Separating Counting from Non-Counting in Fragments of Two-Variable First-Order Logic (Extended Abstract)

Louwe Kuijer¹, Tony Tan¹, Frank Wolter¹ and Michael Zakharyaschev²

¹University of Liverpool, Ashton Street, Liverpool L69 3BX, UK
²Birkbeck, University of London, Malet Street, London WC1E 7HX, UK

Abstract
We consider the problem of deciding whether two disjoint classes of models defined in a fragment of first-order logic (FO) with counting can be separated in the same fragment but without counting. This problem turns out to be hard. We show that separation for the two-variable fragment FO² extended with counting quantifiers by means of plain FO² is undecidable, and the same is true of the pair ALCQI/ALCQI of description logics. On the other hand, we establish 2ExpTime-completeness of the separation problem for the pairs ALCQIu/ALCu and ALCIu/ALCIu.

Keywords
Separation, two-variable first-order logic, counting quantifiers, bisimulation.

1. Introduction
Our concern in this paper is the following separation problems for a pair of languages L and Lₘ:

L/Lₘ-separation: given two mutually exclusive L-formulas ϕ and ψ, decide whether there exists an Lₘ-formula χ—a separator for ϕ and ψ—such that ϕ |= χ and χ |= ¬ψ;

Craig L/Lₘ-separation: decide whether the given L-formulas ϕ and ψ have an Lₘ-separator χ that only contains common non-logical symbols (predicates and functions) of ϕ and ψ.

To illustrate, ϕ could be an ontology O and ψ a concept C that is not satisfiable with respect to O, both given in an expressive language L. Then a separator ontology O' in a weaker, easier to comprehend language Lₘ potentially explains unsatisfiability as it inherits that O |= O' and C is not satisfiable under O'. Also, in the context of concept learning, ϕ and ψ could represent positive and negative examples for a target concept C. Then any separator in an appropriately chosen language Lₘ could represent the concept one aims to learn [1].

Separation generalises definability (aka membership), which asks whether a given L-formula (say, a datalog query) is equivalent to some Lₘ-formula (say, a first-order query), and is regarded...
as one of the main approaches to understanding the expressive power of $L$ relative to $L^*$. For instance, studying separability of regular languages by smaller language classes (e.g., a star-free language) has brought major insights into the respective formal languages, with some fundamental open problems in the area cast as separation questions [2].

Craig $L/L^*$-separation generalises classical Craig interpolation in $L$ [3] because a Craig $L/L$-separator for $\varphi$ and $\psi$ is a Craig interpolant for $\varphi \rightarrow \neg \psi$ in $L$.

Our aim in this paper is to investigate the decidability and complexity of the separation problem for certain fragments $L$ of $C^2$—that is, the two-variable first-order logic $FO^2$ extended with counting quantifiers—and the same fragments $L^*$ but without counting.

**Example 1.** Consider the following $C^2$-formulas:

$$\varphi(x) = \exists x^1 y (R(x, y) \land A(y)) \land \exists x^1 y (R(x, y) \land \neg A(y)).$$

Then $\varphi \models \neg \psi$ and the $FO^2$-formula $\chi(x) = \forall y (R(x, y) \rightarrow A(y)) \land \forall y (R(x, y) \rightarrow \neg A(y))$ is a separator for $\varphi(x)$ and $\psi(x)$. For $\psi'(x) = \exists x^2 y R(x, y)$, we also have $\varphi \models \neg \psi'$, but $\varphi(x)$ and $\psi'(x)$ are not separable in $FO^2$. On the other hand, there is no Craig $FO^2$-separator for $\varphi(x)$ and $\psi(x)$ as it would have to be defined using $R$ only, and so separate $\varphi(x)$ and $\psi'(x)$ as well.

### 2. Logics

The logics we consider here can all be regarded as fragments of first-order logic, $FO$, and are defined as follows. Let $\sigma$ be a signature containing unary and binary relation symbols and possibly constants. Fix a set $var$ comprising two individual variables. Then

$FO^2(\sigma)$, the **two-variable fragment of $FO(\sigma)$**, is defined as the set of formulas that are built from atoms $A(x)$, $R(x, y)$, and $x = y$ with unary $A \in \sigma$, binary $R \in \sigma$, and $x, y \in var$, using the Boolean connectives $\land$ and $\neg$ and quantifier $\exists x$ with $x \in var$ (other Booleans and $\forall x$ are regarded as standard abbreviations);

$C^2(\sigma)$, the **two-variable fragment of $FO^2(\sigma)$ with counting**, extends $FO^2(\sigma)$ with the counting quantifiers $\exists \leq k x$, for $k \in \mathbb{N}$ and $x \in var$ (other counting quantifiers $\exists = k x$, $\exists \leq k x$, $\exists \geq k x$ can be introduced as abbreviations).

In this paper, we are only interested in formulas $\varphi(x)$ with one free variable $x \in var$. The **signature of $\varphi$** is the set $sig(\varphi)$ of relation and constant symbols occurring in $\varphi$.

$FO(\sigma)$ and its fragments are interpreted in $\sigma$-structures $\mathfrak{A} = (\text{dom}(\mathfrak{A}), (R^\mathfrak{A})_{R \in \sigma}, (c^\mathfrak{A})_{c \in \sigma})$ with a domain $\text{dom}(\mathfrak{A}) \neq \emptyset$, relations $R^\mathfrak{A}$ on $\text{dom}(\mathfrak{A})$ of the same arity as $R \in \sigma$, and elements $c^\mathfrak{A} \in \text{dom}(\mathfrak{A})$. A **pointed structure** is a pair $\mathfrak{A}, a$ with $a \in \text{dom}(\mathfrak{A})$.

We also consider a few fragments of $C^2$ that correspond to some standard description logics (DLs). In the context of DLs, unary relation symbols are called **concept names**, binary ones **role names**, and constants **individual names** [4]. A **role** is a role name $r$ or its inverse $r^-$. The **universal role** is denoted by $u$. A **nominal** takes the form $\{c\}$ with an individual name $c$.

An **$ALCOIQ^\omega(\sigma)$-concept** is defined by the grammar

$$C ::= \top \mid A \mid \{c\} \mid \neg C \mid C \cap C' \mid \geq k r.C \mid \exists u.C,$$
We consider several fragments of $\mathcal{ALCOIQ}^u$. The weakest, $\mathcal{ALC}$, is obtained by dropping the universal role (indicated by omitting $\mathcal{I}$ from the name), inverse roles (indicated by omitting $\mathcal{O}$), and only admitting qualified name restrictions of the form $\exists r.C = (\geq 1 r.C)$ (indicated by dropping $\mathcal{Q}$). The languages between $\mathcal{ALC}$ and $\mathcal{ALCOIQ}^u$ are now defined in the obvious way.

The separation and Craig separation problems are

Theorem 1. The separation and Craig separation problems are

- undecidable for the pairs $C^2/\text{FO}^2$ and $\mathcal{ALCOIQ}/\mathcal{ALC}$,
- $2\text{ExpTime}$-complete for the pairs $\mathcal{ALCIQ}^u/\mathcal{ALCI}$ and $\mathcal{ALCQ}^u/\mathcal{ALC}$.

The proof of Theorem 1 is based on the following straightforward model-theoretic characterisation of separation in terms of bisimulations; see [6, 7, 8] and references therein:

Lemma 2. Let $\varphi(x)$ and $\psi(x)$ be any $C^2(\sigma)$-formulas, $\sigma \subseteq \sigma$, and let $L^2$ be any of the languages introduced in Section 2. Then the following conditions are equivalent:

- $\varphi(x)$ and $\psi(x)$ do not have an $L^2(\varrho)$-separator;
- there are pointed $\sigma$-structures $A, a$ and $B, b$ such that
  \[ A \models \varphi(a), \quad B \models \psi(b), \quad A, a \sim_{L^2(\varrho)} B, b. \]

For Craig separation, we additionally require that $\varrho \subseteq \text{sig}(\varphi) \cap \text{sig}(\psi)$.

Here, $A, a \sim_{L^2(\varrho)} B, b$ means that there is an $L^2(\varrho)$-bisimulation $\beta$ between $A$ and $B$ such that $(a, b) \in \beta$, which implies that $A \models \phi(a)$ iff $B \models \phi(b)$, for all $L^2(\varrho)$-formulas $\phi(x)$ [6, 9, 5].

The proof of the characterisation in Lemma 2 is standard and similar to the characterisations of Craig interpolant nonexistence in [7, 8]. The undecidability proofs are by reduction of the halting problem for 2-register machines where the numbers in the registers are represented by the number of $L^2(\varrho)$-bisimilar nodes. The decidability proofs are based on novel adaptations of the mosaic technique for constructing $L^2(\varrho)$-bisimilar models [7, 8]. Preliminary detailed proofs are available at https://www.csc.liv.ac.uk/~frank/publ/publ.html.
4. Related Work

With the exception of separating modal $\mu$-calculus formulas by plain modal formulas \cite{10}, separability has so far been mainly investigated for formal languages \cite{11, 12, 13}. In contrast, definability has been investigated for many logics. For example, the problem of deciding whether a TBox given in a DL $L$ can be equivalently expressed in another DL $L'$ is considered in \cite{14}, the problem of deciding whether a guarded fragment (GF) formula or guarded negation fragment (GNF) formula is equivalent to an existential (or positive existential) GF formula or, respectively, GNF formula is considered in \cite{15, 16}, and there are many results on deciding whether fixpoints can be dropped from a second-order extension of a fragment of FO. Interestingly, the complexity of separation and definability of modal $\mu$-calculus formulas by plain modal formulas are both ExpTime-complete \cite{17, 10}. Variants of definability explored in description logic are approximation \cite{18} and conservative rewritability \cite{19}.

Craig separators are a generalisation of Craig interpolants where $L^s = L$. If the logic $L$ has the Craig interpolation property (CIP), then the Craig separator existence problem for $\varphi$ and $\psi$ reduces to checking whether $\varphi \models \psi$ and is thus not harder than entailment. Only recently the problem of deciding the existence of Craig interpolants has been considered for logics without the CIP \cite{20, 21}. In fact, the bisimulation-based method employed here makes heavy use of techniques introduced for checking Craig interpolant existence \cite{7, 8}.

5. Discussion

We have started investigating the separation problem for fragments of FO with counting by formulas in the same fragments but without counting. Many problems remain to be addressed; we mention a few of them below:

1. Our decidability proofs are non-constructive, and it would be of interest to develop algorithms that construct separators whenever they exist and determine bounds on their size.

2. With the exception of $\mathcal{ALCQ}$, the logics with counting we considered do not have the finite model property. It would be of interest to investigate whether our results also hold on finite structures. In that case, the bisimulation criterion does not hold as formulated (because its proof uses compactness) and one has to employ a different criterion that holds on finite structures (say, bounded bisimulations).

3. Our logics have the universal role. We conjecture that without the universal role $\mathcal{ALCQ}/\mathcal{ALC}$- and $\mathcal{ALCIQ}/\mathcal{ALCI}$-separation becomes $\text{coNExpTime}$-complete.

4. Is definability less complex than separation for the pairs of languages considered here. For example, is $C^2/FO^2$-definability decidable?
References


Appendix: Supplementary Material

A. Logics

In this section, we define the syntax and semantics of the fragments of first-order logic, FO, we deal with in what follows. Let \( \sigma \) be a signature containing unary and binary relation symbols and possibly constants. Fix a set \( \text{var} \) comprising two individual variables. Then

\[ \text{FO}^2(\sigma), \] the two-variable fragment of \( \text{FO}(\sigma) \), is defined as the set of formulas \( \varphi \) that are built from atoms \( A(x), R(x, y), \) and \( x = y \) with unary \( A \in \sigma \), binary \( R \in \sigma \), and \( x, y \in \text{var} \), using the Boolean connectives \( \land \) and \( \neg \) and quantifier \( \exists x \) with \( x \in \text{var} \) (other Boolean connectives and \( \forall x \) are regarded as standard abbreviations);

\[ \text{C}^2(\sigma), \] the two-variable fragment of \( \text{FO}^2(\sigma) \) with counting, extends \( \text{FO}^2(\sigma) \) with the counting quantifiers \( \exists^{<k}x \), for \( k \in \mathbb{N} \) and \( x \in \text{var} \) (other counting quantifiers \( \exists^{=k}x, \exists^{\leq k}x, \exists^{\geq k}x \) can be introduced as abbreviations).

In this paper, we are only interested in formulas \( \varphi(x) \) with one free variable \( x \in \text{var} \). The signature of \( \varphi \) is the set \( \text{sig}(\varphi) \) of relation and constant symbols occurring in \( \varphi \). We denote by \( \text{sub}(\varphi) \) the set of subformulas of \( \varphi \) together with their negations, setting \( |\varphi| = |\text{sub}(\varphi)| \).

\( \text{FO}(\sigma) \) and its fragments are interpreted in \( \sigma \)-structures \( \mathfrak{A} = (\text{dom}(\mathfrak{A}), (R^\mathfrak{A})_{R \in \sigma}, (c^\mathfrak{A})_{c \in \sigma}) \) with a domain \( \text{dom}(\mathfrak{A}) \neq \emptyset \), relations \( R^\mathfrak{A} \) on \( \text{dom}(\mathfrak{A}) \) of the same arity as \( R \in \sigma \), and elements \( c^\mathfrak{A} \in \text{dom}(\mathfrak{A}) \). A pointed structure is a pair \( \mathfrak{A}, a \) with \( a \in \text{dom}(\mathfrak{A}) \).
We also consider a few fragments of $C^2$ that correspond to some standard description (or modal) logics. In the context of DLs, unary relation symbols are called concept names, binary ones role names, and constants individual names [4]. A role is a role name $r$ or its inverse $r^-$, with $(r^-)^- = r$. The universal role is denoted by $\forall$. A nominal takes the form $\{c\}$ with an individual name $c$. An $ALCQI$ concept is defined by the grammar

$C ::= \top \mid A \mid \{c\} \mid \neg C \mid C \cap C' \mid \geq k r. C \mid \exists u. C,$

where $A \in \sigma$ is a concept name, $c \in \sigma$ an individual name, $r$ a role name in $\sigma$ or its inverse, and $k > 0$. The construct $(\geq k r. C)$ is known as the qualified number restriction. As usual, we set $\exists r. C = (\geq 1 r. C)$ and use $C \sqcup D$ as an abbreviation for $¬(¬C \cap D)$, $C \rightarrow D$ for $¬C \sqcup D$, $C \leftrightarrow D$ for $(C \rightarrow D) \cap (D \rightarrow C)$, and $\forall r. C$ for $¬\exists r. (¬C)$. Other counting concepts such as $\leq k r. C$ or $= k r. C$ can also be introduced as abbreviations in an obvious way.

