Living without Beth and Craig: Definitions and Interpolants in Description and Modal Logics with Nominals and Role Inclusions

ALESSANDRO ARTALE, Free University of Bozen-Bolzano, Italy
JEAN CHRISTOPH JUNG, TU Dortmund University, Germany
ANDREA MAZZULLO, Free University of Bozen-Bolzano, Italy
ANA OZAKI, University of Bergen, Norway
FRANK WOLTER, University of Liverpool, United Kingdom

The Craig interpolation property (CIP) states that an interpolant for an implication exists iff it is valid. The projective Beth definability property (PBDP) states that an explicit definition exists iff a formula stating implicit definability is valid. Thus, the CIP and PBDP reduce potentially hard existence problems to entailment in the underlying logic. Description (and modal) logics with nominals and/or role inclusions do not enjoy the CIP nor the PBDP, but interpolants and explicit definitions have many applications, in particular in concept learning, ontology engineering, and ontology-based data management. In this article, we show that, even without Beth and Craig, the existence of interpolants and explicit definitions is decidable in description logics with nominals and/or role inclusions such as $\mathcal{ALCO}$, $\mathcal{ALCH}$, and $\mathcal{ALCHOI}$ and corresponding hybrid modal logics. However, living without Beth and Craig makes these problems harder than entailment: the existence problems become 2ExpTime-complete in the presence of an ontology or the universal modality, and coNExpTime-complete otherwise. We also analyze explicit definition existence if all symbols (except the one that is defined) are admitted in the definition. In this case, the complexity depends on whether one considers individual or concept names. Finally, we consider the problem of computing interpolants and explicit definitions if they exist and turn the complexity upper bound proof into an algorithm computing them, at least for description logics with role inclusions.

CCS Concepts: • **Theory of computation** → **Description logics**; **Modal and temporal logics;**

Additional Key Words and Phrases: Description logic, modal logic, craig interpolants, beth definability, explicit definitions, computational complexity

ACM Reference format:
Alessandro Artale, Jean Christoph Jung, Andrea Mazzullo, Ana Ozaki, and Frank Wolter. 2023. Living without Beth and Craig: Definitions and Interpolants in Description and Modal Logics with Nominals and Role Inclusions. *ACM Trans. Comput. Logic* 24, 4, Article 34 (October 2023), 51 pages.
https://doi.org/10.1145/3597301

Andrea Mazzullo acknowledges the support of the MUR PNRR project FAIR - Future AI Research (PE00000013) funded by the NextGenerationEU.
Frank Wolter was supported by EPSRC grant EP/S032207/1. Ana Ozaki was supported by NFR grant 316022.
Authors' addresses: A. Artale and A. Mazzullo, Free University of Bozen-Bolzano, Piazza Domenicani, 3, Bolzano 39100, Italy; emails: artale@inf.unibz.it, mazzullo@inf.unibz.it; J. C. Jung, TU Dortmund, Otto-Hahn-Str. 12, D-44227 Dortmund, Germany; email: jean.jung@tu-dortmund.de; A. Ozaki, University of Oslo & University of Bergen, Gaustadalléen 23B, Oslo, 0373, Norway; email: anaoz@uio.no; F. Wolter, University of Liverpool, Department of Computer Science, Ashton Building, Ashton Street, Liverpool L69 3BX, UK; email: wolter@liverpool.ac.uk.

1529-3785/2023/10-ART34 $15.00
https://doi.org/10.1145/3597301

1 INTRODUCTION

The ***Craig Interpolation Property*** (**CIP**) for a logic $\mathcal{L}$ states that an implication $\varphi \Rightarrow \psi$ is valid in $\mathcal{L}$ iff there exists a formula χ in $\mathcal{L}$ using only the common symbols of φ and ψ such that $\varphi \Rightarrow \chi$ and $\chi \Rightarrow \psi$ are both valid in $\mathcal{L}$. The intermediate formula χ is then called an $\mathcal{L}$-interpolant for $\varphi \Rightarrow \psi$ [27]. The CIP is generally regarded as one of the most important and useful properties in formal logic [94], with numerous applications ranging from formal verification [74] and software specification [29] to theory combinations [21, 22, 25, 39] and query reformulation and rewriting in databases [13, 90]. A particularly important consequence of the CIP is the ***projective Beth definability property*** (**PBDP**), which states that a relation is implicitly definable using a signature Σ of symbols iff it is explicitly definable using Σ. If Σ is the set of all symbols distinct from that relation, then we speak of the (non-projective) ***Beth definability property*** (**BDP**) [16].

In this article, we investigate interpolants and explicit definitions in **description logics** (**DLs**), and we also highlight consequences in modal logic. In DLs, one distinguishes essentially two forms of interpolation, both of which are relevant and have their applications. Given an entailment $\mathcal{O} \models C \sqsubseteq D$, that is, C is subsumed by D w.r.t. some background knowledge in the form of a DL ontology $\mathcal{O}$, one might either be interested in an interpolant between the concepts C and D or in an interpolant between $\mathcal{O}$ and the **concept inclusion** (**CI**) $C \sqsubseteq D$. In the first case, the interpolant is a concept, whereas in the second case, the interpolant is an ontology. We refer with *CI-interpolation* to the latter form and call the interpolant a *CI-interpolant*. The CIP for CI-interpolation has been shown to be the most important logical property that ensures the robust behavior of ontology modules and decompositions [54, 55].

In this article, we mostly focus on interpolation (in the former sense of an interpolating concept) and only derive some corollaries for CI-interpolation. Hence, unless stated otherwise, here and in what follows, we speak about interpolating concepts and the corresponding CIP. For explicit definability, one asks for definitions of concepts, possibly with respect to an ontology; these explicit definitions are strongly related to interpolants and as stated above BDP and PBDP follow from the CIP. In DLs, the BDP and PBDP have been used in ontology engineering to extract explicit definitions of concepts and obtain equivalent acyclic terminologies from ontologies [88, 89], they have been investigated in ontology-based data management to equivalently rewrite ontology-mediated queries [34, 35, 85, 91, 92], and they have been proposed to support the construction of alignments between ontologies [47]. Interpolants have been used to study P/NP dichotomies in ontology-based query answering [69].

The CIP, PBDP, and BDP are so powerful because potentially very hard existence questions are reduced to straightforward entailment questions: an interpolant *exists* iff an implication is valid and an explicit definition *exists* iff a straightforward formula stating implicit definability is valid. The existence problems are thus not harder than validity. Many basic DLs such as $\mathcal{ALC}$, $\mathcal{ALCI}$, and $\mathcal{ALCIQ}$ enjoy the CIP and PBDP [89], and consequently, the existence of an interpolant or an explicit definition can be decided in ExpTime simply because entailment checking in these DLs is in ExpTime (and without ontology even in PSpace). Unfortunately, the CIP and the PBDP fail to hold for some important DLs. The most basic examples are the extension $\mathcal{ALCO}$ of $\mathcal{ALC}$ with nominals (concepts of the form $\{a\}$ with a an individual name), the extension $\mathcal{ALCH}$ of $\mathcal{ALC}$ with **role inclusions** (**RIs**) (inclusions $r \sqsubseteq s$ between binary relations/role names r and s), and all standard DLs containing either $\mathcal{ALCO}$ or $\mathcal{ALCH}$ [55, 89]. It follows that for these DLs the existence of interpolants and explicit definitions cannot be reduced (directly) to entailment checking.

The aim of this article is to explore the consequences of the failure of the CIP and PBDP for interpolant and explicit definition existence. To this end, we investigate the complexity of deciding the existence of interpolants and explicit definitions for the set $\mathsf{DL}_{\mathsf{nr}}$ of DLs containing $\mathcal{ALCO}$,

$\mathcal{ALCH}$, and their extensions by inverse roles and/or the universal role. We discuss next two more applications of interpolants and explicit definitions for $\mathcal{ALCO}$ and its extensions.

Data Separability and Concept Learning. We show that interpolants are essentially the same as concepts separating positive and negative data examples in DL **knowledge bases** (**KBs**). Recall that a DL KB is a pair $(\mathcal{O}, \mathcal{D})$ with $\mathcal{O}$ a DL ontology and $\mathcal{D}$ a set of data items of the form $A(a)$ and $r(a, b)$ with a, b individuals, A a concept name, and r a role name. Let $\mathcal{O}$ be an ontology and P and N sets of positively and negatively labeled pairs $(\mathcal{D}, a)$ with $\mathcal{D}$ a set of data items and a an individual in $\mathcal{D}$. Then, the aim of supervised concept learning is to determine a concept C in a signature Σ of relevant symbols such that C separates P and N in the sense that $(\mathcal{O}, \mathcal{D}) \models C(a)$ for all positive examples $(\mathcal{D}, a) \in P$ and $(\mathcal{O}, \mathcal{D}) \models \neg C(a)$ for all negative examples $(\mathcal{D}, a) \in N$.[1] Concept learning has received significant interest over the past 15 years, where the focus has been on developing and analyzing refinement based algorithms for finding separating concepts [30, 62–65, 82, 84]. Prominent concept learning systems include the DL Learner [19, 20], DL-Foil [31] and its extension DL-Focl [83], SPaCEL [93], YinYang [45]. The existence problem for separating concepts has been investigated recently [36, 48–50]. For DLs extending $\mathcal{ALCO}$, we establish a one-to-one correspondence between interpolants and separating concepts, modulo a rather straightforward polynomial time translation. Hence the existence of separating concepts reduces to the existence of interpolants and finding small such concepts or concepts of a certain syntactic shape, as is often useful in supervised learning, also reduces to the same task for interpolants. We emphasize that the presence of nominals in the DL is critical as they are required to encode the individuals used in $\mathcal{D}$ into concepts.

Referring Expressions. The computation of explicit definitions of concept names has been explored in detail since at least [88], see also [7]. Only recently, the focus on defining concept names has been extended to defining individual names, also called referring expression generation in computational linguistics and data management [3, 17, 60]. In fact, it has been convincingly argued that very often in applications, the individual names used in ontologies or datasets are insufficient "to allow humans to figure out what real-world objects they refer to" [18]. A natural way to address this problem is to check for such an individual name a whether there exists a concept C over a set of relevant symbols Σ that provides an explicit definition of $\{a\}$ and present such a concept C to the human user. Observe that one has to work with DLs extending $\mathcal{ALCO}$ to formulate this problem as an explicit definition existence problem.

To conclude, data separation, concept learning, and referring expression generation are challenging research problems which directly benefit from a better understanding of interpolant and explicit definition existence in extensions of $\mathcal{ALCO}$. We now discuss the main results of this article, formulated in an informal way. Precise formulations are given later. Recall that DL_{nr} is the set of DLs $\mathcal{ALCO}$, $\mathcal{ALCH}$, and their extensions with inverse roles and the universal role, and that we assume the presence of a background DL ontology. Our first main result is as follows.

Theorem 1.1. *Let $\mathcal{L} \in \text{DL}_{nr}$. Then $\mathcal{L}$-interpolant existence and $\mathcal{L}$-definition existence are* 2ExpTime*-complete.*

Theorem 1.1 confirms the suspicion that interpolant and definition existence are much harder problems than entailment if one has to live without Beth and Craig. On the positive side, these problems are still decidable. Interestingly, for DLs in DL_{nr} with nominals, the 2ExpTime lower

[1]This condition is called strong separation [36, 49, 50]. A weaker version, called weak separation, only demands that $(\mathcal{O}, \mathcal{D}) \not\models C(a)$ for all negative examples $(\mathcal{D}, a) \in N$. Concept learning systems have been developed for both the weak and the strong notion.

bound for definition existence already holds if one asks for an explicit definition of an individual over the signature containing all symbols distinct from that individual. In contrast, the same problem for concept names is shown to be ExpTime-complete and thus not harder than entailment. Hence, in contrast to concept name definitions, referring expression existence does not become less complex in the non-projective case when all symbols are allowed in definitions.

We next consider the same problems if the background ontology is empty, or, in the case of DLs in $\mathsf{DL}_{\mathsf{nr}}$ without nominals, if the ontology contains only RIs. Observe that if the DL admits the universal role or both nominals and inverse roles, then the ontology can be encoded as a concept using spy points [1], so nothing changes compared to the case with ontologies covered in Theorem 1.1. For the remaining cases, we show the following.

Theorem 1.2. *(1) If $\mathcal{L} \in \{\mathcal{ALCO}, \mathcal{ALCHO}\}$, then for the empty ontology and ontologies containing RIs only, $\mathcal{L}$-interpolant existence and $\mathcal{L}$-definition existence are both* coNExpTime*-complete;*
(2) If $\mathcal{L} \in \{\mathcal{ALCH}, \mathcal{ALCHI}\}$, then for ontologies containing RIs only $\mathcal{L}$-interpolant existence and $\mathcal{L}$-definition existence are both coNExpTime*-complete.*

It follows that without ontology and ontologies containing RIs only interpolant existence and explicit definition existence are still harder than entailment, which is PSpace-complete.

The proofs of Theorems 1.1 and 1.2 can be adapted to also obtain results about CI-interpolation and interpolation in modal logic. Regarding the former, we show that for the DLs $\mathcal{L}$ which extend $\mathcal{ALCO}$ with the universal role or with the universal role and inverse roles the problem of deciding the existence of a CI-interpolant for $\mathcal{O} \models C \sqsubseteq D$ is 2ExpTime-complete. It follows that again failure of the CIP leads to an exponentially harder interpolant existence problem than entailment. We conjecture that the same can be proved for all DLs in $\mathsf{DL}_{\mathsf{nr}}$, but leave a proof for future work.

In modal logic, the CIP and PBDP have been investigated for many years. In fact, the CIP and PBDP of DLs such as $\mathcal{ALC}$ and $\mathcal{ALCI}$ follows rather directly from earlier results on the CIP and PBDP in modal logic [71, 72, 81]. Also, the fact that nominals lead to failure of the CIP and PBDP, and how this could be repaired by adding logical connectives, was first analyzed in depth in the literature on hybrid modal logic, in particular [2, 86]. In our investigation of interpolant existence in modal logic, we first consider basic modal logic with nominals and show that as a direct consequence of Theorem 1.2, the problem of deciding interpolant existence is coNExpTime-complete for the standard local consequence relation. We also show using Theorem 1.1 that if one adds the universal modality, then interpolant existence becomes 2ExpTime-complete. In modal logic, nominals are often considered in tandem with the @-operator, where $@_a\varphi$ states that formula φ holds at the world denoted by nominal a. The resulting language is more expressive than modal logic with nominals and less expressive than modal logic with nominals and the universal modality. We show that for the modal logic with both nominals and the @-operator interpolant existence is still coNExpTime-complete. Our complexity results also hold for the modal language with a single modal operator (and the universal modality, if present).

While the focus in this article is on the decision problem, we also make initial observations regarding the problem of actually computing interpolants or explicit definitions if they exist. More specifically, for DLs in $\mathsf{DL}_{\mathsf{nr}}$ that do not admit nominals, we present a modification of the decision procedure from the proof of Theorem 1.1 that returns in double exponential time the DAG representation of an interpolant (if it exists). This corresponds to interpolants of worst case triple exponential size, which we conjecture to be optimal.

Overview of the Article. In the following Section 2, we discuss further related work. In Sections 3 and 4, we introduce the preliminaries on DLs and Craig interpolation and Beth definability, respectively. In Section 5, we provide model-theoretic characterizations of the definition and interpolation existence problems and formulate our main results in detail. The subsequent four sections are devoted to the proofs of these main results. In more detail, Section 6 provides the upper bound proof for the case with ontologies and Section 7 provides the matching lower bounds. Sections 8 and 9 cover the ontology-free case and the case of ontologies containing only RIs. In Section 10, we investigate the problem of actually computing interpolants and explicit definitions in case they exist, and in Section 11, we draw the connections of our results on DLs to modal logic. Finally, we conclude and point out directions for future work in Section 12.

An appendix available as supplementary material provides a few proofs that were left out of the article. Here we prove, in particular, our main result about the computational complexity of non-projective definition existence of concept names.

2 RELATED WORK

This article is an extended version of our earlier work [5, 6, 51]. We include detailed proofs and additionally discuss the link to concept learning, interpolants between ontologies and CIs, and applications to modal logic.

Related work on Craig interpolation and the Beth definability property has been discussed already in the introduction. We, therefore, focus on work on deciding interpolant and explicit definition existence. These decision problems have only very recently been investigated. A notable exception is linear temporal logic, LTL, for which the CIP fails and for which decidability of interpolant existence has been shown both over finite linear orderings [41, 42] and over the natural numbers [79]. Note that these results are formulated as separability results for formal languages of finite and, respectively, infinite words: given two regular languages R_1 and R_2, does there exist a first-order definable language L separating R_1 and R_2 in the sense that $R_1 \subseteq L$ and $L \cap R_2 = \emptyset$. Neither LTL nor Craig interpolation are mentioned in this work. Using the fact that regular languages are projectively LTL definable and that LTL and first-order logic are equivalent over the natural numbers, it is, however, easy to see that interpolant existence is the same problem as separability of regular languages in first-order logic, modulo the representation of the inputs. We note that this result is just one instance of an ongoing exploration of separation between languages in automata theory. The problem of deciding separation is interesting in this context because obtaining an algorithm for separation yields a far deeper understanding of the class under consideration than just membership [78, 80]. We conjecture that deciding interpolant existence could well play a similar role for understanding fragments of first-order logic.

Indeed, interpolant existence has recently also been studied for the **guarded fragment** (**GF**), the two-variable fragment (FO^2) of FO [52], for Horn DLs extending $\mathcal{EL}$ [33], and for first-order modal logics [61]. While GF is a good generalization of modal and description logic in many respects, it neither enjoys the CIP [44] nor the PBDP [10]. Failure of the CIP for FO^2 was shown using algebraic [26, 76] and model-theoretic techniques [73]. Using techniques that are similar to those introduced, in this article, it is shown in [52] that, in GF, explicit definability and interpolant existence are both 3ExpTime-complete in general, and 2ExpTime-complete if the arity of relation symbols is bounded by a constant $c \geq 3$. In FO^2, explicit definability and interpolant existence are in coN2ExpTime and 2ExpTime-hard [52]. Failure of the CIP and PBDP for first-order modal logics with constant domain is shown in the literature [32, 73]. Both properties also fail for their otherwise well-behaved one-variable and monodic fragments [38]. Kurucz et al. [61], investigate the complexity of interpolant existence for first-order S5 with one variable (and some monodic fragments) and for first-order K with one variable. For S5, explicit definability and interpolant existence turn out to be in coN2ExpTime and 2ExpTime-hard while for K only a non-elementary upper bound is shown. These results confirm that for many logics not enjoying the CIP and PBDP, interpolant and explicit definition existence are harder than entailment.

It turns out that this is not always the case. It is shown by Fortin et al. [33] that extensions of the description logic $\mathcal{EL}$ with any combination of the universal role, nominals, or inverse roles do not enjoy the CIP nor PBDP, but that interpolant existence and explicit definition existence still have the same complexity as entailment (in PTime for those that do not admit inverse roles and ExpTime-complete for those that admit inverse roles). The proofs are rather different from those given in this article, as they make use of the universal/canonical model that only exists for Horn logics.

We note that for logics that do not enjoy the CIP nor PBDP it is also of interest to look for "small" extensions that enjoy the CIP and PBDP and are decidable. For example, the guarded negation fragment of FO is a decidable extension of GF that enjoys the CIP and the PBDP [11, 12, 14, 15]. Also, the two-variable fragment of GF is a decidable extension of $\mathcal{ALCH}$ enjoying both properties [43, 44]. In both cases, the complexity of entailment does not increase for the extension (2ExpTime-complete for the guarded negation fragment and ExpTime-complete for the two-variable fragment of FO). On the other hand, under mild conditions, there is no decidable extension of $\mathcal{ALCO}$ with the universal role nor of modal logic with nominals and the @-operator enjoying the CIP [86].

While the problem of deciding interpolant and explicit definition existence for logics that neither enjoy the CIP nor the PBDP has only been considered rather recently, the problem of computing and deciding the existence of uniform interpolants for logics that do not enjoy the **uniform interpolation property** (**UIP**) has been investigated before. Recall that uniform interpolants generalize Craig interpolants in the sense that a uniform interpolant is an interpolant for a fixed φ and all ψ which are entailed by φ and share with φ a fixed set of symbols. First-order logic enjoys the CIP but not the UIP. Propositional intuitionistic logic, local modal logic, and the modal mu-calculus are examples of expressive logics that enjoy the UIP [28, 77, 95], see [46, 59] for more recent results. In description logic, uniform interpolants of ontologies (extending what we call CI-interpolants in this article) are of particular importance but do not always exist for any standard description logic, including $\mathcal{ALC}$. The complexity of deciding their existence has been investigated in depth [68, 70], their size determined [58, 75], and various approaches to computing them have been developed and implemented [56–58, 96].

3 PRELIMINARIES

We introduce the syntax and semantics of the relevant DLs [8]. Let $\mathsf{N_C}$, $\mathsf{N_R}$, and $\mathsf{N_I}$ be mutually disjoint and countably infinite sets of *concept*, *role*, and *individual names*. A *role* is a role name s, or an *inverse role* s^-, with s a role name and $(s^-)^- = s$. We use u to denote the *universal role*. A *nominal* takes the form $\{a\}$, with a an individual name. An $\mathcal{ALCOI}^u$*-concept* is defined according to the syntax rule

$$C, D ::= \top \mid A \mid \{a\} \mid \neg C \mid C \sqcap D \mid \exists r.C$$

where A ranges over concept names, a ranges over individual names, and r over roles and the universal role. We use $C \sqcup D$ as abbreviation for $\neg(\neg C \sqcap \neg D)$, $C \rightarrow D$ for $\neg C \sqcup D$, $C \leftrightarrow D$ for $(C \rightarrow D) \sqcap (D \rightarrow C)$, and $\forall r.C$ for $\neg\exists r.\neg C$. We use several fragments of $\mathcal{ALCOI}^u$, including $\mathcal{ALCOI}$, obtained by dropping the universal role, $\mathcal{ALCO}^u$, obtained by dropping inverse roles, $\mathcal{ALCO}$, obtained from $\mathcal{ALCO}^u$ by dropping the universal role, and $\mathcal{ALC}$, obtained from $\mathcal{ALCO}$ by dropping nominals. If $\mathcal{L}$ is any of the DLs above, then an $\mathcal{L}$*-concept inclusion ($\mathcal{L}$-CI)* takes the form $C \sqsubseteq D$ with C and D $\mathcal{L}$-concepts. An $\mathcal{L}$*-ontology* is a finite set of $\mathcal{L}$-CIs. We also consider DLs with *RIs*, expressions of the form $r \sqsubseteq s$, where r and s are roles. As usual, the addition of RIs is indicated by adding the letter $\mathcal{H}$ to the name of the DL, where inverse roles occur in RIs only if the DL admits inverse roles. Thus, for example, $\mathcal{ALCH}$*-ontologies* are finite sets of $\mathcal{ALC}$-CIs and RIs not using inverse roles and $\mathcal{ALCHOI}^u$*-ontologies* are finite sets of $\mathcal{ALCOI}^u$-CIs and RIs. In what follows, we use $\mathsf{DL}_{\mathsf{nr}}$ to denote the set of DLs $\mathcal{ALCO}$, $\mathcal{ALCOI}$, $\mathcal{ALCH}$, $\mathcal{ALCHI}$, $\mathcal{ALCHO}$, $\mathcal{ALCHOI}$, and their extensions with the universal role. To simplify notation, we do not drop the letter $\mathcal{H}$ when speaking about the concepts and CIs of a DL with RIs. Thus, for example, we sometimes use the expressions $\mathcal{ALCHO}$-concept and $\mathcal{ALCHO}$-CI to denote $\mathcal{ALCO}$-concepts and CIs, respectively. An *RI-ontology* is an ontology containing RIs only.

