{\mathcal{I}_L^{M_d}}$, then add $(p|_{\overline{V}}, g|_{\overline{V}\cap\mathsf{var}(E_p)})$ to $f_{\mathsf{in}}^{M_d}(L)$.

Strictly speaking, the tuple M_d defined above is not a mosaic as the domain elements are not from the fixed set Δ. Since, however, $|\mu^*(d)| \leq ||\mathcal{O}||^{||\mathcal{O}||}$, this can easily be achieved by renaming. If we renamed an element $d \in \Delta^{\mathcal{I}}$ to $e \in \Delta^{\mathcal{I}^M}$, then we use $e^{\downarrow}$ to denote d. We might also use this notation for tuples $\bar{e}$, where $\bar{e}^{\downarrow}$ implies that $\cdot^{\downarrow}$ is applied to all elements in the tuple, or functions f, where $f^{\downarrow}$ is the function obtained by applying $\cdot^{\downarrow}$ to all elements in $\mathsf{ran}(f)$.

Now that we have defined how we read mosaics out of the model we first need to prove that they are indeed mosaics and second that they don't get eliminated in the mosaic removal subroutine of our algorithm. If we proved these two things, then showing that our algorithm must returns 'satisfiable' will be trivial.

Lemma 13. *M_d is a mosaic.*

Proof. We go through all seven Conditions of mosaics. Conditions 1 follows directly from $\mathcal{I}$ being a model and the definition of M_d.

For Condition 2 consider a concept abstraction $L'{:}A$ abstracts $L{:}q(\bar{x})$ in $\mathcal{O}$, a non-empty $V \subsetneq \mathsf{var}(q)$, and a homomorphism h from $q|_V$ to $\mathcal{I}_L^{M_d}$. There are now two cases. The first one is that $h^{\downarrow}$ can be extended to a homomorphism from q to $\mathcal{I}_L$. Then $(p, h|_{V\cap\mathsf{var}(p)}) \in f_{\mathsf{out}}^{M_d}(L)$ for all components p of q w.r.t. V, according to Case 1 of the M_d construction.

If $h^{\downarrow}$ can not be extended to a homomorphism from q to $\mathcal{I}_L$ then there has to be at least one component p of q w.r.t. V such that $h^{\downarrow}$ cannot be extended to a homomorphism from $q|_V \cup p$ to $\mathcal{I}_L$. Case 2 of the M_d construction then yields $(p, h|_{V\cap\mathsf{var}(p)}) \in f_{\mathsf{out}}^{M_d}(L)$. Thus Condition 2 is satisfied. Condition 3 can be proven analogously.

For Condition 4 let $(q_0, h_0) \in f_{\mathsf{in}}^{M_d}(L)$, $V_0 \subseteq \mathsf{var}(q_0)$, and g_0 be a homomorphism from $q_0|_{V_0}$ to $\mathcal{I}_L^{M_d}$ that extends h_0. We have to show that for some component p_0 of q_0 w.r.t. V_0 there is $(p_0, g_0|_{V_0\cap\mathsf{var}(p_0)}) \in f_{\mathsf{out}}^{M_d}$.

Let us assume (q_0, h_0) was added to $f_{\mathsf{in}}^{M_d}(L)$ because of Case 3 in the construction of M_d. This implies that there is

1. a concept abstraction $L'{:}A$ abstracts $L{:}q(\bar{x})$ in $\mathcal{O}$,
2. a non-empty $V \subsetneq \mathsf{var}(q)$,
3. a component p of q w.r.t. V,
4. and a homomorphism g from $q|_V \cup E_p$ to $\mathcal{I}_L$ that cannot be extended to a homomorphism from $q|_V \cup p$ to $\mathcal{I}_L$

such that $p|_{\overline{V}} = q_0^{\downarrow}$ and $g|_{\overline{V}\cap\mathsf{var}(E_p)} = h_0^{\downarrow}$.

By our assumption $\mathsf{ran}(g_0) \subseteq \Delta^{\mathcal{I}_L^{M_d}}$. The definition of M_d implies that $g_0^{\downarrow}$ is also a homomorphism from $q|_{V_0}$ to $\mathcal{I}_L$. Point 4 from above yields that there has to be a component p_0 of q_0 w.r.t. V_0 such that we cannot extend $g_0^{\downarrow}$ to a homomorphism from $q|_{V_0} \cup p_0$ to $\mathcal{I}_L$.

These are exactly the conditions required for Case 2 in the construction of M_d. Thus by Case 2 we have $(p_0, g_0|_{V_0\cap\mathsf{var}(p_0)}) \in f_{\mathsf{out}}^{M_d}$, as required. If (q_0, h_0) was added to $f_{\mathsf{in}}^{M_d}(L)$ because of Case 6 then the proof is analogous. ❑

Now we are ready to prove that all these M_d are contained in the final set of mosaics our algorithm arrives at. Formally we prove this by doing an induction on the number of iterations in our mosaic removal subroutine.

Lemma 14. *Let n be the number of iterations of our algorithm. For every $i \leq n$: $\{M_d \mid d \in \Delta^{\mathcal{I}}\} \subseteq \mathcal{M}_i$.*

Proof. For $i = 0$, this follows from Lemma 13.

For $i > 0$, choose any $M_d \in \mathcal{M}_i$ with $d \in \Delta^{\mathcal{I}}$ and set $M = M_d$.

Let $d' \in \Delta^{\mathcal{I}_L^M}$ with $d' \in A^{\mathcal{I}_L^M}$ and $d' \notin (\exists R.B)^{\mathcal{I}_L^M}$ for any $A \sqsubseteq_L \exists R.B \in \mathcal{O}$. We have to show that there is a mosaic $M' \in \mathcal{M}_i$, an element $e' \in B^{\mathcal{I}_L^{M'}}$, and an edge candidate E such that $R(d', e') \in E$.

From the definition of M_d it follows that $d'^{\downarrow} \in (\exists R.C)^{\mathcal{I}_L}$ and because $\mathcal{I}$ is a model there must be an element $e \in \Delta^{\mathcal{I}_L}$ such that $e \in B^{\mathcal{I}_L}$ and $(d'^{\downarrow}, e) \in R^{\mathcal{I}_L}$. We choose M_e as M' and use e' to denote the $e' \in \Delta^{\mathcal{I}^{M_e}}$ with $e'^{\downarrow} = e$. Now we construct an edge candidate E for M_d and M_e that contains $R(d', e')$:

$$\begin{aligned} E = \{ & R(d', e') \mid d' \in \Delta^{\mathcal{I}_L^{M_d}}, e' \in \Delta^{\mathcal{I}_L^{M_e}}, \text{and} \\ & (d'^{\downarrow}, e'^{\downarrow}) \in R^{\mathcal{I}_L} \text{ for any } L \in \mathbf{A}_{\mathcal{O}} \text{ and } R \in \mathbf{R}\} \end{aligned}$$

What remains to show is that E satisfies all edge candidate conditions. Conditions 1 to 2' and 4 to 5' follow directly from $\mathcal{I}$ being a model and the construction of M_d, M_e and E.

For Condition 3 consider an $L \in \mathbf{A}_{\mathcal{O}}$, a pair $(q_0, h_0) \in f_{\mathsf{out}}^{M_d}(L)$ where $q_0 = E_{q_0} \uplus q_0|_{\overline{V_0}}$ with $V_0 = \mathsf{dom}(h_0)$, and a function g_0 from $\overline{V_0} \cap \mathsf{var}(E_{q_0})$ to $\Delta^{\mathcal{I}_L^{M_e}}$ such that $R(h_0(x), g_0(y)) \in E$ for all $R(x, y) \in E_{q_0}$. We need to show that $(q_0|_{\overline{V_0}}, g_0) \in f_{\mathsf{in}}^{M_e}(L)$.

There are six cases in the construction of M_d concerning $f_{\mathsf{in}}^{M_d}$ and $f_{\mathsf{out}}^{M_d}$. For the following proof we only consider the first three, that is, the ones for concept abstractions. Extending the proof to consider all six cases is, however, trivial as Case 4 behaves analogously to Case 1 and likewise for Cases 2 and 5, and 3 and 6.

If (q_0, h_0) was added to $f_{\mathsf{out}}^{M_d}(L)$ because of Case 1, then

1. q_0 is a component of q w.r.t. some non-empty $V \subsetneq \mathsf{var}(q)$ for some concept abstraction $L'{:}A$ abstracts $L{:}q(\bar{x})$ in $\mathcal{O}$;
2. $h_0^{\downarrow} = h|_{V \cap \mathsf{var}(q_0)}$ for some homomorphism h from $q|_V$ to $\mathcal{I}_L$;
3. h can be extended to a homomorphism h^* from q to $\mathcal{I}_L$;
4. $V_0 = V \cap \mathsf{var}(q_0)$.

Point 3 and the construction of M_d imply that $\mathsf{ran}(h^*) \subseteq \{d'^{\downarrow} \mid d' \in \Delta^{\mathcal{I}_L^{M_d}}\}$ and thus by the definition of M_e and refinement functions we cannot extend $h \cup g_0^{\downarrow}$ to a homomorphism from $q|_V \cup q_0$ to $\mathcal{I}_L$ as this would make the elements in $\mathsf{ran}(g_0^{\downarrow})$ part of two L-ensembles. By definition $\mathsf{ran}(g_0) \subseteq \Delta^{\mathcal{I}_L^{M_e}}$ and as such Case 3 of M_d construction yields $(q_0|_{\overline{V}}, g_0|_{\overline{V} \cap \mathsf{var}(E_{q_0})}) \in f_{\mathsf{in}}^{M_e}$. Point 1 and Point 4 imply that $\overline{V_0} \cap \mathsf{var}(E_{q_0}) \subseteq \overline{V} \cap \mathsf{var}(E_{q_0})$ and since the left side of the subset relation is exactly the domain of g_0 we can simplify the previous expression to $(q_0|_{\overline{V}}, g_0) \in f_{\mathsf{in}}^{M_e}$. Again we can use Point 4 to arrive at the desired result of $(q_0|_{\overline{V_0}}, g_0) \in f_{\mathsf{in}}^{M_e}$.

If (q_0, h_0) was added to $f_{\mathsf{out}}^{M_d}(L)$ because of Case 2, then

1. q_0 is a component of q w.r.t. some non-empty $V \subsetneq \mathsf{var}(q)$ for some concept abstraction $L'{:}A$ abstracts $L{:}q(\bar{x})$ in $\mathcal{O}$;
2. $h_0^{\downarrow} = h|_{V \cap \mathsf{var}(q_0)}$ for some homomorphism h from $q|_V$ to $\mathcal{I}_L$;
3. $V_0 = V \cap \mathsf{var}(q_0)$.