We consider several fragments of $ALCQI$. The weakest, $ALC$, is obtained by dropping the universal role (indicated by omitting $\forall$ from the name), inverse roles (indicated by omitting $\exists$), nominals (indicated by omitting $\top$), and only admitting qualified number restrictions of the form $\exists r. C$ (indicated by dropping $Q$). The languages between $ALC$ and $ALCQ$ are now defined in the obvious way.

The semantics of DLs can be defined via the standard translation $^\sharp$ into $C^2$ with constants. For any $ALCQI$-concept $C$, we denote by $C^\sharp_2$ the $C^2$-formula with constants and free variable $x \in \text{var}$ defined inductively by taking

$\top^\sharp_x = (x = x),$  \quad $A^\sharp_x = A(x),$  \quad $\{c\}^\sharp_x = (x = c),$  \quad $(\neg C)^\sharp_x = \neg C^\sharp_x,$  \quad $(C \cap D)^\sharp_x = C^\sharp_x \land D^\sharp_x,$

$(\exists u. C)^\sharp_x = \top^\sharp_x \land \exists \bar{x} \ C^\sharp_x,$  \quad $(\geq k r. C)^\sharp_x = \exists \bar{x} \ (r(\bar{x}, \bar{y}) \land C^\sharp_x),$  \quad $(\exists r. C)^\sharp_x = \top^\sharp_x \land \exists \bar{x} \ C^\sharp_x,$

where $\bar{x} = y, \bar{y} = x$ and $\{x, y\} = \text{var}$. Then the extension $C^\sharp_\mathfrak{A}$ of a concept $C$ in $\mathfrak{A}$ is defined as

$C^\sharp_\mathfrak{A} = \{a \in \text{dom}(\mathfrak{A}) \mid \mathfrak{A} \models a \in C^\sharp_x(\{a\})\}.$

In this paper, DL concepts $C$ are always regarded as FO-formulas $C^\sharp_x$ with one free variable $x$, though we often use the more succinct DL notation (denoting roles by small letters like $r$ and $s$ rather than $R$ and $S$ as in FO-formulas). A formula $\varphi(x)$ is called satisfiable if there is a pointed structure $\mathfrak{A}, a$ such that structure $\mathfrak{A} \models \varphi(a)$. Given $\sigma$-formulas $\varphi(x)$ and $\psi(x)$, we write $\mathfrak{A} \models \varphi(x)$ if, for any pointed $\sigma$-structure $\mathfrak{A}, a$,

$\mathfrak{A} \models \varphi(a)$  \quad \text{implies}  \quad \mathfrak{A} \models \psi(a).$

Finally, we remind the reader of the complexity of the satisfiability problem for the logics in question [4, 5]:

- FO$^2$, $C^2$, and $ALCQI$ are all NExpTime-complete;
- $ALC$, $ALCQ$, $ALCIQ$, and $ALCIQI$ are ExpTime-complete.

In the next section, we define the separation problems for the logics defined above and give a model-theoretic criterion of separability in terms of bisimulations.
B. Separation and Bisimulation

Let \( \varphi(x) \) and \( \psi(x) \) be FO-formulas. An FO-formula \( \chi(x) \) is called a separator for \( \varphi(x) \) and \( \psi(x) \) if \( \varphi(x) \models \chi(x) \) and \( \chi(x) \models \neg \psi(x) \). If, in addition, \( \text{sig}(\chi) \subseteq \text{sig}(\varphi) \cap \text{sig}(\psi) \), we say that \( \chi(x) \) is a Craig separator for \( \varphi(x) \) and \( \psi(x) \).

Given two languages \( L \) and \( L' \), the separation problem for \( L \) by \( L' \)—or \( L/L' \)-separation, for short—is to decide whether any two given \( L \)-formulas have an \( L' \)-separator. If we are only interested in Craig separators, we refer to the problem as Craig \( L/L' \)-separation.

Lemma 3. Let \( L/L' \) be any of the pairs \( C^2/\text{FO}^2 \), \( \text{ALCQ}/\text{ALCOI} \), \( \text{ALCQ}^n/\text{ALCT}^n \), \( \text{ALCQ}^n/\text{ALC}^n \). Then \( L/L' \)-separation is polynomial-time reducible to Craig \( L/L' \)-separation.

Proof. Given \( L \)-formulas \( \varphi(x) \) and \( \psi(x) \), consider the formulas

\[
\varphi'(x) = \varphi(x) \land \bigwedge_{A \in \text{sig}(\psi) \setminus \text{sig}(\varphi)} (A(x) \rightarrow A(x)) \land \bigwedge_{R \in \text{sig}(\psi) \setminus \text{sig}(\varphi)} (R(x, x) \rightarrow R(x, x)) \land \bigwedge_{c \in \text{sig}(\psi) \setminus \text{sig}(\varphi)} (c = c),
\]

\[
\psi'(x) = \psi(x) \land \bigwedge_{A \in \text{sig}(\varphi) \setminus \text{sig}(\psi)} (A(x) \rightarrow A(x)) \land \bigwedge_{R \in \text{sig}(\varphi) \setminus \text{sig}(\psi)} (R(x, x) \rightarrow R(x, x)) \land \bigwedge_{c \in \text{sig}(\varphi) \setminus \text{sig}(\psi)} (c = c).
\]

It is readily seen that \( \varphi(x) \) and \( \psi(x) \) have an \( L' \)-separator iff \( \varphi'(x) \) and \( \psi'(x) \) have a Craig \( L' \)-separator.

Our main tool for determining the decidability and complexity of \( L/L' \)-separation is based on the notion of bisimulation.

Let \( L \) be any of the fragments of FO defined above and let \( \varrho \subseteq \sigma \). Given pointed \( \sigma \)-structures \( \mathfrak{A} \), \( a \) and \( \mathfrak{B} \), \( b \), we write \( \mathfrak{A}, a \equiv_{L/\varrho} \mathfrak{B}, b \) and say that \( \mathfrak{A}, a \) and \( \mathfrak{B}, b \) are \( L(\varrho) \)-equivalent if \( \mathfrak{A} \models \varphi(a) \iff \mathfrak{B} \models \varphi(b) \), for all \( L(\varrho) \)-formulas \( \varphi(x) \).

A binary relation \( \beta \subseteq \text{dom}(\mathfrak{A}) \times \text{dom}(\mathfrak{B}) \) is called an FO\(^2(\varrho) \)-bisimulation between \( \mathfrak{A} \) and \( \mathfrak{B} \) if \( \beta \) is global in the sense that \( \text{dom}(\mathfrak{A}) \subseteq \{ a \mid (a, b) \in \beta \} \) and \( \text{dom}(\mathfrak{B}) \subseteq \{ b \mid (a, b) \in \beta \} \) and, for every \( (a, b) \in \beta \), the following conditions are satisfied:

- for every \( a' \in \text{dom}(\mathfrak{A}) \), there is a \( b' \in \text{dom}(\mathfrak{B}) \) such that \( (a', b') \in \beta \) and \( (a, a') \mapsto (b, b') \) is a partial \( \varrho \)-isomorphism between \( \mathfrak{A} \) and \( \mathfrak{B} \);

- for every \( b' \in \text{dom}(\mathfrak{B}) \), there is a \( a' \in \text{dom}(\mathfrak{A}) \) such that \( (a', b') \in \beta \) and \( (a, a') \mapsto (b, b') \) is a partial \( \varrho \)-isomorphism between \( \mathfrak{A} \) and \( \mathfrak{B} \).

We write \( \mathfrak{A}, a \sim_{\text{FO}(\varrho)} \mathfrak{B}, b \) if \( a \mapsto b \) is a partial \( \varrho \)-isomorphism between \( \mathfrak{A} \) and \( \mathfrak{B} \) and there is an \( \text{FO}(\varrho) \)-bisimulation \( \beta \) between \( \mathfrak{A} \) and \( \mathfrak{B} \) such that \( (a, b) \in \beta \).

A non-empty binary relation \( \beta \subseteq \text{dom}(\mathfrak{A}) \times \text{dom}(\mathfrak{B}) \) is called an ALC\(^\varrho \)-bisimulation between \( \mathfrak{A} \) and \( \mathfrak{B} \) if the following conditions are satisfied:

1. if \( (a, b) \in \beta \), then \( a \in A^\mathfrak{A} \) iff \( b \in A^\mathfrak{B} \) for all \( A \in \varrho \);

2. if \( (a, b) \in \beta \) and \( (a, a') \in R^\mathfrak{A} \), for \( R \in \varrho \), then there exists \( b' \in \text{dom}(\mathfrak{B}) \) such that \( (a', b') \in \beta \) and \( (b, b') \in R^\mathfrak{B} \).
3. if \((a, b) \in \beta\) and \((b, b') \in R^B\), for \(R \in \wp\), then there exists \(a' \in \text{dom}(A)\) such that \((a', b') \in \beta\) and \((a, a') \in R^A\).

We write \(A, a \sim_{\mathcal{ALC}(\wp)} B, b\) if there is an \(\mathcal{ALC}(\wp)\)-bisimulation \(\beta\) between \(A\) and \(B\) such that \((a, b) \in \beta\). We say that an \(\mathcal{ALC}(\wp)\)-bisimulation \(\beta\) between \(A\) and \(B\) is

- an \(\mathcal{ALC}^n(\wp)\)-bisimulation if it is global;
- an \(\mathcal{ALC}I(\wp)\)-bisimulation if conditions 2 and 3 also hold for \(r = s^-\) with \(s \in \wp\).

Further, we call \(\beta\) an \(\mathcal{ALC}^I(\wp)\)-bisimulation if it is both an \(\mathcal{ALC}^n(\wp)\)- and an \(\mathcal{ALC}I(\wp)\)-bisimulation; \(\beta\) is an \(\mathcal{ALC}^I(\wp)\)-bisimulation if it is both an \(\mathcal{ALC}^n(\wp)\)-bisimulation and \((c^A, c^B) \in \beta\) for all \(c \in \wp\). Finally, \(\beta\) is an \(\mathcal{ALC}I(\wp)\)-bisimulation if it is both an \(\mathcal{ALC}^I(\wp)\)- and an \(\mathcal{ALC}O(\wp)\)-bisimulation. The following characterisation is well-known; see, e.g., [?9, 5]:

**Lemma 4.** Let \(L\) be any of the languages introduced in Section 2. For any pointed \(\sigma\)-structures \(A, a\) and \(B, b\),

\[
A, a \sim_{L(\wp)} B, b \quad \text{implies} \quad A, a \equiv_{L(\wp)} B, b
\]

and, conversely, if structures \(A\) and \(B\) are \(\omega\)-saturated, then

\[
A, a \equiv_{L(\wp)} B, b \quad \text{implies} \quad A, a \sim_{L(\wp)} B, b.
\]

**Lemma 5.** Let \(\varphi(x)\) and \(\psi(x)\) be any \(C^2(\sigma)\)-formulas, \(\wp \subseteq \sigma\), and let \(L^\wp\) be any of the languages introduced in Section 2. Then the following conditions are equivalent:

- \(\varphi(x)\) and \(\psi(x)\) do not have an \(L^\wp(\wp)\)-separator;
- there are pointed \(\sigma\)-structures \(A, a\) and \(B, b\) such that

\[
A \models \varphi(a), \quad B \models \psi(b), \quad A, a \sim_{L^\wp(\wp)} B, b.
\]

**Proof.** The proof is similar to the characterisations of Craig interpolant existence in [7, 8].

### C. Undecidable Separation

We show that \(\mathcal{ALC}I\mathcal{Q}/\mathcal{ALC}I\)-(Craig) separation is undecidable with and without the universal role and that \(C^2/F\mathcal{O}^2\)-(Craig) separation undecidable.

To this end we reduce the halting problem for two-register machines. A (deterministic) two-register machine (2RM) is a pair \(M = (Q, P)\) with \(Q = q_0, \ldots, q_\ell\) a set of states and \(P = I_0, \ldots, I_{\ell-1}\) a sequence of instructions. By definition, \(q_0\) is the initial state, and \(q_\ell\) the halting state. For all \(i < \ell\),

- either \(I_i = +p, q_j\) is an incrementation instruction with \(p \in \{0, 1\}\) a register and \(q_j\) the subsequent state;
or \( I_i = -(p, q_j, q_k) \) is a decrementation instruction with \( p \in \{0, 1\} \) a register, \( q_j \) the subsequent state if register \( p \) contains 0, and \( q_k \) the subsequent state otherwise.

A configuration of \( M \) is a triple \((q, m, n)\), with \( q \) the current state and \( m, n \in \omega \) the register contents. We write \((q_i, n_1, n_2) \Rightarrow_M (q_j, m_1, m_2)\) if one of the following holds:

- \( I_i = +(p, q_j), m_p = n_p + 1, \) and \( m_1 = m_1 - p; \)
- \( I_i = -(p, q_j, q_k), n_p = m_p = 0, \) and \( m_1 = m_1 - p; \)
- \( I_i = -(p, q_k, q_j), n_p > 0, m_p = n_p - 1, \) and \( m_1 = m_1 - p. \)

The computation of \( M \) on input \((n, m) \in \omega^2\) is the unique longest configuration sequence \((p_0, n_0, m_0) \Rightarrow_M (p_1, n_1, m_1) \Rightarrow_M \cdots\) such that \( p_0 = q_0, n_0 = n, \) and \( m_0 = m. \) The halting problem for 2RMs is to decide, given a 2RM \( M \), whether its computation on input \((0, 0)\) is finite (which implies that its last state is \( q_\ell). \)

### C.1. \( \text{ALCOTQ}/\text{ALCOT}-\text{separation} \) is undecidable.

We show that \( \text{ALCOTQ}^u/\text{ALCOT}^u \)-separation is undecidable and then use spy-points to eliminate the universal role and obtain that \( \text{ALCOTQ}/\text{ALCOT}-\text{separation} \) is undecidable. We also obtain undecidability of Craig separation since the concepts used in the proof use the same signature.