The semantics is defined in terms of *interpretations* $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where $\Delta^{\mathcal{I}}$ is a non-empty set, called *domain* of $\mathcal{I}$, and $\cdot^{\mathcal{I}}$ is a function mapping every $A \in \mathsf{N_C}$ to a subset of $\Delta^{\mathcal{I}}$, every $s \in \mathsf{N_R}$ to a subset of $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$, the universal role u to $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$, and every $a \in \mathsf{N_I}$ to an element in $\Delta^{\mathcal{I}}$. Given a role name $s \in \mathsf{N_R}$, we set $(s^-)^{\mathcal{I}} = \{(d,e) \in \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \mid (e,d) \in s^{\mathcal{I}}\}$. Moreover, the *extension* $C^{\mathcal{I}}$ *of an* $\mathcal{L}$*-concept* C *in* $\mathcal{I}$ is defined as follows, where r ranges over roles and the universal role:

$$\begin{aligned}
\top^{\mathcal{I}} &= \Delta^{\mathcal{I}},\\
\{a\}^{\mathcal{I}} &= \{a^{\mathcal{I}}\},\\
\neg C^{\mathcal{I}} &= \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}},\\
(C \sqcap D)^{\mathcal{I}} &= C^{\mathcal{I}} \cap D^{\mathcal{I}},\\
(\exists r.C)^{\mathcal{I}} &= \{d \in \Delta^{\mathcal{I}} \mid \text{there exists } e \in C^{\mathcal{I}} : (d,e) \in r^{\mathcal{I}}\}.
\end{aligned}$$

An interpretation $\mathcal{I}$ *satisfies* an $\mathcal{L}$-CI $C \sqsubseteq D$ if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ and an RI $r \sqsubseteq s$ if $r^{\mathcal{I}} \subseteq s^{\mathcal{I}}$. We say that $\mathcal{I}$ is a *model* of an ontology $\mathcal{O}$ if it satisfies all inclusions in it. We say that an inclusion α follows from an ontology $\mathcal{O}$, in symbols $\mathcal{O} \models \alpha$, if every model of $\mathcal{O}$ satisfies α. We write $\mathcal{O} \models C \equiv D$ if $\mathcal{O} \models C \sqsubseteq D$ and $\mathcal{O} \models D \sqsubseteq C$. We drop $\mathcal{O}$ if it is empty and write $\models C \sqsubseteq D$ for $\emptyset \models C \sqsubseteq D$. A concept C is *satisfiable* w.r.t. an ontology $\mathcal{O}$ if there is a model $\mathcal{I}$ of $\mathcal{O}$ with $C^{\mathcal{I}} \neq \emptyset$. We use a few well known complexity bounds for reasoning in DLs from $\mathsf{DL}_{\mathsf{nr}}$. The $\mathcal{L}$*-subsumption problem* is the problem to decide for any $\mathcal{L}$-ontology $\mathcal{O}$ and $\mathcal{L}$-CI $C \sqsubseteq D$ whether $\mathcal{O} \models C \sqsubseteq D$. The *ontology-free* $\mathcal{L}$*-subsumption problem* and the *RI-ontology* $\mathcal{L}$*-subsumption problem* are the sub-problems of the $\mathcal{L}$-subsumption problem in which the ontology is empty or an RI-ontology, respectively. For any $\mathcal{L} \in \mathsf{DL}_{\mathsf{nr}}$, the $\mathcal{L}$-subsumption problem is ExpTime-complete [1, 9]. If $\mathcal{L}$ admits the universal role or both inverse roles and nominals, then ontologies can be encoded in concepts and so ontology-free $\mathcal{L}$-subsumption and RI-ontology $\mathcal{L}$-subsumption are also ExpTime-complete. In the remaining cases, that is for $\mathcal{ALCO}$, $\mathcal{ALCH}$, $\mathcal{ALCHO}$, and $\mathcal{ALCHI}$, $\mathcal{L}$-subsumption becomes PSpace-complete [1, 9].

A *signature* Σ is a set of concept, role, and individual names, uniformly referred to as *symbols*. Following standard practice, we do not regard the universal role as a symbol but as a logical connective. Thus, the universal role is not contained in any signature. We use $\mathrm{sig}(X)$ to denote the set of symbols used in any syntactic object X such as a concept or an ontology. An $\mathcal{L}(\Sigma)$*-concept* is an $\mathcal{L}$-concept C with $\mathrm{sig}(C) \subseteq \Sigma$. A Σ-role r is a role with $\mathrm{sig}(r) \subseteq \Sigma$. The *size* of a (finite) syntactic object X, denoted $||X||$, is the number of symbols needed to represent it as a word.

We next recall model-theoretic characterizations when elements in interpretations are indistinguishable by concepts formulated in one of the DLs $\mathcal{L}$ introduced above. A *pointed interpretation* is a pair $\mathcal{I}, d$ with $\mathcal{I}$ an interpretation and $d \in \Delta^{\mathcal{I}}$. For pointed interpretations $\mathcal{I}, d$ and $\mathcal{J}, e$ and a signature Σ, we write $\mathcal{I}, d \equiv_{\mathcal{L},\Sigma} \mathcal{J}, e$ and say that $\mathcal{I}, d$ and $\mathcal{J}, e$ are $\mathcal{L}(\Sigma)$*-equivalent* if $d \in C^{\mathcal{I}}$ iff $e \in C^{\mathcal{J}}$, for all $\mathcal{L}(\Sigma)$-concepts C.

As for the model-theoretic characterizations, we start with $\mathcal{ALC}$. Let Σ be a signature. A relation $S \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{J}}$ is an $\mathcal{ALC}(\Sigma)$*-bisimulation* if conditions [AtomC], [Forth], and [Back] from Figure 1 hold, where A and r range over all concept and role names in Σ, respectively. We write $\mathcal{I}, d \sim_{\mathcal{ALC},\Sigma} \mathcal{J}, e$ and call $\mathcal{I}, d$ and $\mathcal{J}, e$ $\mathcal{ALC}(\Sigma)$*-bisimilar* if there exists an $\mathcal{ALC}(\Sigma)$-bisimulation S such that $(d,e) \in S$. For $\mathcal{ALCO}$, we define $\sim_{\mathcal{ALCO},\Sigma}$ analogously, but now demand that, in Figure 1,

[AtomC]	for all $(d, e) \in S$: $d \in A^{\mathcal{I}}$ iff $e \in A^{\mathcal{J}}$
[AtomI]	for all $(d, e) \in S$: $d = a^{\mathcal{I}}$ iff $e = a^{\mathcal{J}}$
[Forth]	if $(d, e) \in S$ and $(d, d') \in r^{\mathcal{I}}$, then there is a e' with $(e, e') \in r^{\mathcal{J}}$ and $(d', e') \in S$.
[Back]	if $(d, e) \in S$ and $(e, e') \in r^{\mathcal{J}}$, then there is a d' with $(d, d') \in r^{\mathcal{I}}$ and $(d', e') \in S$.

Fig. 1. Conditions on $S \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{J}}$.

also condition [AtomI] holds for all individual names $a \in \Sigma$. For languages $\mathcal{L}$ with inverse roles, we demand that, in Figure 1, r additionally ranges over inverse roles. For languages $\mathcal{L}$ with the universal role we extend the respective conditions by demanding that the domain dom(S) and range ran(S) of S contain $\Delta^{\mathcal{I}}$ and $\Delta^{\mathcal{J}}$, respectively. If a DL $\mathcal{L}$ has RIs, then we use $\mathcal{I}, d \sim_{\mathcal{L},\Sigma} \mathcal{J}, e$ to state that $\mathcal{I}, d \sim_{\mathcal{L}',\Sigma} \mathcal{J}, e$ for the fragment $\mathcal{L}'$ of $\mathcal{L}$ without RIs.

The next lemma summarizes the model-theoretic characterizations for all relevant DLs [40, 67]. For the definition of ω-saturated structures, we refer the reader to the literature [24].

Lemma 3.1. *Let $\mathcal{I}, d$ and $\mathcal{J}, e$ be pointed interpretations and ω-saturated. Let $\mathcal{L} \in DL_{nr}$ and Σ a signature. Then*

$$\mathcal{I}, d \equiv_{\mathcal{L},\Sigma} \mathcal{J}, e \quad \textit{iff} \quad \mathcal{I}, d \sim_{\mathcal{L},\Sigma} \mathcal{J}, e.$$

For the "if"-direction, the ω-saturatednesses condition can be dropped.

4 CRAIG INTERPOLATION AND BETH DEFINABILITY

We introduce interpolants and the CIP as well as implicit and explicit definitions and the (projective) Beth definability property ((P)BDP). Recall from the introduction that there are two forms of interpolants, one pertaining to concepts and the other pertaining to CIs. We start the discussion here with the former one, and discuss CI-interpolants later. For concept interpolants, we establish a close link between interpolants and separators of positive and negative data examples, show that logics in $\mathrm{DL}_{\mathrm{nr}}$ do not enjoy the CIP nor PBDP, and determine which DLs in $\mathrm{DL}_{\mathrm{nr}}$ enjoy the BDP.

Let $\mathcal{O}$ be an $\mathcal{L}$-ontology, C_1, C_2 be $\mathcal{L}$-concepts, and let Σ be a signature. Then, an $\mathcal{L}$-concept D is an *$\mathcal{L}(\Sigma)$-interpolant for $C_1 \sqsubseteq C_2$ under $\mathcal{O}$*, if sig$(D) \subseteq \Sigma$, $\mathcal{O} \models C_1 \sqsubseteq D$, and $\mathcal{O} \models D \sqsubseteq C_2$. If $\mathcal{O}$ is empty, then we drop it and call an $\mathcal{L}(\Sigma)$-interpolant for $C_1 \sqsubseteq C_2$ under $\mathcal{O}$ simply an *$\mathcal{L}(\Sigma)$-interpolant for $C_1 \sqsubseteq C_2$*. Observe that Σ is arbitrary in this definition, so it does not follow from $\mathcal{O} \models C_1 \sqsubseteq C_2$ that an $\mathcal{L}(\Sigma)$-interpolant for $C_1 \sqsubseteq C_2$ under $\mathcal{O}$ exists. If $\mathcal{O}$ is empty, then we obtain the standard definition of the CIP by demanding that for $\Sigma = \mathrm{sig}(C_1) \cap \mathrm{sig}(C_2)$ from $\models C_1 \sqsubseteq C_2$ it follows that there exists an $\mathcal{L}(\Sigma)$-interpolant for $C_1 \sqsubseteq C_2$. The obvious generalization of this definition to non-empty ontologies, however, does not work. Consider, for instance, $\mathcal{O} = \{A_1 \sqsubseteq A_2, A_2 \sqsubseteq A_3\}$ and $A_1 \sqsubseteq A_3$. Then for $\Sigma = \mathrm{sig}(A_1) \cap \mathrm{sig}(A_3) = \emptyset$, we have $\mathcal{O} \models A_1 \sqsubseteq A_3$, but there does not exist any $\mathcal{L}(\Sigma)$-interpolant for $A_1 \sqsubseteq A_3$ under $\mathcal{O}$. In fact, to generalize the CIP to non-empty ontologies, one has to split the ontology $\mathcal{O}$ into two. Hence, we adopt here the following definition of the CIP in DLs from the literature [89]. We set sig$(\mathcal{O}, C) = \mathrm{sig}(\mathcal{O}) \cup \mathrm{sig}(C)$, for any ontology $\mathcal{O}$ and concept C.

Definition 4.1. A DL $\mathcal{L}$ *CIP* if for any $\mathcal{L}$-ontologies $\mathcal{O}_1, \mathcal{O}_2$ and $\mathcal{L}$-concepts C_1, C_2 such that $\mathcal{O}_1 \cup \mathcal{O}_2 \models C_1 \sqsubseteq C_2$ there exists an $\mathcal{L}(\Sigma)$-interpolant for $C_1 \sqsubseteq C_2$ under $\mathcal{O}_1 \cup \mathcal{O}_2$, where $\Sigma = \mathrm{sig}(\mathcal{O}_1, C_1) \cap \mathrm{sig}(\mathcal{O}_2, C_2)$. If $\mathcal{O}_1, \mathcal{O}_2$ range over $\mathcal{L}$-ontologies containing RIs only or $\mathcal{O}_1 = \mathcal{O}_2 = \emptyset$, then we say that $\mathcal{L}$ *enjoys the CIP for RI-ontologies* and *the CIP for the empty ontology*, respectively.

Note that the CIP for the empty ontology coincides with the standard definition of the CIP mentioned before. It is shown in [89] that the DLs $\mathcal{ALC}$ and $\mathcal{ALCI}$ and their extensions with qualified number restrictions and the universal role all enjoy the CIP. In contrast, no DL in $\mathrm{DL}_{\mathrm{nr}}$ enjoys the CIP. This is implicitly proved in the literature [89] and is shown in Theorem 4.9 below. The following illustrating example is folklore and shows that this holds even for the empty ontology for logics admitting nominals.

Example 4.2. Consider $C_1 = \{a\} \sqcap \exists r.\{a\}$ and $C_2 = \{b\} \rightarrow \exists r.\{b\}$. Then $\models C_1 \sqsubseteq C_2$ but there does not exist any $\mathcal{ALCHOI}^u(\{r\})$-interpolant for $C_1 \sqsubseteq C_2$. Intuitively, no such interpolant D exists as it would have to be true in exactly the elements x with $r(x,x)$ and no such $\mathcal{ALCHOI}^u(\{r\})$-concept exists. A formal proof is given in Example 5.7 below. ⊣

As discussed in the introduction, the close link between separation of data examples and interpolants is one of our main motivations for studying interpolants for DLs with nominals. We next formalize this link. A *database* $\mathcal{D}$ is a finite set of assertions of the form $A(a)$ and $r(a,b)$ with a, b individuals, A a concept name, and r a role name. By $\mathrm{ind}(\mathcal{D})$, we denote the set of individual names in $\mathcal{D}$. A *knowledge base (KB)* is a pair $\mathcal{K} = (\mathcal{O}, \mathcal{D})$ consisting of an ontology $\mathcal{O}$ and database $\mathcal{D}$. An interpretation $\mathcal{I}$ is a *model of* $\mathcal{K}$ if it is a model of $\mathcal{O}$, $a^{\mathcal{I}} \in A^{\mathcal{I}}$ for all $A(a) \in \mathcal{D}$, and $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$ for all $r(a,b) \in \mathcal{D}$. An assertion $C(a)$ with C a concept and a an individual *follows from* $\mathcal{K}$, in symbols $\mathcal{K} \models C(a)$, if every model $\mathcal{I}$ of $\mathcal{K}$ satisfies $a^{\mathcal{I}} \in C^{\mathcal{I}}$. A *labeled dataset* consists of two sets, P and N, of positive and negative examples each containing pairs $(\mathcal{D}, a)$ with $a \in \mathrm{ind}(\mathcal{D})$. Let $\mathcal{O}$ be an ontology. An $\mathcal{L}(\Sigma)$-*separator* for $\mathcal{O}, P, N$ is an $\mathcal{L}(\Sigma)$-concept C such that $(\mathcal{O}, \mathcal{D}) \models C(a)$ for all $(\mathcal{D}, a) \in P$ and $(\mathcal{O}, \mathcal{D}) \models \neg C(a)$ for all $(\mathcal{D}, a) \in N$. The following result establishes a one-to-one correspondence between interpolants and separators, modulo straightforward polynomial time reductions. Note that we do not require the frequent assumption that the database is uniform across the examples in the sense that $\mathcal{D} = \mathcal{D}'$ for all $(\mathcal{D}, a), (\mathcal{D}', a') \in P \cup N$ [50].

Theorem 4.3. *Let $\mathcal{L} \in DL_{nr}$ admit nominals. Then one can construct for any ontology $\mathcal{O}$, labeled datasets P, N, and signature Σ with $\Sigma \cap \{a \mid (\mathcal{D}, a) \in P \cup N\} = \emptyset$ in polynomial time $\mathcal{L}$-ontologies $\mathcal{O}_1$, $\mathcal{O}_2$ and $\mathcal{L}$-concepts C_1, C_2 such that $\Sigma = sig(\mathcal{O}_1, C_1) \cap sig(\mathcal{O}_2, C_2)$ and the following conditions are equivalent for all $\mathcal{L}$-concepts C:*

(1) C is an $\mathcal{L}(\Sigma)$-separator for $\mathcal{O}, P, N$;
(2) C is an $\mathcal{L}(\Sigma)$-interpolant for $C_1 \sqsubseteq C_2$ under $\mathcal{O}_1 \cup \mathcal{O}_2$.

Conversely, assume that $\mathcal{L}$-ontologies $\mathcal{O}_1, \mathcal{O}_2$, and $\mathcal{L}$-concepts C_1, C_2 are given. Then one can construct in polynomial time an ontology $\mathcal{O}$ and labeled datasets P, N such that Conditions (1) and (2) are equivalent for $\Sigma = sig(\mathcal{O}_1, C_1) \cap sig(\mathcal{O}_2, C_2)$.

Proof. Assume $\mathcal{O}, P, N$, and Σ are given. Let $P = \{(\mathcal{D}_1, a_1), \ldots, (\mathcal{D}_n, a_n)\}$ and $N = \{(\mathcal{D}_{n+1}, a_{n+1}), \ldots, (\mathcal{D}_{n+m}, a_{n+m})\}$. If $\mathcal{L} \in \mathrm{DL}_{\mathrm{nr}}$ admits nominals and the universal role, then a pair $(\mathcal{D}, a)$ can be represented using the $\mathcal{L}$-concept $C_{\mathcal{D},a} = \{a\} \sqcap \exists u.C_{\mathcal{D}}$, where $C_{\mathcal{D}}$ is the conjunction of all $\{b\} \sqcap A$ with $A(b) \in \mathcal{D}$ and $\{b\} \sqcap \exists r.\{c\}$ with $r(b,c) \in \mathcal{D}$. Pick for any symbol X not in Σ a fresh copy X'. Let $\mathcal{O}_1 = \mathcal{O}$ and obtain $\mathcal{O}_2$ from $\mathcal{O}$ by replacing all symbols not in Σ by their copies. Let $C_1 = C_{\mathcal{D}_1, a_1} \sqcup \cdots \sqcup C_{\mathcal{D}_n, a_n}$ and obtain C_2 from $\neg(C_{\mathcal{D}_{n+1}, a_{n+1}} \sqcup \cdots \sqcup C_{\mathcal{D}_{n+m}, a_{n+m}})$ by replacing all symbols not in Σ by their copies. If $\mathcal{L}$ admits the universal role, then $\mathcal{O}_1, \mathcal{O}_2$ and C_1, C_2 are as required. Otherwise, replace the universal role in any $C_{\mathcal{D}_i, a_i}$ by fresh role names not in Σ. The resulting C_1, C_2 are still as required.

Conversely, assume that $\mathcal{O}_1, \mathcal{O}_2$ and C_1, C_2 are given. Introduce fresh individual names a, b and fresh concept names A, B and let $P = \{(\{A(a)\}, a)\}$, $N = \{(\{B(b)\}, b)\}$ and $\mathcal{O} = \mathcal{O}_1 \cup \mathcal{O}_2 \cup \{A \sqsubseteq C_1, B \sqsubseteq \neg C_2\}$. Then $\mathcal{O}, P, N$ is as required. □

We next introduce the relevant definability notions. Let $\mathcal{O}$ be an ontology and C, C_0 be concepts. Let Σ be a signature. An $\mathcal{L}(\Sigma)$-concept D is an *explicit $\mathcal{L}(\Sigma)$-definition of C_0 under $\mathcal{O}$ and C* if

$\mathcal{O} \models C \sqsubseteq (C_0 \leftrightarrow D)$. We call C_0 *explicitly* $\mathcal{L}(\Sigma)$*-definable under* $\mathcal{O}$ *and* C if there is an explicit $\mathcal{L}(\Sigma)$-definition of C_0 under $\mathcal{O}$ and C. If $C = \top$ or $\mathcal{O}$ is empty, then we drop $\mathcal{O}$ and C, respectively. For instance, an $\mathcal{L}(\Sigma)$-concept D is an explicit $\mathcal{L}(\Sigma)$-definition of C_0 under $\mathcal{O}$ if $\mathcal{O} \models C_0 \equiv D$. The following example illustrates the link between explicit definitions of nominals and referring expressions discussed in the introduction and also indicates that often one can single out an individual from a set of individuals using an explicit definition without being able to provide an "absolute" explicit definition of that individual.