This time Case 2 of M_d construction itself already implies that we cannot extend h (and the same must hold true for $h \cup g_0^{\downarrow}$) to a homomorphism from $q|_V \cup q_0$ to $\mathcal{I}_L$. By definition $\mathsf{ran}(g_0) \subseteq \Delta^{\mathcal{I}_L^{M_e}}$ and as such Case 3 of M_e construction yields $(q_0|_{\overline{V}}, g_0|_{\overline{V}}) \in f_{\mathsf{in}}^{M_e}$. We can apply the same reasoning as for Case 1 of M_e construction to arrive at $(q_0|_{\overline{V_0}}, g_0) \in f_{\mathsf{in}}^{M_e}$, as required.

Condition 3' of edge candidates can be proven in an analogous way and consequently E is an edge candidate for M_d and M_e. That $M_e \in \mathcal{M}_i$ follows from the induction hypothesis and thus M_d is good in $\mathcal{M}_i$ for all $d \in \Delta^{\mathcal{I}}$. ❑

By our initial assumtion there is at least one element $d \in \Delta^{\mathcal{I}_L}$ for all $L \in \mathbf{A}_{\mathcal{O}}$, and also an element $d_0 \in {C_0}^{\mathcal{I}_{L_0}}$. We can now use Lemma 14 to imply that $M_{\{d\}} \in \mathcal{M}^*$ for all these domain element and thus the conditions for our algorithm to return 'satisfiable' are met.

C.3 Running Time

Since the domain of a mosaic has a maximum of $||\mathcal{O}||^{||\mathcal{O}||}$ elements, there are at most $2^{2^{\mathsf{poly}(||\mathcal{O}||)}}$ different mosaics. Consequently the length of the constructed sequence $\mathcal{M}_0, \mathcal{M}_1, \ldots$ is also bounded by $2^{2^{\mathsf{poly}(||\mathcal{O}||)}}$. Checking goodness of mosaics requires us to compute answers to CQs. Doing this in a brute force way results in checking at most $||\mathcal{O}||^{||\mathcal{O}||^{||\mathcal{O}||}} \in 2^{2^{\mathsf{poly}(||\mathcal{O}||)}}$ candidates for homomorphisms from a CQ into a mosaic (for a fixed answer). In summary, the running time of our algorithm is at most $2^{2^{\mathsf{poly}(||\mathcal{O}||)}}$.

D Proofs for Section 5.1

We provide the detailed proof of Theorem 3, which we repeat here for the reader's convenience.

Theorem 3. *Satisfiability in $\mathcal{ALC}^{\mathsf{abs}}[\mathrm{ra}]$ is 2*EXPTIME*-hard.*

Let $\mathcal{O}, A_0, q$ be the input for simple CQ evaluation. We construct an $\mathcal{ALC}^{\mathsf{abs}}[\mathrm{ra}]$ ontology $\mathcal{O}'$ with three abstraction levels $L_1 \prec L_2 \prec L_3$ such that A_0 is L_1-satisfiable w.r.t. $\mathcal{O}'$ iff $\mathcal{O}, A_0 \not\models q$.

We may assume w.l.o.g. that the $\mathcal{ALCI}$-ontology $\mathcal{O}$ is in *normal form*, meaning that every CI in it is of the form

$$A_1 \sqcap \cdots \sqcap A_n \sqsubseteq B_1 \sqcup \cdots \sqcup B_m \quad \text{or} \quad A \sqsubseteq C$$

where A_i, B_i, A are concept names and C ranges over concepts of the form $\exists R.B$, $\forall R.B$, or $\neg B$ with R a (potentially inverse) role and B a concept name. It is routine to show that every $\mathcal{ALCI}$-ontology $\mathcal{O}$ can be converted in polynomial time into an $\mathcal{ALCI}$-ontology $\mathcal{O}'$ in this form that is a conservative extension of $\mathcal{O}$.

We next show that CIs of the form $A \sqsubseteq \forall r^-.B$ can be removed. Let $\widehat{\mathcal{O}}$ be obtained from $\mathcal{O}$ by replacing every such CI with

$$\top \sqsubseteq B \sqcup \overline{B},\ \overline{B} \sqsubseteq \forall r.\overline{A},\ \text{and}\ \overline{A} \sqsubseteq \neg A$$

where $\overline{B}$ and $\overline{A}$ are fresh concept names. It is routine to prove the following.

Lemma 15. *$\mathcal{O}, A \models q$ iff $\widehat{\mathcal{O}}, A \models q$.*

Proof. We prove this by showing that $\widehat{\mathcal{O}}$ is a conservative extension of $\mathcal{O}$. Point 1 of conservative extensions is straightforward as $\mathsf{sig}(\widehat{\mathcal{O}}) = \mathsf{sig}(\mathcal{O}) \cup \{\overline{B}, \overline{A} \mid A \sqsubseteq \forall r^-.B \in \mathcal{O}\}$. For Point 2 consider a model $\mathcal{I}$ of $\widehat{\mathcal{O}}$ and domain element $d \in A^{\mathcal{I}}$ with $A \sqsubseteq \forall r^-.B \in \mathcal{O}$. The three CIs we added when replacing $A \sqsubseteq \forall r^-.B$ imply that there is no $e \in \Delta^{\mathcal{I}}$ with both $e \in (\neg B)^{\mathcal{I}}$ and $(e, d) \in r^{\mathcal{I}}$. Thus $d \in (\forall r^-.D)^{\mathcal{I}}$ and consequently $\mathcal{I}$ is a model of $\mathcal{O}$.

For the third point we construct a model $\mathcal{I}_2$ of $\widehat{\mathcal{O}}$ given a model $\mathcal{I}_1$ of $\mathcal{O}$ as follows:

$$\begin{aligned}
\Delta^{\mathcal{I}_2} &= \Delta^{\mathcal{I}_1} \\
A^{\mathcal{I}_2} &= A^{\mathcal{I}_1} \text{ for all } A \in \mathsf{sig}(\mathcal{O}) \cap \mathbf{C} \\
\overline{A}^{\mathcal{I}_2} &= \{d \mid d \in \Delta^{\mathcal{I}_2} \setminus A^{\mathcal{I}_2}\} \text{ for all } A \sqsubseteq \forall r^-.B \in \mathcal{O} \\
\overline{B}^{\mathcal{I}_2} &= \{d \mid d \in \Delta^{\mathcal{I}_2} \setminus B^{\mathcal{I}_2}\} \text{ for all } A \sqsubseteq \forall r^-.B \in \mathcal{O} \\
r^{\mathcal{I}_2} &= r^{\mathcal{I}_1} \text{ for all } r \in \mathsf{sig}(\mathcal{O}) \cap \mathbf{R}
\end{aligned}$$

We are left arguing that $\mathcal{I}_2$ is a model of $\widehat{\mathcal{O}}$.

It is easy to see that the extensions of concept and role names from $\mathsf{sig}(\mathcal{O})$ coincide in the two interpretations and hence the CIs that simply got copied from $\mathcal{O}$ to $\widehat{\mathcal{O}}$ are all still satisfied. Let $\overline{A}$ and $\overline{B}$ be the concept names used in one replacement step for a CI $A \sqsubseteq \forall r^-.B \in \mathcal{O}$. It is clear by the construction of $\overline{A}^{\mathcal{I}_2}$ and $\overline{B}^{\mathcal{I}_2}$ that the CIs $\top \sqsubseteq B \sqcup \overline{B}$, and $\overline{A} \sqsubseteq \neg A$ are satisfied.

For $\overline{B} \sqsubseteq \forall r.\overline{A}$, we argue by contradiction. Assume there is a $d \in \overline{B}^{\mathcal{I}_2}$, $e \in (\neg\overline{A})^{\mathcal{I}_2}$, and $(d,e) \in r$. By the construction of $\overline{A}^{\mathcal{I}_2}$ we know $e \in A^{\mathcal{I}_2}$. Hence in the original model there are now $d \in (\neg B)^{\mathcal{I}_1}$, $e \in A^{\mathcal{I}_1}$ and $(e,d) \in r^{\mathcal{I}_1}$. This contradicts the assumption that $\mathcal{I}_1$ is a model, since the CI $A \sqsubseteq \forall r^-.B \in \mathcal{O}$ is now no longer satisfied. ❏

We may thus assume that $\mathcal{O}$ contains no CIs of the form $A \sqsubseteq \forall r^-.B$. We now convert $\mathcal{O}$ into an $\mathcal{ALC}$-ontology $\mathcal{O}_{\mathcal{ALC}}$. Introduce a fresh role name $\widehat{r}$ for every role name r used in $\mathcal{O}$. Then start from $\mathcal{O}$, replace every CI $A \sqsubseteq \exists r^-.B$ with $A \sqsubseteq \exists\widehat{r}.B$, and add $\exists\widehat{r}.A \sqsubseteq B$ for every CI $A \sqsubseteq \forall r.B$ in $\mathcal{O}$.[2] Clearly, representing an inverse role r^- with a fresh role name r is problematic from the perspective of the CQ q.

We now assemble the $\mathcal{ALC}^{\mathsf{abs}}$[ra]-ontology $\mathcal{O}'$. On level L_1, we want a model of $\mathcal{O}_{\mathcal{ALC}}$ and thus include $C \sqsubseteq_{L_1} D$ for all $C \sqsubseteq D$ in $\mathcal{O}_{\mathcal{ALC}}$. The interpretation on level L_2 is a copy of that on level L_1, but we add an r-edge for every $\widehat{r}^-$-edge. This is achieved by adding the role abstractions

$$L_2{:}r \ \underline{\mathsf{abstracts}}\ L_1{:}r(x,y) \quad \text{and} \quad L_2{:}r \ \underline{\mathsf{abstracts}}\ L_1{:}\widehat{r}(y,x).$$

We also need to copy the concept names but do not want to use concept abstractions. We thus introduce a fresh role name r_A for every concept name A and put $A \equiv_{L_i} \exists r_A.\top$ for $i \in \{1,2\}$ and

$$L_2{:}r_A \ \underline{\mathsf{abstracts}}\ L_1{:}r_A(x,y)$$

for all concept names A in $\mathcal{O}_{\mathcal{ALC}}$. Let $\widehat{q}(\bar{x},z)$ be obtained from q by dequantifying all variables and adding the atom $\top(z)$ for a fresh variable z. Furthermore, let r_q be a fresh role name and add the following statements to $\mathcal{O}'$:

$$L_3{:}r_q \ \underline{\mathsf{abstracts}}\ L_1{:}\widehat{q}(\bar{x},z)$$
$$\exists r_q.\top \sqsubseteq_{L_3} \bot$$

This concludes the construction of $\mathcal{O}'$ and now we will prove that we can use it to reduce from the complement of the simple CQ evaluation problem.