We reduce the non-halting problem for 2RM to the non-existence of a separator. To this purpose, let a two-register machine \( M = (Q, P) \) be given. We assume without loss of generality that the initial state \( q_0 \) only occurs initially and that \( I_0 \) is a decrementation instruction.

We introduce two concepts, \( C_M \) and \( D_M \), both conjunctions, with the conjuncts best presented in a few groups. For \( C_M \), the first group of conjuncts are as follows.

1. \( \{a\} \cap Q_0 \)
2. \( \forall u. (F \cup \bigcup_{1 \leq i \leq \ell} Q_i) \)
3. \( \forall u. (F \rightarrow \bigcup_{1 \leq i \leq \ell} \neg Q_i) \)
4. \( \forall u. \left( \prod_{1 \leq i \leq \ell} (Q_i \rightarrow \bigcap_{i < j \leq \ell} \neg Q_j) \right) \)

These conjuncts describe the various concept names that we need: every element satisfies exactly one of \( F, Q_0, \cdots, Q_\ell \), and the unique node satisfying \( \{a\} \) satisfies \( Q_0 \). We will use models of \( C_M \) to represent runs of the machine \( M \); elements satisfying some \( Q_i \) will represent snapshots of the computations where the machine is in state \( q_i \). The elements satisfying \( F \) are auxiliary, their purpose will become more clear once we discuss \( D_M \).

The next group of conjuncts of \( C_M \) is as follows.

5. \( \{a\} \rightarrow \forall r^{-1}. \bot \)
6. \( \forall u. (\neg F \rightarrow (= 1 \: r \: \top)) \)
7. \( \forall u.((\neg F \land \neg \{a\}) \rightarrow (1 \cdot r^{-1}, \top)) \)

8. \( \forall u.((\neg F \rightarrow (\forall r. \neg F \land \forall r^{-1}. \neg F)) \)

9. \( \forall u.((\neg F \rightarrow \exists s. F) \)

These conjuncts describe the relations \( r \) and \( s \): every non-auxiliary element (i.e., every \( \neg F \) element) has exactly one \( r \)-successor, and every such element has exactly one \( r \)-predecessor, except for the node satisfying \( \{a\} \) which has no \( r \)-predecessors. Furthermore, if an element is non-auxiliary then so are its \( r \)-successor and \( r \)-predecessor. Finally, every non-auxiliary element has an \( s \)-successor that is auxiliary.

It follows from these conditions that every pointed model \( A, x_0 \) with \( x_0 \in C_M^A \) contains an infinite \( r \)-chain starting at \( x_0 \) and each element of the chain satisfies exactly one \( Q_i \). We will interpret these elements as representing a run of \( M \).

The next group of conjuncts depends on the instructions of the machine \( M \). If \( I_i = +(p, q_j) \) we add the conjunct

10. \( \forall u. (Q_i \rightarrow \forall r. Q_j) \)

and if \( I_i = -(p, q_j, q_k) \) we add the conjuncts

11. \( \forall u.((Q_i \land E_p) \rightarrow \forall r. Q_j) \)

12. \( \forall u.((Q_i \land \neg E_p) \rightarrow \forall r. Q_k) \)

These conjuncts express that the instructions of \( M \) are followed, where \( E_p \) (with \( p \in \{0, 1\} \)) indicates that register \( p \) is empty. Note that we do not, at this stage, keep track of the value in each register. But if \( E_p \) holds if and only if register \( p \) is empty, then these conjuncts ensure that every pointed model \( A, x_0 \) of \( C_M \) encodes the run of \( M \).

Finally, we add one more, very simple, conjunct,

13. \( \forall u. \neg Q_\ell \)

which expresses that the halting state \( Q_\ell \) is not reached. So if \( C_M \) encodes a run of \( M \), it must be a non-halting run.

The concept \( D_M \) can similarly be divided into groups of conjuncts. Before discussing these conjuncts, it is convenient to first define a few abbreviations:

- \( U := (1 \cdot s. F) \sqcup (3 \cdot s. F) \)
- \( R_0 := (1 \cdot s. F) \sqcup (2 \cdot s. F) \)
- \( R_1 := \neg R_0. \)

Note that, because \( U \), \( R_0 \) and \( R_1 \) are mere abbreviations, they are not considered to be part of the signature. Note also that \( U \sqcap R_0 \) and \( U \sqcap R_1 \) are both consistent, while \( R_0 \sqcap R_1 \) obviously is not. Now, let us introduce the conjuncts of \( D_M \).

The first group is the same as the first group of conjuncts of \( C_M \):

These conjuncts express that the instructions of \( M \) are followed, where \( E_p \) (with \( p \in \{0, 1\} \)) indicates that register \( p \) is empty. Note that we do not, at this stage, keep track of the value in each register. But if \( E_p \) holds if and only if register \( p \) is empty, then these conjuncts ensure that every pointed model \( A, x_0 \) of \( C_M \) encodes the run of \( M \).
1. \( \{a\} \cap Q_0 \)

2. \( \forall u. (F \cup \bigcup_{1 \leq i \leq \ell} Q_i) \)

3. \( \forall u. (F \rightarrow \bigcup_{1 \leq i \leq \ell} \neg Q_i) \)

4. \( \forall u. \bigcap_{1 \leq i \leq \ell} (Q_i \rightarrow \bigcap_{i < j \leq \ell} \neg Q_j) \)

So, as with \( C_M \), in every pointed model \( B \), \( y_0 \) of \( D_M \), all elements satisfy exactly one of \( F, Q_0, \ldots, Q_\ell \). The remaining conjuncts are quite different, however.

5. \( (= 2 r, \top) \)

6. \( \exists r. (U \cap R_0) \cap \exists r. (U \cap R_1) \)

7. \( E_0 \cap E_1 \)

8. \( \forall u. (R_p \rightarrow (\forall r. R_p \cap \forall r^{-1}. (R_p \cup \{a\}))) \) for \( p \in \{0, 1\} \)

9. \( \forall u. ((U \cap \neg \{a\}) \rightarrow ((= 1 r, U) \cap (= 1 r^{-1}, U))) \)

Conjunct 1 implies that in any pointed model \( B \), \( y_0 \) of \( D_M \) we have that \( y_0 \) satisfies \( \{a\} \), so conjuncts 5 and 6 imply that \( y_0 \) has exactly two \( r \)-successors, with one satisfying \( U \cap R_0 \) and the other \( U \cap R_1 \). We will use these branches to represent registers 0 and 1, respectively. Conjunct 7, meanwhile, says that \( y_0 \) satisfies both \( E_0 \) and \( E_1 \). Conjuncts 8 and 9 handle propagation: 8 says that \( R_p \) propagates forward and back along \( r \) (except to \( \{a\} \)), while 9 says that \( U \) propagates to exactly one \( r \)-successor and \( r \)-predecessor (except from \( \{a\} \)).

The next group of conjuncts depends on the instructions of \( M \). If \( I_i = +(p, q_j) \),

10. \( \forall u. ((Q_i \cap R_p) \rightarrow ((= 2 r, \top) \cap \forall r.(= 1 r^{-1}, \top))) \)

11. \( \forall u. ((Q_i \cap \neg R_p) \rightarrow ((= 1 r, \top) \cap \forall r.(= 1 r^{-1}, \top))) \)

If \( i \neq 0 \) and \( I_i = -(p, q_j, q_k) \),

12. \( \forall u. ((Q_i \cap R_p \cap E_p) \rightarrow ((= 1 r, \top) \cap \forall r.(= 1 r^{-1}, \top))) \)

13. \( \forall u. ((Q_i \cap R_p \cap \neg E_p) \rightarrow ((= 1 r, \top) \cap \forall r.(= 2 r^{-1}, \top))) \)

14. \( \forall u. ((Q_i \cap \neg R_p) \rightarrow ((= 1 r, \top) \cap \forall r.(= 1 r^{-1}, \top))) \)

Conjunct 10 expresses that if \( Q_i \) is an incrementation instruction, then an element satisfying \( Q_i \cap R_p \) will have two \( r \)-successors that each have one \( r \)-predecessor. We will use \( r \) to represent the computation, so this means that the number of elements will double in a state with an incrementation instruction. With respect to conjuncts 12 and 13, recall that we use \( E_p \) as a marker for register \( p \) being empty. So these conjuncts say that if a decrementation instruction is given with an empty register (conjunct 12), then for every \( R_p \) element there is one \( r \)-successor that has one \( r \)-predecessor so the number of elements remains the same, and if a decrementation
instruction is given with a non-empty register (conjunct 13) then for every \( R_p \) element there is a single \( r \)-successor that has two \( r \)-predecessors, so the number of elements is halved.

Conjuncts 11 and 14 state that elements in the other register (i.e., those satisfying \( \neg R_p \)) have exactly one successor that has exactly one predecessor, so the number of elements remains the same.

Note that we treat \( Q_0 \) separately. This is because we first need to create the two registers before we can increase or decrease the number of elements in each register. This is done by conjuncts 5 and 6. After one step of the computation, our two registers are then initialized with one element each, which we interpret as the value 0 for that register. This is why we assume that \( q_0 \) is decrementing (so the registers are empty after one step of the computation) and only occurs initially (so we do not perform the initialization again at later times).

Finally, we add one more conjunct

\[
15. \forall u((E_p \cap R_p) \rightarrow U)
\]

which states that the combination of \( E_p \) and \( R_p \) is possible only when \( U \) also holds.

The concepts \( C_M \) and \( D_M \) have the same signature. Hence any separator is a Craig separator. We show that a separator exists if and only if \( M \) is non-halting, from which it follows that separation is undecidable. More precisely, the reduction shows that non-separation is co-RE hard, and therefore the separation problem is RE hard.

Before proving our reduction, let us first introduce the \textit{intended pointed models} for \( C_M \) and \( D_M \), where \( M \) is non-halting. A pointed model \( \mathfrak{A}, x_0 \) of \( C_M \) will be used to represent a clock, and the state the machine is in at each time step. Recall that \( C_M \) guarantees that \( \mathfrak{A}, x_0 \) follows the instructions of \( M \), as long as \( E_p \) holds when register \( p \) is empty. In our intended models, we will make sure that (i) \( E_p \) only holds if register \( p \) is empty and (ii) if register \( p \) is empty and the current instruction is to decrement register \( p \), then \( E_p \) holds (to indicate that decrementing is impossible). If register \( p \) is empty and the current instruction is anything other than to decrement register \( p \), we do not care whether \( E_p \) holds.

A pointed model \( \mathfrak{B}, y_0 \) of \( D_M \) is used to represent the content of the registers. Every point \( t \) of the computation is represented by some element \( e_t \) of \( \mathfrak{A} \), and this \( e_t \) is \( \text{ALCOT}^u \)-bisimilar to a set of elements in \( \mathfrak{B} \). These elements of \( \mathfrak{B} \) can be divided into those that satisfy \( R_0 \) and those that satisfy \( R_1 \). If \( e_t \) is \( \text{ALCOT}^u \)-bisimilar to \( m \) elements that satisfy \( R_0 \) and \( n \) elements that satisfy \( R_1 \), then we say that registers 0 and 1 contains \( \log_2(m) \) and \( \log_2(n) \), respectively, at time \( t \).

An example of (part of) these intended models is shown in Figure 1. In this example, \( I_0 = -(0, q_1, q_1), I_1 = +(0, q_4) \text{ and } I_4 = -(0, q_2, q_4) \). So the machine starts in state \( q_0 \) by decrementing the (already empty) register 0, and goes to state \( q_1 \). In this state, it increments register 0, and continues to state \( q_4 \). In \( q_4 \), it first decrements register 0 while staying in \( q_4 \). Then it tries to decrement register 0 again, but now that register is empty so the next state is \( q_2 \) instead of \( q_4 \).

It is straightforward (if somewhat labour intensive) to verify that the intended pointed models satisfy \( C_M \) and \( D_M \), and that they are \( \text{ALCOT}^u \)-bisimilar.

**Proposition 6.** If \( M \) is non-halting, then there are pointed models \( \mathfrak{A}, x_0 \) of \( C_M \) and \( \mathfrak{B}, y_0 \) of \( D_M \) such that \( \mathfrak{A}, x_0 \sim_{\text{ALCOT}^u} \mathfrak{B}, y_0 \).
Figure 1: Example pointed models of $C_M$ (above the dashed line) and $D_M$ (below the dashed line). Elements that are drawn below each other are $\mathcal{ALCOI}^u$-bisimilar. Recall that $R_0$, $R_1$ and $U$ are abbreviations that depend on the number of $s$-successors satisfying $F$. All elements in the upper branch of the pointed model of $D_M$ have 1 or 2 $s$-successors, so they satisfy $R_0$. In the lower branch they have more than 2 $s$-successors, so they satisfy $R_1$. All but one elements have 1 or 3 $s$-successors, and therefore satisfy $U$. The one exception is where the upper branch splits; there can be at most one $R_0$-element that satisfies $U$, so the lower of the two $R_0$ elements has 2 $s$-successors, which is a $\neg U$ amount.

Next, we need to show the converse.

**Proposition 7.** If there are pointed models $\mathfrak{A}, x_0$ of $C_M$ and $\mathfrak{B}, y_0$ of $D_M$ such that $\mathfrak{A}, x_0 \sim_{\mathcal{ALCOI}^u} \mathfrak{B}, y_0$, then $M$ is non-halting.

**Proof.** We will show that $\mathfrak{A}$ and $\mathfrak{B}$ contain intended models as sub-models. $\mathfrak{A}$ contains a sequence of elements $x_0, x_1, \ldots$ where $x_0 = a^\mathfrak{A}$, $x_{t+1}$ is the unique $r$-successor of $x_t$ and $x_t$ is the unique $r$-predecessor of $x_{t+1}$. Furthermore, on each $x_t$ some $Q_i$ (with $0 \leq i \leq \ell$) holds.

In $\mathfrak{B}$ we also have $y_0 = a^\mathfrak{B}$. Let $Y_0 = \{y_0\}$, and for every $t$ let $Y_{t+1}$ be the $r$-successors of the elements of $Y_t$. Let $\beta \subseteq \text{dom}(\mathfrak{A}) \times \text{dom}(\mathfrak{B})$ be the $\mathcal{ALCOI}^u$-bisimulation between $\mathfrak{A}, x_0$ and $\mathfrak{B}, y_0$.