Example 4.4. Let L be an abbreviation for the $\mathcal{ALCO}$-concept

$$\{\mathcal{ALC}\} \sqcup \{\mathcal{ALCO}\} \sqcup \{\mathcal{ALCO}^u\} \sqcup \{\mathrm{ML}\} \sqcup \{\mathrm{ML}_n\} \sqcup \{\mathrm{ML}_n^u\},$$

where ML, ML_n, and ML_n^u are modal logics introduced below in Section 11. Let $\mathcal{O}$ be the ontology consisting of the following CIs:

$$\begin{aligned}
\{\mathcal{ALC}\} \sqcup \{\mathcal{ALCO}\} \sqcup \{\mathcal{ALCO}^u\} &\sqsubseteq \exists \mathsf{hasOperator}.\mathsf{DLOperator} \sqcap \neg\exists \mathsf{hasOperator}.\mathsf{MLOperator},\\
\{\mathrm{ML}\} \sqcup \{\mathrm{ML}_n\} \sqcup \{\mathrm{ML}_n^u\} &\sqsubseteq \exists \mathsf{hasOperator}.\mathsf{MLOperator} \sqcap \neg\exists \mathsf{hasOperator}.\mathsf{DLOperator},\\
\{\mathcal{ALC}\} \sqcup \{\mathrm{ML}\} &\sqsubseteq \mathsf{EnjoysCIP},\\
\mathsf{EnjoysCIP} &\sqsubseteq \mathsf{EnjoysPBDP},\\
\mathsf{EnjoysPBDP} &\sqsubseteq \mathsf{EnjoysBDP},\\
\{\mathcal{ALCO}^u\} \sqcup \{\mathrm{ML}_n^u\} &\sqsubseteq \mathsf{EnjoysBDP} \sqcap \neg\mathsf{EnjoysPBDP},\\
\{\mathcal{ALCO}\} \sqcup \{\mathrm{ML}_n\} &\sqsubseteq \neg\mathsf{EnjoysBDP}.
\end{aligned}$$

Then $\mathcal{O} \models \mathsf{L} \sqsubseteq (\{\mathcal{ALC}\} \leftrightarrow \mathsf{EnjoysCIP} \sqcap \exists\mathsf{hasOperator}.\mathsf{DLOperator})$. Hence, $\{\mathcal{ALC}\}$ is explicitly $\mathcal{ALCO}(\Sigma_0)$-definable under $\mathcal{O}$ and L, with $\Sigma_0 = \{\mathsf{EnjoysCIP}, \mathsf{DLOperator}, \mathsf{hasOperator}\}$. However, $\{\mathcal{ALC}\}$ is not explicitly $\mathcal{ALCO}(\Sigma)$-definable under $\mathcal{O}$, for any signature Σ with $\mathcal{ALC} \notin \Sigma$, since there does not exist an $\mathcal{ALCO}(\Sigma)$-concept C such that $\mathcal{O} \models \{\mathcal{ALC}\} \equiv C$. ⊣

We next define when a concept is implicitly definable. For a signature Σ, the Σ*-reduct* $\mathcal{I}_{|\Sigma}$ of an interpretation $\mathcal{I}$ coincides with $\mathcal{I}$ except that no non-Σ symbol is interpreted in $\mathcal{I}_{|\Sigma}$. A concept C_0 is called *implicitly* Σ*-definable under* $\mathcal{O}$ *and* C if the Σ-reduct of any pointed model $\mathcal{I}, d$ with $\mathcal{I}$ a model of $\mathcal{O}$ and $d \in C^{\mathcal{I}}$ determines whether $d \in C_0^{\mathcal{I}}$. More formally, C_0 is implicitly Σ-definable under $\mathcal{O}$ and C if the following holds for all models $\mathcal{I}$ and $\mathcal{J}$ of $\mathcal{O}$ and $d \in \Delta^{\mathcal{I}} = \Delta^{\mathcal{J}}$: if $\mathcal{I}_{|\Sigma} = \mathcal{J}_{|\Sigma}$ and $d \in C^{\mathcal{I}}$, then $d \in C_0^{\mathcal{I}}$ iff $d \in C_0^{\mathcal{J}}$. If $C = \top$, then we drop C and say that C_0 is implicitly Σ-definable under $\mathcal{O}$. To illustrate, observe that in Example 4.4, $\{\mathcal{ALC}\}$ is not implicitly Σ-definable, for any Σ such that $\mathcal{ALC} \notin \Sigma$ under $\mathcal{O}$. Implicit definability can be reformulated as a standard reasoning problem as follows: a concept C_0 is implicitly Σ-definable under $\mathcal{O}$ and C iff

$$\mathcal{O} \cup \mathcal{O}_\Sigma \models C \sqcap C_0 \sqsubseteq C_\Sigma \rightarrow C_{0\Sigma}, \tag{1}$$

where $\mathcal{O}_\Sigma$, C_Σ, and $C_{0\Sigma}$ are obtained from $\mathcal{O}$, C and, respectively, C_0, by replacing every non-Σ symbol uniformly by a fresh symbol. If a concept is explicitly $\mathcal{L}(\Sigma)$-definable under $\mathcal{O}$ and C, then it is implicitly Σ-definable under $\mathcal{O}$ and C, for any language $\mathcal{L}$. A logic enjoys the PBDP if the converse implication holds as well.

Definition 4.5. A DL $\mathcal{L}$ *enjoys the PBDP*, if for any $\mathcal{L}$-ontology $\mathcal{O}$, $\mathcal{L}$-concepts C and C_0, and signature $\Sigma \subseteq \mathrm{sig}(\mathcal{O}, C)$ the following holds: if C_0 is implicitly Σ-definable under $\mathcal{O}$ and C, then C_0 is explicitly $\mathcal{L}(\Sigma)$-definable under $\mathcal{O}$ and C. If $\mathcal{O}$ ranges over $\mathcal{L}$-ontologies containing RIs only or $\mathcal{O} = \emptyset$, then we say that $\mathcal{L}$ *enjoys the PBDP for RI-ontologies* and *the PBDP for the empty ontology*, respectively.

The DLs $\mathcal{ALC}$, $\mathcal{ALCI}$, and their extensions with qualified number restrictions and the universal role all enjoy the PBDP [89]. The following example shows that, in contrast, $\mathcal{ALCH}$ does not.

Example 4.6. Consider $\mathcal{O} = \{r \sqsubseteq r_1, r \sqsubseteq r_2\}$ and let

$$C = \Big((\neg\exists r.\top \sqcap \exists r_1.A) \rightarrow \forall r_2.\neg A\Big) \sqcap \Big((\neg\exists r.\top \sqcap \exists r_1.\neg A) \rightarrow \forall r_2.A\Big).$$

Let $\Sigma = \{r_1, r_2\}$ and $C_0 = \exists r.\top$. Then, the concept $D = \exists r_1 \cap r_2.\top$ is an explicit definition of C_0 under $\mathcal{O}$ and C in the extension of $\mathcal{ALCH}$ with role intersection (the semantics of $r_1 \cap r_2$ is defined in the obvious way). Hence C_0 is implicitly Σ-definable under $\mathcal{O}$ and C. There does not exist an explicit $\mathcal{ALCH}(\Sigma)$-definition of C_0 under $\mathcal{O}$ and C, however. Intuitively, the reason is that role intersection cannot be expressed in $\mathcal{ALCH}$ (see Example 5.9 below for a proof).

Note that an example without "background concept" C can be obtained by taking the ontology

$$\mathcal{O}' = \{\ r \sqsubseteq r_1, \quad r \sqsubseteq r_2, \quad \neg\exists r.\top \sqcap \exists r_1.A \sqsubseteq \forall r_2.\neg A, \quad \neg\exists r.\top \sqcap \exists r_1.\neg A \sqsubseteq \forall r_2.A\ \}$$

and asking for an explicit $\mathcal{ALCH}(\{r_1, r_2\})$-definition of $\exists r.\top$ under $\mathcal{O}'$. ⊣

It is known that the CIP and PBDP are tightly linked [89]. We state the inclusion for logics in $\mathrm{DL}_{\mathrm{nr}}$ only, but the proof shows that it holds under rather mild conditions.

Lemma 4.7. *If $\mathcal{L} \in DL_{nr}$ enjoys the CIP, then $\mathcal{L}$ enjoys the PBDP.*

Proof. Assume that an $\mathcal{L}$-concept C_0 is implicitly Σ-definable under an $\mathcal{L}$-ontology $\mathcal{O}$ and $\mathcal{L}$-concept C, for some signature Σ. Then Equation (1) holds. Take an $\mathcal{L}(\Sigma)$-interpolant D for $C \sqcap C_0 \sqsubseteq C_\Sigma \rightarrow C_{0\Sigma}$ under $\mathcal{O} \cup \mathcal{O}_\Sigma$. Then D is an explicit $\mathcal{L}(\Sigma)$-definition of C_0 under $\mathcal{O}$ and C. □

An important special case of explicit definability is the explicit definability of a concept name A from $\mathrm{sig}(\mathcal{O}, C) \setminus \{A\}$ under an ontology $\mathcal{O}$ and concept C. For this case, we also consider the following *non-projective version* of the Beth definability property.

Definition 4.8. A DL $\mathcal{L}$ *enjoys the BDP* if for any $\mathcal{L}$-ontology $\mathcal{O}$, concept C, and any concept name A the following holds: if A is implicitly $(\mathrm{sig}(\mathcal{O}, C) \setminus \{A\})$-definable

under $\mathcal{O}$ and C, then A is explicitly $\mathcal{L}(\mathrm{sig}(\mathcal{O}, C) \setminus \{A\})$-definable under $\mathcal{O}$ and C. If $\mathcal{O}$ ranges over $\mathcal{L}$-ontologies containing RIs only or $\mathcal{O} = \emptyset$, then we say that $\mathcal{L}$ *enjoys the BDP for RI-ontologies* and *the BDP for the empty ontology*, respectively.

Clearly, the PBDP entails the BDP, but we will see below that the converse direction does not always hold. In fact, the following theorem states that no DL in $\mathrm{DL}_{\mathrm{nr}}$ enjoys the CIP or PBDP, but that quite a few DLs in $\mathrm{DL}_{\mathrm{nr}}$ enjoy the BDP. Moreover, all DLs in $\mathrm{DL}_{\mathrm{nr}}$ enjoy the BDP for RI-ontologies and for the empty ontology.

As mentioned before, the theorem is mostly folklore and therefore proved in the appendix.

Theorem 4.9. *The following statements hold.*

(1) No $\mathcal{L} \in DL_{nr}$ enjoys the CIP nor the PBDP. The CIP and PBDP also do not hold for RI-ontologies and, if $\mathcal{L}$ admits nominals, the empty ontology.
(2) All $\mathcal{L} \in DL_{nr} \setminus \{\mathcal{ALCO}, \mathcal{ALCHO}\}$ enjoy the BDP. $\mathcal{ALCO}$ and $\mathcal{ALCHO}$ do not enjoy the BDP.
(3) All $\mathcal{L} \in DL_{nr}$ enjoy the BDP for RI-ontologies and the BDP for the empty ontology.

We have seen that all $\mathcal{L} \in \mathrm{DL}_{\mathrm{nr}} \setminus \{\mathcal{ALCO}, \mathcal{ALCHO}\}$ enjoy the BDP. One might be tempted to conjecture that this holds as well if concept names are replaced by nominals; that is to say, a nominal $\{a\}$ that is implicitly definable using symbols distinct from a is explicitly definable using

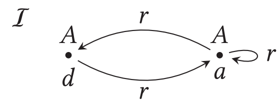

Fig. 2. Failure of the PBDP for $\mathcal{L} \in \mathsf{DL}_{\mathsf{nr}}$ with nominals.

symbols distinct from a. Rather surprisingly, the following example shows that this is not the case for any DL in $\mathsf{DL}_{\mathsf{nr}}$ with nominals (for DLs without nominals, this notion is clearly meaningless).

Example 4.10. Let $\mathcal{L} \in \mathsf{DL}_{\mathsf{nr}}$ admit nominals and assume that

$$\begin{aligned} \mathcal{O} = \{&\{a\} \sqsubseteq \exists r.\{a\}, \\ &A \sqcap \neg\{a\} \sqsubseteq \forall r.(\neg\{a\} \rightarrow \neg A), \\ &\neg A \sqcap \neg\{a\} \sqsubseteq \forall r.(\neg\{a\} \rightarrow A)\}. \end{aligned}$$

Thus, $\mathcal{O}$ implies that a is r-reflexive and that no element distinct from a is r-reflexive. Let $\Sigma = \{r, A\}$. Then $\{a\}$ is implicitly Σ-definable under $\mathcal{O}$ since we have the following explicit definition in first-order logic:

$$\mathcal{O} \models \forall x((x = a) \leftrightarrow r(x, x)),$$

but one can show that $\{a\}$ is not explicitly $\mathcal{L}(\Sigma)$-definable under $\mathcal{O}$ for any $\mathcal{L} \in \mathsf{DL}_{\mathsf{nr}}$ with nominals. Indeed, the interpretation $\mathcal{I}$ in Figure 2 +(where $a^{\mathcal{I}} = a$) is a model of $\mathcal{O}$ and the relation $S = \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ is an $\mathcal{L}(\Sigma)$-bisimulation on $\mathcal{I}$. Thus, Lemma 3.1 implies $\mathcal{I}, a^{\mathcal{I}} \equiv_{\mathcal{L},\Sigma} \mathcal{I}, d$ and there is no explicit $\mathcal{L}(\Sigma)$-definition for $\{a\}$ under $\mathcal{O}$, as any such definition would apply to d as well. ⊣

The CIP defined above is concerned with interpolating concepts. In the context of modular ontologies and forgetting there is also an interest in interpolating CIs [55]. For simplicity, we only consider DLs without RIs. Let $\mathcal{L}$ not admit RIs and let $\mathcal{O}$ and $\mathcal{O}'$ be $\mathcal{L}$-ontologies. We write $\mathcal{O} \models \mathcal{O}'$ if $\mathcal{O} \models \alpha$ for all $\alpha \in \mathcal{O}'$. Then an $\mathcal{L}$-ontology $\mathcal{O}''$ is called an *$\mathcal{L}$-CI interpolant for $\mathcal{O}$ and $\mathcal{O}'$* if $\mathrm{sig}(\mathcal{O}'') \subseteq \mathrm{sig}(\mathcal{O}) \cap \mathrm{sig}(\mathcal{O}')$, $\mathcal{O} \models \mathcal{O}''$, and $\mathcal{O}'' \models \mathcal{O}'$. If the particular language $\mathcal{L}$ is clear from the context or not important we drop it and call $\mathcal{L}$-CI interpolants simply CI-interpolants.

Definition 4.11. Let $\mathcal{L}$ be a DL that does not admit RIs. Then $\mathcal{L}$ has the *CI-interpolation property* if for all $\mathcal{L}$-ontologies $\mathcal{O}$ and $\mathcal{O}'$ such that $\mathcal{O} \models \mathcal{O}'$ there exists an $\mathcal{L}$-CI-interpolant for $\mathcal{O}$ and $\mathcal{O}'$.

Observe that for any $\mathcal{L} \in \mathsf{DL}_{\mathsf{nr}}$ that does not admit RIs and $\mathcal{L}$-ontology $\mathcal{O}$ one can construct in linear time an $\mathcal{L}$-concept D such that $\mathcal{O}$ and $\{\top \sqsubseteq D\}$ are equivalent in the sense that $\mathcal{O} \models \top \sqsubseteq D$ and $\{\top \sqsubseteq D\} \models \mathcal{O}$. Hence $\mathcal{L}$ has the CI-interpolation property if for all $\mathcal{L}$-ontologies $\mathcal{O}$ and $\mathcal{L}$-CIs $C \sqsubseteq D$ such that $\mathcal{O} \models C \sqsubseteq D$ there exists an $\mathcal{L}$-CI-interpolant for $\mathcal{O}$ and $\{C \sqsubseteq D\}$. It is known that $\mathcal{ALC}$ and its extensions with inverse roles, qualified number restrictions, and the universal role enjoy the CI-interpolation property [55]. The following example shows that no DL in $\mathsf{DL}_{\mathsf{nr}}$ that does not admit RIs enjoys the CI-interpolation property.

Example 4.12. We modify the ontology given in Example 4.10. Let

$$\mathcal{O}' = \{A \sqsubseteq \forall r.\neg A, \neg A \sqsubseteq \forall r.A\}.$$

We have that $\mathcal{O}' \models \{a\} \sqsubseteq \neg\exists r.\{a\}$, but there does not exist an $\mathcal{ALCOI}^u$-CI-interpolant for $\mathcal{O}'$ and $\{a\} \sqsubseteq \neg\exists r.\{a\}$, since one cannot express using an $\mathcal{ALCOI}^u(\{r\})$-CI that $\forall x\ \neg r(x, x)$. Indeed, assume for a proof by contradiction that there exists an $\mathcal{ALCOI}^u$-CI-interpolant $\mathcal{O}''$ for $\mathcal{O}'$ and $\{\{a\} \sqsubseteq \neg\exists r.\{a\}\}$. Consider the interpretations $\mathcal{I}_1, \mathcal{I}_2$ in Figure 3, where $\mathcal{I}_1 \models \mathcal{O}'$, $a^{\mathcal{I}_2} \in (\{a\} \sqcap \exists r.\{a\})^{\mathcal{I}_2}$, and $\mathcal{I}_1, d \sim_{\mathcal{ALCOI}^u,\{r\}} \mathcal{I}_2, a^{\mathcal{I}_2}$. Since $\mathcal{I}_1 \models \mathcal{O}'$, we have that $\mathcal{I}_1 \models \mathcal{O}''$, where $\mathcal{O}''$ can be assumed to be of the form $\{\top \sqsubseteq D\}$, with $\mathrm{sig}(D) \subseteq \{r\}$. Thus, $d \in (\forall u.D)^{\mathcal{I}_1}$ and, from $\mathcal{I}_1, d \sim_{\mathcal{ALCOI}^u,\{r\}} \mathcal{I}_2, a^{\mathcal{I}_2}$, we obtain by Lemma 3.1 that $a^{\mathcal{I}_2} \in (\forall u.D)^{\mathcal{I}_2}$. This implies $\mathcal{I}_2 \models \mathcal{O}''$, and hence $\mathcal{I}_2 \models \{a\} \sqsubseteq \neg\exists r.\{a\}$, contrary to the assumption that $a^{\mathcal{I}_2} \in (\{a\} \sqcap \exists r.\{a\})^{\mathcal{I}_2}$. ⊣

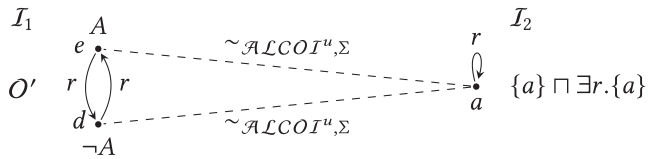

Fig. 3. Failure of the CI-interpolation property for $\mathcal{L} \in \mathrm{DL}_{\mathrm{nr}}$ without RIs.

In this article, we focus on interpolating concepts and not CIs. The main reasons are that the corresponding notion of an explicit definition of an ontology appears to be less useful than definitions of concepts and nominals and that while the CI-interpolation property is crucial for robust decompositions of ontologies and for robust forgetting [55], checking the existence of an interpolant or computing it for concrete ontologies and CIs appears to not have found any applications yet. Regarding the first point, observe that CI-interpolants correspond to the following notion of an explicit CI-definition of an ontology. Let Σ be a signature and $\mathcal{O}$ and $\mathcal{O}'$ ontologies. Then an $\mathcal{L}(\Sigma)$-ontology $\mathcal{O}''$ is called an *explicit $\mathcal{L}(\Sigma)$-definition of $\mathcal{O}'$ under $\mathcal{O}$* if $\mathcal{O} \cup \mathcal{O}' \models \mathcal{O}''$ and $\mathcal{O} \cup \mathcal{O}'' \models \mathcal{O}'$. In particular, if $\mathcal{O}$ is empty, then one asks for an ontology using symbols in Σ only that is equivalent to $\mathcal{O}$. While the existence of such ontologies is an interesting theoretical question that could well have applications in the future, investigating this problem is beyond the focus of this article. In what follows, we only consider aspects of CI-interpolants that are closely related to concept interpolants, leaving their detailed investigation for future work.

5 MAIN RESULTS

The failure of CIP and (P)BDP reported in Theorem 4.9 imply that interpolant existence and projective and non-projective definition existence cannot be directly polynomially reduced to subsumption checking. This motivates studying the respective decision problems of interpolant existence and projective and non-projective definition existence. In this section, we introduce the decision problems, formulate model-theoretic characterizations of the problems that play a fundamental role in our proofs, and we formulate the main results.

We start with interpolant existence for which we take the definition used in the formulation of the CIP.

Definition 5.1. Let $\mathcal{L}$ be a DL. Then *$\mathcal{L}$-interpolant existence* is the problem to decide for any $\mathcal{L}$-ontologies $\mathcal{O}_1, \mathcal{O}_2$ and $\mathcal{L}$-concepts C_1, C_2, whether there exists an $\mathcal{L}(\Sigma)$-interpolant for $C_1 \sqsubseteq C_2$ under $\mathcal{O}_1 \cup \mathcal{O}_2$, where $\Sigma = \mathrm{sig}(\mathcal{O}_1, C_1) \cap \mathrm{sig}(\mathcal{O}_2, C_2)$.

In our proofs, we actually focus on a more general version of interpolant existence which has been discussed in the previous section and in which we do not split $\mathcal{O}$ into two ontologies and in which Σ is arbitrary.

Definition 5.2. Let $\mathcal{L}$ be a DL. Then *generalized $\mathcal{L}$-interpolant existence* is the problem to decide for any $\mathcal{L}$-ontology $\mathcal{O}$, $\mathcal{L}$-concepts C_1, C_2, and signature Σ whether there exists an $\mathcal{L}(\Sigma)$-interpolant for $C_1 \sqsubseteq C_2$ under $\mathcal{O}$.

We also consider (generalized) $\mathcal{L}$-interpolant existence with empty ontologies, called *ontology-free (generalized) $\mathcal{L}$-interpolant existence*, and with RI-ontologies, called *RI-ontology (generalized) $\mathcal{L}$-interpolant existence*, both defined in the obvious way. Observe that in the ontology-free case there is no difference between generalized interpolant existence and interpolant existence. In fact, also with ontologies generalized interpolant existence and interpolant existence are interreducible.

Lemma 5.3. *Let $\mathcal{L} \in DL_{nr}$. There are mutual polynomial time reductions between generalized $\mathcal{L}$-interpolant existence and $\mathcal{L}$-interpolant existence.*

Proof. The reduction from $\mathcal{L}$-interpolant existence to generalized $\mathcal{L}$-interpolant existence is trivial: for input O_1, O_2, C_1, C_2 to $\mathcal{L}$-interpolant existence, set $O = O_1 \cup O_2$ and $\Sigma = \mathrm{sig}(O_1, C_1) \cap \mathrm{sig}(O_2, C_2)$.

For the converse reduction from generalized interpolant existence to interpolant existence, assume that an $\mathcal{L}$-ontology O, $\mathcal{L}$-concepts C_1, C_2, and Σ are given. Then, there exists an $\mathcal{L}(\Sigma)$-interpolant for $C_1 \sqsubseteq C_2$ under O iff there exists an $\mathcal{L}(\Sigma)$-interpolant for $C_1 \sqsubseteq C_{2\Sigma}$ under $O \cup O_\Sigma$, where O_Σ and $C_{2\Sigma}$ are obtained from O and C_2 by replacing every non-Σ symbol uniformly by a fresh symbol. The latter is an instance of $\mathcal{L}$-interpolant existence. □

Note that the reduction above works for all standard DLs including $\mathcal{ALC}$. Recall that interpolant existence reduces to checking $O_1 \cup O_2 \models C_1 \sqsubseteq C_2$ for logics with the CIP. Hence, for DLs which enjoy the CIP such as $\mathcal{ALC}$, interpolant existence and generalized interpolant existence are ExpTime-complete and ontology-free interpolant existence and generalized ontology-free interpolant existence are PSpace-complete.