Lemma 16. *$\mathcal{O}, A_0 \not\models q$ iff A_0 is L_1-satisfiable w.r.t. $\mathcal{O}'$.*

To prepare for the proof of Lemma 16, we first establish an auxiliary lemma. For interpretations $\mathcal{I}, \mathcal{J}$, we write $\mathcal{I} \subseteq \mathcal{J}$ if $\mathcal{I}$ can be found in $\mathcal{J}$, that is, if the following conditions are satisfied:

$\Delta^{\mathcal{I}} \subseteq \Delta^{\mathcal{J}}$,

$A^{\mathcal{I}} \subseteq A^{\mathcal{J}}$ for all concept names A, and

$r^{\mathcal{I}} \subseteq r^{\mathcal{J}}$ for all role names r.

Similarly, for a signature Σ we use $\mathcal{I}_{|\Sigma}$ to denote the restriction of $\mathcal{I}$ to the concept and role names in Σ. Let $\mathcal{O}'^-$ be the fragment of $\mathcal{O}'$ in which all statements have been removed that refer to level L_3. Then we have the following.

Lemma 17.

1. *If $\mathcal{I}$ is a model of $\mathcal{O}$, then there is a model $\mathcal{J}$ of $\mathcal{O}'^-$ such that $\mathcal{J}_{L_2|\Sigma} = \mathcal{I}$ with $\Sigma = \mathsf{sig}(\mathcal{O})$;*
2. *If $\mathcal{J}$ is a model of $\mathcal{O}'^-$, then there is a model $\mathcal{I}$ of $\mathcal{O}$ such that $\mathcal{I} \subseteq \mathcal{J}_{L_2}$.*

[2]It is not important to achieve normal form here.

Proof. "Point 1". Let $\mathcal{I}$ be a model of $\mathcal{O}$. We construct $\mathcal{J} = (\prec, \mathcal{J}_{L_1}, \mathcal{J}_{L_2}, \rho)$ as follows:

$$\begin{aligned}
\Delta^{\mathcal{J}_{L_1}} &= \Delta^{\mathcal{I}} && \\
A^{\mathcal{J}_{L_1}} &= A^{\mathcal{I}} && \text{for all } A \in \mathsf{sig}(\mathcal{O}) \cap \mathbf{C}\\
r_A^{\mathcal{J}_{L_1}} &= \{(d,d) \mid d \in A^{\mathcal{I}}\} && \text{for all } A \in \mathsf{sig}(\mathcal{O}) \cap \mathbf{C}\\
r^{\mathcal{J}_{L_1}} &= r^{\mathcal{I}} && \text{for all } r \in \mathsf{sig}(\mathcal{O}) \cap \mathbf{R}\\
\widehat{r}^{\mathcal{J}_{L_1}} &= \{(e,d) \mid (d,e) \in r^{\mathcal{I}}\} && \text{for all } r \in \mathsf{sig}(\mathcal{O}) \cap \mathbf{R}
\end{aligned}$$

We let $\mathcal{J}_{L_2}$ be an isomorphic copy of $\mathcal{J}_{L_1}$ and let ρ map each domain element in $\Delta^{\mathcal{J}_{L_2}}$ to its isomorphic copy in $\Delta^{\mathcal{J}_{L_1}}$. Now we prove that $\mathcal{J}$ is a model of $\mathcal{O}'^-$.

Let us first consider the labeled CIs in $\mathcal{J}_{L_1}$. All the CIs that didn't change going from $\mathcal{O}$ to $\mathcal{O}_{\mathcal{ALC}}$ are still satisfied. For each CI of the form $A \sqsubseteq \exists r^-.B$ in $\mathcal{O}$, we introduced a CI $A \sqsubseteq_{L_1} \exists\widehat{r}.B$ in $\mathcal{O}'^-$. It is trivial to verify that our new CI is satisfied by the construction of $\widehat{r}^{\mathcal{J}_{L_1}}$.

For each CI of the form $A \sqsubseteq \forall r.B$ in $\mathcal{O}$, we added a CI $\exists\widehat{r}.A \sqsubseteq_{L_1} B$ to $\mathcal{O}'^-$. Assume we have a domain element $d \in (\exists\widehat{r}.A)^{\mathcal{J}_{L_1}}$. Our construction of $\widehat{r}^{\mathcal{J}_{L_1}}$ then implies that there is an $e \in A^{\mathcal{J}_{L_1}}$ with $(e,d) \in r^{\mathcal{J}_{L_1}}$. Now we can again use the construction of $\mathcal{J}$ to jump back to the model $\mathcal{I}$ and obtain $e \in (\forall r.B)^{\mathcal{I}}$ and $d \in B^{\mathcal{I}}$. Thus also $d \in B^{\mathcal{J}_{L_1}}$, as required.

The rest of the labeled CIs are all of the form $A \equiv_{L_1} \exists r_a.\top$ and it is clear that they are satisfied by our construction. Since $\mathcal{J}_{L_2}$ is an isomorphic copy we can argue for the L_2-CIs in exactly the same way. Again $\mathcal{J}_{L_2}$ being an isomorphic copy and the way we defined ρ makes it easy to see that the two types of role abstractions are satisfied as well. Note that we already defined $\widehat{r}$ to point in the inverse direction of r for any two domain elements that are connected by r. Looking at the construction it is straightforward to see that $\mathcal{J}_{L_2|\Sigma} = \mathcal{I}$ with $\Sigma = \mathsf{sig}(\mathcal{O})$.

"Point 2". Let $\mathcal{J}$ be a model of $\mathcal{O}'^-$. We construct $\mathcal{I}$ as follows:

$$\begin{aligned}
\Delta^{\mathcal{I}} &= \{d \mid d \in \Delta^{\mathcal{J}_{L_1}} \text{ and } d \text{ is in the extension of}\\
&\qquad \text{a concept or role name in } \mathcal{J}_{L_1}\}\\
A^{\mathcal{I}} &= A^{\mathcal{J}_{L_1}} \qquad \text{for all } A \in \mathsf{sig}(\mathcal{O}) \cap \mathbf{C}\\
r^{\mathcal{I}} &= r^{\mathcal{J}_{L_1}} \cup \{(d,e) \mid (e,d) \in \widehat{r}^{\mathcal{J}_{L_1}}\}\\
&\qquad \text{for all } r \in \mathsf{sig}(\mathcal{O}) \cap \mathbf{R}
\end{aligned}$$

All the CIs in $\mathcal{O}$ that do not contain any universal or existential restriction are obviously still satisfied since we copied them to $\mathcal{O}'$. Same for CIs of the form $A \sqsubseteq \exists r.B$. Since $\mathcal{O}$ is normalized we then only need to consider CIs of the form $A \sqsubseteq \forall r.B$ and $A \sqsubseteq \exists r^-.B$.

Consider a CI of the form $A \sqsubseteq \exists r^-.B \in \mathcal{O}$ and $d \in A^{\mathcal{I}}$. Since there is a CI $A \sqsubseteq \exists\widehat{r}.B \in \mathcal{O}'$ and $d \in A^{\mathcal{J}_{L_1}}$ there is an $e \in B^{\mathcal{J}_{L_1}}$ with $(d,e) \in \widehat{r}^{\mathcal{J}_{L_1}}$ and thus by our construction of $\mathcal{I}$ also $(e,d) \in r^{\mathcal{I}}$ which satisfies the mentioned CI.

On the other hand let there be a CI $A \sqsubseteq \forall r.B \in \mathcal{O}$, $d \in A^{\mathcal{I}}$ and $(d,e) \in r^{\mathcal{I}}$ an edge such that $(e,d) \in \widehat{r}^{\mathcal{J}_{L_1}}$. By our construction $d \in A^{\mathcal{J}_{L_1}}$ and therefore $e \in (\exists\widehat{r}.A)^{\mathcal{J}_{L_1}}$. The CI $\exists\widehat{r}.A \sqsubseteq_{L_1} B \in \mathcal{O}'$ then gives us $e \in B^{\mathcal{J}_{L_1}}$ and consequently $e \in B^{\mathcal{I}}$. Thus $\mathcal{I}$ is a model of $\mathcal{O}$.

To show $\mathcal{I} \subseteq \mathcal{J}_{L_2}$ we construct an isomorphic copy $\mathcal{I}'$

of $\mathcal{I}$ with $\mathcal{I}' \subseteq \mathcal{J}_{L_2}$ that is by nature still a model of $\mathcal{O}$. We do this by replacing each domain element $d \in \Delta^{\mathcal{I}}$ (implies $d \in \Delta^{\mathcal{J}_{L_1}}$) with its abstraction, that is, the $e \in \Delta^{\mathcal{J}_{L_2}}$ with $\rho(e) = d$. This works because the role abstractions $L_2{:}r$ abstracts $L_1(r(x,y))$ force ρ to be such a one to one mapping for all domain elements in the extension of a role name or concept name (concept names because of the CIs $A \equiv_{L_1} \exists r_A.\top$).

Now, obviously, $\Delta^{\mathcal{I}'} \subseteq \Delta^{\mathcal{J}_{L_2}}$. The only interesting case for the subset relation between all the concept and role name extensions are the roles we added with $\{(d,e) \mid (e,d) \in \widehat{r}^{\mathcal{J}_{L_1}}\}$. Let $(d,e) \in r^{\mathcal{I}'}$ be such a role. The construction of $r^{\mathcal{I}'}$ implies that $(\rho(e), \rho(d)) \in \widehat{r}^{\mathcal{J}_{L_1}}$ and with the role abstraction $L_2{:}r$ abstracts $L_1{:}\widehat{r}(y,x) \in \mathcal{O}'$ we get $(d,e) \in r^{\mathcal{J}_{L_2}}$. Thus $\mathcal{I}' \subseteq \mathcal{J}_{L_2}$. ❑

Now Lemma 16 is a direct consequence of Lemma 17 and the statements in $\mathcal{O}' \setminus \mathcal{O}'^{-}$.

E Proofs for Section 5.2

Our aim is to prove Theorem 4. We again repeat the theorem for the reader's convenience.

Theorem 4. *Satisfiability in $\mathcal{ALC}^{\mathsf{abs}}[\mathrm{ca}]$ is 2EXPTIME-hard.*

Recall that $\mathcal{ALC}^{\mathsf{sym}}$ is $\mathcal{ALC}$ with a single role name s that must be interpreted as a reflexive and symmetric relation. We reduce simple CQ evaluation in $\mathcal{ALC}^{\mathsf{sym}}$.

Let $\mathcal{O}$ be an $\mathcal{ALC}^{\mathsf{sym}}$ ontology, A_0 a concept name, and q a Boolean CQ such that it is to be decided whether $\mathcal{O}, A_0 \models q$. We assume w.l.o.g. that $\mathcal{O}$ is in *negation normal form* (NNF), that is, negation is only applied to concept names, but not to compound concepts. For an $\mathcal{ALC}^{\mathsf{sym}}$ concept C we use $\overline{C}$ to denote the result of converting $\neg C$ to NNF. With $\mathsf{cl}(\mathcal{O})$ we denote the smallest set that contains all concepts used in $\mathcal{O}$ (possibly inside a CQ) and is closed under subconcepts and under $\overline{\cdot}$. Let $\exists s.C_1, \dots, \exists s.C_n$ be all concepts in $\mathsf{cl}(\mathcal{O})$ that quantify existentially over s.