**Claim 1:** For every $t \in \mathbb{N}$, we have (i) if $(x_t, y) \in \beta$ then $y \in Y_t$ and (ii) if $y \in Y_t$, then $(x_t, y) \in \beta$. 

Proof of claim 1: By induction on \( t \). As base case, note that \( x_0 \) and \( y_0 \) are the unique elements satisfying \( \{ a \} \) in their respective models, so \( x_0 \) and \( y_0 \) can only be \( \mathit{ALCOT}^n \)-bisimilar to each other. Since \( \beta \) is, by assumption, a \( \mathit{ALCOT}^n \)-bisimulation, we must have \( (x_0, y_0) \in \beta \). Assume then as induction hypothesis that \( t > 0 \) and that the claim holds for all \( t' < t \).

Take any \( y \) such that \( (x_t, y) \in \beta \). Because \( \beta \) is an \( \mathit{ALCOT}^n \)-bisimulation and \( x_t \) has \( x_{t-1} \) as an \( r \)-predecessor, \( y \) must have some \( y' \) as \( r \)-predecessor such that \( (t_{x-1}, y') \in \beta \). By the induction hypothesis, \( y' \in Y_{t-1} \). This implies that \( y \) is an \( r \)-successor of an element from \( Y_{t-1} \) so, by definition, \( y \in Y_t \).

Now, take any \( y \in Y_t \). By the definition of \( Y_t \), there is some \( y' \in Y_{t-1} \) that is an \( r \)-predecessor of \( y \). By the induction hypothesis, \( (x_{t-1}, y') \in \beta \). Because \( \beta \) is a bisimulation and \( y \) is an \( r \)-successor of \( y' \), there must be some \( x' \) that is an \( r \)-successor of \( x_{t-1} \) such that \( (x', y') \in \beta \). The only \( r \)-successor of \( x_{t-1} \) is \( x_t \), so \( y' = x_t \), and hence \( (x_t, y) \in \beta \). This completes the induction step, and thereby the proof of Claim 1.

Claim 2: For every \( y \in Y_t \), if \( y' \) is an \( r \)-predecessor of \( y \), then \( y' \in Y_{t-1} \).

Proof of claim 2: By claim 1, \( (x_t, y) \in \beta \). Since \( y' \) is an \( r \)-predecessor of \( y \), we must have \( (y', x') \in \beta \) for some \( r \)-predecessor \( x' \) of \( x_t \). The only \( r \)-predecessor of \( x_t \) is \( x_{t-1} \), so we have \( (x_{t-1}, y') \in \beta \). By claim 1, this implies that \( y' \in Y_{t-1} \). This completes the proof of claim 2.

Claim 3: For every \( t \), there is exactly one \( i_t \) such that \( Q_{i_t} \) holds on any of \( x_t \) and \( Y_t \). This \( Q_{i_t} \) holds on all of \( x_t \) and \( Y_t \). Furthermore, \( E_p \) either holds on all of \( x_t \) and \( Y_t \) or on none.

Proof of claim 3: Because \( \mathfrak{M} \) is a model of \( C_M \), every \( x_t \) satisfies exactly one \( Q_{i_t} \). The rest of the claim follows immediately from \( \beta \) being a bisimulation.

Before continuing with further claims, let use define \( Y_t^P \), for \( t > 0 \) and \( p \in \{ 0, 1 \} \), by \( Y_t^P = \{ y \in Y_t \mid \mathfrak{M}, y \models R_p \} \). If \( p \in \{ 0, 1 \} \), we write \( \overline{p} \) for the other element of \( \{ 0, 1 \} \), i.e., \( \overline{0} = 1 \) and \( \overline{1} = 0 \).

Claim 4: For \( p \in \{ 0, 1 \} \) and \( t > 0 \), the \( r \)-successors of \( Y_t^P \) are in \( Y_{t+1}^P \) and the \( r \)-predecessors of \( Y_{t-1}^P \) are in \( Y_t^P \).

Proof of claim 4: Conjuncts 8 says that \( R_p \) propagates forward and back through \( r \).

Claim 5: For every \( t > 0 \),

- if \( I_{i_t} = +(p, q_j) \) then \( i_{t+1} = j, |Y_{t+1}^P| = 2 \times |Y_t^P| \),
- if \( I_{i_t} = -(p, q_j, q_k) \) and \( E_p \) holds on \( x_t \), then \( i_{t+1} = j, |Y_{t+1}^P| = |Y_t^P| \),
- if \( I_{i_t} = -(p, q_j, q_k) \) and \( E_p \) does not hold on \( x_t \), then \( i_{t+1} = k, |Y_{t+1}^P| = \frac{1}{2} \times |Y_t^P| \),

and, in each case, \( |Y_{t+1}^P| = |Y_t^P| \).

Proof of claim 5: Conjuncts 10, 11 and 12 of \( C_M \) guarantee that \( i_{t+1} \) has the appropriate value.

Consider, then the sets \( Y_t^P \) and \( Y_{t+1}^P \). By claim 4, all the \( r \)-successor of \( Y_t^P \) are in \( Y_{t+1}^P \), and all the \( r \)-predecessors of \( Y_{t-1}^P \) are in \( Y_t^P \). Furthermore, conjunct 10 of \( D_M \) implies that, if \( I_{i+t} = +(p, q_j) \) then every \( y \in Y_t^P \) has exactly two \( r \)-successors that each have exactly one \( r \)-predecessor. It follows that \( |Y_{t+1}^P| = 2 \times |Y_t^P| \). Similarly, if \( I_{i_t} = -(p, q_j, q_k) \) and \( E_p \) holds, then conjunct 12 says that every \( y \in Y_t^P \) has exactly one \( r \)-successor that has exactly one \( r \)-predecessor, so \( |Y_{t+1}^P| = |Y_t^P| \). If \( I_{i+t} = -(p, q_j, q_k) \) and \( E_p \) does not hold, then conjunct 13 makes sure that every \( y \in Y_t^P \) has exactly one \( r \)-successor that has exactly two \( r \)-predecessors, so \( |Y_{t+1}^P| = \frac{1}{2} \times |Y_t^P| \).
Finally, the conjuncts 11 and 14 guarantee that every \( y \in Y_t^P \) has exactly one \( r \)-successor that has exactly one \( r \)-predecessor, so \( |Y_{t+1}^P| = |Y_t^P| \).

Claim 6: \( |Y_{t+1}^P| = |Y_t^P| = 1 \).

Proof of claim 6: By conjuncts 5 and 6 of \( D_M \), \( \{a\} \) has exactly two successors, one of which satisfies \( U \sqcap R_0 \) while the other satisfies \( U \sqcup R_1 \).

Claim 7: For every \( t > 0 \) and \( p \in \{0, 1\} \), there is exactly one \( y \in Y_t^P \) that satisfies \( U \).

Proof of claim 7: For \( t = 1 \), the claim holds because, by conjunct 6 of \( D_M \), there is at least one \( y \in Y_1^P \) that satisfies \( U \). As \( Y_1^P \) is a singleton (by claim 6), there must be exactly one such \( y \). Conject 9 implies that every \( U \) element (other than \( \{a\} \)) has exactly one \( r \)-predecessor and one \( r \)-successor that satisfies \( U \), so if \( Y_t^P \) has exactly one element satisfying \( U \) then so does \( Y_{t+1}^P \).

Claim 8: For every \( t > 0 \) and \( p \in \{0, 1\} \),

- if \( E_p \) holds on \( x_t \) and \( Y_t \), then \( Y_t^P \) is a singleton and
- if \( I_t = -(p, q_j, q_k) \) and \( Y_t^P \) is a singleton, then \( E_p \) holds on \( x_t \) and \( Y_t \).

Proof of claim 8: Suppose that \( E_p \) holds on \( x_t \) and \( Y_t \). Then \( E_p \sqcap R_p \) is true for every element of \( Y_t^P \). By conjunct 15, this implies that \( U \) is true for all of \( Y_t^P \). Claim 7 says that only one element of \( Y_t^P \) can satisfy \( U \), so \( Y_t^P \) must be a singleton.

Suppose then that \( I_t = -(p, q_j, q_k) \) and \( Y_t^P \) is a singleton. Assume towards a contradiction that \( E_p \) does not hold on \( x_t \). Then, by claim 5, \( |Y_{t+1}^P| = \frac{1}{2} \times |Y_t^P| \). But \( Y_{t+1}^P \) cannot have half as many elements as \( Y_t^P \), since \( Y_t^P \) is a singleton. From this contradiction, we conclude that \( E_p \) holds on \( x_t \) (and therefore also on \( Y_t \)).

Claim 9: For every \( t \) and \( p \in \{0, 1\} \), let \( v_t^p \) be the value in register \( p \) at time \( t \) in the run of \( M \) and \( q_t \) the state \( M \) is in at time \( t \). The following hold:

- if \( Q_{t_i} \) holds on \( x_t \) and \( Y_t \),
- if \( t > 0 \), then \( |Y_t^P| = 2^{|v_t^p|} \).

Proof of claim 9: As we remarked when we introduced them, conjuncts 10–12 of \( C_M \) guarantee that the state transition instruction of \( M \) are obeyed, if \( E_p \) holds in the appropriate places. Claim 5 shows that \( B \) follows the incrementation/decrementation instructions of \( M \), again under the assumption that \( E_p \) holds when appropriate. Finally, claim 8 shows that \( E_p \) holds where appropriate.

Claim 10: \( M \) is not halting.

Proof of claim 10: Follows from conjunct 13 of \( C_M \) together with claim 9.

We have shown the following.

**Theorem 8.** \( \text{ALCQ}^u/\text{ALCQ}^u \)-(Craig) separation is RE-complete.

We observe that the proof above can be adapted so that the universal role \( u \) is not used. Let \( C_M \) and \( D_M \) be as defined above, except that we treat \( u \) not as the universal role but instead as a role name. Now, add the following additional conjuncts to both \( C_M \) and \( D_M \):

- \( \exists u. \{a\} \)
• $\forall u. (\forall r. \exists u^- \cdot \{a\} \cap \forall r^- . \exists u^- \cdot \{a\})$

• $\forall u. (\forall s. \exists u^- \cdot \{a\} \cap \forall s^- . \exists u^- \cdot \{a\})$

These additional conjuncts guarantee that while $u$ is not necessarily interpreted as the universal relation, all elements that are relevant in the proof are $u$-successors of $a^\mathcal{A}$ and $a^\mathcal{B}$, respectively, and hence all conjuncts that start with $\forall u$ have the desired effect. Undecidability of $\mathcal{ALCOIQ}/\mathcal{ALCOI}$-(Craig) separation follows immediately.

**Theorem 9.** $\mathcal{ALCOIQ}/\mathcal{ALCOI}$-(Craig) separation is RE-complete.

### C.2. $C^2/FO^2$-separation is undecidable

We show that $C^2/FO^2$-(Craig) separation is undecidable. Our reduction is very similar to the one presented in the previous Section for $\mathcal{ALCOIQ}/\mathcal{ALCOI}$-separation. The reduction is given below, but before we consider the details we discuss the modifications that are required.

Since we are now working in $C^2$, instead of $\mathcal{ALCOIQ}^\omega$, we will use different notation, but the main idea remains the same. We use two formulas $\varphi_M$ and $\psi_M$, where the pointed model $\mathcal{A}$, $x_0$ of $\phi_M$ is essentially an $r$-chain encoding the run of $M$, and the model $\mathcal{B}$, $y_0$ of $\psi_M$ represents the contents of the registers. Synchronisation between $\mathcal{A}$ and $\mathcal{B}$ is done through the bisimulation between them. All of this is exactly as in the $\mathcal{ALCOIQ}^\omega/\mathcal{ALCOIQ}^\omega$ case.

Where things get more complicated is that now the bisimulation has to be a FO$^2$-bisimulation, which is more constraining than a $\mathcal{ALCOIQ}^\omega$-bisimulation. Recall that a FO$^2$-bisimulation requires that if $(x, y) \in \beta$, then for every $x'$ there is an $y'$ such that $(x', y') \in \beta$ and $(x, x') \mapsto (y, y')$ is a partial isomorphism, and similarly for every $y'$ there must be an $x'$ with the same properties. Here $(x, x') \mapsto (y, y')$ being a partial isomorphism means that (i) for every unary predicate $P$ in the relevant signature, $P(x)$ iff $P(y)$ and $P(x')$ iff $P(y')$, (ii) for every binary predicate $R$ in the relevant signature, $R(x, x')$ iff $R(y, y')$ and $R(x', x)$ iff $R(y', y')$ and (iii) $x = x'$ iff $y = y'$.

Perhaps surprisingly, it is only condition (iii) that will give us trouble, and that requires us to make the current proof slightly more complicated than the $\mathcal{ALCOIQ}^\omega/\mathcal{ALCOIQ}^\omega$ one. On the intended pointed models of $M$ and $D_M$, we can already choose an $y'$ (or $x'$) that satisfies properties (i) and (ii). Condition (iii), however, is not satisfied for those models.

This is because condition (iii) implies that a FO$^2$-bisimulation can never relate a singleton set to a multi-element set (i.e., if $(x, y) \in \beta$ and $(x', y') \in \beta$ with $y \neq y'$ then there must be a $x' \neq x$ such that $(x', y) \in \beta$ and $(x', y') \in \beta$). After all, if $(x, y), (x, y') \in \beta$ with $y \neq y'$ then there must be some $x'$ such that $(x', y') \in \beta$ and $(x, x') \mapsto (y, y')$ is a partial isomorphism. In particular, we must then have $x = x'$ iff $y = y'$, so from $y \neq y'$ it follows that $x \neq x'$.

In order to solve this issue, we make a small modification to the intended models for $M$ and $D_M$. Specifically, we create two copies of everything: the intended models of $\phi_M$ and $\psi_M$ are simply two disjoint copies of $C_M$ and $D_M$, respectively.