We next introduce the relevant definition existence problems.

Definition 5.4. Let $\mathcal{L}$ be a DL. *Projective $\mathcal{L}$-definition existence* is the problem to decide for any $\mathcal{L}$-ontology O, $\mathcal{L}$-concepts C and C_0, and signature Σ, whether there exists an explicit $\mathcal{L}(\Sigma)$-definition of C_0 under O and C.

(*Non-projective*) *$\mathcal{L}$-definition existence of concept names (nominals)* is the sub-problem where C_0 ranges only over concept names A (nominals $\{a\}$) and $\Sigma = \mathrm{sig}(O, C) \setminus \{A\}$ (and $\Sigma = \mathrm{sig}(O, C) \setminus \{a\}$, respectively).

We also consider the (projective) $\mathcal{L}$-definition existence problems with empty ontologies, called *ontology-free (projective) $\mathcal{L}$-definition existence*, and with RI-ontologies, called *RI-ontology (projective) $\mathcal{L}$-definition existence*, both defined in the obvious way. Similar to the case of interpolant existence, definition existence reduces to checking implicit definability for logics with the PBDP. We provide model-theoretic characterizations for the non-existence of generalized interpolants and explicit definitions in terms of bisimulations.

Definition 5.5 (Joint Consistency). Let $\mathcal{L} \in \mathrm{DL}_{\mathrm{nr}}$. Let O be an $\mathcal{L}$-ontology, C_1, C_2 be $\mathcal{L}$-concepts, and Σ a signature. Then C_1 and C_2 are called *jointly consistent under O modulo $\mathcal{L}(\Sigma)$-bisimulations* if there exist pointed interpretations $\mathcal{I}_1, d_1$ and $\mathcal{I}_2, d_2$ such that $\mathcal{I}_i$ is a model of O, $d_i \in C_i^{\mathcal{I}_i}$, for $i = 1, 2$, and $\mathcal{I}_1, d_1 \sim_{\mathcal{L},\Sigma} \mathcal{I}_2, d_2$.

The associated decision problem, *joint consistency modulo $\mathcal{L}$-bisimulations*, is defined in the expected way. The following result characterizes the existence of interpolants using joint consistency modulo $\mathcal{L}(\Sigma)$-bisimulations. The proof uses Lemma 3.1.

Theorem 5.6. *Let $\mathcal{L} \in DL_{nr}$. Let O be an $\mathcal{L}$-ontology, C_1, C_2 be $\mathcal{L}$-concepts, and Σ a signature. Then the following conditions are equivalent:*

(1) there is no $\mathcal{L}(\Sigma)$-interpolant for $C_1 \sqsubseteq C_2$ under O;
(2) C_1 and $\neg C_2$ are jointly consistent under O modulo $\mathcal{L}(\Sigma)$-bisimulations.

Proof. The proof is standard, and we refer the reader to the literature for similar proofs [40]. We only provide a sketch.

"1 ⇒ 2". Assume there is no $\mathcal{L}(\Sigma)$-interpolant for $C_1 \sqsubseteq C_2$ under O. Let

$$\Gamma = \{D \mid O \models C_1 \sqsubseteq D, D \in \mathcal{L}(\Sigma)\}.$$

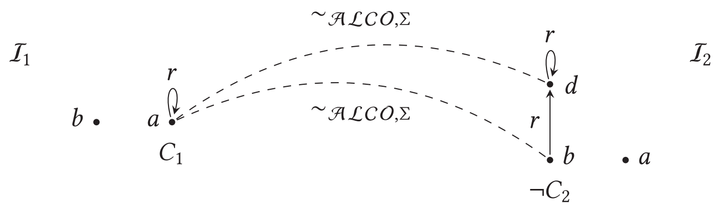

Fig. 4. Interpretations $\mathcal{I}_1$ and $\mathcal{I}_2$ illustrating Example 5.7.

Then $\mathcal{O} \not\models D \sqsubseteq C_2$, for any $D \in \Gamma$. As Γ is closed under conjunction and by compactness (recall that $\mathcal{ALCHOI}^u$ is a fragment of first-order logic), there exists a model $\mathcal{J}$ of $\mathcal{O}$ and an element $d \in \Delta^{\mathcal{J}}$ such that $d \in D^{\mathcal{J}}$ for all $D \in \Gamma$ but $d \notin C_2^{\mathcal{J}}$. Consider the set $t_{\mathcal{J}}(d) = \{D \in \mathcal{L}(\Sigma) \mid d \in D^{\mathcal{J}}\}$. Then, using again compactness, there exists a model $\mathcal{I}$ of $\mathcal{O}$ and an element $e \in \Delta^{\mathcal{I}}$ such that $e \in C_1^{\mathcal{I}}$ and $e \in D^{\mathcal{I}}$ for all $D \in t_{\mathcal{J}}(d)$. Thus $\mathcal{I}, e \equiv_{\mathcal{L},\Sigma} \mathcal{J}, d$. For every interpretation $\mathcal{I}$, there exists an ω-saturated elementary extension $\mathcal{I}'$ of $\mathcal{I}$ [24]. Thus, it follows from the fact that $\mathcal{ALCHOI}^u$ is a fragment of first-order logic that we may assume that both $\mathcal{I}$ and $\mathcal{J}$ are ω-saturated. By Lemma 3.1, $\mathcal{I}, e \sim_{\mathcal{L},\Sigma} \mathcal{J}, d$.

"2 $\Rightarrow$ 1". Assume an $\mathcal{L}(\Sigma)$-interpolant D for $C_1 \sqsubseteq C_2$ under $\mathcal{O}$ exists. Assume that Condition 2 holds, that is, there are models $\mathcal{I}_1$ and $\mathcal{I}_2$ of $\mathcal{O}$ and $d_i \in \Delta^{\mathcal{I}_i}$ for $i = 1, 2$ such that $d_1 \in C_1^{\mathcal{I}_1}$ and $d_2 \notin C_2^{\mathcal{I}_2}$ and $\mathcal{I}_1, d_1 \sim_{\mathcal{L},\Sigma} \mathcal{I}_2, d_2$. Then, by Lemma 3.1, $\mathcal{I}_1, d_1 \equiv_{\mathcal{L},\Sigma} \mathcal{I}_2, d_2$. But then from $d_1 \in C^{\mathcal{I}_1}$ we obtain $d_1 \in D^{\mathcal{I}_1}$ and so $d_2 \in D^{\mathcal{I}_2}$ which implies $d_2 \in C_2^{\mathcal{I}_2}$, a contradiction. □

Example 5.7. Consider again $C_1 = \{a\} \sqcap \exists r.\{a\}$ and $C_2 = \{b\} \rightarrow \exists r.\{b\}$ from Example 4.2 and set $\Sigma = \{r\}$. The interpretations $\mathcal{I}_1, \mathcal{I}_2$ depicted in Figure 4 (where we set $a^{\mathcal{I}_i} = a$ and $b^{\mathcal{I}_i} = b$, for $i = 1, 2$) show that C_1 and $\neg C_2$ are jointly consistent modulo $\mathcal{ALCO}(\Sigma)$-bisimulations. By extending the bisimulation in Figure 4 to a relation S such that $(b^{\mathcal{I}_1}, a^{\mathcal{I}_2}) \in S$ (so that the domain and range of S contain $\Delta^{\mathcal{I}_1}$ and $\Delta^{\mathcal{I}_2}$, respectively), one can show that C_1 and $\neg C_2$ are jointly consistent modulo $\mathcal{ALCO}^u(\Sigma)$-bisimulations. Moreover, by introducing an element e in $\mathcal{I}_2$ so that $(e, b^{\mathcal{I}_2}) \in r^{\mathcal{I}_2}$ and $(e, e) \in r^{\mathcal{I}_2}$, and further extending S by adding $(a^{\mathcal{I}_1}, e) \in S$, it can be seen that C_1 and $\neg C_2$ are jointly consistent modulo $\mathcal{ALCOI}^u(\Sigma)$-bisimulations (and hence $\mathcal{ALCHOI}^u(\Sigma)$-bisimulations). ⊣

The following characterization of the existence of explicit definitions can be proved similarly to Theorem 5.6.

Theorem 5.8. *Let $\mathcal{L} \in DL_{nr}$. Let $\mathcal{O}$ be an $\mathcal{L}$-ontology, C and C_0 $\mathcal{L}$-concepts, and $\Sigma \subseteq sig(\mathcal{O}, C)$ a signature. Then the following conditions are equivalent:*

(1) there is no explicit $\mathcal{L}(\Sigma)$-definition of C_0 under $\mathcal{O}$ and C;
(2) $C \sqcap C_0$ and $C \sqcap \neg C_0$ are jointly consistent under $\mathcal{O}$ modulo $\mathcal{L}(\Sigma)$-bisimulations.

Example 5.9. Consider $\mathcal{O}$, C, and Σ from Example 4.6. The interpretations $\mathcal{I}_1, \mathcal{I}_2$ depicted in Figure 5 show that $C \sqcap \exists r.\top$ and $C \sqcap \neg\exists r.\top$ are jointly consistent under $\mathcal{O}$ modulo $\mathcal{ALCH}(\Sigma)$-bisimulations. Note that the $\mathcal{ALCH}(\Sigma)$-bisimulation in Figure 5 is also an $\mathcal{ALCH}^u(\Sigma)$-bisimulation, but it is not an $\mathcal{ALCHI}(\Sigma)$-bisimulation, since e_1 has both an r_1- and an r_2-predecessor, whereas e_2 and e_2' lack an r_2- and an r_1-predecessor, respectively. To repair this, we replace $\mathcal{I}_2$ with an interpretation $\mathcal{J}$ that is obtained by taking the union of $\mathcal{I}_2$ with a copy $\overline{\mathcal{I}_1}$ of $\mathcal{I}_1$, and further adding $(\overline{d}_1, e_2) \in r_2^{\mathcal{J}}$ and $(\overline{d}_1, e_2') \in r_1^{\mathcal{J}}$ (where $\overline{d}_1$ is the copy of d_1 in $\mathcal{J}$). Then, we extend the $\mathcal{ALCH}(\Sigma)$-bisimulation in Figure 5 to a relation S that also connects the elements of $\mathcal{I}_1$ with the

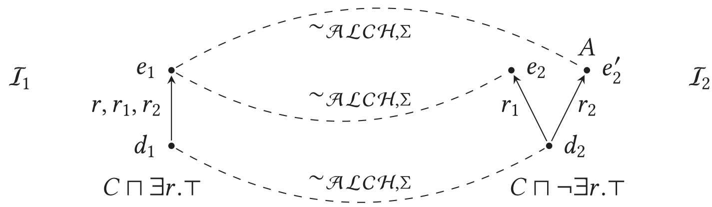

Fig. 5. Interpretations $\mathcal{I}_1$ and $\mathcal{I}_2$ illustrating Example 5.9.

respective copies in $\mathcal{J}$. It can be seen that $\mathcal{J}$ is a model of $\mathcal{O}$, $d_2 \in (C \sqcap \neg\exists r.\top)^{\mathcal{J}}$, and $(d_1, d_2) \in S$, where S is an $\mathcal{ALCHI}^u(\Sigma)$-bisimulation. ⊣

Interpolant existence and explicit definition existence are closely linked. We use Theorems 5.6 and 5.8 to show the following reductions.

Lemma 5.10. *Let $\mathcal{L} \in DL_{nr}$, $\mathcal{O}$ be an $\mathcal{L}$-ontology, C, C_0, C_1, and C_2 be $\mathcal{L}$-concepts, and Σ a signature. Then the following conditions are equivalent:*

(1) there is an explicit $\mathcal{L}(\Sigma)$-definition of C_0 under $\mathcal{O}$ and C;
(2) there is an $\mathcal{L}(\Sigma)$-interpolant for $C \sqcap C_0 \sqsubseteq C \rightarrow C_0$ under $\mathcal{O}$.

Conversely, the following conditions are also equivalent:

(1) there is an $\mathcal{L}(\Sigma)$-interpolant for $C_1 \sqsubseteq C_2$ under $\mathcal{O}$;
(2) $\mathcal{O} \models C_1 \sqsubseteq C_2$ and there is an explicit $\mathcal{L}(\Sigma)$-definition of C_2 under $\mathcal{O}$ and $C_2 \rightarrow C_1$.

Proof. We show the second equivalence. Assume that (1) does not hold. To show that (2) does not hold, assume $\mathcal{O} \models C_1 \sqsubseteq C_2$ (otherwise we are done). By Theorem 5.6, there exist pointed interpretations $\mathcal{I}_1, d_1$ and $\mathcal{I}_2, d_2$ such that $\mathcal{I}_i$ is a model of $\mathcal{O}$, $d_1 \in C_1^{\mathcal{I}_1}$, $d_2 \notin C_2^{\mathcal{I}_2}$, and $\mathcal{I}_1, d_1 \sim_{\mathcal{L},\Sigma} \mathcal{I}_2, d_2$. But then $d_1 \in ((C_2 \rightarrow C_1) \sqcap C_2)^{\mathcal{I}_1}$ and $d_2 \in ((C_2 \rightarrow C_1) \sqcap \neg C_2)^{\mathcal{I}_2}$ which shows that (2) does not hold by Theorem 5.8. The other direction is shown similarly. □

Hence, we obtain the following corollary.

Theorem 5.11. *Let $\mathcal{L} \in DL_{nr}$. Then there is a polynomial time reduction of projective $\mathcal{L}$-definition existence to $\mathcal{L}$-interpolant existence (and thus to generalized $\mathcal{L}$-interpolant existence). Conversely, there is a polynomial time reduction of generalized $\mathcal{L}$-interpolant existence (and thus $\mathcal{L}$-interpolant existence) to projective $\mathcal{L}$-definition existence if an oracle for $\mathcal{L}$-subsumption is admitted.*

Both reductions also exist for the ontology-free case and for RI-ontologies.

We now formulate the main complexity results proved in this article.

Theorem 5.12. *Let $\mathcal{L} \in DL_{nr}$. Then $\mathcal{L}$-interpolant existence, generalized $\mathcal{L}$-interpolant existence, and projective $\mathcal{L}$-definition existence are all* 2ExpTime*-complete.*

It follows that interpolant existence and projective definition existence are one exponential harder than subsumption for logics in $\mathsf{DL}_{\mathsf{nr}}$. Our lower bound proofs rely on the presence of ontologies. To understand the ontology-free case (and the case with RI-ontologies) we first recall from our introduction of DLs in $\mathsf{DL}_{\mathsf{nr}}$ above that for DLs with the universal role or with both inverse roles and nominals, the ontology can be encoded in a concept and so interpolant existence and projective definition existence are still 2ExpTime-complete with empty ontologies and RI-ontologies, respectively. For the remaining DLs in $\mathsf{DL}_{\mathsf{nr}}$, interpolant existence and projective definition existence become coNExpTime-complete. Thus, less complex than with ontologies, but still harder than subsumption (which is PSpace-complete), under standard complexity theoretic assumptions.

THEOREM 5.13. *Let $\mathcal{L} \in DL_{nr}$.*

(1) If $\mathcal{L}$ admits nominals and the universal role, or nominals and inverse roles, then ontology-free $\mathcal{L}$-interpolant existence, generalized $\mathcal{L}$-interpolant existence, and projective $\mathcal{L}$-definition existence are all 2EXPTIME*-complete.*
(2) If $\mathcal{L}$ admits the universal role and RIs, then RI-ontology $\mathcal{L}$-interpolant existence, generalized $\mathcal{L}$-interpolant existence, and projective $\mathcal{L}$-definition existence are all 2EXPTIME*-complete.*
(3) If $\mathcal{L} \in \{\mathcal{ALCO}, \mathcal{ALCHO}\}$, then ontology-free and RI-ontology $\mathcal{L}$-interpolant existence, generalized $\mathcal{L}$-interpolant existence, and projective $\mathcal{L}$-definition existence are all coNEXPTIME*-complete.*
(4) If $\mathcal{L} \in \{\mathcal{ALCH}, \mathcal{ALCHI}\}$, then RI-ontology $\mathcal{L}$-interpolant existence, generalized $\mathcal{L}$-interpolant existence, and projective $\mathcal{L}$-definition existence are all coNEXPTIME*-complete.*

We have seen that with the exception of $\mathcal{ALCO}$ and $\mathcal{ALCHO}$, all DLs in $\mathrm{DL}_{\mathrm{nr}}$ enjoy the non-PBDP. Hence checking the existence of a non-projective definition of a concept name is polynomial time reducible to subsumption checking and so EXPTIME-complete in the presence of an ontology. The following result states that even for $\mathcal{ALCO}$ and $\mathcal{ALCHO}$ checking the existence of non-projective definitions of concept names is not harder than subsumption.

THEOREM 5.14. *Let $\mathcal{L} \in \{\mathcal{ALCO}, \mathcal{ALCHO}\}$. Then non-projective $\mathcal{L}$-definition existence of concept names is* EXPTIME*-complete.*

Interestingly, Theorem 5.14 is the only result where the lack of either the CIP of (P)BDP does not lead to an increase in complexity of the interpolant/explicit definition existence problem. We show Theorem 5.14 in the appendix provided as supplementary material as it uses techniques that are slightly different from our other main results.

We next consider the non-projective explicit definability of nominals. We have seen in Example 4.10 above that for nominals even the non-PBDP does not hold for any DL in $\mathrm{DL}_{\mathrm{nr}}$. In fact, the following result states that the non-projective definability of nominals is as hard as their projective definability.

THEOREM 5.15. *Let $\mathcal{L} \in DL_{nr}$ admit nominals. Then non-projective $\mathcal{L}$-definition existence of nominals is* 2EXPTIME*-complete.*

Observe that the characterizations given in Theorems 5.6 and 5.8 provide mutual polynomial time reductions of generalized interpolant and definition existence to the complement of joint consistency modulo $\mathcal{L}$-bisimulations. Hence, to prove Theorems 5.12 to 5.15, it suffices to prove the corresponding complexity bounds for joint consistency.

We finally discuss an interesting consequence for CI-interpolants. Let $\mathcal{L}$ be a DL in $\mathrm{DL}_{\mathrm{nr}}$ that does not admit RIs. The *CI-interpolant existence problem in $\mathcal{L}$* is the problem to decide for $\mathcal{L}$-ontologies $\mathcal{O}$ and $\mathcal{O}'$ whether there exists an $\mathcal{L}$-CI-interpolant for $\mathcal{O}$ and $\mathcal{O}'$.

THEOREM 5.16. *Let $\mathcal{L} \in \{\mathcal{ALCO}^u, \mathcal{ALCOI}^u\}$. Then CI-interpolant existence in $\mathcal{L}$ is* 2EXPTIME*-complete.*

Observe that the 2EXPTIME upper bound is an immediate consequence of Point 1 of Theorem 5.13 as we can give a polynomial time reduction of CI-interpolant existence to ontology-free interpolant existence. Assume $\mathcal{L}$-ontologies $\mathcal{O}$ and $\mathcal{O}'$ are given. Let $\Sigma = \mathrm{sig}(\mathcal{O}) \cap \mathrm{sig}(\mathcal{O}')$. We find $\mathcal{L}$-concepts D and D' such that $\mathcal{O}$ is equivalent to $\{\top \sqsubseteq D\}$ and $\mathcal{O}'$ is equivalent to $\{\top \sqsubseteq D'\}$, respectively. Then, there exists a $\mathcal{L}$-CI-interpolant for $\mathcal{O}$ and $\mathcal{O}'$ iff there exists an $\mathcal{L}$-interpolant for $\forall u.D \sqsubseteq \forall u.D'$.

The 2EXPTIME lower bound is proved in Section 7 (Lemma 7.7) by adapting the 2EXPTIME lower bound proof for interpolant existence in $\mathcal{L}$.

6 UPPER BOUND PROOFS WITH ONTOLOGY

We show the double exponential upper bound of Theorem 5.12 (and thus of Theorem 5.15) using a new mosaic elimination procedure that decides joint consistency modulo $\mathcal{L}$-bisimulations, for all $\mathcal{L} \in \mathsf{DL}_{\mathsf{nr}}$.

THEOREM 6.1. *Let $\mathcal{L} \in DL_{nr}$. Then joint consistency modulo $\mathcal{L}$-bisimulations is in* 2EXPTIME.

To motivate our approach, reconsider Example 5.7. Notice that in interpretations $\mathcal{I}_1, \mathcal{I}_2$ witnessing joint consistency of C_1 and $\neg C_2$, $a^{\mathcal{I}_1}$ is bisimilar to both $b^{\mathcal{I}_2}$ and d. Moreover, it can be easily verified that there are no witnessing interpretations where $a^{\mathcal{I}_1}$ is bisimilar to a single element in $\mathcal{I}_2$. Using an ontology, one can extend this example so that $a^{\mathcal{I}_1}$ is enforced to be bisimilar to exponentially many elements in $\mathcal{I}_2$ in any interpretations $\mathcal{I}_1, \mathcal{I}_2$ witnessing joint consistency of two concepts (in fact, this will be the basis for showing the lower bound in the subsequent section). Thus, we cannot consider (pairs of) elements in isolation, but instead need to consider sets of elements. As usual in DLs, we abstract elements in interpretations by *types*, which syntactically describe the behavior of these elements by listing the relevant concepts that are satisfied there. Correspondingly, sets of elements are abstracted to sets of types. Since we need to coordinate two interpretations $\mathcal{I}_1, \mathcal{I}_2$, we thus consider *mosaics*, which are pairs (T_1, T_2) of sets of types. The intuitive meaning of such a pair is that it describes collections of elements in two interpretations $\mathcal{I}_1$ and $\mathcal{I}_2$ which realize precisely the types in T_1 and T_2, respectively, and are all mutually bisimilar. Naturally, not all possible mosaics (T_1, T_2) can be realized in this way and the goal is to determine the realizable ones. For this task, we use an elimination procedure. We start with the set of all possible mosaics and drop the "bad" ones until a fixed point is reached. We will see that the elimination conditions extend the conditions known from standard type elimination procedures in a relatively natural way to mosaics. Then, concepts C_1, C_2 will be jointly consistent under an ontology $\mathcal{O}$ modulo bisimulations if there is a surviving mosaic (T_1, T_2) such that C_1 is contained in some type in T_1 and C_2 is contained in some type in T_2.