Let $C_0 = A_0 \sqcap (E^0 \sqcup E^1)$. We construct (in polynomial time) an $\mathcal{ALC}[\mathrm{ca}]$ ontology $\mathcal{O}'$ such that $\mathcal{O}, A_0 \not\models q$ iff C_0 is L-satisfiable w.r.t. $\mathcal{O}'$. In $\mathcal{O}'$, we represent the role name s by the composition $r^{-};r$ where r is a normal (neither reflexive nor symmetric) role name. We use all concept names from $\mathsf{sig}(\mathcal{O})$ as well as

- a concept name A_C for every $C \in \mathsf{cl}(\mathcal{O})$;
- concept names E^0, E^1 that represent endpoints of symmetric edges;
- concept names N^i_j for all $i \in \{0,1\}$ and $j \in \{0,\dots,n\}$ where j denotes the number of s-children of the current node (in a tree-shaped model),
- concept names $M^{i,j}_k$ for all $i \in \{0,1\}$, $j \in \{1,\dots,n\}$, and $k \in \{1,\dots,j\}$ that represent midpoints of the composition $r^{-};r$;
- auxiliary role names $\widehat{r}$ an u.

W.l.o.g. we assume that $A_0 \in \mathsf{sub}(\mathcal{O})$. For every concept $C \in \mathsf{cl}(\mathcal{O})$, $\mathcal{O}'$ contains CIs that axiomatize the semantics

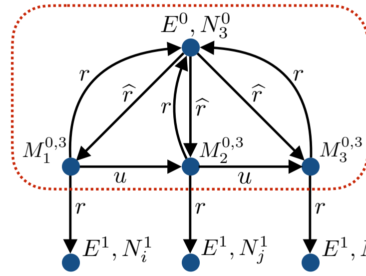


Figure 4: Example of the symmetric structure of r between endpoints (E^i). Inside the dotted line is an L-ensemble for the case of three children (N^0_3).

of the corresponding concept names A_C:

$$
\begin{aligned}
A_B &\equiv_L B && \text{for all } B \in \mathsf{cl}(\mathcal{O}) \cap \mathbf{C} && (1)\\
A_{\overline{C}} &\equiv_L \neg A_C && \text{for all } C \in \mathsf{cl}(\mathcal{O}) && (2)\\
A_{C \sqcap D} &\equiv_L A_C \sqcap A_D && \text{for all } C \sqcap D \in \mathsf{cl}(\mathcal{O}) && (3)\\
A_{C \sqcup D} &\equiv_L A_C \sqcup A_D && \text{for all } C \sqcup D \in \mathsf{cl}(\mathcal{O}) && (4)\\
A_C &\sqsubseteq_L A_D && \text{for all } C \sqsubseteq D \in \mathcal{O}. && (5)
\end{aligned}
$$

At endpoints of symmetric edges, we guess the number of children and introduce corresponding midpoints and endpoints. Note that each midpoint has two r-successor endpoints and we will later merge one of them with the $\widehat{r}$-predecessor. For all $i \in \{0,1\}$, $j \in \{1,\dots,n\}$, and $k \in \{1,\dots,j\}$ we add the following CIs to $\mathcal{O}'$:

$$E^i \sqsubseteq_L N^i_0 \sqcup \dots \sqcup N^i_n \tag{6}$$

$$N^i_j \sqsubseteq_L \bigsqcap_{1 \le k \le j} \exists \widehat{r}.M^{i,j}_k \tag{7}$$

$$M^{i,j}_k \sqsubseteq_L \exists r.E^0 \sqcap \exists r.E^1 \tag{8}$$

For an existential restriction $\exists s.C$, there has to be a child that satisfies C or the element itself satisfies C (reflexivity). Similarly, if we have a universal restriction $\forall s.C$ then C is satisfied in all children and the element itself. For all $i \in \{0,1\}$ and $j \in \{0,\dots,n\}$ we add the following CIs to $\mathcal{O}'$:

$$A_{\exists s.C} \sqcap N^i_j \sqsubseteq_L A_C \sqcup \bigsqcup_{1 \le k \le j} \forall \widehat{r}.(M^{i,j}_k \to \forall r.(E^{1-i} \to A_C)) \tag{9}$$

$$A_{\forall s.C} \sqsubseteq_L A_C \tag{10}$$

$$\exists r.A_{\forall s.C} \sqsubseteq_L \forall r.A_C \tag{11}$$

For utility purposes (because CQs in abstractions have to be connected), we connect successive midpoints with the u-role. For all $i \in \{0,1\}$, $j \in \{2,\dots,n\}$, and $k \in \{1,\dots,j-1\}$ we add the following CI to $\mathcal{O}'$:

$$M^{i,j}_k \sqsubseteq_L \exists u.M^{i,j}_{k+1} \tag{12}$$

Now we construct the concept abstractions that will "create" L-ensembles as depicted in Figure 4. First, we add concept abstractions such that the CQ in it matches onto L-ensembles consisting of one N^i_j element and the j corresponding $\widehat{r}$-successors (midpoints). For all $i \in \{0,1\}$, $j \in \{1,\dots,n\}$, and $k \in \{1,\dots,j\}$ we add the following concept abstraction to $\mathcal{O}'$:

$$L'{:}\top \ \underline{\mathsf{abstracts}}\ L{:}N^i_j(x_0) \wedge \bigwedge_{1\le k\le j} \widehat{r}(x_0,x_k) \wedge M^{i,j}_k(x_j) \tag{13}$$

Intuitively for a midpoint $M^{i,j}_k$ we want the r-successor satisfying E^i to match back onto the $\widehat{r}$-predecessor also satisfying E^i. For this purpose, the CQs in the concept abstractions below contain for each midpoint an r-edge from the midpoint to the mentioned endpoint. As the concept abstractions below match onto the same L-ensembles as the ones above and partial overlaps are forbidden by the refinement function semantics, we achieve the desired structure.

For each $i \in \{0,1\}$ and $j \in \{1,\dots,n\}$ we add the following j concept abstractions to $\mathcal{O}'$:

$$L'{:}\top \ \underline{\mathsf{abstracts}}\ L{:}E^i(x_0) \wedge r(x_1,x_0) \wedge \bigwedge_{1\le k<j} M^{i,j}_k(x_k) \wedge u(x_k,x_{k+1}) \wedge M^{i,j}_{k+1}(x_{k+1})$$

$$\vdots$$

$$L'{:}\top \ \underline{\mathsf{abstracts}}\ L{:}E^i(x_0) \wedge r(x_j,x_0) \wedge \bigwedge_{1\le k<j} M^{i,j}_k(x_k) \wedge u(x_k,x_{k+1}) \wedge M^{i,j}_{k+1}(x_{k+1}) \tag{14}$$

Let $\widehat{q}$ be obtained from q by first replacing every concept atom $C(x) \in q$ with $A_C(x)$, then adding $D(x)$ with $D = E^0 \sqcup E^1$ for every variable $x \in \mathsf{var}(q)$, and lastly replacing every role atom $s(x,y) \in q$ with $r(z,x) \wedge r(z,y)$, where z is a fresh variable. Now we add a concept abstraction that checks whether the query matches into level L:

$$L'{:}\bot \ \underline{\mathsf{abstracts}}\ L{:}\widehat{q} \tag{15}$$

This concludes the construction of $\mathcal{O}'$ and now we will prove that we can use it to reduce from the complement of the simple CQ evaluation problem.

Lemma 18. *$\mathcal{O}, A_0 \not\models q$ iff C_0 is L-satisfiable w.r.t. $\mathcal{O}'$.*

We split the proof of Lemma 18 into a soundness part ("if" direction) and a completeness part ("only if" direction).

E.1 Soundness

Assume that $\mathcal{O}, A_0 \not\models q$. Then there exists a model $\mathcal{I}$ of $\mathcal{O}$ with $A_0^{\mathcal{I}} \neq \emptyset$ and $\mathcal{I} \not\models q$. Using a standard unraveling construction, we can obtain from $\mathcal{I}$ a tree-shaped interpretation $\mathcal{J}$ with $A_0^{\mathcal{J}} \neq \emptyset$ and $\mathcal{J} \not\models q$, see for example (Lutz 2008). Here, an interpretation $\mathcal{I}$ (of $\mathcal{ALC}^{\mathsf{sym}}$) is *tree-shaped* if the undirected graph $G_{\mathcal{I}} = (V,E)$ with $V = \Delta^{\mathcal{I}}$ and $E = \{\{d,e\} \mid (d,e) \in s^{\mathcal{I}} \text{ and } d \neq e\}$ is a tree. We may choose an element as the root and then speak about children and parents.

We may thus assume that $\mathcal{I}$ itself is tree-shaped. By adding subtrees, we may easily achieve that if $d \in (\exists s.C_i)^{\mathcal{I}}$, then there is a *child* e of d such that $d \in C_i^{\mathcal{I}}$, that is, existential restrictions are always satisfied in children in the tree-shaped $\mathcal{I}$. By dropping subtrees, we may then additionally achieve that the outdegree of every node is bounded by n.

We have to construct an A-interpretation $\mathcal{J} = (\mathbf{A}_{\mathcal{O}'}, \prec, (\mathcal{J}_L)_{L\in\mathbf{A}_{\mathcal{O}'}}, \rho)$ that is a model of $\mathcal{O}'$ and such that $C_0^{\mathcal{J}_L} \neq \emptyset$.