Now, let us consider the precise formulas $\varphi_M$ and $\varphi_M$ that achieve this effect. We begin with $\varphi_M$, which is the conjunction of the following formulas:

1. $\exists^=2 x Q_0(x)$
2. \( \forall x (F(x) \lor \bigvee_{1 \leq i \leq \ell} Q_i(x)) \)

3. \( \forall x (F(x) \rightarrow \bigwedge_{1 \leq i \leq \ell} \neg Q_i(x)) \)

4. \( \forall x \bigwedge_{1 \leq i \leq \ell} (Q_i(x) \rightarrow \bigwedge_{i<j \leq \ell} \neg Q_j(x)) \)

5. \( \forall x (Q_0(x) \rightarrow \forall y \neg R(y, x)) \)

6. \( \forall x (\neg F(x) \rightarrow \exists_{=1} y R(x, y)) \)

7. \( \forall x ((\neg F(x) \land \neg Q_0(x)) \rightarrow \exists_{=1} y R(y, x)) \)

8. \( \forall x (\neg F(x) \rightarrow \forall y ((R(x, y) \lor R(y, x)) \rightarrow \neg F(y))) \)

9. \( \forall x (\neg F \rightarrow \exists_{\geq 2} y (S(x, y) \land F(y))) \)

if \( I_i = +(p, q_j) \) then

10. \( \forall x (Q_i(x) \rightarrow \forall y (R(x, y) \rightarrow Q_j(y))) \)

and if \( I_i = -(p, q_j, q_k) \) then

11. \( \forall x ((Q_i(x) \land E_p(x)) \rightarrow \forall y (R(x, y) \rightarrow Q_j(y))) \)

12. \( \forall x ((Q_i(x) \land \neg E_p(x)) \rightarrow \forall y (R(x, y) \rightarrow Q_k(y))) \)

And, finally,

13. \( \forall x \neg Q_\ell(x) \)

The reader may notice that most conjuncts of \( \varphi_M \) are simply translations of the conjunct of \( C_M \) with the same number to \( C^2 \). The only exceptions, and these are very minor exceptions, are conjuncts 1 and 9. The corresponding \( C_M \) conjuncts guarantee the existence of one \( Q_0 \) element and one \( s \)-successor, while here we guarantee the existence of two \( Q_0 \) elements and two \( S \)-successors.

The formula \( \psi_M \) uses abbreviations \( U, R_0 \) and \( R_1 \), similar to how we used them in \( D_M \). We need to scale the numbers up a bit, however, due to the aforementioned issue regarding bisimulations and singletons.

- \( U(x) := \exists_{=2} y (S(x, y) \land F(y)) \lor \exists_{=4} y (S(x, y) \land F(y)) \)
- \( R_0(x) := \exists_{=2} y (S(x, y) \land F(y)) \lor \exists_{=3} y (S(x, y) \land F(y)) \)
- \( R_1(x) := \neg R_0(x) \).

Using these abbreviations, we can define the conjuncts of \( \psi_M \).

1. \( \exists_{=2} x Q_0(x) \)

2. \( \forall x (F(x) \lor \bigvee_{1 \leq i \leq \ell} Q_i(x)) \)

3. \( \forall x (F(x) \rightarrow \bigwedge_{1 \leq i \leq \ell} \neg Q_i(x)) \)
And finally

Apart from the renaming of $\phi$ discussed above, a singleton set can never be FO that we discussed previously. The required modifications are as follows:

**Proposition 10.** If $I_0(x) \rightarrow \exists y R(x, y)$ then

10. $\forall x ((Q_1(x) \land R_0(x)) \rightarrow (\exists = 2 y (\forall (R(x, y)) \land R(p(y))))

11. $\forall x ((Q_1(x) \land \neg R_0(x)) \rightarrow (\exists = 2 y (\forall (R(x, y)) \land \exists = 1 y (R(y, x) \land U(y))))

If $I_1 = +(p, q_j)$ then

12. $\forall x ((Q_1(x) \land R_0(x)) \land E_p) \rightarrow (\exists = 1 y (R(x, y) \land \forall (R(x, y) \land \exists = 1 x (R(x, y))))

13. $\forall x ((Q_1(x) \land R_0(x)) \land \neg E_p) \rightarrow (\exists = 1 y (R(x, y) \land \forall (R(x, y) \land \exists = 2 x (R(x, y))))

14. $\forall x ((Q_1(x) \land \neg R_0(x)) \rightarrow (\exists = 1 y (R(x, y) \land \forall (R(x, y) \land \exists = 1 x (R(x, y))))

And finally

15. $\forall x ((E_p(x) \land R_0(x)) \rightarrow U(x))$

Note that $\varphi_M$ and $\psi_M$ use the same signature $\varphi$.

**Proposition 10.** If $M$ is non-halting, then there are pointed models $\mathfrak{A}, x_0$ and $\mathfrak{B}, y_0$ with $\mathfrak{A} \models \varphi_M(x_0)$ and $\mathfrak{B} \models \psi_M(y_0)$ such that $\mathfrak{A}, x_0 \sim_{FO^2(\varphi)} \mathfrak{B}, y_0$.

**Proof.** The models in question are slight variations on the intended models for $C_M$ and $D_M$ that we discussed previously. The required modifications are as follows:

- For notation reasons, we call the relations $R$ and $S$, instead of $r$ and $s$.
- In the intended model of $C_M$, we gave every $\neg F$ element a single $s$-successor that satisfies $F$, in the intended model of $\varphi_M$ we give every $\neg F$ element two $S$-successors that satisfy $F$.
- Similarly, in the intended model of $D_M$, we gave every $\neg F$ element between 1 and 4 $s$-successors satisfying $F$, with the exact amount determined by whether $U, R_0$ and $R_1$ should hold. In the intended model for $\psi_M$ we increase this amount by one, so we add between 2 and 5 $S$-successors satisfying $F$ for every $\neg F$ element.
- Finally, we take two disjoint copies of the structure obtained so far.

Apart from the renaming of $r$ and $s$ to $R$ and $S$, these changes are necessary because, as discussed above, a singleton set can never be FO^2-bisimilar to a multi-element set.

Verifying that these structures are indeed models of $\mathfrak{A}$ and $\mathfrak{B}$, and that they are FO^2-bisimilar, is straightforward.
Next, the other direction.

**Proposition 11.** If there are pointed models $\mathfrak{A}, x_0$ and $\mathfrak{B}, y_0$ with $\mathfrak{A} \models \varphi_M(x_0)$ and $\mathfrak{B} \models \psi_M(y_0)$ such that $\mathfrak{A}, x_0 \sim_{\mathcal{FO}(\omega)} \mathfrak{B}, y_0$, then $M$ is non-halting.

**Proof.** This proof is mostly similar to that of Proposition 7, except for the first few claims, in which we show certain elements to be bisimilar. We therefore prove the first few claims in detail, and refer to the proof of Proposition 7 for details on the remainder.

Let $\beta$ be the bisimulation between $\mathfrak{A}$ and $\mathfrak{B}$, and let $\alpha_0$ be the elements in $\mathfrak{A}$ that satisfy $Q_0$ and $\beta_0$ the ones in $\mathfrak{B}$. Then, for $n \in \mathbb{N}$, let $\alpha_{n+1}$ be the set of $R$-successors of elements of $\alpha_n$ and $\beta_{n+1}$ the set of $R$-successors of $\beta_n$. The remained of this proof is organized by several numbered claims.

**Claim 1:** $\beta$ can only relate an element of $\alpha_0$ to elements of $\beta_0$, and vice versa.

**Proof of claim 1:** By induction. We show the base case for one direction, the other direction can be shown similarly. Suppose, towards a contradiction, that $(a, b) \in \beta$ with $a \in \alpha_0$ and $b \notin \beta_0$. Then, for any $x$, $(a, a) \mapsto (b, x)$ is not a partial isomorphism, since $Q_0(a)$ but $\neg Q_0(b)$. This contradicts $\beta$ a bisimulation, so such $a$ and $b$ cannot exist.

Assume then, as induction hypothesis, that the claim holds for all $n' < n$. Again, we show one direction; that $(a, b) \in \beta$ and $a \in \alpha_n$. The element $a$ has an $a$-predecessor $a' \in \alpha_{n-1}$. Because $\beta$ is a bisimulation, there must be a $b'$ such that $(b, b') \in \beta$ and $(a, a') \mapsto (b, b')$ is a partial isomorphism. We have $R(a', a)$, so we must also have $R(b', b)$. Furthermore, by the induction hypothesis, $(a', b') \in \beta$, together with $a' \in \alpha_{n-1}$, implies that $b' \in \beta_{n-1}$. Hence $b$ is the $R$-successor of an element in $\beta_{n-1}$, which, by definition, means $b \in \beta_n$. This completes the induction step, thereby proving the claim.

**Claim 2:** If $a \in \alpha_{n+1}$ and $R(a', a)$ then $a' \in \alpha_n$, and if $b \in \beta_{n+1}$ and $R(b', b)$ then $b' \in \beta_n$.

**Proof of claim 2:** In $\mathfrak{A}$, every element, other than the two that satisfy $Q_0$, has exactly one $R$-predecessor. So it immediately follows that every $R$-predecessor of $a \in \alpha_{n+1}$ is in $\alpha_n$.

Now, take any $b \in \beta_{n+1}$ and $b'$ such that $R(b', b)$. There must be some $a$ such that $(a, b) \in \beta$. By Claim 1, this implies that $a \in \alpha_{n+1}$. Furthermore, there is some $a'$ such that $(a', b') \in \beta$ and $(a', a) \mapsto (b', b)$ is a partial isomorphism. As $R(b', b)$, this implies that $R(a', a)$ as well, so we have $a' \in \alpha_n$. By Claim 1, this implies that $b' \in \beta_n$.

**Claim 3:** If $a_0 \in \alpha_0$ and $b_0 \in \beta_0$, then $(a_0, b_0) \in \beta$.

**Proof of claim 3:** Let $b_0$ be one of the two elements of $\beta_0$. Then there is at least one $a_0 \in \alpha_0$ such that $(a_0, b_0) \in \beta$. Let $a'_0$ be the other element of $\alpha_0$. This $b_0$ has exactly two $R$-successors $b_1, b'_1 \in \beta_1$ (with $b_1 \neq b'_1$). There must be $a_1, a'_1$ such that $(a_1, b_1) \in \beta$ and $(a'_1, b'_1) \in \beta$, and $(a_0, a_1) \mapsto (b_0, b_1)$ and $(a_0, a'_1) \mapsto (b_0, b'_1)$ are partial isomorphisms. From $R(b_0, b_1)$ and $R(b_0, b'_1)$ it therefore follows that $R(a_0, a_1)$ and $R(a_0, a'_1)$. As $a_0$ has exactly one $R$-successor, this implies that $a_0 = a'_0$.

Now, consider the pair $(b_1, b'_1)$. We have $(a_1, b_1) \in \beta$, so there must be some $a''_1$ such that $(a''_1, b_1) \in \beta$ and $(a''_1, b'_1) \mapsto (b_1, b'_1)$ is a partial isomorphism. This implies that, in particular, $a_1 = a''_1$ if and only if $b_1 = b'_1$. We have $b_1 \neq b'_1$, so $a_1 \neq a''_1$. From $(a''_1, b'_1) \in \beta$ and Claim 1, it follows that $a''_1 \in \alpha_1$. Because $a_1 \neq a''_1$, this $a''_1$ must be the other element of $\alpha_1$, which is the $R$-successor of $a'_0$.

Because $(a''_1, b'_1) \in \beta$ and $R(a''_0, a''_1)$, there must be some $b'_0$ such that $(a''_0, b'_0) \in \beta$ and $R(b'_0, b'_1)$. But $b'_1$ is an $R$-successor of $b_0$ and has only one $R$-predecessor, so $b'_0 = b'_1$. Hence
\((a'_0, b'_0) \in \beta\) implies \((a'_0, b_0) \in \beta\). We have now shown that, for arbitrary \(b_0 \in \beta_0\), we have \((a_0, b_0) \in \beta\) and \((a'_0, b_0) \in \beta\), with \(a_0 \neq a'_0\). The claim follows immediately.

**Claim 4:** If \(a_n \in \alpha_n\) and \(b_n \in \beta_n\), then \((a_n, b_n) \in \beta\).

**Proof of claim 4:** By induction. The base case is Claim 3. So assume as induction hypothesis that \(n > 0\) and that Claim 4 holds for all \(n' \leq n\). Take any \(a_n \in \alpha_n\) and \(b_n \in \beta_n\). There are \(a_{n-1} \in \alpha_{n-1}\) and \(\beta_{n-1} \in \beta_{n-1}\) such that \(R(a_{n-1}, a_n)\) and \(R(b_{n-1}, b_n)\). By the induction hypothesis, \((a_{n-1}, b_{n-1}) \in \beta\). As we have \((a'_0, b_0) \in \beta\), this shows that \((a_n, b_n) \in \beta\), thereby proving the claim.

The remainder of the proof proceeds in the same way as the proof of Proposition 7, so we omit it here.

**Corollary 12.** \(C^2/FO^2\)-separability is RE-complete.

**D. \(ALCQ^u/ALC^u\)-separability is 2ExpTime-complete**

In this section, we prove the following:

**Theorem 13.** \(ALCQ^u/ALC^u\)-(Craig) separation is 2ExpTime-complete.

We begin with the upper bound proof. Let \(L\) be one of the fragments of \(ALCIQ^u\) defined in Section 2, \(\varphi\) a signature, and let \(C_1, C_2\) be any \(L\)-concepts. To simplify presentation, we assume that \(C_1, C_2\) contain only one role name \(r \in \varphi\). It is straightforward to extend the upper bound proofs given in this and the next section to arbitrarily-many role names in and out of \(\varphi\).

Denote by \(sub(C_1, C_2)\) the closure under single negation of the set of subconcepts in \(C_1\) and \(C_2\). By a **type**, \(t\), we mean any subset of \(sub(C_1, C_2)\) for which there exists a structure \(\mathfrak{A}\) with some \(d \in \text{dom}(\mathfrak{A})\) such that \(t = \text{tp}(\mathfrak{A}, d)\), where

\[
\text{tp}(\mathfrak{A}, d) = \{C \in \text{sub}(C_1, C_2) \mid d \in C^{\mathfrak{A}}\}
\]

is the type realised by \(d\) in \(\mathfrak{A}\). Denote by \(\text{Tp}\) the set of all types. Clearly, \(|\text{Tp}| \leq 2^{|C_1|+|C_2|}\) and, moreover, \(\text{Tp}\) can be computed in time exponential in \(|C_1| + |C_2|\) for all considered logics (fragments of \(ALCIQ^u\)).