We will now formalize our approach and start by introducing the relevant notions. Assume $\mathcal{L} \in \mathsf{DL}_{\mathsf{nr}}$ and consider an $\mathcal{L}$-ontology $\mathcal{O}$, $\mathcal{L}$-concepts C_1, C_2, and a signature Σ. Let $\Xi = \mathrm{sub}(\mathcal{O}, C_1, C_2)$ denote the closure under single negation of the set of subconcepts of concepts in $\mathcal{O}, C_1, C_2$. A *Ξ-type* t is a subset of Ξ such that there exists a model $\mathcal{I}$ of $\mathcal{O}$ and $d \in \Delta^{\mathcal{I}}$ with $t = \mathrm{tp}_\Xi(\mathcal{I}, d)$, where

$$\mathrm{tp}_\Xi(\mathcal{I}, d) = \{C \in \Xi \mid d \in C^{\mathcal{I}}\}$$

is the *Ξ-type realized at d in $\mathcal{I}$*. Let $\mathrm{Tp}(\Xi)$ denote the set of all Ξ-types. We remark that the number of Ξ-types is at most exponential in $||\mathcal{O}|| + ||C_1|| + ||C_2||$ and, moreover, the set of all Ξ-types can be computed in time exponential in $||\mathcal{O}|| + ||C_1|| + ||C_2||$ for all considered logics [1, 9]. A *mosaic* is a pair (T_1, T_2) of sets of types $T_1, T_2 \subseteq \mathrm{Tp}(\Xi)$. For interpretations $\mathcal{I}_1, \mathcal{I}_2$ and $i \in \{1, 2\}$, the *mosaic defined by $d \in \Delta^{\mathcal{I}_i}$* in $\mathcal{I}_1, \mathcal{I}_2$ is the pair $(T_1(d), T_2(d))$ where

$$T_j(d) = \{\mathrm{tp}_\Xi(\mathcal{I}_j, e) \mid e \in \Delta^{\mathcal{I}_j}, \mathcal{I}_i, d \sim_{\mathcal{L},\Sigma} \mathcal{I}_j, e\},$$

for $j = 1, 2$. We say that a pair (T_1, T_2) of sets T_1, T_2 of types is a *mosaic defined by* $\mathcal{I}_1, \mathcal{I}_2$ if there exists $d \in \Delta^{\mathcal{I}_1} \cup \Delta^{\mathcal{I}_2}$ such that $(T_1, T_2) = (T_1(d), T_2(d))$. Clearly, there are at most doubly exponentially many mosaics.

Example 6.2. Recall C_1, C_2, Σ, and $\mathcal{I}_1, \mathcal{I}_2$ from Example 5.7, and let $\mathcal{O} = \emptyset$. The set Ξ consists of the concepts $\{a\}$, $\exists r.\{a\}$, $\{b\}$, $\exists r.\{b\}$, C_1, C_2, and negations thereof. We have that:

$$\mathrm{tp}_\Xi(\mathcal{I}_1, a^{\mathcal{I}_1}) = \{\{a\}, \exists r.\{a\}, \neg\{b\}, \neg\exists r.\{b\}, C_1, C_2\}$$
$$\mathrm{tp}_\Xi(\mathcal{I}_2, b^{\mathcal{I}_2}) = \{\neg\{a\}, \neg\exists r.\{a\}, \{b\}, \neg\exists r.\{b\}, \neg C_1, \neg C_2\}$$
$$\mathrm{tp}_\Xi(\mathcal{I}_2, d) = \{\neg\{a\}, \neg\exists r.\{a\}, \neg\{b\}, \neg\exists r.\{b\}, \neg C_1, C_2\}$$

The mosaic defined by $a^{\mathcal{I}_1}$ in $\mathcal{I}_1, \mathcal{I}_2$ is $(T_1(a^{\mathcal{I}_1}), T_2(a^{\mathcal{I}_1}))$, where

$$T_1(a^{\mathcal{I}_1}) = \{\mathrm{tp}_\Xi(\mathcal{I}_1, a^{\mathcal{I}_1})\} \quad \text{and} \quad T_2(a^{\mathcal{I}_1}) = \{\mathrm{tp}_\Xi(\mathcal{I}_2, b^{\mathcal{I}_2}), \mathrm{tp}_\Xi(\mathcal{I}_2, d)\}. \qquad \dashv$$

As announced above, the aim of the mosaic elimination procedure is to determine all mosaics (T_1, T_2) such that all $t \in T_1 \cup T_2$ can be realized in mutually $\mathcal{L}(\Sigma)$-bisimilar elements of models $\mathcal{I}_1, \mathcal{I}_2$ of $\mathcal{O}$. In order to formulate the elimination conditions, we define several compatibility conditions between types and between mosaics, similar to the compatibility conditions that are used in standard type elimination procedures. Throughout the rest of the section, we treat the universal role u as a role name contained in Σ, in case $\mathcal{L}$ admits the universal role. Note that u^- is equivalent to u, and that $\mathcal{O} \models r \sqsubseteq u$, for every role r.

Let t_1, t_2 be Ξ-types. We call t_1, t_2 *u-equivalent* if $\exists u.C \in t_1$ iff $\exists u.C \in t_2$, for every $\exists u.C \in \Xi$. Notice that the condition is trivially satisfied if $\mathcal{L}$ does not admit the universal role. For a role r, we call t_1, t_2 *r-coherent for $\mathcal{O}$*, in symbols $t_1 \rightsquigarrow_r t_2$, if t_1, t_2 are u-equivalent and the following conditions hold for all roles s with $\mathcal{O} \models r \sqsubseteq s$: (1) if $\neg\exists s.C \in t_1$, then $C \notin t_2$ and (2) if $\neg\exists s^-.C \in t_2$, then $C \notin t_1$. Note that $t \rightsquigarrow_r t'$ iff $t' \rightsquigarrow_{r^-} t$. We lift the definition of r-coherence from types to mosaics $(T_1, T_2), (T_1', T_2')$. Specifically, we call (T_1, T_2), (T_1', T_2') *r-coherent*, in symbols $(T_1, T_2) \rightsquigarrow_r (T_1', T_2')$, if for $i = 1, 2$:

- for every $t \in T_i$ there exists a $t' \in T_i'$ such that $t \rightsquigarrow_r t'$, and
- if $\mathcal{L}$ admits inverse roles, then for every $t' \in T_i'$, there is a $t \in T_i$ such that $t \rightsquigarrow_r t'$.

Note that $(T_1, T_2) \rightsquigarrow_r (T_1', T_2')$ iff $(T_1', T_2') \rightsquigarrow_{r^-} (T_1, T_2)$ in case $\mathcal{L}$ admits inverse roles. Also, notice that $(T_1, T_2) \rightsquigarrow_r (T_1', T_2')$ implies $(T_1, T_2) \rightsquigarrow_u (T_1', T_2')$, for every role r.

Example 6.3. Consider again interpretations $\mathcal{I}_1, \mathcal{I}_2$ from Example 5.7 and the types $t_1 = \mathrm{tp}_\Xi(\mathcal{I}_1, a^{\mathcal{I}_1})$, $t_2 = \mathrm{tp}_\Xi(\mathcal{I}_2, b^{\mathcal{I}_2})$, and $t_3 = \mathrm{tp}_\Xi(\mathcal{I}_2, d)$. Then, $t_1 \rightsquigarrow_r t_1$, $t_2 \rightsquigarrow_r t_3$, and $t_3 \rightsquigarrow_r t_3$. Moreover, the mosaic (T_1, T_2) defined by $a_1^{\mathcal{I}}$ in $\mathcal{I}_1, \mathcal{I}_2$ satisfies $(T_1, T_2) \rightsquigarrow_r (T_1, T_2)$. $\dashv$

We are now in the position to formulate the mosaic elimination conditions. Let $\mathcal{S} \subseteq 2^{\mathrm{Tp}(\Xi)} \times 2^{\mathrm{Tp}(\Xi)}$ be a set of mosaics. We call $(T_1, T_2) \in \mathcal{S}$ *bad* if it violates one of the following conditions.

Σ-concept name coherence $A \in t$ iff $A \in t'$, for every concept name $A \in \Sigma$ and every $t, t' \in T_1 \cup T_2$;

Existential saturation for $i = 1, 2$ and $\exists r.C \in t \in T_i$, there exists $(T_1', T_2') \in \mathcal{S}$ such that (1) there exists $t' \in T_i'$ with $C \in t'$ and $t \rightsquigarrow_r t'$ and (2) if $\mathcal{O} \models r \sqsubseteq s$ for a Σ-role s, then $(T_1, T_2) \rightsquigarrow_s (T_1', T_2')$.

For didactic purposes and because we need it later in Section 10, we first give the mosaic elimination procedure for logics $\mathcal{L}$ that do not admit nominals. The procedure starts with the set $\mathcal{S}_0$ of all mosaics. Then obtain, for $i \geq 0$, $\mathcal{S}_{i+1}$ from $\mathcal{S}_i$ by eliminating all mosaics (T_1, T_2) that are bad in $\mathcal{S}_i$. Let $\mathcal{S}^*$ be where the sequence stabilizes. The elimination procedure decides joint consistency in the following sense.

Lemma 6.4. *If $\mathcal{L}$ does not admit nominals, the following conditions are equivalent:*

(1) C_1, C_2 are jointly consistent under $\mathcal{O}$ modulo $\mathcal{L}(\Sigma)$-bisimulations;
(2) there exist $(T_1, T_2) \in \mathcal{S}^*$ and Ξ-types $t_1 \in T_1, t_2 \in T_2$ with $C_1 \in t_1$ and $C_2 \in t_2$.

We refrain from giving the proof of Lemma 6.4 since it will follow from Lemma 6.5 below. We note, however, that for $\mathcal{L}$ as in the lemma, Theorem 6.1 is an immediate consequence of the procedure: there are only double exponentially many mosaics, so the elimination terminates after at most double exponentially steps. It remains to observe that every elimination step can be executed in double exponential time.

This relatively straightforward elimination procedure does not quite work in the presence of nominals. Intuitively, the reason is that in any two interpretations $\mathcal{I}_1, \mathcal{I}_2$, every nominal a is realized (modulo bisimulation) in exactly one mosaic. Now, if the set $\mathcal{S}$ contains several mosaics mentioning a, they possibly witness existential saturation of each other which, however, cannot be reflected in an interpretation. Thus, for the mosaic elimination procedure to work (in the sense of Lemma 6.4) one has to "guess" for every nominal a exactly one mosaic that describes a.

To formalize this idea, let us call a set $\mathcal{S}$ of mosaics *good for nominals* if for every individual name $a \in \mathrm{sig}(\Xi)$ and $i = 1, 2$ there exists exactly one t_a^i with $\{a\} \in t_a^i \in \bigcup_{(T_1,T_2)\in\mathcal{S}} T_i$ and exactly one pair $(T_1, T_2) \in \mathcal{S}$ with $t_a^i \in T_i$. Moreover, if $a \in \Sigma$, then that pair takes the form

- $(\{t_a^1\}, \{t_a^2\})$ in case $\mathcal{L}$ admits the universal role, and
- $(\{t_a^1\}, \{t_a^2\})$, $(\{t_a^1\}, \emptyset)$, or $(\emptyset, \{t_a^2\})$, otherwise.

We can now formulate the more general lemma.

Lemma 6.5. *The following conditions are equivalent:*

(1) C_1, C_2 are jointly consistent under $\mathcal{O}$ modulo $\mathcal{L}(\Sigma)$-bisimulations;
(2) there exists a set $\mathcal{S}^*$ of mosaics that is good for nominals and does not contain a bad mosaic, such that there exist $(T_1, T_2) \in \mathcal{S}^*$ and Ξ-types $t_1 \in T_1, t_2 \in T_2$ with $C_1 \in t_1$ and $C_2 \in t_2$.

Proof. "1 ⇒ 2". Let $\mathcal{I}_1, d_1 \sim_{\mathcal{L},\Sigma} \mathcal{I}_2, d_2$ for models $\mathcal{I}_1$ and $\mathcal{I}_2$ of $\mathcal{O}$ such that d_1, d_2 realize Ξ-types t_1, t_2 and $C_1 \in t_1, C_2 \in t_2$. Let $\mathcal{S}^*$ be the set of all mosaics defined by $\mathcal{I}_1, \mathcal{I}_2$. It is routine to show that no (T_1, T_2) in $\mathcal{S}^*$ is bad and that $\mathcal{S}^*$ is good for nominals. Now, the mosaic (T_1, T_2) defined by $d_1^{\mathcal{I}}$ in $\mathcal{I}_1, \mathcal{I}_2$ witnesses Condition (2).

"2 ⇒ 1". Suppose there exist a good set $\mathcal{S}^*$ of mosaics and $(S_1, S_2) \in \mathcal{S}^*$ and Ξ-types $s_1 \in S_1, s_2 \in S_2$ with $C_1 \in s_1$ and $C_2 \in s_2$. Let $\mathcal{I}_i$, for $i = 1, 2$ be interpretations defined by setting:

$$\begin{aligned}
\Delta^{\mathcal{I}_i} &:= \{(t, (T_1, T_2)) \mid (T_1, T_2) \in \mathcal{S}^*, t \in T_i, \text{ and} \\
&\qquad \text{if } \mathcal{L} \text{ admits the universal role, then } (S_1, S_2) \rightsquigarrow_u (T_1, T_2) \text{ and } t, s_i \text{ are } u\text{-equivalent}\} \\
r^{\mathcal{I}_i} &:= \{((t,p),(t',p')) \in \Delta^{\mathcal{I}_i} \times \Delta^{\mathcal{I}_i} \mid t \rightsquigarrow_r t' \text{ and for all } \Sigma\text{-roles } s : ((\mathcal{O} \models r \sqsubseteq s) \Rightarrow p \rightsquigarrow_s p')\} \\
A^{\mathcal{I}_i} &:= \{(t,p) \in \Delta^{\mathcal{I}_i} \mid A \in t\} \\
a^{\mathcal{I}_i} &:= (t, (T_1, T_2)) \in \Delta^{\mathcal{I}_i}, \{a\} \in t \in T_i
\end{aligned}$$

Note that the interpretation of nominals is well-defined since $\mathcal{S}^*$ is good for nominals.

We verify that interpretations $\mathcal{I}_1$ and $\mathcal{I}_2$ witness Condition (1). □

Claim 1. For $i = 1, 2$, all $C \in \Xi$, and all $(t, p) \in \Delta^{\mathcal{I}_i}$, we have $(t, p) \in C^{\mathcal{I}_i}$ iff $C \in t$.

Proof of Claim 1. Let $i \in \{1, 2\}$. The proof is by induction on the structure of concepts in Ξ.

- The claim holds for concept names $C = A$ and all nominals $C = \{a\}$, by definition of $\mathcal{I}_i$.
- The Boolean cases, $\neg C$ and $C \sqcap C'$, are immediate consequences of the hypothesis.
- Let $C = \exists r.D$. (Recall that r is possibly the universal role u.)
  "if": Suppose $\exists r.D \in t$. By existential saturation, there is a $p' = (T_1', T_2') \in \mathcal{S}^*$ such that (1) there exists $t' \in T_i'$ with $D \in t'$ and $t \rightsquigarrow_r t'$ and (2) if $\mathcal{O} \models r \sqsubseteq s$ for some Σ-role s, then $p \rightsquigarrow_s p'$. Note that t, t' are thus also u-equivalent, so $(t', p') \in \Delta^{\mathcal{I}_i}$. We distinguish cases:
  - If r is a role name, then by definition of $r^{\mathcal{I}_i}$, we have $((t,p),(t',p')) \in r^{\mathcal{I}_i}$. Since $D \in t'$, induction yields $(t', p') \in D^{\mathcal{I}_i}$. Overall, we get $(t, p) \in (\exists r.D)^{\mathcal{I}_i}$.
  - If $r = r_0^-$ is an inverse role, then (1) and (2) above imply (1') $t' \rightsquigarrow_{r_0} t$ and (2') if $\mathcal{O} \models r_0 \sqsubseteq s$ for some Σ-role s, then $p \rightsquigarrow_s p'$. As before, we can then conclude that $((t',p'),(t,p)) \in r_0^{\mathcal{I}_i}$. Since $D \in t'$, induction yields $(t', p') \in D^{\mathcal{I}_i}$. Overall, we get $(t, p) \in (\exists r_0^-.D)^{\mathcal{I}_i}$.

"only if": Suppose $(t,p) \in (\exists r.D)^{\mathcal{I}_i}$. Then, there is $(t',p') \in \Delta^{\mathcal{I}_i}$ with $((t,p),(t',p')) \in r^{\mathcal{I}_i}$ and $(t',p') \in D^{\mathcal{I}_i}$. By induction, the latter implies $D \in t'$. We distinguish cases:
- If r is a role name, then by definition of $r^{\mathcal{I}_i}$, $t \rightsquigarrow_r t'$ and thus $\exists r.D \in t$.
- If $r = r_0^-$ is an inverse role, then by definition of $r_0^{\mathcal{I}_i}$, $t' \rightsquigarrow_{r_0} t$. Thus, also $\exists r_0^-.D \in t$.

This finishes the proof of Claim 1. Claim 1 implies that $(s_1,(S_1,S_2)) \in C_1^{\mathcal{I}_1}$ and $(s_2,(S_1,S_2)) \in C_2^{\mathcal{I}_2}$. Claim 1 also implies that the type realized by (t,p) in $\mathcal{I}_i$ is t, for all $(t,p) \in \Delta^{\mathcal{I}_i}$. Since types are, by definition, realized in models of $\mathcal{O}$, it follows that both $\mathcal{I}_1$ and $\mathcal{I}_2$ are models of $\mathcal{O}$.

Claim 2. The relation R defined by

$$R = \{((t,p),(t',p)) \mid (t,p) \in \Delta^{\mathcal{I}_1}, (t',p) \in \Delta^{\mathcal{I}_2}\}$$

is an $\mathcal{L}(\Sigma)$-bisimulation.

Proof of Claim 2. Clearly, R satisfies Condition [AtomC] due to Σ-concept name coherence. Condition [AtomI] follows from the fact that $\mathcal{S}^*$ is good for nominals in case $\mathcal{L}$ admits nominals.

For Condition [Forth], let $((t,p),(t',p)) \in R$ and $((t,p),(t_1,p_1)) \in r^{\mathcal{I}_1}$, for some Σ-role r, and let $p = (T_1,T_2)$ and $p_1 = (T_1',T_2')$. We distinguish cases:

- If r is a role name, then by definition of $r^{\mathcal{I}_1}$, we have (1) $t \rightsquigarrow_r t_1$ and (2) for all Σ-roles s with $\mathcal{O} \models r \sqsubseteq s$, we have $p \rightsquigarrow_s p_1$. Since $t' \in T_2$ and $p \rightsquigarrow_r p_1$ there is some $t'' \in T_2'$ with $t' \rightsquigarrow_r t''$. Thus, in particular, t'' is u-equivalent to t' (and thus to s_2), which implies $(t'',p_1) \in \Delta^{\mathcal{I}_2}$. The definition of $r^{\mathcal{I}_2}$ then implies that $((t',p),(t'',p_1)) \in r^{\mathcal{I}_2}$. It remains to note that the definition of R yields $((t_1,p_1),(t'',p_1)) \in R$.
- If $r = r_0^-$ is an inverse role, then by definition of $r_0^{\mathcal{I}_1}$, we have (1) $t_1 \rightsquigarrow_{r_0} t$ and (2) for all Σ-roles s with $\mathcal{O} \models r_0 \sqsubseteq s$, we have $p_1 \rightsquigarrow_s p$. Since $t' \in T_2$ and $p_1 \rightsquigarrow_{r_0} p$ there is some $t'' \in T_2'$ with $t'' \rightsquigarrow_{r_0} t'$. Thus, in particular, t'' is u-equivalent to t' (and thus to s_2), which implies $(t'',p_1) \in \Delta^{\mathcal{I}_2}$. The definition of $r_0^{\mathcal{I}_2}$ then implies that $((t'',p_1),(t',p)) \in r_0^{\mathcal{I}_2}$. It remains to note that the definition of R yields $((t_1,p_1),(t'',p_1)) \in R$.

Condition [Back] is dual.

Finally, we verify that R and R^- are surjective if $\mathcal{L}$ admits the universal role. Let $(t,(T_1,T_2)) \in \Delta^{\mathcal{I}_1}$. Then, $(S_1,S_2) \rightsquigarrow_u (T_1,T_2)$, by definition of $\Delta^{\mathcal{I}_1}$. This implies that there is a type $t' \in T_2$ which is u-equivalent to s_2 and thus $(t',(T_1,T_2)) \in \Delta^{\mathcal{I}_2}$. The definition of R implies $((t,(T_1,T_2)),(t',(T_1,T_2))) \in R$. The other direction is dual.

This finishes the proof of Claim 2. By definition of R, $((s_1,(S_1,S_2)),(s_2,(S_1,S_2))) \in R$, and thus $\mathcal{I}_1,(s_1,(S_1,S_2)) \sim_{\mathcal{L},\Sigma} \mathcal{I}_2,(s_2,(S_1,S_2))$. □

It remains to argue that we can find in double exponential time a set $\mathcal{S}^*$ as in Condition (2) of Lemma 6.5. We use a suitable variant of the elimination procedure described after Lemma 6.4.

Lemma 6.6. *Let $\mathcal{L} \in DL_{nr}$. Then it is decidable in time double exponential in $||\mathcal{O}|| + ||C_1|| + ||C_2||$ whether for an $\mathcal{L}$-ontology $\mathcal{O}$, $\mathcal{L}$-concepts C_1, C_2, and a signature $\Sigma \subseteq sig(\Xi)$ there exists some $\mathcal{S}^*$ satisfying Condition (2) of Lemma 6.5.*

Proof. Let $\mathcal{L} \in \mathrm{DL_{nr}}$, and assume $\mathcal{O}$, C_1, C_2, and Σ are given. We can enumerate in double exponential time the maximal good sets $\mathcal{U} \subseteq 2^{T(\Xi)} \times 2^{T(\Xi)}$ by picking, for each nominal $a \in \mathrm{sig}(\Xi)$ and $i = 1,2$, a type t_a^i, and a mosaic (T_1,T_2) with $t_a^i \in T_i$. In doing so, we make sure that $(\{t_a^1\},\{t_a^2\})$ is selected in case $a \in \Sigma$. Crucially, there are only double exponentially many possibilities to make this choice. Remove all mosaics that mention a nominal and have not been selected. The resulting set is good for nominals.