In the following, we construct a sequence $\mathcal{J}^0, \mathcal{J}^1, \dots$ of A-interpretations and obtain the desired model $\mathcal{J}$ in the limit. Let $v_0 \in \Delta^{\mathcal{I}}$ be the root of $\mathcal{I}$. We start with defining $\mathcal{J}^0 = (\prec, (\mathcal{J}^0_L)_{L\in\mathbf{A}_{\mathcal{O}}}, \rho^0)$:

$$\begin{aligned}
\Delta^{\mathcal{J}^0_L} &= \{v_0\} \\
A^{\mathcal{J}^0_L} &= \{v_0\} \quad \text{for all } A \in \mathsf{cl}(\mathcal{O}) \cap \mathbf{C} \text{ with } v_0 \in A^{\mathcal{I}} \\
(A_C)^{\mathcal{J}^0_L} &= \{v_0\} \quad \text{for all } C \in \mathsf{cl}(\mathcal{O}) \text{ with } v_0 \in C^{\mathcal{I}} \\
(E^0)^{\mathcal{J}^0_L} &= \{v_0\} \\
\Delta^{\mathcal{J}^0_{L'}} &= \{d'\} \quad \text{with } d' \text{ being a fresh domain element}
\end{aligned}$$

To construct $\mathcal{J}^{i+1} = (\prec, (\mathcal{J}^{i+1}_L)_{L\in\mathbf{A}_{\mathcal{O}}}, \rho^{i+1})$ from $\mathcal{J}^i$ we start with $\mathcal{J}^{i+1} = \mathcal{J}^i$. Now we construct a structure similar to Figure 4. Let $d \in \Delta^{\mathcal{J}^i_L}$ with $d \in E^i$ for some $i \in \{0,1\}$ but $d \notin N^i_j$ for all $i \in \{0,1\}$ and $j \in \{0,\dots,n\}$. Let $Q = e_1,\dots,e_m$ be the children of d in $\mathcal{I}$. We do the following (disjoint) operations for all such d :

- add d to $(N^i_m)^{\mathcal{J}^0_L}$;
- for all $l \in \{1,\dots,m\}$ we
  - add a fresh domain element e_l to $\Delta^{\mathcal{J}^i_L}$,
  - add e_l to $(E^{1-i})^{\mathcal{J}^i_L}$,
  - add e_l to A for all $A \in \mathsf{cl}(\mathcal{O}) \cap \mathbf{C}$ with $e_l \in A^{\mathcal{I}}$,
  - add e_l to A_C for all $C \in \mathsf{cl}(\mathcal{O})$ with $e_l \in C^{\mathcal{I}}$,
  - add a fresh domain element v_l to $\Delta^{\mathcal{J}^i_L}$,
  - add v_l to $(M^{i,j}_l)^{\mathcal{J}^i_L}$,
  - add (d, v_l) to $\widehat{r}^{\mathcal{J}^i_L}$,
  - add (v_l, d) to $r^{\mathcal{J}^i_L}$,
  - add (v_l, e_l) to $r^{\mathcal{J}^i_L}$ and
  - add (v_{l-1}, v_l) to $u^{\mathcal{J}^i_L}$ if $l \ge 2$;
- if $m \ge 0$, then
  - add a fresh element d' to $\Delta^{\mathcal{J}^i_{L'}}$, and
  - add $(d, v_1, \dots, v_l)$ to $\rho^i_L(d')$.

Note that $m \le n$ is always true because of how we defined tree-shapedness for $\mathcal{I}$ above. As announced, we take $\mathcal{J}$ to be the limit of the constructed sequence of $\mathcal{J}^0, \mathcal{J}^1, \dots$.

What remains to show is that $\mathcal{J}$ is a model of $\mathcal{O}'$ and that C_0 is satisfied on L. We do this step by step starting with the basic condition that $\mathcal{J}$ is an A-interpretation. Subsequently, we will show that C_0 is satisfied on L and that all statements in $\mathcal{O}$ are satisfied.

Lemma 19. *$\mathcal{J} = (\mathbf{A}_{\mathcal{O}'}, \prec, (\mathcal{J}_L)_{L\in\mathbf{A}_{\mathcal{O}'}}, \rho)$ is an A-interpretation.*

Proof. To prove the lemma we go through the three conditions of A-interpretations.

- we only have two abstraction levels with $L \prec L'$. Consequently $(\mathcal{A}_{\mathcal{J}}, \{(L', L) \mid L \prec L'\})$ is obviously a tree;
- by definition, $\Delta^{\mathcal{J}^0_L}$ $\Delta^{\mathcal{J}^0_{L'}}$ are non-empty and thus the same holds for $\Delta^{\mathcal{J}_L}$ and $\Delta^{\mathcal{J}_{L'}}$;

- whenever we add a tuple to the range of ρ in the construction of $\mathcal{J}$, the elements in it are one endpoint e and a number of fresh midpoints. In the same construction step, we add e to N^i_j for some $i \in \{1,2\}$ and $j \in \{1,\ldots,n\}$ which implies that we will never consider it again in another step of constructing $\mathcal{J}$. Thus there is always at most one $d \in \Delta^{\mathcal{I}}$ such that e occurs in $\rho_L(d)$.

❑

Now we are ready to prove the following lemma which establishes the soundness of the reduction.

Lemma 20. *$\mathcal{J} = (\mathbf{A}_{\mathcal{O}'}, \prec, (\mathcal{J}_L)_{L\in\mathbf{A}_{\mathcal{O}'}}, \rho)$ is a model of $\mathcal{O}'$ and $C_0^{\mathcal{J}_L} \neq \emptyset$.*

Proof. We use the fact that $\mathcal{J}$ is indeed an A-interpretation as proven in the previous lemma. First, we prove the second part of the claim. Since we assumed that $C_0^{\mathcal{I}} \neq \emptyset$, the construction of $\mathcal{J}_L$ implies the desired $C_0^{\mathcal{J}_L} \neq \emptyset$.

Now we show that $\mathcal{J}$ is a model of $\mathcal{O}'$. We prove below that the CIs and CA of (9) to (11) and (15) are satisfied in $\mathcal{J}$. For all other statements it is trivial to verify that they are satisfied since it follows directly from the construction of $\mathcal{J}$.

For (9) let us assume $d \in (A_{\exists s.C} \sqcap N^i_j)^{\mathcal{J}_L}$ for some $i \in \{0,1\}$ and $j \in \{1,\ldots,n\}$. Then by the definition of $\mathcal{J}$, we have $d \in (\exists s.C)^{\mathcal{I}}$. The tree-shapedness, as defined above, implies that there is a child e of d in $\mathcal{I}$ such that $e \in C^{\mathcal{I}}$. The construction of $\mathcal{J}$ then lets us obtain a midpoint $v \in \Delta^{\mathcal{J}_L}$ with $(d,v) \in \widehat{r}^{\mathcal{J}_L}$, $(v,e) \in r^{\mathcal{J}_L}$, and $e \in A_C$ and thus (9) is satisfied.

For (10) let us assume $d \in (A_{\forall s.C})^{\mathcal{J}_L}$. By definition of $\mathcal{J}$ we have $d \in (A_{\forall s.C})^{\mathcal{I}}$ and because $\mathcal{I}$ is a model of $\mathcal{O}$ the construction of $\mathcal{J}$ immediately implies $d \in A_C$ and thus (10) is satisfied.

For (11) let us assume $d \in \exists r.(A\forall s.C)^{\mathcal{J}_L}$. By the construction of $\mathcal{J}$, this can only be the case for a midpoint, that is for all $e \in \Delta^{\mathcal{J}_L}$ with $(d,e) \in r^{\mathcal{J}_L}$ and $e \in A_{\forall s.C}$ we have $e \in \Delta^{\mathcal{I}}$ and thus $e \in (\forall s.C)^{\mathcal{I}}$. The construction then also implies that for all $e' \in \Delta^{\mathcal{J}_L}$ with $(d,e') \in r^{\mathcal{J}_L}$ we have $(e,e') \in s^{\mathcal{I}}$ (and $(e',e) \in s^{\mathcal{I}}$). Now we can use that $\mathcal{I}$ is a model to obtain $e' \in C^{\mathcal{I}}$ and thus $e' \in (A_C)^{\mathcal{J}_L}$, as required.

For (15) we argue by contradiction. Assume that there is a homomorphism h from $\widehat{q}$ to $\mathcal{J}_L$. We show that $h|_{\mathsf{var}(q)}$ is a homomorphism from q to $\mathcal{I}$ contradicting our original assumtion that $\mathcal{I} \not\models q$. Consider any $C(x) \in q$. By definition of $\widehat{q}$ we have $A_C(x) \in \widehat{q}$ and $(E^0 \sqcup E^1)(x) \in \widehat{q}$. Consequently $\widehat{h}$ maps x to an endpoint $e = h(x)$ in $\mathcal{J}_L$ with $e \in A_C^{\mathcal{J}_L}$ and the construction of $\mathcal{J}_L$ then implies that $e \in C^{\mathcal{I}}$.

Consider any $s(x,y) \in q$. By definition of $\widehat{q}$, we have $r(z,x) \in \widehat{q}$ and $r(z,y) \in \widehat{q}$ with z a variable not occurring in q. Furthermore, x and y are again mapped to endpoints like in the concept atom case. The construction of $\mathcal{J}_L$ then implies that either $h(x)$ is a child of $h(y)$ or the inverse is true in $\mathcal{I}$. Thus $(h(x),h(y)) \in s^{\mathcal{I}}$ and $(h(y),h(x)) \in s^{\mathcal{I}}$ which proves that $h|_{\mathsf{var}(q)}$ is a homomorphism from q to $\mathcal{I}$ and consequently leading to a contradiction since we assumed that $\mathcal{I} \not\models q$.

❑

E.2 Completeness

Assume that C_0 is L-satisfiable w.r.t. $\mathcal{O}'$ and let $\mathcal{J} = (\mathbf{A}_{\mathcal{O}'}, \prec, (\mathcal{J}_L)_{L\in\mathbf{A}_{\mathcal{O}'}}, \rho)$ be a model of $\mathcal{O}'$ with $C_0^{\mathcal{J}_L} \neq \emptyset$. We need to construct a model $\mathcal{I}$ of $\mathcal{O}$ such that $A_0^{\mathcal{I}} \neq \emptyset$ and $\mathcal{I} \not\models q$.

We define $\mathcal{I}$ as follows:

$$\begin{aligned}
\Delta^{\mathcal{I}} &= \{d \mid d \in \Delta^{\mathcal{J}_L} \text{ and } d \in (E^0 \sqcup E^1)^{\mathcal{J}_L}\}\\
s^{\mathcal{I}} &= \{(d,d) \mid d \in (E^0 \sqcup E^1)^{\mathcal{J}_L}\} \cup\\
&\quad\ \{(d,d') \mid d,d' \in (E^0 \sqcup E^1)^{\mathcal{J}_L} \text{ and } e \in \Delta^{\mathcal{J}_L}\}\\
&\quad\ \text{with } (e,d) \in r^{\mathcal{J}} \text{ and } (e,d') \in r^{\mathcal{J}}\}\\
A^{\mathcal{I}} &= A^{\mathcal{J}} \qquad \text{for all } A \in \mathsf{sig}(\mathcal{O}) \cap \mathbf{C}
\end{aligned}$$

What remains to show is that $\mathcal{I}$ is a model of $\mathcal{O}$, $A_0^{\mathcal{I}} \neq \emptyset$, and $\mathcal{I} \not\models q$. To prove this we first prove an intermediary lemma which intuitively says that the A_C satisfied at the endpoints in $\mathcal{J}$ coincide with the concepts C satisfied by the corresponding element in $\mathcal{I}$.