A **mosaic** is a set \(M\) of types. For structures \(\mathfrak{A}_1, \mathfrak{A}_2\) and \(i \in \{1, 2\}\), the mosaic defined by \(d \in \text{dom}(\mathfrak{A}_i)\) in \(\mathfrak{A}_1, \mathfrak{A}_2\) is the set \(T(d)\), where

\[
T(d) = \{\text{tp}(\mathfrak{A}_j, e) \mid e \in \text{dom}(\mathfrak{A}_j), \ \mathfrak{A}_i, d \sim_{ALC^u(\varphi)} \mathfrak{A}_j, e, j = 1, 2\}.
\]

We then say that \(M\) is a **mosaic defined by** \(\mathfrak{A}_1, \mathfrak{A}_2\) if there exists \(d \in \text{dom}(\mathfrak{A}_1) \cup \text{dom}(\mathfrak{A}_2)\) such that \(M = T(d)\). Clearly, there are at most doubly exponentially many mosaics.

By Lemmas 5 and 3, we can prove the \(2\text{ExpTime}\) upper bound of Theorem 13 by checking in double-exponential time whether there exist pointed structures \(\mathfrak{A}_1, d_1\) and \(\mathfrak{A}_2, d_2\) such that...
We can now define the mosaic elimination procedure. Let $\mathcal{S}$ with $t \mathcal{R} \subseteq \mathcal{S}$ we have constructed in double-exponential time.

Lemma 15. Given a set $\mathcal{S}$ of mosaics and $M \in \mathcal{S}$, it can be decided in double-exponential time in $|C_1| + |C_2|$ whether $M$ is bad in $\mathcal{S}$.

Let $\mathcal{S}_0$ be the set of all mosaics. Then we obtain the set $\mathcal{S}_{i+1}$ from $\mathcal{S}_i$, for $i \geq 0$, by eliminating mosaics that are bad in $\mathcal{S}_i$. Denote by $\mathcal{S}^*$ the resulting set without bad mosaics, which can be constructed in double-exponential time.

Lemma 15. The following conditions are equivalent:

1. $C_1$ and $C_2$ are satisfied in $\mathcal{ALC}_n(\emptyset)$-bisimilar pointed structures;
2. there is $M \in S^*$ with $t_1, t_2 \in M$ such that $C_1 \in t_1$ and $C_2 \in t_2$.

**Proof.** ($1 \Rightarrow 2$) Suppose $\mathfrak{A}_1, d_1 \sim_{\mathcal{ALC}_n(\emptyset)} \mathfrak{A}_2, d_2, d_1 \in C_{\mathfrak{A}_1}^\emptyset$ and $d_2 \in C_{\mathfrak{A}_2}^\emptyset$. Let $S$ be the set of all mosaics defined by $\mathfrak{A}_1, \mathfrak{A}_2$. It is easy to see that $S$ does not contain any bad $M$ in $S$, and so $S \subseteq S^*$.

($2 \Rightarrow 1$) We construct structures $\mathfrak{A}_1, \mathfrak{A}_2$ from $S^*$ and a mosaic $M^* \in S^*$ with $t_1^*, t_2^* \in M^*$ such that $C_1 \in t_1^*$ and $C_2 \in t_2^*$. For every $M \in S^*$, take an existential witness $M_M$ for $M$ with a copying function $F_M$ and a witnessing relation $R_M \subseteq \Delta_M \times \Gamma_M$. The domain of $\mathfrak{A}_i$ contains all words

$$s = (t_0, M_0)(t_1, M_1, i_1) \ldots (t_n, M_n, i_n)$$

such that

- $M_0 \in S^*$;
- $t_0, t_1, \ldots, t_n, t_i^*$ are $u$-equivalent;
- $t_j \in M_j$ for all $j \leq n$;
- $M_{j+1} \in M_{M_j}$ and $i_{j+1} < f_{M_j}(t_{j+1}, M_{j+1})$.

We next interpret the concept names $A$ by setting, for $s$ defined by (1), $s \in \mathcal{A}\mathcal{L}\mathcal{C}_i$ iff $A \in t_n$. We let $(s, s') \in r_{\mathcal{A}\mathcal{L}\mathcal{C}_i}$, for $s$ defined by (1) and

$$s' = (t_0', M_0')(t_1', M_1', i_1') \ldots (t_m', M_m', i_m') \in \text{dom(}\mathfrak{A}_i\text{)},$$

iff $s$ is an initial part of $s'$ with $m = n + 1$ and $(t_n, M_n)R_M(t_{n+1}', M_{n+1}', i_{n+1}')$. It is not hard to check that, for all $C \in \text{sub}(C_1, C_2)$ and all $s$ given by (1), we have $s \in C_{\mathfrak{A}_i}$ iff $C \in t_n$.

Finally, we define an $\mathcal{ALC}_n(\emptyset)$-bisimulation $\beta$ between $\mathfrak{A}_1$ and $\mathfrak{A}_2$ by setting, for $s$ given by (1) and $s'$ by (2), $s\beta s'$ iff $n = m$ and $M_i = M'_i$ for all $i \leq n$. One can show that $\beta$ is an $\mathcal{ALC}_n(\emptyset)$-bisimulation between $\mathfrak{A}_1$ and $\mathfrak{A}_2$, as required.

We now prove the $\mathcal{E}\mathcal{X}\mathcal{P}\mathcal{T}\mathcal{I}\mathcal{M}$-lower bound, first for Craig-separation and then for separation. The proof is by reduction of acceptance for alternating Turing machines and is very similar to the proof that interpolant existence is $\mathcal{E}\mathcal{X}\mathcal{P}\mathcal{T}\mathcal{I}\mathcal{M}$-hard for concept inclusions under $\mathcal{ALC}_H$ ontologies given in [8].

An alternating Turing machine (ATM) is a tuple $M = (Q, \Theta, \Gamma, q_0, \Delta)$ where $Q = Q_\exists \cup Q_\forall$ is a finite set of states partitioned into existential states $Q_\exists$ and universal states $Q_\forall$. Further, $\Theta$ is the input alphabet and $\Gamma$ is the tape alphabet that contains a blank symbol $\square \notin \Theta$, $q_0 \in Q_\forall$ is the initial state, and $\Delta \subseteq Q \times \Gamma \times Q \times \Gamma \times \{L, R\}$ is the transition relation. We assume without loss of generality that the set $\Delta(q, a) := \{(q', a', M) \mid (q, a, q', a', M) \in \Delta\}$ contains exactly two or zero elements for every $q \in Q$ and $a \in \Gamma$. Moreover, the state $q'$ must be from $Q_\forall$ if $q \in Q_\exists$ and from $Q_\exists$ otherwise, that is, existential and universal states alternate. Acceptance of ATMs is defined in a slightly unusual way, without using accepting states. Intuitively, an ATM accepts if it runs forever on all branches and rejects otherwise. More formally, a configuration of an ATM is a word $wqvw'$ with $w, w' \in \Gamma^*$ and $q \in Q$. We say that $wqv'$ is existential if $q$ is, and likewise for universal. Successor configurations are defined in the usual way. Note that every
configuration has exactly zero or two successor configurations. A computation tree of an ATM $M$ on input $w$ is a (possibly infinite) tree whose nodes are labeled with configurations of $M$ such that

- the root is labeled with the initial configuration $q_0w$;
- if a node is labeled with an existential configuration $wq_w'$, then it has a single successor which is labeled with a successor configuration of $wq_w'$;
- if a node is labeled with a universal configuration $wq_w''$, then it has two successors which are labeled with the two successor configurations of $wq_w''$.

An ATM $M$ accepts an input $w$ if there is a computation tree of $M$ on $w$. Note that we can convert any ATM $M$ in which acceptance is based on accepting states to our model by assuming that $M$ terminates on any input and then modifying it to enter an infinite loop from the accepting states. It is well-known that there are $2^n$-space bounded ATMs which recognize a 2ExpTime-hard language $\mathcal{L}$, where $n$ is the length of the input $w$.

Let us fix such an ATM $M = (Q, \Theta, \Gamma, q_0, \delta)$ and an input $w = a_0 \ldots a_{n-1}$ of length $n$. We aim to construct $C_1, C_2$ such that $M$ accepts $w$ iff $C_1, C_2$ are satisfied in $\mathcal{ALC}^n(\varrho)$-bisimilar pointed models, where

$$\varrho = \{r, s, Z, B_r, B_1^2, B_2^2\} \cup \{A_\sigma \mid \sigma \in \Gamma \cup (Q \times \Gamma)\}.$$  

$C_1$ generates an $r$-chain of length $n$ as follows:

$$C_1 = \exists r^n.T \cap \forall r^n.\varr.\bot \sqcap \bigwedge_{0 \leq i < n} \forall r_i(= 1 r.\top)$$

$C_2$ first generates a binary $r$-tree of depth $n$ with leaves satisfying counter values from 0 to $2^n - 1$ represented by concept names $A_0, \ldots, A_{n-1}$. Hence $C_1$ contains the following conjuncts:

1. $\forall r_i.(= 2 r.\top)$ for $0 \leq i < n$;
2. $\forall r_i.(\exists r.A_i \sqcap \exists r.\lnot A_i)$ for $0 \leq i < n$;
3. $\forall r_i.((A_j \rightarrow \forall r.A_j) \sqcap (\lnot A_j \rightarrow \forall r.\lnot A_j))$, for $0 \leq j < i < n$.

Next, we make a concept name $L_R$ true in all leaves of the tree

$$\forall r^n.L_R$$

and start from the leaves $s$-trees with two counters, realised using concept names $U_i$ and $V_i$, $0 \leq i < n$, and initialised to 0 and the value of the $A_0, \ldots, A_{n-1}$ counter, respectively:

1. $\forall u.(L_R \rightarrow (U = 0))$;
2. $\forall u.(L_R \rightarrow (A_j \leftrightarrow V_j))$ for $0 \leq j < n$;
3. $\forall u.(L_R \rightarrow \exists s.\top)$.  

Observe that if \( d \in C_1^{\delta_1}, e \in C_2^{\delta_2}, \) and \( \mathfrak{A}_1, d \sim_{\mathcal{ALC}^n(\varnothing)} \mathfrak{A}_2, e \) for \( C_2 \) the conjuncts of \( C_2 \) constructed up to now, then there is a node \( d' \) in \( \mathfrak{A}_1 \) which is reachable via an \( r \)-chain of length \( n \) from \( d \) and there are leaves \( e_0, \ldots, e_{2^n-1} \in L_{R}^{\delta_2} \) of the binary tree in \( \mathfrak{A}_2 \) with root \( e \) such that
\[
\mathfrak{A}_1, d' \sim_{\mathcal{ALC}^n(\varnothing)} \mathfrak{A}_2, e_0 \sim_{\mathcal{ALC}^n(\varnothing)} \cdots \sim_{\mathcal{ALC}^n(\varnothing)} \mathfrak{A}_2, e_{2^n-1}.
\]

Now we add further conjuncts to \( C_2 \) that start from \( L_R s \)-trees encoding the computation of \( M \) on input \( w \) using the counters \( U \) and \( V \) in exactly the same way as in [8] such that the following conditions are equivalent:

1. \( M \) accepts \( w \);
2. there exist structures \( \mathfrak{A}_1 \) and \( \mathfrak{A}_2 \) such that \( \mathfrak{A}_1, d \sim_{\mathcal{ALC}^n(\varnothing)} \mathfrak{A}_2, e \), for some \( d \in C_1^{\delta_1} \) and \( e \in C_2^{\delta_2} \).

This completes the proof of \( 2 \text{ExpTime} \)-hardness of Craig separation. To prove \( 2 \text{ExpTime} \)-hardness of separation, we replace all concept names \( A \) in \( C_2 \) that are not in \( \varnothing \) by \( (= 2r_A, \top) \) with \( R_A \) a fresh role name and denote the resulting concept by \( D_2 \). Let \( \sigma \) be the set of all concept and role names in \( C_1, D_2 \). Then one can show that the following conditions are equivalent:

1. there exist structures \( \mathfrak{A}_1 \) and \( \mathfrak{A}_2 \) such that \( \mathfrak{A}_1, d \sim_{\mathcal{ALC}^n(\varnothing)} \mathfrak{A}_2, e \), for some \( d \in C_1^{\delta_1} \) and \( e \in C_2^{\delta_2} \).
2. there exist structures \( \mathfrak{A}_1 \) and \( \mathfrak{A}_2 \) such that \( \mathfrak{A}_1, d \sim_{\mathcal{ALC}^n(\sigma)} \mathfrak{A}_2, e \), for some \( d \in C_1^{\delta_1} \) and \( e \in D_2^{\delta_2} \).

This complete the hardness proof.

\[\]

**E. \( \mathcal{ALCT}^u/\mathcal{ALCT}^u \) separation is \( 2 \text{ExpTime-complete} \)**

In this section, we prove the following:

**Theorem 16.** \( \mathcal{ALCT}^u/\mathcal{ALCT}^u \)-(Craig) separation is \( 2 \text{ExpTime-complete} \).

The lower bound proof is essentially the same as in the previous section. Hence we focus on the upper bound proof. Assume \( \mathcal{ALCT}^u \)-concepts \( C_1, C_2 \) and a signature \( \varnothing \) are given. As before, we assume that \( C_1, C_2 \) contain only one role name \( r \in \varnothing \). It is straightforward to extend the upper bound proofs given in this section to arbitrarily many role names in and out of \( \varnothing \).

We use the notion of types introduced in the previous section. Other notions such as that of a mosaic are a bit different. Let \( m_{C_1,C_2} \) be the maximal parameter in \( C_1, C_2 \).

We first introduce the notion of an extended type as a pair \((t, P)\) with \( t \) a type and \( P \) a set of expressions containing, for each type \( t' \), exactly one expression of the form \((= n \ r.t')\) with \( 0 \leq n \leq m_{C_1,C_2} \) or \((> m_{C_1,C_2} \ r.t')\) and exactly one expression of the form \((= n \ r^- t')\) with \( 0 \leq n \leq m_{C_1,C_2} \) or \((> m_{C_1,C_2} \ r^- t')\). The semantics of these expressions is defined in the obvious way. Note that \( t \) and \((= n \ s.t')\) with \( n > 0 \) can only be satisfied if \( t \) and \( t' \) are \( u \)-equivalent.
For $s \in \{r, r^{-1}\}$, an extended type specifies exactly how many $s$-successors there are that satisfy type $t'$. In some places we only care about a coarser distinction, however, namely whether the number of $s$-successors that have type $t'$ is greater than 0. We therefore write $(> 0 s.t') \in P$ as a shorthand for ‘$(= n s.t') \in P$ for some $1 \leq n \leq m_{C_1,C_2}$ or $(> m_{C_1,C_2} s.t') \in P$’.