Then, we eliminate from any set $\mathcal{U}$ obtained in that process recursively all bad mosaics. Let $\mathcal{S}_{\mathcal{U}} \subseteq \mathcal{U}$ be the largest fixpoint of that procedure. Then one can easily show that there exists a set $\mathcal{S}^*$ satisfying Condition (2) of Lemma 6.5 iff there exists a set $\mathcal{U}$ that can be obtained by the process described above such that the largest fixpoint $\mathcal{S}_{\mathcal{U}}$ satisfies Condition (2) of Lemma 6.5. Since elimination terminates after double exponential time, and there are only double exponentially many possible choices for $\mathcal{U}$, the lemma follows. □

Theorem 6.1 is a direct consequence of Lemmas 6.5 and 6.6.

7 LOWER BOUND PROOFS WITH ONTOLOGY

The goal of this section is to provide the proofs of the lower bounds in Theorems 5.12, 5.15, and 5.16. We start with the former two. By Lemma 5.10 and Theorem 5.8, it suffices to consider joint consistency. We will provide two reductions: in Section 7.1, we provide the reduction for DLs in DL_{nr} that admits nominals and, in Section 7.3, the one for DLs that admits RIs. In Section 7.2, we will investigate the shape of the interpolants/explicit definitions that arise in the preceding lower bound proof. In Section 7.4, we then show how to adapt the lower bound proof from Section 7.1 to the case of CI-interpolant existence. In all cases, we reduce the word problem for languages recognized by exponentially space bounded **alternating Turing machines** (**ATMs**), which we introduce next.

An *ATM* is a tuple $M = (Q, \Theta, \Gamma, q_0, \Delta)$ where $Q = Q_\exists \uplus Q_\forall$ is a finite set of states partitioned into *existential states* $Q_\exists$ and *universal states* $Q_\forall$. Further, Θ is the input alphabet and Γ 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 wqw' with $w, w' \in \Gamma^*$ and $q \in Q$. We say that wqw' 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 $q_0 w$;
- if a node is labeled with an existential configuration wqw', then it has a single successor which is labeled with a successor configuration of wqw';
- if a node is labeled with a universal configuration wqw', then it has two successors which are labeled with the two successor configurations of wqw'.

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 [23], where n is the length of the input w.

7.1 DLs with Nominals

We start with DLs supporting nominals. By Theorem 5.8, it suffices to prove the following result.

Lemma 7.1. *Let $\mathcal{L} \in DL_{nr}$ admit nominals. It is 2ExpTime-hard to decide for an $\mathcal{L}$-ontology $\mathcal{O}$, individual name b, and signature $\Sigma \subseteq sig(\mathcal{O}) \setminus \{b\}$ whether $\{b\}$ and $\neg\{b\}$ are jointly consistent under $\mathcal{O}$ modulo $\mathcal{L}(\Sigma)$-bisimulations. This is true even if b is the only individual in $\mathcal{O}$ and $\Sigma = sig(\mathcal{O}) \setminus \{b\}$.*

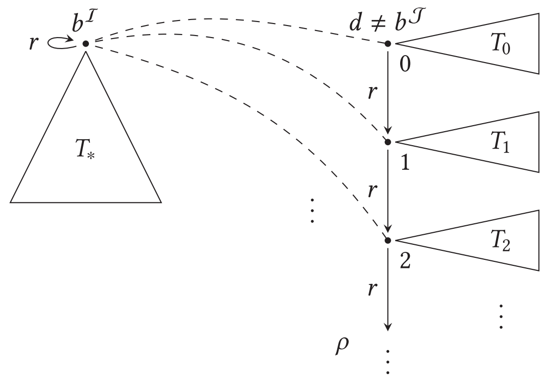

Fig. 6. Enforced bisimulation in lower bound.

As announced, we reduced the word problem for 2^n-space bounded ATMs. Let us fix such an ATM $M = (Q, \Theta, \Gamma, q_0, \Delta)$ and an input $w = a_0 \dots a_{n-1}$ of length n. We first provide the reduction for $\mathcal{L} = \mathcal{ALCO}$ using an ontology $\mathcal{O}$ and a signature Σ such that $\mathcal{O}$ contains concept names that are not in Σ and uses two role names r, s, and show later how to adapt this proof to $\Sigma = \text{sig}(\mathcal{O}) \setminus \{b\}$ and DLs supporting inverses and/or the universal role.

The idea of the reduction is as follows. We aims to construct an ontology $\mathcal{O}$ such that M accepts w iff $\{b\}$ and $\neg\{b\}$ are jointly consistent under $\mathcal{O}$ modulo $\mathcal{L}(\Sigma)$-bisimulations, where

$$\Sigma = \{r, s, Z, B_\forall, B^1_\exists, B^2_\exists\} \cup \{A_\sigma \mid \sigma \in \Gamma \cup (Q \times \Gamma)\}.$$

The ontology $\mathcal{O}$ enforces that $r(b, b)$ holds in any model $\mathcal{O}$ using the CI $\{b\} \sqsubseteq \exists r.\{b\}$. Moreover, it enforces that any element distinct from $b^{\mathcal{I}}$ with an r-successor lies on an infinite r-path ρ enforced by the CIs:

$$\neg\{b\} \sqcap \exists r.\top \sqsubseteq I_s \qquad\qquad I_s \sqsubseteq \exists r.\top \sqcap \forall r.I_s$$

with I_s a concept name. Thus, if there exist models $\mathcal{I}, \mathcal{J}$ of $\mathcal{O}$ with $\mathcal{I}, b^{\mathcal{I}} \sim_{\mathcal{ALCO},\Sigma} \mathcal{J}, d$ for some $d \neq b^{\mathcal{J}}$ and $d \in (\exists r.\top)^{\mathcal{J}}$, it follows that all elements on the path ρ are $\mathcal{ALC}(\Sigma)$-bisimilar to $b^{\mathcal{I}}$ and thus mutually $\mathcal{ALC}(\Sigma)$-bisimilar. The situation is depicted in Figure 6, where the trees T_* and T_i, $i \geq 0$ starting in $b^{\mathcal{I}}$ and on the path elements, respectively, are also mutually $\mathcal{ALC}(\Sigma)$-bisimilar. These trees shall represent the computation tree of M on input w (using symbols from Σ) as follows, cf. Figure 7, which shows the skeleton of a single tree T_i. Configurations of M are represented as paths of length 2^n over a role s in which every element is labeled with a symbol A_σ, $\sigma \in \Gamma \cup (Q \times \Gamma)$ that represents the content of a single tape cell (omitted in the figure for the sake of readibility). In Figure 7, the start of a configuration is indicated by $\circ$ and the s-path between consecutive $\circ$ has length 2^n. Every configuration is marked as existential or universal using concept names $B_\forall, B^1_\exists, B^2_\exists$; the superscript $\cdot^1/\cdot^2$ indicates which successor is chosen for an existential configuration. Existential configurations have a single successor configuration and universal configurations have two successor configurations.

The structure of this computation tree can easily be enforced in $\mathcal{ALC}$ using standard techniques (as we detail below). The difficulty is to achieve synchronization between successor configurations in the tree. That is, if a configuration c in the computation tree is followed by another configuration c', then c' is actually a successor configuration of c according to M. To achieve this, we first ensure that in T_i, for $i \leq 2^n$, the $(2^n - i)$th cell of each configuration in the computation tree is synchronized with the $(2^n - i)$th cell of the next configuration(s), as indicated by the dotted lines in Figure 7. This

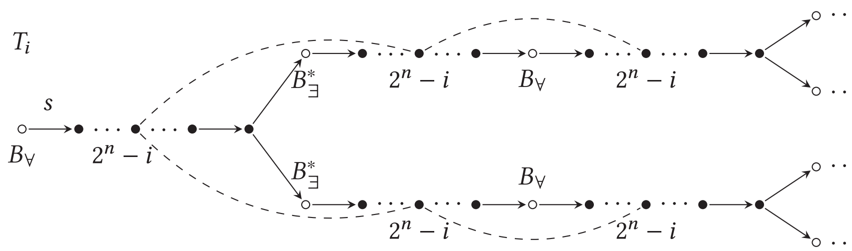

Fig. 7. Computation tree of M.

can be realized in $\mathcal{ALC}$ using a set of concept names not in Σ. Then we exploit the fact that the trees T_i are mutually $\mathcal{ALC}(\Sigma)$-bisimilar, which implies that in *all* T_i *all* cells of *all* configurations are synchronized. In more detail, we do this by using several counters modulo 2^n as follows.

The first counter counts modulo 2^n along the path ρ using concept names not in Σ. As announced, each point of ρ starts an infinite tree along role s that is supposed to mimick the computation tree of M on input w. Along this tree, two more counters are maintained:

— one counter starting at 0 and counting modulo 2^n, and
— another counter starting at *the value of the counter on* ρ and also counting modulo 2^n.

The first counter is used to divide the tree into configurations of length 2^n and the second counter is used to link the $(2^n - i)$th cell of successive configurations in T_i as described above.

We will next provide the CIs in $\mathcal{O}$ in more detail. The counter along ρ is realized using concept names A_i, $0 \leq i < n$ and by including the following (standard) CIs, for every i with $0 \leq i < n$:

$$I_s \sqcap A_i \sqcap \bigsqcap_{j<i} A_j \sqsubseteq \forall r.\neg A_i \qquad I_s \sqcap \neg A_i \sqcap \bigsqcap_{j<i} A_j \sqsubseteq \forall r.A_i$$
$$I_s \sqcap A_i \sqcap \bigsqcup_{j<i} \neg A_j \sqsubseteq \forall r.A_i \qquad I_s \sqcap \neg A_i \sqcap \bigsqcup_{j<i} \neg A_j \sqsubseteq \forall r.\neg A_i$$

Using again the concept name I_s, we start the s-trees with two counters, realized using concept names U_i and V_i, $0 \leq i < n$, and initialized to 0 and the value of the A-counter, respectively, by including the following CIs for every j with $0 \leq j < n$:

$$I_s \sqsubseteq (U = 0)$$
$$I_s \sqsubseteq A_j \leftrightarrow V_j$$
$$\top \sqsubseteq \exists s.\top$$

Here, $(U = 0)$ is an abbreviation for the concept $\bigsqcap_{i=0}^{n-1} \neg U_i$; we use similar abbreviations below without further notice. The counters U_i and V_i are incremented along s in the same way as A_i is incremented along r, so we omit details. Configurations of M are represented between two consecutive points having U-counter value 0. We next enforce the structure of the computation tree (recall that $q_0 \in Q_\forall$):

$$I_s \sqsubseteq B_\forall \tag{†}$$
$$(U < 2^n - 1) \sqcap B_\forall \sqsubseteq \forall s.B_\forall$$
$$(U < 2^n - 1) \sqcap B_\exists^i \sqsubseteq \forall s.B_\exists^i \qquad i \in \{1, 2\}$$
$$(U = 2^n - 1) \sqcap B_\forall \sqsubseteq \forall s.(B_\exists^1 \sqcup B_\exists^2)$$
$$(U = 2^n - 1) \sqcap B_\exists^i \sqsubseteq \forall s.B_\forall \qquad i \in \{1, 2\}$$
$$(U = 2^n - 1) \sqcap B_\forall \sqsubseteq \exists s.Z \sqcap \exists s.\neg Z$$

These CIs enforce that all points, which represent a configuration satisfy one of $B_\forall, B^1_\exists, B^2_\exists$ indicating the kind of configuration and, if existential, also a choice of the transition function. The symbol $Z \in \Sigma$ enforces the branching.

We next set the initial configuration, for input $w = a_0, \dots, a_{n-1}$.

$$\begin{aligned} I_s &\sqsubseteq A_{q_0, a_0} \\ I_s &\sqsubseteq \forall s^k.A_{a_k} && 0 < i < n \\ I_s &\sqsubseteq \forall s^n.\mathsf{Blank} \\ \mathsf{Blank} &\sqsubseteq A_\square \\ \mathsf{Blank} \sqcap (U < 2^n - 1) &\sqsubseteq \forall s.\mathsf{Blank} \end{aligned}$$

To coordinate successor configurations, we associate with M functions f_i, $i \in \{1, 2\}$ that map the content of three consecutive cells of a configuration to the content of the middle cell in the ith successor configuration (assuming an arbitrary order on the set $\Delta(q, a)$, for all q, a). In what follows, we ignore the corner cases that occur at the border of configurations; they can be treated in a similar way. Clearly, for each possible triple $(\sigma_1, \sigma_2, \sigma_3) \in (\Gamma \cup (Q \times \Gamma))^3$, the $\mathcal{ALC}$-concept $C_{\sigma_1,\sigma_2,\sigma_3} = A_{\sigma_1} \sqcap \exists s.(A_{\sigma_2} \sqcap \exists s.A_{\sigma_3})$ is true at an element a of the computation tree iff a is labeled with A_{σ_1}, an s-successors b of a is labeled with A_{σ_2}, and an s-successors c of b is labeled with A_{σ_3}. In each configuration, we synchronize elements with V-counter 0 by including for every $(\sigma_1, \sigma_2, \sigma_3)$ and $i \in \{1, 2\}$ the following CIs:

$$\begin{aligned} (V = 2^n - 1) \sqcap (U < 2^n - 2) \sqcap C_{\sigma_1,\sigma_2,\sigma_3} \sqcap B_\forall &\sqsubseteq \forall s.A^1_{f_1(\sigma_1,\sigma_2,\sigma_3)} \sqcap \forall s.A^2_{f_2(\sigma_1,\sigma_2,\sigma_3)} \\ (V = 2^n - 1) \sqcap (U < 2^n - 2) \sqcap C_{\sigma_1,\sigma_2,\sigma_3} \sqcap B^i_\exists &\sqsubseteq \forall s.A^i_{f_i(\sigma_1,\sigma_2,\sigma_3)} \end{aligned}$$

At this point, the importance of the superscript in $B^*_\exists$ becomes apparent: since different cells of a configuration are synchronized in different trees T_k the superscript makes sure that all trees rely on the same choice for existential configurations. The concept names A^i_σ are used as markers (not in Σ) and are propagated along s for 2^n steps, exploiting the V-counter. The superscript $i \in \{1, 2\}$ determines the successor configuration that the symbol is referring to. After crossing the end of a configuration, the symbol σ is propagated using concept names A'_σ (the superscript is not needed anymore because the branching happens at the end of the configuration, based on Z).

$$\begin{aligned} (U < 2^n - 1) \sqcap A^i_\sigma &\sqsubseteq \forall s.A^i_\sigma \\ (U = 2^n - 1) \sqcap B_\forall \sqcap A^1_\sigma &\sqsubseteq \forall s.(Z \to A'_\sigma) \\ (U = 2^n - 1) \sqcap B_\forall \sqcap A^2_\sigma &\sqsubseteq \forall s.(\neg Z \to A'_\sigma) \\ (U = 2^n - 1) \sqcap B^i_\exists \sqcap A^i_\sigma &\sqsubseteq \forall s.A'_\sigma && i \in \{1, 2\} \\ (V < 2^n - 1) \sqcap A'_\sigma &\sqsubseteq \forall s.A'_\sigma \\ (V = 2^n - 1) \sqcap A'_\sigma &\sqsubseteq \forall s.A_\sigma \end{aligned}$$

For those (q, a) with $\Delta(q, a) = \emptyset$, we add the CI

$$A_{q,a} \sqsubseteq \bot.$$

The following lemma establishes correctness of the reduction.

Lemma 7.2. *The following conditions are equivalent:*

(1) M accepts w;
(2) there exist models $\mathcal{I}$ and $\mathcal{J}$ of $\mathcal{O}$ such that $\mathcal{I}, b^\mathcal{I} \sim_{\mathcal{ALCO},\Sigma} \mathcal{J}, d$, for some $d \neq b^\mathcal{J}$.

Proof. "1 ⇒ 2". If M accepts w, there is a computation tree of M on w. We construct a single interpretation $\mathcal{I}$ with $\mathcal{I}, b^{\mathcal{I}} \sim_{\mathcal{ALCO},\Sigma} \mathcal{I}, d$ for some $d \neq b^{\mathcal{I}}$ as follows. Let $\widehat{\mathcal{J}}$ be the infinite tree-shaped interpretation that represents the computation tree of M on w as described above, that is, configurations are represented by sequences of 2^n elements linked by role s and labeled by $B_\forall, B^1_\exists, B^2_\exists$ depending on whether the configuration is universal or existential, and in the latter case the superscript indicates which choice has been made for the existential state. Finally, the first element of the first successor configuration of a universal configuration is labeled with Z. Observe that $\widehat{\mathcal{J}}$ interprets only the symbols in Σ as non-empty. Now, we obtain interpretations $\mathcal{I}_k$, $k < 2^n$ from $\widehat{\mathcal{J}}$ by interpreting non-Σ-symbols as follows:

— the root of $\mathcal{I}_k$ satisfies I_s;
— the U-counter starts at 0 at the root and counts modulo 2^n along each s-path;
— the V-counter starts at k at the root and counts modulo 2^n along each s-path;
— the auxiliary concept names of the shape A^i_σ and A'_σ are interpreted in a minimal way so as to satisfy the CIs starting from CI (†). Note that, by definition of these CIs, there is a unique result.

Now obtain $\mathcal{I}$ from $\widehat{\mathcal{J}}$ and the $\mathcal{I}_k$ by creating an infinite outgoing r-path ρ from some element $d \neq b^{\mathcal{I}}$ (with the corresponding A-counter) and adding $\mathcal{I}_k$, $k < 2^n$ to every element with A-counter value k on the r-path, identifying the roots of the $\mathcal{I}_k$ with the element on the path. Additionally, include $(b^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$ and add $\widehat{\mathcal{J}}$ to $\mathcal{I}$ by identifying $b^{\mathcal{I}}$ with the root of $\widehat{\mathcal{J}}$. It should be clear that $\mathcal{I}$ is as required. In particular, the reflexive, transitive, and symmetric closure of

— all pairs $(b^{\mathcal{I}}, e)$, with e on ρ, and
— all pairs (e, e'), with e in $\widehat{\mathcal{J}}$ and e' a copy of e in some tree $\mathcal{I}_k$

is an $\mathcal{ALCO}(\Sigma)$-bisimulation S on $\mathcal{I}$ with $(b^{\mathcal{I}}, d) \in S$.

"2 ⇒ 1". Assume that $\mathcal{I}, b^{\mathcal{I}} \sim_{\mathcal{ALCO},\Sigma} \mathcal{J}, d$ for some $d \neq b^{\mathcal{J}}$. As argued above, due to the r-self loop at $b^{\mathcal{I}}$, from d there has to be an outgoing infinite r-path on which all s-trees are $\mathcal{ALCO}(\Sigma)$-bisimilar. Since $\mathcal{I}$ is a model of $\mathcal{O}$, all these s-trees are additionally labeled with some auxiliary concept names not in Σ, depending on the distance from their roots on ρ. Using the CIs in $\mathcal{O}$ and the arguments given in their description, it can be shown that all s-trees contain a computation tree of M on input w (which is solely represented with concept names in Σ). □

The same ontology $\mathcal{O}$ can be used for the remaining DLs with nominals. For $\mathcal{ALCO}^u$, exactly the same proof works; in particular, note that both the bisimulation S constructed in "1 ⇒ 2" and its inverse are surjective. For the DLs with inverse roles, the (one-way) infinite r-path ρ has to replaced by a two-way infinite path in "1 ⇒ 2".

Using the ontology $\mathcal{O}$ defined above, we define a new ontology $\mathcal{O}'$ to obtain the 2ExpTime lower bound for signatures $\Sigma' = \text{sig}(\mathcal{O}') \setminus \{b\}$. Fix a role name r_E for any concept name $E \in \text{sig}(\mathcal{O}) \setminus \Sigma$. Now replace in $\mathcal{O}$ any occurrence of $E \in \text{sig}(\mathcal{O}) \setminus \Sigma$ by $\exists r_E.\{b\}$ and denote the resulting ontology by $\mathcal{O}'$.

Lemma 7.3. *The following conditions are equivalent:*

(1) M accepts w;
(2) there exist models $\mathcal{I}$ and $\mathcal{J}$ of $\mathcal{O}'$ such that $\mathcal{I}, b^{\mathcal{I}} \sim_{\mathcal{ALCO},\Sigma'} \mathcal{J}, d$, for some $d \neq b^{\mathcal{J}}$.

Proof. "1 ⇒ 2". We modify the interpretation $\mathcal{I}$ defined in the proof of Lemma 7.2 in such a way that we obtain a model of $\mathcal{O}'$ and such that the $\mathcal{ALCO}(\Sigma)$-bisimulation S on $\mathcal{I}$ defined in that proof is, in fact, an $\mathcal{ALCO}(\Sigma')$-bisimulation on the new interpretation. Formally, obtain $\mathcal{I}'$ from $\mathcal{I}$ by interpreting every r_E, $E \in \mathrm{sig}(\mathcal{O}) \setminus \Sigma$ as follows:

(i) there is an r_E-edge from e to $b^\mathcal{I}$, for all $e \in E^\mathcal{I}$;
(ii) there is an r_E-edge from e to all elements on the path ρ, for all $(e, e') \in S$ and $e' \in E^\mathcal{I}$;
(iii) there are no more r_E-edges.

Note that, by (i), $\mathcal{I}'$ is a model of $\mathcal{O}'$. By (ii), the relation S defined in the proof of Lemma 7.2 is an $\mathcal{ALCO}(\Sigma')$-bisimulation. In particular, by (i), elements $e' \in E^\mathcal{I}$ have now an r_E-edge to $b^\mathcal{I}$, so any element e bisimilar to e', that is, $(e, e') \in S$, needs an r_E-successor to some element bisimilar to $b^\mathcal{I}$. Since all elements on the path ρ are bisimilar to $b^\mathcal{I}$, these r_E-successors exist due to (ii).

"2 $\Rightarrow$ 1". This direction remains the same as in the proof of Lemma 7.2. □

The extension to DLs with inverse roles and the universal role and the restriction to a single role name are again straightforward.

We conclude the section with an observation that will be relevant for the application of our results to modal logic in Section 11. More specificially, we strengthen the lower bound for the case of $\mathcal{L} = \mathcal{ALCO}^u$ as follows:

Lemma 7.4. *Let $\mathcal{L} \in \{\mathcal{ALCO}, \mathcal{ALCO}^u\}$. Then, it is* 2ExpTime*-hard to decide for an $\mathcal{L}$-ontology $\mathcal{O}$, individual name b, and signature $\Sigma \subseteq sig(\mathcal{O}) \setminus \{b\}$ whether $\{b\}$ and $\neg\{b\}$ are jointly consistent under $\mathcal{O}$ modulo $\mathcal{L}(\Sigma)$-bisimulations, even if $\mathcal{O}$ is allowed to use only a single role name.*

Proof. We modify the ontology $\mathcal{O}$ and signature Σ used in the proof of Lemma 7.1. Let $\mathcal{O}'$ be the ontology obtained from $\mathcal{O}$ by:

— replacing every subconcept of the shape $\exists r.C$ with $\exists r.(X_r \sqcap C)$ and
— replacing every subconcept of the shape $\exists s.C$ with $\exists r.(X_s \sqcap C)$,

for fresh concept names X_r, X_s, and set $\Sigma' = \Sigma \cup \{X_r, X_s\}$. It is routine to verify that Lemma 7.2 holds for $\mathcal{O}', \Sigma'$ instead of $\mathcal{O}, \Sigma$. In particular, we can obtain an interpretation $\mathcal{I}'$ from $\mathcal{I}$ as constructed in "1 $\Rightarrow$ 2" as follows.