Lemma 21. *For every element $d \in (E^0 \sqcup E^1)^{\mathcal{J}_L}$ and $\mathcal{ALC}^{\mathsf{sym}}$ concept $C \in \mathsf{cl}(\mathcal{O})$ we have $d \in (A_C)^{\mathcal{J}_L}$ iff $d \in C^{\mathcal{I}}$*

Proof. To prove the "if" direction of the lemma, we do an induction on C.

- If $C = B$, then (1) and the definition of $\mathcal{I}$ give us $d \in B^{\mathcal{I}}$.
- If $C = \neg B$, then (2) and the definition of $\mathcal{I}$ give us $d \in (\neg B)^{\mathcal{I}}$.
- If $C = D \sqcap D'$, then (3) implies $d \in (A_D \sqcap A_{D'})^{\mathcal{J}_L}$. Now we can apply the IH twice to get $d \in (D \sqcap D')^{\mathcal{I}}$.
- If $C = D \sqcup D'$, then (4) implies $d \in (A_D \sqcup A_{D'})^{\mathcal{J}_L}$. Now we can apply the IH twice to get $d \in (D \sqcup D')^{\mathcal{I}}$.
- If $C = \exists s.D$, then we use multiple CIs and concept abstractions for our reasoning. Firstly, $d \in N^i_j$ for some $i \in \{0,1\}$ and $j \in \{0,\ldots,n\}$ because of (6). Next, we consider (9). If $j = 0$, then $d \in A_D$ and by the IH $d \in D^{\mathcal{I}}$ and thus $d \in (\exists s.D)^{\mathcal{I}}$, as required. If $j > 0$, then (7) results in j (at least one) $\widehat{r}$-successors of d that satisfy $M^{i,j}_k$ for all $k \in \{1,\ldots,j\}$.

  Let $M = \{e \mid (d,e) \in \widehat{r}^{\mathcal{J}_L} \text{ and } e \in M^{i,j}_k \text{ for some } k \in \{1,\ldots,j\}\}$ be the set of those $\widehat{r}$-successors. Point (8) tells us that each $e \in M$ has two r-successors, one that satisfies E^i and one that satisfies E^{1-i}. For the next step, we are specifically interested in the E^{1-i} successor in conjunction with (9).

  Point (9) together with the previous part of the proof imply that from d there is a $\widehat{r};r$ path via one element $v \in M$ to an element $e \in (A_D \sqcap E^{1-i})^{\mathcal{J}_L}$. If we apply the IH on e we get $e \in D^{\mathcal{I}}$. What remains to show is that there is also an r edge pointing from v to d.

  The concept abstractions in (13) make d and all the $v' \in M$ an L-ensemble $\bar{e}$. In particular, v is part of $\bar{e}$. Point (12) implies that there is a chain $\{(v_1,v_2),\ldots,(v_{j-1},v_j)\} \subseteq u^{\mathcal{J}_L}$ such that $v_l \in M^{i,j}_l$ for all $l \in \{1,\ldots,j\}$. Point (14) takes this chain together with one r-successor that satisfies E^i and makes it an L-ensemble $\bar{e}'$. It is easy to see that $\bar{e}$ and $\bar{e}'$ must overlap partially and the semantics of refinement functions then implies that they overlap fully, that is, $\bar{e} = \bar{e}'$. Thus the elements in M are connected by a chain of u-edges.

Now consider the following concept abstraction of Point (14).

$$L'{:}\top \ \underline{\mathsf{abstracts}}\ L{:}E^i(x_0) \wedge r(x_k, x_0) \wedge \bigwedge_{1 \le l < j} M_l^{i,j}(x_l) \wedge u(x_l, x_{l+1}) \wedge M_{l+1}^{i,j}(x_{l+1})$$

This forces another L-ensemble $\bar{e}''$ into existence. In this ensemble, v has an r-successor that satisfies E^i. Again it is straightforward to see that $\bar{e}, \bar{e}'$ and $\bar{e}''$ partially overlap and must therefore be equal. This implies that the r-successor of v is d. We have now shown that there is an $r^-;r$ path from d via v to e and $e \in D^{\mathcal{I}}$. Thus by the definition of $\mathcal{I}$, we have $d \in (\exists s.D)^{\mathcal{I}}$, as required.

- If $C = \forall s.D$, then first let us consider d itself. By (10) we have $d \in (A_D)^{\mathcal{J}_L}$ and with the IH we obtain $d \in D^{\mathcal{I}}$. Otherwise let $e \in \Delta^{\mathcal{I}}$ be an element with $e \neq d$ and $(d, e) \in s^{\mathcal{I}}$. We have to show that $e \in D^{\mathcal{I}}$. The definition of $\mathcal{I}$ implies that $e \in (E^i)^{\mathcal{J}_L}$ for some $i \in \{0, 1\}$ and that there is a $v \in \Delta^{\mathcal{J}_L}$ with $(v, d) \in r^{\mathcal{J}_L}$ and $(v, e) \in r^{\mathcal{J}_L}$. This v then has to satisfy $\exists r.A_{\forall s.D}$ and due to (11) we then have $e \in (A_D)^{\mathcal{J}_L}$. Now we can use the IH to obtain $e \in D^{\mathcal{I}}$ and thus $d \in (\forall s.D)^{\mathcal{I}}$, as required.

"only if". We argue by contrapositive. If $d \notin (A_C)^{\mathcal{J}_L}$, then $d \in (A_{\overline{C}})^{\mathcal{J}_L}$ by (2). We can now apply the '$\Leftarrow$' direction to get $d \in \overline{C}^{\mathcal{I}}$, and thus $d \notin C^{\mathcal{I}}$. ❑

Now proving that $\mathcal{I}$ is a model of $\mathcal{O}$ is very straightforward. Let $d \in C^{\mathcal{I}}$ for any $d \in \Delta^{\mathcal{I}}$. We have to show that $d \in D^{\mathcal{I}}$ for all CIs $C \sqsubseteq D \in \mathcal{O}$. Lemma 21 implies that $d \in (A_C)^{\mathcal{J}_L}$. Point (5) in the construction of $\mathcal{O}'$ implies $d \in (A_D)^{\mathcal{J}_L}$ and now we can again apply Lemma 21 to obtain $d \in D^{\mathcal{I}}$. Thus $\mathcal{I}$ is a model of $\mathcal{O}$.

By assumption $C_0^{\mathcal{J}} \neq \emptyset$ and thus there is a $d_0 \in (A_0 \sqcap (E^0 \sqcup E^1))^{\mathcal{J}_L}$. The construction of $\mathcal{I}$ then implies $d_0 \in C_0^{\mathcal{I}}$.

Lastly, we have to show that $\mathcal{I} \not\models q$. We prove this by contradiction. Assume that there is a homomorphism h from q to $\mathcal{I}$. We want to show that we can extend h to also be a homomorphism from $\widehat{q}$ to $\mathcal{J}_L$ which would make $\mathcal{J}$ no longer a model of $\mathcal{O}'$ because of (15).

First, consider h as a partial homomorphism from $\widehat{q}$ to $\mathcal{J}_L$. Let $C(x) \in \widehat{q}$ be a concept atom. There are two cases. If $C = A_D$ for some $D(x) \in q$ then $h(x) \in C^{\mathcal{J}_L}$ because of Lemma 21. Otherwise $C = E^0 \sqcup E^1$ and then $h(x) \in C^{\mathcal{J}_L}$ simply by the construction of $\mathcal{I}$. What remains to show is that all role atoms in $\widehat{q}$ are satisfied.

Let $r(z, x) \in \widehat{q}$ be a role atom. The definition of $\widehat{q}$ implies that there is a second role atom $r(z, y)$ in $\widehat{q}$ and that $s(x, y)$ or $s(y, x) \in q$. The two cases are analogous so let us assume w.l.o.g. that $s(x, y) \in q$. Since the construction of $s^{\mathcal{I}}$ is defined in such a way that $(h(x), h(y)) \in s^{\mathcal{I}}$ if and only if there is an $e \in \Delta^{\mathcal{J}_L}$ with $(e, h(x)) \in r^{\mathcal{J}_L}$ and $(e, h(y)) \in r^{\mathcal{J}_L}$ we can simply extend h by $h(z) = e$ and now we have $(h(z), h(x)) \in r^{\mathcal{J}_L}$, as required. Thus $\mathcal{J}_L \models \widehat{q}$ which leads to a contradiction since (15) would now imply that $\mathcal{J}$ is not a model of $\mathcal{O}'$.

F Proofs for Section 6.2

We prove Proposition 1 and complete the proof of Theorem 6, starting with the former.

Proposition 1. *For every $\mathcal{ALCI}$-concept C, abstraction level L, and $\mathcal{ALCHI}^{\mathsf{abs}}[\mathrm{cr}, \mathrm{rr}]$-ontology $\mathcal{O}$: C is L-satisfiable w.r.t. $\mathcal{O}$ iff C is L-satisfiable w.r.t. $\mathcal{O}$ under the repetition-free semantics.*

Proof. Assume that C_0 is L_0-satisfiable w.r.t. $\mathcal{O}$. Then there is a model $\mathcal{I} = (\mathbf{A}_{\mathcal{O}}, \prec, (\mathcal{I}_L)_{L \in \mathbf{A}_{\mathcal{I}}}, \rho)$ of $\mathcal{O}$ with $C_0^{\mathcal{I}_{L_0}} \neq \emptyset$. Assume that for some $L_1, L_2 \in \mathbf{A}_{\mathcal{O}}$ and $d_0 \in \Delta^{\mathcal{I}_{L_1}}$ we have $\rho_{L_2}(d_0) = \bar{d}$ with $\bar{d} = d_1 \cdots d_n$ such that an element e_0 occurs in positions $i_1, \ldots, i_m$ of $\bar{d}$ with $m \geq 2$.

We construct a model $\mathcal{I}' = (\mathcal{I}_{\mathcal{O}}, \prec, (\mathcal{I}'_L)_{L \in \mathbf{A}_{\mathcal{I}}}, \rho')$ of $\mathcal{O}$ with $C^{\mathcal{I}'_L} \neq \emptyset$ such that e_0 no longer occurs multiple times in $\rho'_{L_2}(d_0)$ (in fact, it does not occur at all anymore). Let $e_1, \ldots, e_m$ be fresh domain elements. We define an A-interpretation $\mathcal{I}' = (\mathbf{A}_{\mathcal{O}}, \prec, (\mathcal{I}'_L)_{L \in \mathbf{A}_{\mathcal{I}}}, \rho')$ that differs from $\mathcal{I}$ only in $\mathcal{I}'_{L_2}$ and ρ'. To start, set

$$\Delta^{\mathcal{I}'_{L_2}} = \Delta^{\mathcal{I}_{L_2}} \cup \{e_1, \ldots, e_m\}.$$

For every $d \in \Delta^{\mathcal{I}'_{L_2}}$, put $d^{\uparrow} = d$ if $d \in \Delta^{\mathcal{I}_{L_2}}$ and $d^{\uparrow} = e_0$ if $d \notin \Delta^{\mathcal{I}_{L_2}}$. Then define

$$\begin{aligned} A^{\mathcal{I}'_{L_2}} &= \{d \in \Delta^{\mathcal{I}'_{L_2}} \mid d^{\uparrow} \in A^{\mathcal{I}_{L_2}}\} \\ r^{\mathcal{I}'_{L_2}} &= \{(d, e) \in \Delta^{\mathcal{I}'_{L_2}} \times \Delta^{\mathcal{I}'_{L_2}} \mid (d^{\uparrow}, e^{\uparrow}) \in r^{\mathcal{I}_{L_2}}\} \end{aligned}$$

for all concept names A and role names r and define ρ' like ρ except that $\rho'(d_0)$ is obtained from $\rho(d_0)$ by replacing the occurrence of e in position i_j with e_j, for $1 \leq j \leq m$.