The extended type $(tp(A, d), e(A, d, D))$ realised in $A$ at a pair $(d, D)$ with $d \in \text{dom}(A)$ and $D \subseteq \text{dom}(A)$ is defined by setting

- $(= n s.t') \in e(A, d, D)$ if $n \leq m_{C_1,C_2}$ is the number of $d' \in D$ with $(d, d') \in s^A$ such that $t' = tp(A, d')$;

- $(> m_{C_1,C_2} s.t') \in e(A, d, D)$ if the number of $d' \in D$ with $(d, d') \in s^A$ such that $t' = tp(A, d')$ is larger than $m_{C_1,C_2}$.

An extended type $(t, P)$ is called a root if $n = 0$ for all $(= n s.t') \in P$ and $P$ contains no $(> m_{C_1,C_2} s.t')$.

We are going to construct bisimilar models that are (almost) tree-shaped. When constructing these models, we distinguish between types that are realised in a parent node and types that are realised in child nodes. Hence we define a two-way extended type as a tuple of the form $(t, E)$ with $E = (E_{up}, E_{down})$, where $(t, E_{up})$ and $(t, E_{down})$ are extended types.

Given $d \in A$ and $D \subseteq \text{dom}(A)$, the two-way extended type realised in $A$ at $(d, D)$, is defined as

$$\text{tw}(A, d, D) = (tp(A, d), e(A, d, D), e(A, d, \text{dom}(A) \setminus D))$$

We next aim at spelling out when adding up the number of witnesses for the number restrictions in $E_{up}$ and $E_{down}$ ensures that we satisfy the number restrictions in $t$. Assume $P \in \{E_{up}, E_{down}\}$ is given. The $s$-profile of $P$, $Pr_s(P)$ contains for every $C \in \text{sub}(C_1, C_2)$

- $(= n s.C)$ if $n$ is the sum over all $k$ with $(= k s.t') \in P$ and $C \in t'$;

- $(> m_{C_1,C_2} s.C)$ if the sum above exceeds $m_{C_1,C_2}$ or $(> m_{C_1,C_2} s.t') \in P$ for some $t'$ with $C \in t'$.

The joint $s$-profile of $E_{up}, E_{down}$, $Pr_s(E_{up}, E_{down})$, contains for $C \in \text{sub}(C_1, C_2)$

- $(= n s.C)$ if $n = n_1 + n_2 \leq m_{C_1,C_2}$ for $(= n_1 s.C)$ in the $s$-profile of $E_{up}$ and $(= n_2 s.C)$ in the $s$-profile of $E_{down}$;

- $(> m_{C_1,C_2} s.C)$ otherwise

We call $(t, E_{up}, E_{down})$ consistent if for $s \in \{r, r^{-1}\}$

- if $(\geq n s.C) \in t$ and $(= n' s.C) \in Pr_s(E_{up}, E_{down})$, then $n' \geq n$,

- if $(\leq n s.C) \in t$ and $(= n' s.C) \in Pr_s(E_{up}, E_{down})$, then $n' \leq n$,

- if $(\leq n s.C) \in t$, then $(> m_{C_1,C_2} s.C) \notin Pr_s(E_{up}, E_{down})$.

- if $(> 0 s.t')$ occurs in $E_{up} \cup E_{down}$, then $t$ and $t'$ are $u$-equivalent.
Clearly any realised two-way extended type $\text{tw}(\mathfrak{A}, d, D)$ is consistent.

A mosaic $M$ is a set of consistent two-way extended types. A mosaic is a root if $(t, E_{\text{up}})$ is a root for all $(t, E_{\text{up}}, E_{\text{down}}) \in m$.

For structures $\mathfrak{A}_1, \mathfrak{A}_2$ and $i \in \{1, 2\}$, the mosaic defined by $(d, d')$ with $d, d' \in \text{dom}(\mathfrak{A}_i)$ for some $i \in \{1, 2\}$ in $\mathfrak{A}_1, \mathfrak{A}_2$, is defined as

$$T(d, d') = \{ \text{tw}(\mathfrak{A}_j, e, D) \mid e \in \text{dom}(\mathfrak{A}_j), \mathfrak{A}_i, d \sim_{\text{ALCCU}, e} \mathfrak{A}_j, e, j \in \{1, 2\} \},$$

for $D = \{ e' \in \text{dom}(\mathfrak{A}_j) \mid \mathfrak{A}_i, e' \sim_{\text{ALCCU}, e} \mathfrak{A}_j, d' \}$, for $j = 1, 2$.

We say that a mosaic $M$ of two-way extended types is a mosaic defined by $\mathfrak{A}_1, \mathfrak{A}_2$ if there exist $d, d' \in \text{dom}(\mathfrak{A}_1) \cup \text{dom}(\mathfrak{A})$ such that $m = (T(d, d'))$.

Observe that such a mosaic is a root if and only if there do not exist $d, d'$ defining $m$ with $(d, d') \in s^{\mathfrak{A}_1}$ for some $s \in \{r, r^{-1}\}$.

Given a mosaic $M$, we now provide criteria for when a set $M$ of mosaics provides appropriate witnesses for the number restrictions in $E_{\text{down}}$ for all two-way extended types $(t, E_{\text{up}}, E_{\text{down}}) \in M$. For extended types $(t, P)$ and $(t', P')$, we write $(t, P) \rightarrow_s (t', P')$ and say that $(t, P)$ and $(t', P')$ are $s$-coherent if $(> 0 s t') \in P$ and $(> 0 s^{-1} t) \in P'$ and $t, t'$ are $u$-equivalent. Intuitively, $s$-coherence captures that we can draw an $s$-edge between nodes representing $(t, P)$ and $(t', P')$. For two-way extended types $(t, E)$ and $(t', E')$ we write $(t, E) \rightarrow_s (t', E')$ and say that they are $s$-coherent if $(t, E_{\text{down}}) \rightarrow_s (t', E'_{\text{up}})$.

Let $M$ be a mosaic and $M$ a set of mosaics, in what follows we assume the two-way extended types they contain are consistent. Then $M$ is a syntactic existential witness for $M$ if there are relations $r, r^{-1} \subseteq M \times \{ (t', E'), M' \} \mid (t', E') \in M' \} \text{ such that for every } s \in \{r, r^{-1}\}$

1. if $((t, E), ((t', E'), M')) \in s$ then $(t, E) \rightarrow_s (t', E')$,

2. for every $M' \in M$, either
   
   - for every $(t, E) \in M$ there is some $(t', E') \in M'$ such that $((t, E), ((t', E'), M')) \in s$ and
   - for every $(t', E') \in M'$ there is some $(t, E) \in M$ such that $((t, E), ((t', E'), M')) \in s$

   in which case we say that $M'$ is a $s$-successor of $M$, or
   
   - for every $(t, E) \in M$ and every $(t', E') \in M', ((t, E), ((t', E'), M')) \notin s$

   in which case $M'$ is not an $s$-successor of $M$,

3. for every $(t, E) \in M$, if $(= n s t') \in E_{\text{down}}$ or $(> m_{C_1, C_2} s t') \in E_{\text{down}}$ then there are $E'$ and $M'$ such that $((t, E), ((t', E'), M')) \in s$,

4. for every $M' \in M$ and every $(t', E') \in M'$, if $(= n s^{-1} t) \in E_{\text{up}}$ or $(> m_{C_1, C_2} s^{-1} t) \in E_{\text{up}}$, then there is some $(t, E) \in M$ such that $((t, E), ((t', E'), M')) \in s$.

5. for every $((t, E), ((t', E'), M')) \in s$, there is a function $f$ that assigns to each $s$-successor $M''$ of $M$ an extended type $(t'', E'') \in M''$ such that
   
   - $f(M') = (t', E')$, 

   etc.
We next provide a definition of existential witnesses that is closer to the intended semantics than syntactic existential witnesses. We call $\mathcal{M}$ an existential witness for $M$ if there are relations $r, r^{-1} \subseteq \Delta \times \Gamma$, where $\Delta = \{(t, E), j) \mid j < \omega, (t, E) \in M\}$ and $\Gamma = \{(t', E'), j, M' \mid j < \omega, (t', E') \in M, M' \in \mathcal{M}\}$ such that for every $s \in \{r, r^{-1}\}$,

- if $((t, E), j) \in \Delta$ and $(= n \ s, t') \in E_{\text{down}}$ then there are exactly $n$ different $E', k$ and $M'$ such that $((t', E'), k, M')$ is an $s$-successor of $((t, E), j)$,

- if $((t, E), j) \in \Delta$ and $(> m_{c_1, c_2} \ s, t') \in E_{\text{down}}$ then there are more than $m_{c_1, c_2}$ different $E', k$ and $M'$ such that $((t', E'), k, M')$ is an $s$-successor of $((t, E), j)$,

- if $((t', E'), k, M') \in \Gamma$ and $(\geq n^{-1} \ s, t) \in E_{\text{up}}$, then there are exactly $n$ different $E$ and $j$ such that $((t', E'), k, M')$ is an $s$-successor of $((t, E), j)$,

- if $((t', E'), k, M') \in \Gamma$ and $(>) m_{c_1, c_2} \ s^{-1}, t) \in E_{\text{up}}$, then there are more than $m$ different $E$ and $j$ such that $((t', E'), k, M')$ is an $s$-successor of $((t, E), j)$,

- if $((t, E), j) \in \Delta$ has an $s$-successor $((t', E'), k, M') \in \Gamma$ then for every $(t^*, E^*, j^*) \in \Delta$ there are $t^*, E^*$ and $k^*$ such that $((t^*, E^*), k^*, M')$ is an $s$-successor of $((t^*, E^*), j^*)$,

- if $((t', E'), k, M') \in \Gamma$ has an $s$-predecessor $((t, E), j) \in \Delta$ then for every $((t^*, E^*), k^*, M') \in \Gamma$ there are $t^*$ and $j^*$ such that $((t^*, E^*), k^*, M')$ is an $s$-successor of $((t^*, E^*), j^*)$.

**Lemma 17.** If $\mathcal{M}$ is a syntactic existential witness for $M$ then $\mathcal{M}$ is an existential witness for $M$.

**Proof.** Let $r, r^{-1}$ be relations witnessing the fact that $\mathcal{M}$ is a syntactic existential witness for $M$, and enumerate all elements of $\Gamma \cup \Delta$. We will add $r$ and $r^{-1}$ edges for each $x \in \Gamma \cup \Delta$ in enumeration order. Throughout the process, we will keep invariant the property that, by the time we reach $x \in \Gamma \cup \Delta$, there will be at most one $r$ or $r^{-1}$ edge to or from $x$ already. We first treat the two cases where $x$ does not have any edges yet, then the two cases where one edge has already been added.

**Case 1:** Suppose $x = ((t, E), j) \in \Delta$ and no edges from $x$ have been added yet. For every $s \in \{r, r^{-1}\}$, do the following.

Let $\mathcal{M}_s \subseteq \mathcal{M}$ be the set of $s$-successors of $M$. Take any $M' \in \mathcal{M}_s$. Then there is some $(t', E') \in M'$ such that $((t, E), ((t', E'), M')) \in s$. Let $f$ be the assignment function for $((t, E), ((t', E'), M')) \in s$, and let $\mathcal{E} = \{(t'', E''), M'' \mid f(M'') = (t'', E'')\}$.

Consider any $t''$ such that $(= n.s.t'') \in E_{\text{down}}$. It is a property of $f$ that there are at most $n$ elements in $\mathcal{E}$ that have a $t''$ component. If there are less than $n$ such elements, take any $((t'', E''), M'')$ such that $((t, E), ((t'', E''), M'')) \in s$, and add enough copies of this $((t'', E''), M'')$ to $\mathcal{E}$ to make the total number of elements with a $t''$ component exactly $n$. We turn $\mathcal{E}$ into a multi-set by doing this. Note that $t''$, $E''$, $M''$ with the required property exist by condition 3 of syntactic existential witness.

Similarly, if $(>) m_{c_1, c_2} \ s.t'' \in E_{\text{down}}$ and there are $\leq M_{c_1, c_2}$ elements in $\mathcal{E}$ with $t''$ component, add copies of $((t'', E''), M'')$ until we have more than $m_{c_1, c_2}$ such elements.
Now, for every \(((t', E'), M') \in E\), add an \(s\)-edge from \(x = ((t', E), j)\) to the lowest (in the enumeration) element \(((t', E'), m, M') \in \Gamma\) that does not yet have any edges to it.

**Case 2:** Suppose \(x = ((t', E'), k, M') \in \Gamma\) and no edges to \(x\) have been added yet. For every \(s \in \{r, r^{-1}\}\), do the following.

If \((= n \cdot s^{-1} \cdot t) \in E'_{up}\) or \((> m_{C_1, C_2} \cdot s^{-1} \cdot t) \in E'_{up}\), let \((t, E)\) be such that \(((t, E), ((t'E'), M')) \in s\). Create \(s\) edges to \(x\) from the first \(n\) if \((= n \cdot s^{-1} \cdot t) \in E'_{up}\) or the first \(m_{C_1, C_2} + 1\) if \((> m_{C_1, C_2} \cdot s^{-1} \cdot t) \in E'_{up}\) elements of the form \(((t, E), j) \in \Delta\) that do not yet have any edges going from them.

**Case 3:** Suppose \(x = ((t', E'), k, M') \in \Delta\) and there is already an \(s\) edge from \(x\) to \(((t', E'), k, M') \in \Gamma\). Add edges for \(s^{-1}\) as in case 1. With regard to \(s\), add edges like in case 1 except that instead of taking any \(s\)-successor we now take the successor \(((t', E'), M')\) as a starting point, and take the already existing \(s\) edge from \(((t, E), j)\) to \(((t', E'), k, M')\) instead of one of the edges that we would otherwise add.

**Case 4:** Suppose \(x = ((t', E'), k, M') \in \Gamma\) and there is already an \(s\) edge from \(((t', E'), k, M') \in \Gamma\) to \(x\). Add edges for \(s^{-1}\) as in case 2. For \(s\), do the same except we take the edge from \(((t, E), j)\) to \(x\) instead of one edge that would otherwise be added.