— replace all s-connections by r-connections;
— every element that has an s-predecessor in $\mathcal{I}$ satisfies X_s in $\mathcal{I}'$, that is, $X_s^{\mathcal{I}'} = (\exists s^-.\top)^\mathcal{I}$;
— $b^\mathcal{I}$ and every element on the infinite r-path ρ in $\mathcal{I}$ satisfy X_r in $\mathcal{I}'$, that is, $X_r^{\mathcal{I}'} = (\exists r.\top)^\mathcal{I}$ (the root of the infinite path has to satisfy X_r since it is bisimilar to $b^\mathcal{I}$ which satisfies X_r).

□

7.2 Shape of Explicit Definitions in the Lower Bound

The goal of this subsection is to provide some intuition on the shape of the explicit definitions that arise in the proof of Lemma 7.1. We note first that $r(x, x)$ is an explicit FO(Σ)-definition of $\{b\}$ under $\mathcal{O}$, regardless of whether the ATM accepts its input or not. This means that interpolant and explicit definition existence is 2ExpTime-hard even under the promise that a fixed FO-definition/FO-interpolant exists.

We now analyze the $\mathcal{ALCO}(\Sigma)$-definitions that arise in the proof of Lemma 7.1. Recall that such a definition exists iff the ATM M does not accept its input w. So, for the rest of the dicussion, we assume the latter. Instead of directly providing an explicit $\mathcal{ALCO}(\Sigma)$-definition of $\{b\}$, we give a definition $C_{\neg b}$ of $\neg\{b\}$, since the definition of $C_{\neg b}$ is close to the intuitions provided in the proof of Lemma 7.1. Obviously, $\neg C_{\neg b}$ will be the desired definition of $\{b\}$. Let n be the length of the input word w and let $k = |\Gamma \cup (Q \times \Gamma)|$ be the number of possible labelings of a cell in some configuration of the ATM. Moreover, set $K = k^{2^n} + 2^n$.

The concept $C_{\neg b}$ takes the shape

$$C_{\neg b} = \exists r.\top \rightarrow \left(C_{\text{tree}} \sqcap C_{\text{start}} \sqcap \neg C_{\text{stop}} \sqcap \bigsqcup_{i=0}^{2^n-1} C_i \right).$$

To understand the structure $\exists r.\top \rightarrow C'$ of $C_{\neg b}$, recall that the proof of Lemma 7.1 relies on the assumption that an element $d \neq b^{\mathcal{I}}$ has an r-successor. The concepts $C_{\text{tree}}, C_{\text{stop}}, C_{\text{start}}, C_i$ provide an "approximation" of an accepting computation tree of the ATM M on its input w in the following sense (Note that the definition of $\neg\{b\}$ cannot describe the full accepting computation since it is not entailed).

The concept C_{tree} enforces an s-tree of depth K that acts as the skeleton for encoding (an initial fragment of) a computation tree. It is labeled with concepts $Z, B_\forall, B_\exists^1, B_\exists^2$ in the expected way. Formally, C_{tree} is

$$\text{Path}_{s,B_\forall}^{2^n} \sqcap \bigsqcap_{\substack{i=\ell\cdot 2^n-1\\ i<K}} \forall s^i. \left(B_\forall \rightarrow \left(\exists s.(Z \sqcap \text{Path}_{s,B_\exists^1}^{2^n}) \sqcap \exists s.(\neg Z \sqcap \text{Path}_{s,B_\exists^2}^{2^n}) \right) \sqcap (B_\exists^1 \sqcup B_\exists^2) \rightarrow \exists s.\text{Path}_{s,B_\forall}^{2^n} \right)$$

where $\text{Path}_{s,X}^m$ is a concept that enforces an s-path of length m with each element labeled with X. We refrain from giving the precise definitions of the remaining concepts, and rather provide the intuitions. C_{start} is a concept that enforces the initial configuration to be true in the computation tree, and C_{stop} is a concept that is true if some element within K s-steps is labeled with a concept name $A_{q,a}$ for which $\Delta(q, a) = \emptyset$. Moreover, each C_i is a concept with $\mathcal{O} \models I_s \sqcap (A = i) \sqsubseteq C_i$; recall that we denote with $(A = i)$ that the A-counter has value i. The disjunction over all possible C_i in $C_{\neg b}$ is needed since the A-counter can take any value between 0 and $2^n - 1$ at a given element in $d \neq b^{\mathcal{I}}$. More precisely, each C_i is a conjunction

$$C_i = \bigsqcap_{j=0}^{2^n-1} \forall r^j.C_{\text{sync}}^{i \oplus_{2^n} j},$$

where $\oplus_m$ denotes addition modulo m, and for each m with $0 \leq m < 2^n$, C_{sync}^m is a concept that coordinates the content of the mth cell in every configuration in the computation tree with the same cell in the successor configuration(s). This can be easily realized using value restrictions $\forall s$.

Observe that $\mathcal{O} \models \neg\{b\} \sqsubseteq C_{\neg b}$ regardless of whether the ATM accepts w or not. In particular, in every model of $\mathcal{O}$, each element d satisfying $\neg\{b\} \sqcap \exists r.\top$ satisfies the concepts $C_{\text{tree}}, C_{\text{start}}$, and $\neg C_{\text{stop}}$. Moreover, d satisfies I_s and $(A = i)$ for some i, and thus d also satisfies C_i.

For the converse, $\mathcal{O} \models C_{\neg b} \sqsubseteq \neg\{b\}$, suppose that $C_{\neg b}$ is realizable in a model $\mathcal{I}$ of $\mathcal{O}$ in an element d with $(d, d) \in r^{\mathcal{I}}$. We thus also have $d \in (C_{\text{tree}} \sqcap C_{\text{start}} \sqcap \neg C_{\text{stop}})^{\mathcal{I}}$, and $d \in C_i^{\mathcal{I}}$, for some i. Due to the r-self loop, $d \in (C_{\text{sync}}^m)^{\mathcal{I}}$, for all m with $0 \leq m < 2^n$. But this means that at d starts the initial segment of a computation tree of M which is not labeled with a halting configuration, and all of whose cells are coordinated with the corresponding cell of the successor configuration(s). By the choice of K, on every path there is a configuration that occurs twice. We can thus extend the initial fragment of the computation tree to an infinite computation tree for the word w, in contradiction to the fact that M does not accept w.

We conclude with observing that the size of the definition $C_{\neg b}$ of $\neg\{b\}$ is double exponential in the length n of the input word, due to the depth K of the enforced tree. This is in stark contrast with the (constant!) size of the FO(Σ)-definition. We conjecture that one can enforce explicit definitions of triple exponential size. For example, when using two roles s_1, s_2 instead of s for encoding the computation tree, already the concept C_{tree} will be of triple exponential size. We leave a detailed analysis for future work.

7.3 DLs with Role Inclusions

By Theorem 5.8, it suffices to prove the following.

LEMMA 7.5. *Let $\mathcal{L} \in DL_{nr}$ admit RIs. It is* 2EXPTIME*-hard to decide for an $\mathcal{L}$-ontology O, concept C, and signature $\Sigma \subseteq sig(O)$ whether C and $\neg C$ are jointly consistent under O modulo $\mathcal{L}(\Sigma)$-bisimulations.*

As in the proof of Lemma 7.1, we reduce the word problem for exponentially space bounded ATMs, so let M be a 2^n-space bounded ATM and $w = a_0 \dots a_{n-1}$ an input of length n. In fact, the only difference to the proof of Lemma 7.1 is the way in which we enforce that exponentially many elements are $\mathcal{L}(\Sigma)$-bisimilar. We first provide the reduction for $\mathcal{L} = \mathcal{ALCH}$ and

$$\Sigma = \{r_1, r_2, s, Z, B_\forall, B^1_\exists, B^2_\exists\} \cup \{A_\sigma \mid \sigma \in \Gamma \cup (Q \times \Gamma)\}.$$

The symbols $s, Z, B_\forall, B^1_\exists, B^2_\exists$ and A_σ, $\sigma \in \Gamma \cup (Q\times\Gamma)$, play exactly the same role as above. The main difference is that we replace the nominal b by an r-chain of length n. The ontology O contains the RIs $r \sqsubseteq r_1, r \sqsubseteq r_2$ and the CI $\neg\exists r^n.\top \sqcap \exists r_1^n.\top \sqsubseteq R$. As usual $\exists r^n$ abbreviates a sequence of n times $\exists r$.

To see how we use these inclusions, suppose there exist models $\mathcal{I}$ and $\mathcal{J}$ of O and $d \in \Delta^\mathcal{I}$, $e \in \Delta^\mathcal{J}$ such that

- $d \in (\exists r^n.\top)^\mathcal{I}$;
- $e \in (\neg\exists r^n.\top)^\mathcal{J}$;
- $\mathcal{I}, d \sim_{\mathcal{ALCH},\Sigma} \mathcal{J}, e$.

then it follows that $e \in R^\mathcal{J}$: due to $\mathcal{I}, d \sim_{\mathcal{ALC},\Sigma} \mathcal{J}, e$ and $d \in (\exists r_1^n.\top)^\mathcal{I}$, we also have $e \in (\exists r_1^n.\top)^\mathcal{I}$. Let now d' be an element reachable from d via an r-path of length n (which exists due to $d \in (\exists r^n.\top)^\mathcal{I}$). Since $r \sqsubseteq r_i$ for $i = 1, 2$, there are also arbitrary r_1/r_2-paths of length n from d to d'. Since $\mathcal{I}, d \sim_{\mathcal{ALC},\Sigma} \mathcal{J}, e$, there are also arbitrary r_1/r_2-paths of length n starting in e and whose end points are all $\mathcal{ALCH}(\Sigma)$-bisimilar to d' and thus also mutually $\mathcal{ALCH}(\Sigma)$-bisimilar. The concept name R will enforce that

(∗) the end point of any r_1/r_2-path of length n starting in e carries a counter value that describes the path in a canonical way.

We can thus use these 2^n different, but bisimilar end points to start the infinite trees which mimick the computation tree of M as in the proof of Lemma 7.1. Along these, we maintain the same two counters as there:

- one counter starting at 0 and counting modulo 2^n to divide the tree into configurations of length 2^n;
- another counter starting at *the value of the counter on the leaf* and also counting modulo 2^n.

Formally, the ontology O is constructed as follows. In order to realize (∗) above, we use concept names A_i, $0 \le i < n$ realizing the counter and the following CIs:

$$
\begin{array}{rcll}
R & \sqsubseteq & R_0 & \\
R_i & \sqsubseteq & \forall r_1.(A_i \sqcap R_{i+1}) \sqcap \forall r_2.(\neg A_i \sqcap R_{i+1}) & i < n \\
R_i \sqcap A_j & \sqsubseteq & \forall r_1.A_j \sqcap \forall r_2.A_j & 0 \le j < i < n \\
R_i \sqcap \neg A_j & \sqsubseteq & \forall r_1.\neg A_j \sqcap \forall r_2.\neg A_j & 0 \le j < i < n \\
R_n & \sqsubseteq & L_R &
\end{array}
$$

Using the concept name L_R, we start the s-trees with two counters, realized using concept names U_i and V_i, $0 \leq i < n$, and initialized to 0 and the value of the A-counter, respectively:

$$L_R \sqsubseteq (U = 0)$$
$$L_R \sqsubseteq A_j \leftrightarrow V_j \qquad 0 \leq j < n$$
$$\top \sqsubseteq \exists s.\top$$

The structure of the computation tree, the initial configuration, and the coordination between consecutive configurations is done using the same CIs as in the proof of Lemma 7.1, starting from inclusion (†) and replacing I_s with L_R. We can then prove the following very similarly to Lemma 7.2.

Lemma 7.6. *The following conditions are equivalent:*

(1) M accepts w;
(2) there exist models $\mathcal{I}$ and $\mathcal{J}$ of $\mathcal{O}$ such that $\mathcal{I}, d \sim_{\mathcal{ALCH},\Sigma} \mathcal{J}, e$, for some $d \in (\exists r^n.\top)^{\mathcal{I}}$ and $e \notin (\exists r^n.\top)^{\mathcal{J}}$.

Proof. "$1 \Rightarrow 2$". If M accepts w, there is a computation tree of M on w. We construct a single interpretation $\mathcal{I}$ with $\mathcal{I}, d \sim_{\mathcal{ALCH},\Sigma} \mathcal{I}, e$ for some d, e with $d \in (\exists r^n.\top)^{\mathcal{I}}$ and $e \notin (\exists r^n.\top)^{\mathcal{I}}$ as follows. Let $\widehat{\mathcal{J}}$ be the infinite tree-shaped interpretation that represents the computation tree of M on w as described above, that is, configurations are represented by sequences of 2^n elements linked by role s and labeled by $B_\forall, B_\exists^1, B_\exists^2$ depending on whether the configuration is universal or existential, and in the latter case the superscript indicates which choice has been made for the existential state. Finally, the first element of the first successor configuration of a universal configuration is labeled with Z. Observe that $\widehat{\mathcal{J}}$ interprets only the symbols in Σ as non-empty. Now, we obtain interpretations $\mathcal{I}_k$, $k < 2^n$ from $\widehat{\mathcal{J}}$ by interpreting non-Σ-symbols as follows:

— the root of $\mathcal{I}_k$ satisfies L_R;
— the U-counter starts at 0 at the root and counts modulo 2^n along each s-path;
— the V-counter starts at k at the root and counts modulo 2^n along each s-path;
— the auxiliary concept names of the shape A_σ^i and A'_σ are interpreted in a minimal way so as to satisfy the CIs that enforce the coordination between consecutive configurations (cf. the CIs in proof of Lemma 7.1).

Now obtain $\mathcal{I}$ from $\widehat{\mathcal{J}}$ and the $\mathcal{I}_k$ as follows: First, create a path of length n from some element d so that consecutive elements are connected with r, r_1, r_2, and identify the end point of the path with the root of $\widehat{\mathcal{J}}$. Then create a binary tree of depth n, rooted in e, in which left children are always r_1-successors and right children are always r_2-successors. Label the nodes of the tree with R_i and A_j as described above and identify the leaf having A-counter value k with the root of $\mathcal{I}_k$, for all $k < 2^n$. $\mathcal{I}$ is as required since, by construction, $d \in (\exists r^n.\top)^{\mathcal{I}}$, $e \notin (\exists r^n.\top)^{\mathcal{I}}$, and the reflexive, transitive, and symmetric closure of

— all pairs (d', e') such that d' has distance $\ell \leq n$ from d and e' has distance ℓ from e, and
— all pairs (b, b'), with b in $\widehat{\mathcal{J}}$ and b' a copy of b in some tree $\mathcal{I}_k$

is an $\mathcal{ALCH}(\Sigma)$-bisimulation S on $\mathcal{I}$ with $(d, e) \in S$.

"$2 \Rightarrow 1$". Assume that $\mathcal{I}, d \sim_{\mathcal{ALCH},\Sigma} \mathcal{J}, e$ for models $\mathcal{I}, \mathcal{J}$ of $\mathcal{O}$ and some d, e with $d \in (\exists r^n.\top)^{\mathcal{I}}$ and $e \notin (\exists r^n.\top)^{\mathcal{J}}$. As argued above, there are r_1/r_2-paths of length n whose end points carry all possible counters $< 2^n$ and are all $\mathcal{ALCH}(\Sigma)$-bisimilar. In addition, all these end points root s-trees which are $\mathcal{ALCH}(\Sigma)$-bisimilar. Since $\mathcal{J}$ is a model of $\mathcal{O}$, all these s-trees are additionally labeled with some auxiliary concept names not in Σ, depending on the value of the A-counter of the corresponding leaf. Using the CIs in $\mathcal{O}$ and the arguments given in their description, it can be shown that all s-trees contain a computation tree of M on input w (which is solely represented with concept names in Σ). ◻

The same proof works as well for $\mathcal{ALCH}^u$ as the relation S constructed in the direction "$1 \Rightarrow 2$" above is actually an $\mathcal{ALCH}^u(\Sigma)$-bisimulation. For $\mathcal{L} \in \{\mathcal{ALCHI}, \mathcal{ALCHI}^u\}$, we have to slightly adapt the model construction in "$1 \Rightarrow 2$", following the idea provided in Example 5.9 (except that we do not need to take the union of $\mathcal{I}_1, \mathcal{I}_2$ here, since we construct a single interpretation $\mathcal{I} = \mathcal{I}_1 = \mathcal{I}_2$). Let $d_0, \ldots, d_n$ be the elements on the r-path that starts in d, that is, $d_0 = d$ and d_ℓ has distance ℓ from d. Recall that $(d_\ell, e') \in S$ for every element e' in level ℓ in the binary tree rooted at e. Observe that S is not an $\mathcal{L}(\Sigma)$-bisimulation since, for $\ell > 0$, d_ℓ has both an r_1 and an r_2-predecessor (both are $d_{\ell-1}$), but elements in the binary tree lack either an r_1- or an r_2-predecessor. To repair this, we add for every element e' in level $\ell > 0$ in the binary tree the following connections:

$$(d_{\ell-1}, e') \in r_1^{\mathcal{I}} \quad \text{and} \quad (d_{\ell-1}, e') \in r_2^{\mathcal{I}}.$$

It can be verified that the modified interpretation is still a model of $\mathcal{O}$, and that S is an $\mathcal{L}(\Sigma)$-bisimulation as required.

We conclude the section by remarking that one can analyze the structure of the explicit $\mathcal{ALCH}(\Sigma)$-definitions that arise in the proof of Lemma 7.5 along the lines of Section 7.2. In contrast to that section, the size of the FO-definition

$$\varphi(x_1) = \exists x_2 \ldots \exists x_n. \bigwedge_{i=1}^{n-1} r_1(x_i, x_{i+1}) \wedge r_2(x_i, x_{i+1})$$

of $\exists r^n.\top$ under $\mathcal{O}$ is not constant, but depends on n.

7.4 CI-Interpolant Existence

We show the 2ExpTime lower bound for CI-interpolant existence stated in Theorem 5.16. We employ the ontology $\mathcal{O}$, individual b, and signature Σ constructed in the proof of Lemma 7.2 and remind the reader that the claim of Lemma 7.2 holds also for $\mathcal{ALCO}^u$ and $\mathcal{ALCOI}^u$. Let $\mathcal{O}_1$ be defined as $\mathcal{O}$ without $\{b\} \sqsubseteq \exists r.\{b\}$ and with $\neg\{b\} \sqcap \exists r.\top \sqsubseteq I_s$ replaced by $\exists r.\top \sqsubseteq I_s$. Also, define $\mathcal{O}_2$ as $\mathcal{O}$ without $\{b\} \sqsubseteq \exists r.\{b\}$ and with all concept and role names not in Σ replaced by fresh symbols. Transform $\mathcal{O}_2$ into an equivalent ontology of the form $\{\top \sqsubseteq D\}$. Observe that $\mathcal{O}_1$ does not use b. In fact, the shared symbols of $\mathcal{O}_1$ and the CI $\forall u.D \sqcap \{b\} \sqsubseteq \neg\exists r.\{b\}$ are exactly the symbols in Σ. The 2ExpTime lower bound now follows from the following lemma.

Lemma 7.7. *Let $\mathcal{L} \in \{\mathcal{ALCO}^u, \mathcal{ALCOI}^u\}$. Then the following conditions are equivalent:*

(1) Point 2 of Lemma 7.2 holds; that is, there exist models $\mathcal{I}$ and $\mathcal{J}$ of $\mathcal{O}$ such that $\mathcal{I}, b^{\mathcal{I}} \sim_{\mathcal{L},\Sigma} \mathcal{J}, d$, for some $d \neq b^{\mathcal{J}}$;

(2) there does not exist an $\mathcal{L}$-CI interpolant for $\mathcal{O}_1$ and $\forall u.D \sqcap \{b\} \sqsubseteq \neg\exists r.\{b\}$.

Proof. Assume Point (1) holds and take $\mathcal{I}, \mathcal{J}$, and d witnessing this. We may assume that $\mathcal{I} = \mathcal{J}$ is the interpretation constructed in the proof of "$1 \Rightarrow 2$" of Lemma 7.2. Assume for a proof by contradiction that $\mathcal{O}'$ is an $\mathcal{L}$-CI interpolant for $\mathcal{O}_1$ and $\forall u.D \sqcap \{b\} \sqsubseteq \neg\exists r.\{b\}$. Let $\mathcal{I}'$ denote the restriction of $\mathcal{I}$ to elements that cannot be reached from b along a path following $r^{\mathcal{I}}$ or $s^{\mathcal{I}}$ and reinterpret b as an element of $\Delta^{\mathcal{I}'}$. Then $\mathcal{I}'$ is a model of $\mathcal{O}_1$ by the definition of $\mathcal{I}$ and since $\mathcal{O}_1$ does not contain any CIs with the individual b. Moreover, we have $\mathcal{I}, b^{\mathcal{I}} \sim_{\mathcal{L},\Sigma} \mathcal{I}', d$ since $b \notin \Sigma$. Then, as $\mathcal{L}$ admits the universal role, $\mathcal{I}$ is a model of $\mathcal{O}'$. We now reinterpret in $\mathcal{I}$ the fresh concept and role names in $\mathcal{O}_2$ in the same way as the original ones in $\mathcal{I}$ and obtain a model $\mathcal{I}''$ with $\Delta^{\mathcal{I}} = D^{\mathcal{I}''}$ since $\mathcal{I}$ is a model of $\mathcal{O}$. But then $\mathcal{I}'' \not\models \forall u.D \sqcap \{b\} \sqsubseteq \neg\exists r.\{b\}$ and so (as $\mathcal{I}''$ is still a model of $\mathcal{O}'$ since $\mathrm{sig}(\mathcal{O}') \subseteq \Sigma$) $\mathcal{O}' \not\models \forall u.D \sqcap \{b\} \sqsubseteq \neg\exists r.\{b\}$, a contradiction.