Clearly, $C^{\mathcal{I}_L} \neq \emptyset$ implies $C^{\mathcal{I}'_L} \neq \emptyset$ and $\mathcal{I}'$ satisfies all role inclusions in $\mathcal{O}$ since $\mathcal{I}$ does. Moreover, it is easy to show the following by induction on the structure of C:

Claim. For all $\mathcal{ALCI}$-concepts C and $d \in \Delta^{\mathcal{I}'_{L_2}}$: $d \in C^{\mathcal{I}'_{L_2}}$ iff $d^{\uparrow} \in C^{\mathcal{I}_{L_2}}$.

It follows that $\mathcal{I}$ also satisfies all concept inclusion in $\mathcal{O}$. We next show that it also satisfies all concept and role refinements, and thus is a model of $\mathcal{O}$.

Let $q(\bar{x})$ be a CQ such that $\mathcal{O}$ contains a concept refinement $L_2{:}q(\bar{x})\ \underline{\mathsf{refines}}\ L_1{:}C$. Further assume that $d' \in C^{\mathcal{I}'_{L_1}}$. Then $d' \in C^{\mathcal{I}_{L_1}}$ by the claim and thus $\rho(d')$ is defined and the function h that satisfies $h(\bar{x}) = \rho(d')$ is a homomorphism from q to $\mathcal{I}_{L_2}$. By definition of ρ' and the claim, the function h' that satisies $h'(\bar{x}) = \rho'(d')$ is a homomorphism from q to $\mathcal{I}'_{L_2}$. The case of role refinements is similar.

By applying this construction multiple times, we eventually arrive at a repetition-free model of $\mathcal{O}$, as desired. ❑

We next present the remaining details of the proof of Theorem 6 which we repeat here for the reader's convenience.

Theorem 6. *Under the repetition-free semantics, satisfiability is undecidable in $\mathcal{ALC}^{\mathsf{abs}}[\mathrm{ca}]$, $\mathcal{ALC}^{\mathsf{abs}}[\mathrm{ra}]$, $\mathcal{EL}^{\mathsf{abs}}[\mathrm{rr}, \mathrm{ca}]$, and $\mathcal{EL}^{\mathsf{abs}}[\mathrm{rr}, \mathrm{ra}]$.*

Recall that we want to construct an ontology $\mathcal{O}$ and choose a concept name S and abstraction level L such that

S is L-satisfiable w.r.t. $\mathcal{O}$ iff M does not halt on the empty tape. A part of $\mathcal{O}$ has already been given in the main part of the paper. We proceed with the construction of $\mathcal{O}$.

We first step up the initial configuration:

$$S \sqsubseteq_L A_{q_0} \sqcap A_\Box \sqcap B_\rightarrow)$$
$$B_\rightarrow \sqsubseteq_L \forall t.(A_\Box \sqcap B_\rightarrow)$$

Transitions are again indicated by marker concepts:

$$A_q \sqcap A_\sigma \sqsubseteq_L \forall c.B_{q',\sigma',M}$$

for all $q \in Q$ and $\sigma \in \Gamma$ such that $\delta(q,\sigma) = (q',\sigma',M)$. We implement transitions as follows:

$$B_{q,\sigma,M} \sqsubseteq_L A_\sigma \quad \exists t.B_{q,\sigma,L} \sqsubseteq_L A_q \quad B_{q,\sigma,R} \sqsubseteq_L \forall t.A_q$$

for all $q \in Q$, $\sigma \in \Gamma$, and $M \in \{L,R\}$.

We next mark cells that are not under the head:

$$A_q \sqsubseteq_{L_n} \forall t.H_\leftarrow \qquad \exists t.A_q \sqsubseteq_{L_n} H_\rightarrow$$
$$H_\leftarrow \sqsubseteq_{L_n} \forall t.H_\leftarrow \qquad \exists t.H_\rightarrow \sqsubseteq_{L_n} H_\rightarrow$$

for all $q \in Q$. We can now say that cells that are not under the head do not change:

$$(H_\leftarrow \sqcup H_\rightarrow) \sqcap A_\sigma \sqsubseteq_{L_n} \forall c_i.A_\sigma$$

for all $\sigma \in \Gamma$ and $i \in \{1,2\}$. State, content of tape, and head position must be unique:

$$A_q \sqcap A_{q'} \sqsubseteq_{L_n} \bot \quad A_\sigma \sqcap A_{\sigma'} \sqsubseteq_{L_n} \bot$$
$$(H_\leftarrow \sqcup H_\rightarrow) \sqcap A_q \sqsubseteq_{L_n} \bot$$

for all $q,q' \in Q$ and $\sigma,\sigma' \in \Gamma$ with $q \neq q'$ and $\sigma \neq \sigma'$.

Since we reduce from the complement of the halting problem, we do not want the halting state to be reached:

$$A_{q_h} \sqsubseteq_L \bot.$$

This concludes the construction of $\mathcal{O}$.

Lemma 22. *S is L-satisfiable w.r.t. $\mathcal{O}$ iff M does not halt on the empty tape.*

Proof. "if". Assume that M does not halt on the empty tape. We use $|w|$ to denote the length of a word $w \in \Sigma^*$ and construct an A-interpretation $\mathcal{I}$ as follows:

$$\begin{aligned}
\Delta^{\mathcal{I}_L} &= \{c_i t_j \mid i,j \in \mathbb{N}\}\\
S^{\mathcal{I}_L} &= \{c_0 t_0\}\\
A_q^{\mathcal{I}_L} &= \{c_i t_j \mid K_i = wqw' \text{ and } |w| = j\}\\
A_\sigma^{\mathcal{I}_L} &= \{c_i t_j \mid K_i = \sigma_0 \cdots \sigma_k q \sigma_{k+1} \cdots \sigma_l \text{ and either}\\
&\qquad \sigma = \sigma_j \text{ or } j > l \text{ and } \sigma = \Box\}\\
B_\rightarrow^{\mathcal{I}_L} &= \{c_0 t_j \mid j \in \mathbb{N}\}\\
B_{q',\sigma',M}^{\mathcal{I}_L} &= \{c_i t_j \mid K_{i-1} = wq\sigma w', |w| = j \text{ and}\\
&\qquad \delta(q,\sigma) = (q',\sigma',M)\}\\
H_\rightarrow^{\mathcal{I}_L} &= \{c_i t_j \mid K_i = wqw' \text{ and } j < |w|\}\\
H_\leftarrow^{\mathcal{I}_L} &= \{c_i t_j \mid K_i = wqw' \text{ and } j > |w|\}\\
t^{\mathcal{I}_L} &= \{(c_i t_j, c_i t_{j+1}) \mid i,j \in \mathbb{N}\}\\
c^{\mathcal{I}_L} &= \{(c_i t_j, c_{i+1} t_j) \mid i,j \in \mathbb{N}\}
\end{aligned}$$

for all $q \in Q$, $\sigma \in \Gamma$, and $M \in \{L,R\}$.

For $\mathcal{I}_{L'}$ we simply define a singleton domain $\Delta^{\mathcal{I}_{L'}} = \{d\}$ and no concept extensions; ρ we leave undefined for all elements. It is straightforward to verify that $\mathcal{I}$ is a model of $\mathcal{O}$, and thus S is L-satisfiable w.r.t. $\mathcal{O}$.

"only if". Assume that S is L-satisfiable w.r.t. $\mathcal{O}$ and let $\mathcal{I}$ be a model of $\mathcal{O}$ with $S^{\mathcal{I}_L} \neq \emptyset$. We identify a grid in $\mathcal{I}_L$ in the form of a mapping $\pi : \mathbb{N} \times \mathbb{N} \to \Delta^{\mathcal{I}_L}$ such that for all $i,j \in \mathbb{N}\times\mathbb{N}$, $(\pi(i,j), \pi(i+1,j)) \in t^{\mathcal{I}_L}$ and $(\pi(i,j), \pi(i,j+1)) \in c^{\mathcal{I}_L}$. We construct such a π in several steps:

- Start with choosing some $d \in S^{\mathcal{I}_L}$ and set $\pi(0,0) = d$.
- Complete the diagonal. If $\pi(i,i)$ is defined and $\pi(i+1,i+1)$ undefined, then choose $d,e \in \Delta^{\mathcal{I}_L}$ such that $(\pi(i,i),d) \in t^{\mathcal{I}_L}$ and $(d,e) \in c^{\mathcal{I}_L}$ and set $\pi(i+1,i) = d$ and $\pi(i+1,i+1) = e$.
- Add grid cells in the upwards direction. If $\pi(i,i), \pi(i+1,1), \pi(i+1,i+1)$ are defined and $\pi(i,i+1)$ is undefined, then choose $d \in \Delta^{\mathcal{I}_L}$ such that $(\pi(i,i),d) \in c^{\mathcal{I}_L}$ and set $\pi(i+1,i) = d$. We have to argue that $(\pi(i+1,i), \pi(i+1,i+1)) \in t^{\mathcal{I}_L}$. There is some $e \in \Delta^{\mathcal{I}_L}$ with $(\pi(i+1,i),e) \in t^{\mathcal{I}_L}$. The elements $\pi(i,i), \pi(i+1,i), \pi(i,i+1), \pi(i+1,i+1), e)$ cannot all be distinct because then the first concept abstraction in $\mathcal{O}$ applies, meaning that $\mathcal{I}$ cannot be a model of $\mathcal{O}$. Thus at least two out of these five elements have to be identical; then, the first concept abstraction does not apply since we work under the repetition-free semantics. The additional concept abstraction in $\mathcal{O}$, however, rules out any identifications except that of $\pi(i+1,i+1)$ and e. Consequently, $(\pi(i+1,i), \pi(i+1,i+1)) \in t^{\mathcal{I}_L}$ as required.
- Add grid cells in the downwards direction. Analogous to the upwards case.