It is immediate from the construction of \(r\) and \(r^{-1}\) that the number restriction conditions of an existential witness are satisfied. The bisimilarity conditions follow from \(\mathcal{M}\) being a syntactic existential witness.

The converse direction should be obvious.

**Lemma 18.** If \(\mathcal{M}\) is a existential witness for \(M\) then \(\mathcal{M}\) is a syntactic existential witness for \(M\).

We are going to construct bisimilar models starting with a root mosaic, then satisfying number restrictions in the mosaic by adding existential witnesses, then satisfying number restrictions in the \(E_{down}\) part of the fresh mosaics in existential witnesses, and so on.

Extended types \((t, P)\) and \((t', P')\) are called profile-equivalent if \(t = t'\) and \(Pr_s(P) = Pr_s(P')\) for \(s \in \{r, r^{-1}\}\). Two-way extended types \((t, E_{up}, E_{down})\) and \((t', E'_{up}, E'_{down})\) are

- up profile-equivalent if \((t, E_{up})\) and \((t', E'_{up})\) are profile-equivalent;
- down profile-equivalent if \((t, E_{down})\) and \((t', E'_{down})\) are profile-equivalent;

We say that mosaics \(M, M'\) are

- down profile-equivalent if for any \((t, E_{up}, E_{down}) \in M\) there exists \((t', E'_{up}, E'_{down}) \in M'\) which is down profile-equivalent and vice versa;
- up profile-equivalent if for any \((t, E_{up}, E_{down}) \in M\) there exists \((t', E'_{up}, E'_{down}) \in M'\) which is up profile-equivalent and vice versa.
- down/up profile-equivalent if for any \((t, E_{up}, E_{down}) \in M\) there exists \((t', E'_{up}, E'_{down}) \in M'\) such that \((t, E_{down})\) and \((t', E'_{down})\) are profile-equivalent and vice versa.

**Lemma 19.** Assume that \(\mathcal{M}\) is a syntactic existential witness for \(M\), and let \(\{r, r^{-1}\}\) be the witnessing relations. Let \(M' \subseteq \mathcal{M}\) be such that for every \(s \in \{r, r^{-1}\}\), every \(M' \in \mathcal{M} \setminus \mathcal{M}'\), every \((t, E) \in M\) and every \((t', E') \in M'\) such that \(M'\) is an \(s\)-successor of \(M\) and \((t, E) \rightarrow_s\)
(t′, E′) there are some M′′ ∈ M′ and (t′, E′′) ∈ M′′ such that M′′ is an s-successor of M and
(t, E) →_s (t′, E′′).

Then M′ is also a syntactic existential witness for M.

Proof. We obtain witnessing relations \{r′, r′−1\} for M and M′ by replacing any edge
((t, E), ((t′, E′), M′)) ∈ s where M′ ∈ M \ M′ by the edge ((t, E), ((t′, E′′), M′′)). ⌦

Lemma 20. Assume that M is a syntactic existential witness for M, and let \{r, r−1\} be the
witnessing relations. Let M′ be obtainable from M be replacing some M′ ∈ M by any M′′ with
the properties that (i) M′′ ⊆ M′ and (ii) for every s ∈ \{r, r−1\}, every (t, E) ∈ M and every
(t′, E′) ∈ M′, if (t, E) →_s (t′, E′) then there is a (t′, E′′) ∈ M′′ such that (t, E) →_s (t′, E′′).

Then M′ is also a syntactic existential witness for M.

Proof. We obtain witnessing relations \{r′, r′−1\} for M and M′ by replacing any edge
((t, E), ((t′, E′), M′)) ∈ s by the edge ((t, E), ((t′, E′′), M′′)). ⌦

Lemma 21. Assume that M is a syntactic existential witness for M, and let \{r, r−1\} be the
witnessing relations. Let M* ⊆ M be such that for every s ∈ \{r, r−1\}, every M′ ∈ M, every
(t, E) ∈ M and (t′, E′) ∈ M′, if ((t, E), (t′, E′), M′)) ∈ s then there is some (t, E*) ∈ M*
such that (t, E*), (t′, E′), M′) ∈ s.

Then M is a syntactic existential witness for M*.

Proof. The witnessing relations r* and r*−1 are simply the restrictions of r and r−1 to M*. ⌦

Lemma 22. Assume that M is a syntactic existential witness for M. Then there are M′ and
M* ⊆ M such that for any N′ ∈ M′ there exists M′ ∈ M with N′ ⊆ M′ such that

- M′ is a syntactic existential witness for M′;
- |M′| ≤ K_1, where K_1 is exponential in |C_1| + |C_2|;
- |M*| ≤ K_2, where K_2 is exponential in |C_1| + |C_2|;
- for all N′ ∈ M′, |N′| ≤ K_3, where K_3 is exponential in |C_1| + |C_2|;
- M and M* are up profile-equivalent;
- for every N′ ∈ M′ there exists a down profile-equivalent M′ ∈ M.

Proof. For any M′ ∈ M, let N′ ⊆ M′ be such that (i) for every (t′, E′) ∈ M′ there is a
down profile-equivalent (t′, E′′) ∈ N′, (ii) for every t, if there is a (t′, E′) ∈ M′ such that
(> 0s.t′) ∈ E_u, then there is a (t′, E′′) ∈ N′ such that (> 0s.t′) ∈ E_u and (iii) N′ is at most
exponential in size. Such N′ exists because there are at most exponentially many profiles and
exponentially many types, and we need at most one witness for each.

Let N be the result of replacing every M′ ∈ M by the corresponding N′. By Lemma 20, N
is a syntactic existential witness for M. Let r, r−1 be the witnessing relations.

Now, let M′ be a subset of N such that (i) for every every s ∈ \{r, r−1\}, every M′ ∈ N \ M′,
every (t, E) ∈ M and every (t′, E′) ∈ M′ such that M′ is an s-successor of M and (t, E) →_s
We compute a sequence with \( M'' \) such that \( M'' \) is an \( s \)-successor of \( M \) and \( (t, E) \to_s (t', E') \) and (ii) \( M' \) is at most exponential in size. Such \( M' \) exists because for every \( s \in \{ r, r^{-1} \} \) and every \( t, t' \) we need only one witnessing mosaic. Furthermore, by Lemma 19, \( M' \) is a syntactic existential witness for \( M \).

Now, let \( M^* \) be a subset of \( M \) such that (i) for every \((t, E) \in M\) there is an up profile-equivalent \((t, E^*) \in M^*\), (ii) for every \( s \in \{ r, r^{-1} \} \) and every \((t, E), ((t', E'), M') \in M\) and (iii) \( M^* \) is at most exponential in size. Such \( M^* \) exists because there are at most exponentially-many profiles and exponentially many \( t \) and \((t', E'), M')\), and hence we need at most exponentially-many \( E^* \) such that \((t, E^*), ((t', E'), M')) \in M^* \). By Lemma 21, \( M^* \) is a syntactic existential witness for \( M^* \).

Next we define how existential witnesses are extracted from bisimilar models. We now return to the view of mosaics as tuples \( (T_1, T_2) \). Assume \( A_{\mathcal{L}C\mathcal{T}^\omega} \) \( \mathfrak{A}_1, \mathfrak{A}_2 \). Consider \((d, d') \in \text{dom}(\mathfrak{A}_1)\) realising a mosaic \( M \). Then let \( F_{d,d'} \) be the set of pairs \((f, d)\) such that

1. \((d, f) \in s^{\mathfrak{A}_1}\) for some \( s \in \{r, r^{-1}\}\);
2. \( \mathfrak{A}_1, f \not\sim_{\mathcal{L}C\mathcal{T}^\omega} \mathfrak{A}_1, d' \).

Let \( \mathcal{M} \) be the set of mosaics realised by pairs in \( F_{d,d'} \). It is straightforward to show that \( \mathcal{M} \) is an existential witness for \( M \). We say that \( \mathcal{M} \) is an existential witness defined by \( \mathfrak{A}_1, \mathfrak{A}_2 \) for \( M \). (Note that there can be many existential witnesses for one and the same \( M \) defined by \( \mathfrak{A}_1, \mathfrak{A}_2 \) as \( M \) could be realised in many different ways.)

We are now in a position to give the mosaic elimination procedure. Let \( S_0 \) be the set of all mosaics \( M \) containing only consistent two-way extended types and with \( |M| \leq (K_2 + K_3)^2 \). Clearly \( S_0 \) can be computed in double exponential time in \(|C_1| + |C_2|\).

Let \( S \subseteq S_0 \). We call \( M \in S \) \textit{bad} in \( S \) if it violates at least one of the following conditions:

1. \( A \in t \) iff \( A \in t' \) for all \( A \in \emptyset \) and all \((t, E), (t', E') \in M\);
2. there exists a set \( \mathcal{M} \subseteq S \) with \(|\mathcal{M}| \leq K_1\) such that \( \mathcal{M} \) is a syntactic existential witness for \( M \).
3. \( \exists u. C \in t \) for some \((t, E) \in M \) then there exists a root mosaic \( M' \in S \) such that \( C \in t' \) for some \((t', E') \in M \) such that \( t \) and \( t' \) are \( u \)-equivalent.

We compute a sequence \( S_0, S_1, \ldots \), where we obtain, for \( i \geq 0 \), the set \( S_{i+1} \) from \( S_i \) by eliminating all mosaics that are bad in \( S_i \). Let \( S^* \) be the set of mosaics where the sequence stabilises. Clearly \( S^* \) is obtained from \( S_0 \) in at most double exponential time.

**Lemma 23.** The following conditions are equivalent:

1. \( C_1, C_2 \) are satisfied in \( \mathcal{L}C\mathcal{T}^\omega \)-bisimilar pointed structures.
2. There is a root mosaic \( M \in S^* \) and \((t_1, E_1), (t_2, E_2) \in M \) with \( C_1 \in t_1 \) and \( C_2 \in t_2 \).

**Proof.** Assume Point 1 holds. Assume \( \mathfrak{A}_1, d_1 \sim_{\mathcal{L}C\mathcal{T}^\omega} \mathfrak{A}_2, d_2 \) and \( d_1 \in C_{\mathfrak{A}_1}^1 \) and \( d_2 \in C_{\mathfrak{A}_2}^2 \). Consider the set of all pairs \( p = (M, \mathcal{M}_M) \) such that \( \mathcal{M}_M \) is an existential witness for \( M \) defined by \( \mathfrak{A}_1, \mathfrak{A}_2 \). Next apply Lemma 22 to these pairs and obtain the pairs \((M^p, \mathcal{M}^p_M)\). We
call any $M^p$ down-good (as we have found the relevant witnesses for number restrictions in children) and all mosaics in any $M^u_M$ up-good (as we have found the relevant witnesses for number restrictions in parents). Now define for any up-good $M_1$ and down-good $M_2$ which are up/down profile equivalent a fresh mosaic $M_{M_1, M_2}$ by setting $(t, E_{up}, E_{down}) \in M_{M_1, M_2}$ iff there are $(t, E_{up}, E_{down}') \in M_1$ and $(t, E_{up}', E_{down}) \in M_2$ such that $(t, E_{down}'')$ and $(t, E_{up}')$ are profile equivalent. Let $\mathcal{S}$ be the set of all these mosaics. Then by construction $\mathcal{S} \subseteq \mathcal{S}_0$ and $\mathcal{S}$ does not contain any mosaics that are bad in $\mathcal{S}$. Hence $\mathcal{S} \subseteq \mathcal{S}^*$ and Point 2 follows.

Conversely, assume that Point 2 holds. We construct structures $\mathfrak{A}_1, \mathfrak{A}_2$ from $\mathcal{S}^*$ and the root mosaic $M^* \in \mathcal{S}^*$ with $(t_1', E_1'), (t_2', E_2') \in M^*$ with $C_1 \in t_1'$ and $C_2 \in t_2'$.

Take for every $M \in \mathcal{S}^*$ an existential witness $M_M \subseteq \mathcal{S}^*$ for $M$ and witnessing relations $r_M, r_M^{-1} \subseteq \Delta_M \times \Gamma_M$.

The domain of $\mathfrak{A}_i$ contains all words

$$s = ((t_0, E_0), (t_1, E_1), (t_2, E_2), \ldots, (t_n, E_n), j_n, M_n)$$

such that

- $M_0 \in \mathcal{S}^*$ is a root mosaic;
- $t_0, t_1, \ldots, t_n, t'_n$ are all $u$-equivalent;
- $j_\ell < \omega$, for all $\ell \leq n$;
- $(t_\ell, E_\ell) \in M_\ell$, for all $\ell \leq n$;
- $M_{\ell+1} \in M_M$, for all $\ell < n$.

We next interpret the concept names $A$ by setting for $s$ of the form (3), $s \in A^{\mathfrak{A}_i}$ if $A \in t_n$. We let $(s, s') \in r^{\mathfrak{A}_i}$, for $s$ of the form (3) and

$$s' = ((t'_0, E'_0), (t'_1, E'_1), (t'_2, E'_2), \ldots, (t'_m, E'_m), j'_m, M'_m) \in \text{dom}(\mathfrak{A}_i)$$

if $s$ is an initial part of $s'$ with $m = n + 1$ and

- $((t_n, E_n), j_n, M_n) r_{M_n} ((t'_{n+1}, E'_{n+1}), j'_{n+1}, M'_{n+1})$ or
- $((t'_{n+1}, E'_{n+1}), j'_{n+1}, M'_{n+1}) r_{M_n}^{-1} ((t_n, E_n), j_n, M_n)$.

One can now prove by induction on the construction of $C$ the following

Claim. For all $C \in \text{sub}(C_1, C_2)$ and all $s$ given by (3), $s \in C^{\mathfrak{A}_i}$ iff $C \in t_n$.

We define the $\mathcal{ALC}^u(\varnothing)$-bisimulation $\beta$ by setting for $s$ of the form (3) and $s'$ of the form (4), $s \beta s'$ iff $n = m$ and $M_i = M'_i$ for all $i \leq n$.

Claim. $\beta$ is a $\mathcal{ALC}^u(\varnothing)$-bisimulation between $\mathfrak{A}_1$ and $\mathfrak{A}_2$.

This finishes the proof of Point 1.