We can now read off the computation of M on the empty tape from the grid in a straightforward way, using the concept names A_σ for the tape content and A_q for the state and head position. Since $A_{a_h}^{\mathcal{I}_L} = \emptyset$, the computation is nonterminating. ❑

G Proofs for Section 6.3

We complete the proof of Theorem 7 which we repeat here for the reader's convenience.

Theorem 7. *Under the DAG semantics, satisfiability is undecidable in $\mathcal{ALC}^{\mathsf{abs}}[\mathrm{ca,cr}]$ and $\mathcal{EL}^{\mathsf{abs}}[\mathrm{ca,cr,rr}]$.*

Recall that we want to construct an ontology $\mathcal{O}$ and choose a concept name S and abstraction level L such that S is L-satisfiable w.r.t. $\mathcal{O}$ iff M does not halt on the empty tape. A part of $\mathcal{O}$ has already been given in the main body of the paper. The construction of $\mathcal{O}$ is completed by adding the concept inclusions presented in Appendix F.

Lemma 23. *S is L-satisfiable w.r.t. $\mathcal{O}$ iff M does not halt on the empty tape.*

Proof. "if". Assume that M does not halt on the empty tape. We construct an A-interpretation $\mathcal{I}$ as follows:

$$
\begin{aligned}
\Delta^{\mathcal{I}_L} &= \{c_i t_j \mid i, j \in \mathbb{N}\} \\
A_t^{\mathcal{I}_L} &= \{c_0 t_j \mid j \in \mathbb{N}\} \\
A_c^{\mathcal{I}_L} &= \Delta^{\mathcal{I}_L} \\
S^{\mathcal{I}_L} &= \{c_0 t_0\} \\
A_q^{\mathcal{I}_L} &= \{c_i t_j \mid K_i = wqw' \text{ and } |w| = j\} \\
A_\sigma^{\mathcal{I}_L} &= \{c_i t_j \mid K_i = \sigma_0 \cdots \sigma_k q \sigma_{k+1} \cdots \sigma_l \text{ and either} \\
&\qquad \sigma = \sigma_j \text{ or } j > l \text{ and } \sigma = \square\} \\
B_\rightarrow^{\mathcal{I}_L} &= \{c_0 t_j \mid j \in \mathbb{N}\} \\
B_{q',\sigma',M}^{\mathcal{I}_L} &= \{c_i t_j \mid K_{i-1} = wq\sigma w', |w| = j \text{ and} \\
&\qquad \delta(q,\sigma) = (q', \sigma', M)\} \\
H_\rightarrow^{\mathcal{I}_L} &= \{c_i t_j \mid K_i = wqw' \text{ and } j < |w|\} \\
H_\leftarrow^{\mathcal{I}_L} &= \{c_i t_j \mid K_i = wqw' \text{ and } j > |w|\} \\
X_1^{\mathcal{I}_L} &= \{c_i t_j \mid i \bmod 2 = 0 \text{ and } j \bmod 2 = 0\} \\
X_2^{\mathcal{I}_L} &= \{c_i t_j \mid i \bmod 2 = 0 \text{ and } j \bmod 2 = 1\} \\
X_3^{\mathcal{I}_L} &= \{c_i t_j \mid i \bmod 2 = 1 \text{ and } j \bmod 2 = 0\} \\
X_4^{\mathcal{I}_L} &= \{c_i t_j \mid i \bmod 2 = 1 \text{ and } j \bmod 2 = 1\} \\
t^{\mathcal{I}_L} &= \{(c_i t_j, c_i t_{j+1}) \mid i, j \in \mathbb{N}\} \\
c^{\mathcal{I}_L} &= \{(c_i t_j, c_{i+1} t_j) \mid i, j \in \mathbb{N}\}
\end{aligned}
$$

for all $q \in Q$, $\sigma \in \Gamma$, and $M \in \{L, R\}$.

We use $\bar{e}^i_j$ to denote a 4-tuple $\bar{e}^i_j = c_i t_j \cdot c_i t_{j+1} \cdot c_{i+1} t_j \cdot c_{i+1} t_{j+1}$ with $c_i, t_j \in \Delta^{\mathcal{I}_L}$ and $i, j \in \mathbb{N}$. We define four sets of 4-tuples that represent the answers to the four CQs in the abstractions:

$$
\begin{aligned}
Q_1 &= \{\bar{e}^i_j \mid i \bmod 2 = 0 \text{ and } j \bmod 2 = 0\} \\
Q_2 &= \{\bar{e}^i_j \mid i \bmod 2 = 0 \text{ and } j \bmod 2 = 1\} \\
Q_3 &= \{\bar{e}^i_j \mid i \bmod 2 = 1 \text{ and } j \bmod 2 = 0\} \\
Q_4 &= \{\bar{e}^i_j \mid i \bmod 2 = 1 \text{ and } j \bmod 2 = 1\}
\end{aligned}
$$

Now we define the $\mathcal{I}_{L_k}$ for all $k \in \{1, \ldots, 4\}$:

$$
\begin{aligned}
\Delta^{\mathcal{I}_{L_k}} &= \{d^i_j \mid \bar{e}^i_j \in Q_k\} \\
U_i^{\mathcal{I}_{L_k}} &= \Delta^{\mathcal{I}_{L_k}}
\end{aligned}
$$

Next we add $\bar{e}^i_j$ to $\rho_L(d^i_j)$ for all $k \in \{1, \ldots, 4\}$ and $d^i_j \in \Delta^{\mathcal{I}_{L_k}}$. It is straightforward to verify that $\mathcal{I}$ is a model of $\mathcal{O}$, and thus S is L-satisfiable w.r.t. $\mathcal{O}$.

"only if". Assume that S is L-satisfiable w.r.t. $\mathcal{O}$ and let $\mathcal{I}$ be a model of $\mathcal{O}$ with $S^{\mathcal{I}_L} \neq \emptyset$. We identify a grid in $\mathcal{I}_L$ in the form of a mapping $\pi : \mathbb{N} \times \mathbb{N} \to \Delta^{\mathcal{I}_L}$ such that for all $i, j \in \mathbb{N} \times \mathbb{N}$, $(\pi(i,j), \pi(i{+}1, j)) \in t^{\mathcal{I}_L}$ and $(\pi(i,j), \pi(i, j{+}1)) \in c^{\mathcal{I}_L}$. We construct such a π in several steps:

- Start with choosing some $d \in S^{\mathcal{I}_L}$ and set $\pi(0,0) = d$.
- Complete the bottom horizontal. If $\pi(0,j)$ is defined and $\pi(0, j+1)$ undefined, then choose $d \in \Delta^{\mathcal{I}_L}$ such that $(\pi(0,j), d) \in t^{\mathcal{I}_L}$ and set $\pi(0, j+1) = d$.
- Complete the verticals. If $\pi(i,j)$ is defined and $\pi(i{+}1, j)$ undefined, then choose $d \in \Delta^{\mathcal{I}_L}$ such that $(d, \pi(i,j)) \in c^{\mathcal{I}_L}$ and set $\pi(i+1, j) = d$.
- Complete all horizontals. Now we have to argue that we can find horizontals such that the verticals and horizontals form a grid.

  For this purpose, we do an induction for all $i \geq 0$ proving that $(\pi(i,j), \pi(i, j+1)) \in t^{\mathcal{I}_L}$ and either $\pi(i,j) \in X_k^{\mathcal{I}_L}$ and $\pi(i, j+1) \in X_{k+1}^{\mathcal{I}_L}$ for any $k \in \{1, 3\}$ or $\pi(i,j) \in X_k^{\mathcal{I}_L}$ and $\pi(i, j+1) \in X_{k-1}^{\mathcal{I}_L}$ for any $k \in \{2, 4\}$.

  For $i = 0$ this follows from the bottom horizontal we completed and the CIs $S \sqsubseteq X_1$, $X_1 \sqsubseteq \forall c.X_3 \sqcap \forall t.X_2$, and $X_2 \sqsubseteq \forall c.X_4 \sqcap \forall t.X_1$.

  Now let us argue for $i > 0$. By the IH we have $(\pi(i-1, j), \pi(i-1, j+1)) \in t^{\mathcal{I}_L}$. We can complete the two vertical edges as defined above to obtain $(\pi(i-1, j), \pi(i,j)) \in c^{\mathcal{I}_L}$ and $(\pi(i-1, j+1), \pi(i, j+1)) \in c^{\mathcal{I}_L}$. Now it is straightforward to prove that by the IH and the CIs concerning the X_k that one of the CQs q_k with $k \in \{1, 2, 3, 4\}$ as defined in Figure 3 can be matched onto $\pi(i,j)\pi(i,j+1)\pi(i-1,j)\pi(i-1,j+1)$. The abstraction for q_i followed by the refinement for U_i then imply $(\pi(i,j), \pi(i, j+1)) \in t^{\mathcal{I}_L}$, as required.

We can now read off the computation of M on the empty tape from the grid in a straightforward way, using the concept names A_σ for the tape content and A_q for the state and head position. Since $A_{a_h}^{\mathcal{I}_L} = \emptyset$, the computation is non-terminating.

❏

H Proofs for Section 6.4

We complete the proof of Theorem 8 which we repeat here for the reader's convenience.

Theorem 8. *In the extension with quantified variables, satisfiability is undecidable in* $\mathcal{ALC}^{\mathsf{abs}}$[ca,cr] *and* $\mathcal{EL}^{\mathsf{abs}}$[ca, cr, rr].

Recall that we want to construct an ontology $\mathcal{O}$ and choose a concept name S and abstraction level L such that S is L-satisfiable w.r.t. $\mathcal{O}$ iff M does not halt on the empty tape. We generate an infinite t-path with outgoing infinite c-paths from every node:

$$
\begin{aligned}
S &\sqsubseteq_L \exists t.A_t \qquad & A_t &\sqsubseteq_L \exists t.A_t \\
A_t &\sqsubseteq_L \exists c.A_c \qquad & A_c &\sqsubseteq_L \exists c.A_c.
\end{aligned}
$$

We next form a grid by using quantified variables in a concept abstraction and refinement as mentioned in the main part of the paper:

$$
\begin{aligned}
&L_1 {:}\, U_1 \ \underline{\mathsf{abstracts}}\ L{:}q \\
&L{:}\, q \wedge t(x_3, x_4) \ \underline{\mathsf{refines}}\ L_1 {:}\, U_1 \ \text{ where} \\
&q = \exists x_1 \exists x_2\, c(x_1, x_3) \wedge t(x_1, x_2) \wedge c(x_2, x_4).
\end{aligned}
$$

We complete the construction of $\mathcal{O}$ by adding the CIs presented in Appendix F which simulate the Turing Machine behaviour.

The correctness is captured by the following lemma which can be proved analogously to Lemma 23. In fact, the proof is an even simpler version because we only have two abstraction levels instead of five.

Lemma 24. *S is L-satisfiable w.r.t. $\mathcal{O}$ iff M does not halt on the empty tape.*