Conversely, assume there does not exist an $\mathcal{L}$-CI interpolant for $\mathcal{O}_1$ and $\forall u.D \sqcap \{b\} \sqsubseteq \neg\exists r.\{b\}$. As in the proof of Theorem 5.6 and using the fact that $\mathcal{L}$ admits the universal role, we obtain a model $\mathcal{J}$ of $\mathcal{O}_1$, $d \in \Delta^{\mathcal{J}}$, and an interpretation $\mathcal{I}$ with $\mathcal{I} \not\models \forall u.D \sqcap \{b\} \sqsubseteq \neg\exists r.\{b\}$ and $\mathcal{I}, b^{\mathcal{I}} \sim_{\mathcal{L},\Sigma} \mathcal{J}, d$. We may assume that $\mathcal{I}$ and $\mathcal{J}$ are disjoint. Observe that $\mathcal{J}$ satisfies all CIs in $\mathcal{O}$ with the exception of $\{b\} \sqsubseteq \exists r.\{b\}$. By reinterpreting in $\mathcal{I}$ the original concept and role names in $\mathcal{O}$ in the same way as the fresh concept and role names in $\mathcal{O}_2$, we obtain a model $\mathcal{I}'$ of $\mathcal{O}$. Take the union $\mathcal{I}' \cup \mathcal{J}$ of $\mathcal{I}'$ and $\mathcal{J}$ with $b^{\mathcal{I}' \cup \mathcal{J}}$ defined as $b^{\mathcal{I}}$. Then $\mathcal{I}' \cup \mathcal{J}$ is a model of $\mathcal{O}$ such that $\mathcal{I}' \cup \mathcal{J}, b^{\mathcal{I}' \cup \mathcal{J}} \sim_{\mathcal{L},\Sigma} \mathcal{I}' \cup \mathcal{J}, d$, for some $d \neq b^{\mathcal{J}}$, as required for Point (1). □

8 UPPER BOUND PROOFS WITHOUT ONTOLOGY

The upper bound for Points 1 and 2 of Theorem 5.13 is a consequence of the respective upper bounds in Theorem 5.12. For showing the upper bounds of Points 3 and 4 in Theorem 5.13, we prove that joint consistency is in NExpTime and then apply Theorem 5.6. Indeed, the NExpTime upper bound follows directly from the following exponential size witness model property.

Lemma 8.1. *Let $\mathcal{L} \in DL_{nr}$ admit neither the universal role nor both inverse roles and nominals simultaneously. Let $\mathcal{O}$ be a set of RIs, C_1, C_2 $\mathcal{L}$-concepts, and Σ a signature. If C_1 and C_2 are jointly consistent under $\mathcal{O}$ modulo $\mathcal{L}(\Sigma)$-bisimulations, then there exist pointed interpretations $\mathcal{I}_1, d_1$ and $\mathcal{I}_2, d_2$ with $\mathcal{I}_1, \mathcal{I}_2$ models of $\mathcal{O}$ and of at most exponential size in $||\mathcal{O}|| + ||C_1|| + ||C_2||$ such that $d_1 \in C_1^{\mathcal{I}_1}$, $d_2 \in C_2^{\mathcal{I}_2}$, and $\mathcal{I}_1, d_1 \sim_{\mathcal{L},\Sigma} \mathcal{I}_2, d_2$.*

Before we prove Lemma 8.1, we introduce some notation. The *depth* of a concept C is the number of nestings of existential restrictions in C. For instance, a concept name has depth 0 and $\exists r.\exists r.B$ has depth 2. Given the ontology $\mathcal{O}$, concepts C_1, C_2, and the signature Σ, we use the notation introduced in Section 6. For instance, the set of concepts Ξ, Ξ-types t, and mosaics (T_1, T_2) are defined as in Section 6. While in Section 6, we used the relation $\rightsquigarrow_r$ between mosaics to guide the construction of interpretations, here we use a relation between mosaics that is directly induced by interpretations. Assume interpretations $\mathcal{I}_1$ and $\mathcal{I}_2$ are given. Consider mosaics $p = (T_1(d), T_2(d))$ and $q = (T_1(d'), T_2(d'))$ such that there exists a role name $r \in \Sigma$ with $(d, d') \in r^{\mathcal{I}_i}$, for some $i \in \{1, 2\}$. Then define, for every role name s and $i \in \{1, 2\}$, relations $R^{s,i}_{p,q} \subseteq T_i(d) \times T_i(d')$ by setting $(t, t') \in R^{s,i}_{p,q}$ if there exist e and e' realizing t and t', respectively, with $(T_1(e), T_2(e)) = p$ and $(T_1(e'), T_2(e')) = q$, such that $(e, e') \in s^{\mathcal{I}_i}$.

Now assume that C_1 and C_2 are jointly consistent under $\mathcal{O}$ modulo $\mathcal{L}(\Sigma)$-bisimulations. By definition, there exist pointed models $\mathcal{I}_1, d_1$ and $\mathcal{I}_2, d_2$ of $\mathcal{O}$ such that $d_1 \in C_1^{\mathcal{I}_1}$, $d_2 \in C_2^{\mathcal{I}_2}$, and $\mathcal{I}_1, d_1 \sim_{\mathcal{L},\Sigma} \mathcal{I}_2, d_2$. Let k be the maximum depth of C_1, C_2.

We start with the case involving nominals and without inverse roles. We construct exponential size $\mathcal{J}_1, \mathcal{J}_2$ with the same properties as $\mathcal{I}_1, \mathcal{I}_2$ above. Intuitively, $\mathcal{J}_i$ is obtained via a suitable unraveling operation up to the depth k of the concepts C_1, C_2; during the unraveling, we take care of the nominals and, moreover, restrict the outdegree of the produced interpretation by keeping only necessary successors. Formally, let $\mathcal{B}$ be some minimal set of mosaics defined by $\mathcal{I}_1, \mathcal{I}_2$ such that

- $(T_1(d_1), T_2(d_1)) \in \mathcal{B}$;
- $\mathcal{B}$ contains every mosaic generated by some nominal, or formally, $(T_1(d), T_2(d)) \in \mathcal{B}$ for every $d \in \Delta^{\mathcal{I}_i}$ such that $d = a^{\mathcal{I}_i}$ for some nominal $a \in \text{sig}(C_i)$;
- for every type t realized in $\mathcal{I}_i$ there exists $(T_1, T_2) \in \mathcal{B}$ with $t \in T_i$.

Intuitively, $\mathcal{B}$ serves to describe the behavior of the root of the unraveling (first item), of the nominals (second item), and of potential witnesses for existential restrictions for non-Σ-roles (third item). Observe that the size of $\mathcal{B}$ is at most exponential in the size of $\mathcal{O}, C_1, C_2$. To restrict the

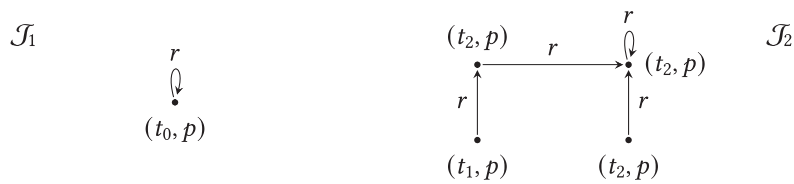

Fig. 8. Interpretations $\mathcal{J}_1$ and $\mathcal{J}_2$ illustrating Example 8.2.

outdegree, select, for any mosaic $p = (T_1, T_2)$ defined by $\mathcal{I}_1, \mathcal{I}_2$ and any $\exists s.C \in t \in T_i$ such that there exists $r \in \Sigma$ with $\mathcal{O} \models s \sqsubseteq r$, a mosaic $q = (T_1', T_2')$ such that $(t, t') \in R^{s,i}_{p,q}$ and $C \in t'$, and denote the resulting set by $\mathcal{S}(p)$. Form the set $\mathcal{T}$ of sequences

$$\sigma = p_0 \cdots p_j = (T_1^0, T_2^0) \cdots (T_1^j, T_2^j)$$

with $j \leq k$, $p_0 \in \mathcal{B}$ and $p_{i+1} \in \mathcal{S}(p_i)$ for $i < j$. Let $\text{tail}(\sigma) = p_j$ and $\text{tail}_i(\sigma) = T_i^j$. We next define the domain of $\mathcal{J}_1$ and $\mathcal{J}_2$ as

$$\begin{aligned}\Delta^{\mathcal{J}_i} &= \{(t, p) \mid t \in \text{tail}_i(p), p \in \mathcal{B}\} \\ &\quad \cup \{(t, \sigma) \mid \sigma \in \mathcal{T}, t \in \text{tail}_i(\sigma), |\sigma| > 1, t \text{ contains no nominal}\}\end{aligned}$$

and define the interpretation of individual, concept, and role names in $\mathcal{J}_1, \mathcal{J}_2$ in the expected way:

— for any individual name a and $(T_1, T_2) \in \mathcal{B}$ with $\{a\} \in t \in T_i$, we set $a^{\mathcal{J}_i} = (t, (T_1, T_2))$;
— for any concept name A, $(t, \sigma) \in A^{\mathcal{J}_i}$ iff $A \in t$;
— for any role name r, we let for $\sigma p \in \mathcal{T}$,
  – $((t, \sigma), (t', \sigma p)) \in r^{\mathcal{J}_i}$ if $(t, t') \in R^{r,i}_{\text{tail}(\sigma),p}$ and t' contains no nominal;
  – $((t, \sigma), (t', p)) \in r^{\mathcal{J}_i}$ if $(t, t') \in R^{r,i}_{\text{tail}(\sigma),p}$ and t' contains a nominal.

  Next assume that $\text{tail}(\sigma) = (T_1, T_2)$ and σ has length k. If $\text{tail}(\sigma') = (T_1, T_2)$ for some $|\sigma'| < k$, then choose as r-successors of any element of the form (t, σ) exactly the r-successors of (t, σ') defined above. If no such σ' exists, then all elements of the form $(t, \text{tail}(\sigma))$ have distance exactly k from the roots (since no nominal occurs in any type in any mosaic in σ) and no successors are added.

  It remains to take care of existential restrictions $\exists r.C$ for the role names r that do not entail any role name in Σ. If $\sigma \in \mathcal{T}$, $\exists r.C \in t \in T_i$ with $\text{tail}_i(\sigma) = T_i$ and $\mathcal{O} \not\models r \sqsubseteq s$ for any $s \in \Sigma$, we add $((t, \sigma), (t', p))$ to $r^{\mathcal{J}_i}$ (and all $s^{\mathcal{J}_i}$ with $\mathcal{O} \models r \sqsubseteq s$) for some $p = (T_1', T_2') \in \mathcal{B}$ and $t' \in T_i'$ with $C \in t'$ such that there are e, e' realizing t, t' in $\mathcal{I}_i$ and $(e, e') \in r^{\mathcal{I}_i}$.

The following example illustrates the construction of $\mathcal{J}_1, \mathcal{J}_2$ using the interpretations $\mathcal{I}_1, \mathcal{I}_2$ introduced in Example 5.7.

Example 8.2. Let $t_0 = \text{tp}_\Xi(\mathcal{I}_1, a^{\mathcal{I}_1})$, $t_1 = \text{tp}_\Xi(\mathcal{I}_2, b^{\mathcal{I}_2})$, and $t_2 = \text{tp}_\Xi(\mathcal{I}_2, d)$. We ignore the types realized by $b^{\mathcal{I}_1}$ in $\mathcal{I}_1$ and by $a^{\mathcal{I}_2}$ in $\mathcal{I}_2$ as they are not relevant for understanding the construction. Then only the mosaic $p = (T_1, T_2)$ with $T_1 = \{t_0\}$ and $T_2 = \{t_1, t_2\}$ remains and $\mathcal{J}_1$ and $\mathcal{J}_2$ are depicted in Figure 8. ⊣

We show that $\mathcal{J}_1, \mathcal{J}_2$ are as required. First, for $i \in \{1, 2\}$, $\mathcal{J}_i \models \mathcal{O}$ follows from the definition of $\mathcal{J}_i$ and the fact that $\mathcal{I}_i \models \mathcal{O}$. Indeed, given $r \sqsubseteq s \in \mathcal{O}$, let $((t, \sigma), (t', \sigma')) \in r^{\mathcal{J}_i}$. This means that $(t, t') \in R^{r,i}_{\text{tail}(\sigma),\text{tail}(\sigma')}$, that is, there exist e, e' realizing t and t', respectively, with $(T_1(e), T_2(e)) = \text{tail}(\sigma)$ and $(T_1(e'), T_2(e')) = \text{tail}(\sigma')$, such that $(e, e') \in r^{\mathcal{I}_i}$. Since $\mathcal{I}_i \models \mathcal{O}$, we obtain that $(e, e') \in s^{\mathcal{I}_i}$ as well, and thus $(t, t') \in R^{s,i}_{\text{tail}(\sigma),\text{tail}(\sigma')}$, meaning that $((t, \sigma), (t', \sigma')) \in s^{\mathcal{J}_i}$. Hence, $\mathcal{J}_i \models r \sqsubseteq s$.

We next prove that, for every $(t, \sigma) \in \Delta^{\mathcal{J}_i}$ and every concept $C \in \Xi$ of depth $\leq k - |\sigma|$,

$$(t, \sigma) \in C^{\mathcal{J}_i} \text{ iff } C \in t.$$

The proof is by induction on the structure of C. We consider the case $C = \exists r.D$, where D has depth $< k - |\sigma|$. We can assume that $|\sigma| < k$, since for $|\sigma| = k$ the claim holds trivially.

($\Rightarrow$) Let $(t, \sigma) \in \exists r.D^{\mathcal{J}_i}$. Then $\exists r.D \in t$ follows by construction of $r^{\mathcal{J}_i}$ as we only have $((t, \sigma), (t', \sigma') \in r^{\mathcal{J}_i}$ if there are e, e' realizing t, t' in $\mathcal{I}_i$ such that $(e, e') \in r^{\mathcal{I}_i}$.

($\Leftarrow$) Let $\mathrm{tail}(\sigma) = p = (T_1, T_2)$ and suppose that $\exists r.D \in t \in T_i$. We distinguish two cases.

— There exists $s \in \Sigma$ such that $\mathcal{O} \models r \sqsubseteq s$. Then, there exists $q = (T_1', T_2') \in \mathcal{S}(p)$ and $t' \in T_i'$ such that $(t, t') \in R^{r,i}_{p,q}$ and $D \in t'$. We distinguish two cases.
  – t' does not contain nominals. Then we have that $((t, \sigma), (t', \sigma q)) \in r^{\mathcal{J}_i}$. By inductive hypothesis, $(t', \sigma q) \in D^{\mathcal{J}_i}$, and thus $(t, \sigma) \in \exists r.D^{\mathcal{J}_i}$.
  – t' contains a nominal. Then we have that $((t, \sigma), (t', q)) \in r^{\mathcal{J}_i}$. By inductive hypothesis, $(t', q) \in D^{\mathcal{J}_i}$, hence $(t, \sigma) \in \exists r.D^{\mathcal{J}_i}$.

— For every $s \in \Sigma$, $\mathcal{O} \not\models r \sqsubseteq s$. By definition of $\mathcal{J}_i$, we have $((t, \sigma), (t', q)) \in r^{\mathcal{J}_i}$, for some $q = (T_1', T_2') \in \mathcal{B}$ and $t' \in T_i'$ such that $D \in t'$. By inductive hypothesis, $(t', q) \in D^{\mathcal{J}_i}$. Thus, $(t, \sigma) \in \exists r.D^{\mathcal{J}_i}$.

Next, observe that the relation

$$S = \{((t, \sigma), (t', \sigma')) \in \Delta^{\mathcal{J}_1} \times \Delta^{\mathcal{J}_2} \mid \mathrm{tail}(\sigma) = \mathrm{tail}(\sigma')\}$$

is an $\mathcal{ALCHO}(\Sigma)$-bisimulation. Indeed, for $((t, \sigma), (t', \sigma')) \in S$, we have the following.

[AtomC] Let $(t, \sigma) \in A^{\mathcal{J}_1}$ and $A \in \Sigma$. By definition of $\mathcal{J}_1$, we have that $(t, \sigma) \in A^{\mathcal{J}_1}$ iff $A \in t \in \mathrm{tail}_1(\sigma)$, and thus $A \in t' \in \mathrm{tail}_2(\sigma) = \mathrm{tail}_2(\sigma')$, by definition of mosaics. But then $(t', \sigma') \in A^{\mathcal{J}_2}$. The converse direction is analogous.

[AtomI] Let $(t, \sigma) = a^{\mathcal{J}_1}$ and $a \in \Sigma$. By definition of $\mathcal{J}_1$, $(t, \sigma) = a^{\mathcal{J}_1}$ iff $\{a\} \in t \in \mathrm{tail}_1(\sigma)$, and thus $\{a\} \in t' \in \mathrm{tail}_2(\sigma) = \mathrm{tail}_2(\sigma')$, by definition of mosaics. But then $(t', \sigma') = a^{\mathcal{J}_2}$.

[Forth] Suppose that $((t, \sigma), (\hat{t}, \hat{\sigma})) \in r^{\mathcal{J}_1}$ with $r \in \Sigma$.

First, consider the case with $|\sigma|, |\sigma'| < k$. We have two possibilities.

— $\hat{t}$ does not contain nominals. The following proof is illustrated in Figure 9. There is a mosaic p with $\hat{\sigma} = \sigma p$, and from $((t, \sigma), (\hat{t}, \sigma p)) \in r^{\mathcal{J}_1}$ we obtain $(t, \hat{t}) \in R^{r,1}_{\mathrm{tail}(\sigma), p}$. This means that there exist $d, \hat{d}$ realizing t and $\hat{t}$, respectively, with $(T_1(d), T_2(d)) = \mathrm{tail}(\sigma)$ and $(T_1(\hat{d}), T_2(\hat{d})) = p$, such that $(d, \hat{d}) \in r^{\mathcal{I}_1}$. As $t' \in T_2(d)$, there exists $e \in \Delta^{\mathcal{I}_2}$ with $\mathcal{I}_1, d \sim_{\mathcal{ALCHO},\Sigma} \mathcal{I}_2, e$ and e realizes t'. By the definition of bisimulations, there exists $\hat{e}$ with $(e, \hat{e}) \in r^{\mathcal{I}_2}$ and $\mathcal{I}_1, \hat{d} \sim_{\mathcal{ALCHO},\Sigma} \mathcal{I}_2, \hat{e}$. Assume that $\hat{e}$ realizes $\hat{t}'$. Then $\hat{t}' \in T_2(\hat{d})$ and $(\hat{t}, \hat{t}') \in R^{r,2}_{\mathrm{tail}(\sigma'), p}$. Now we consider again two possibilities.
  – $\hat{t}'$ does not contain nominals. Then from $(t', \hat{t}') \in R^{r,2}_{\mathrm{tail}(\sigma'), p}$ we obtain $((t', \sigma'), (\hat{t}', \sigma' p)) \in r^{\mathcal{J}_2}$. Since $\mathrm{tail}(\sigma p) = \mathrm{tail}(\sigma' p)$, we also obtain $((\hat{t}, \sigma p), (\hat{t}', \sigma' p)) \in S$.
  – $\hat{t}'$ contains nominals. Then from $(t', \hat{t}') \in R^{r,2}_{\mathrm{tail}(\sigma'), p}$ we obtain $((t', \sigma'), (\hat{t}', p)) \in r^{\mathcal{J}_2}$. Since $\mathrm{tail}(\sigma p) = p$, we get $((\hat{t}, \sigma p), (\hat{t}', p)) \in S$ as well.

  In both cases, we obtain some $(\hat{t}', \hat{\sigma}')$ with $((t', \sigma'), (\hat{t}', \hat{\sigma}')) \in r^{\mathcal{J}_2}$ and $((\hat{t}, \hat{\sigma}), (\hat{t}', \hat{\sigma}')) \in S$, as required.

— $\hat{t}$ contains nominals. In this case, $\hat{\sigma} = p$ for some mosaic p, and from $((t, \sigma), (\hat{t}, p)) \in r^{\mathcal{J}_1}$ we obtain $(t, \hat{t}) \in R^{r,1}_{\mathrm{tail}(\sigma), p}$. Now we can reason as above.

Now consider the case with $|\sigma| = k$ and $|\sigma'| < k$. As $\mathrm{tail}(\sigma) = \mathrm{tail}(\sigma')$, there exists σ'' such that $|\sigma''| < k$ and $\mathrm{tail}(\sigma'') = \mathrm{tail}(\sigma)$ and the r-successors of any node of the form (t, σ) are

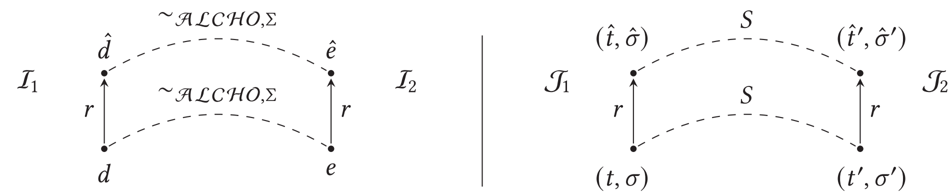

Fig. 9. Proof step to show that S satisfies [Forth], with $\hat{\sigma} = \sigma p$ and $\hat{\sigma}' = \sigma' p$ (if $\hat{t}'$ does not contain nominals) and $\hat{\sigma}' = p$ (if $\hat{t}'$ contains nominals).

exactly the r-successors of (t, σ''), and thus to show [Forth] one can proceed as above. The same argument applies if $|\sigma| < k$ and $|\sigma'| = k$ and if $|\sigma| = |\sigma'| = k$ and there exists σ'' with $\text{tail}(\sigma'') = \text{tail}(\sigma') = \text{tail}(\sigma)$ and $|\sigma''| < k$. Finally, if $|\sigma| = |\sigma'| = k$ but there does not exist any σ'' with $\text{tail}(\sigma'') = \text{tail}(\sigma') = \text{tail}(\sigma)$ and $|\sigma''| < k$, then there are no r-successors to consider.

[Back] Dual to [Forth].

Observe that the models $\mathcal{J}_i$, $i = 1, 2$, are at most exponential in the size of $\mathcal{O}, C_1, C_2$. Moreover, we have $(T_1(d_1), (T_2(d_1)) \in \mathcal{B}$ and so $(\text{tp}_\Xi(\mathcal{I}_1, d_1), T_1(d_1)) \in C_1^{\mathcal{J}_1}$, $(\text{tp}_\Xi(\mathcal{I}_2, d_2), T_2(d_1)) \in C_2^{\mathcal{J}_2}$, and

$$((\text{tp}_\Xi(\mathcal{I}_1, d_1), T_1(d_1)), (\text{tp}_\Xi(\mathcal{I}_2, d_2), T_2(d_1)) \in S,$$

as required.

We next consider the case with inverse roles, but without nominals. In this case, we let $\mathcal{B}$ be some minimal set of mosaics defined by $\mathcal{I}_1, \mathcal{I}_2$ containing $(T_1(d_1), T_2(d_1))$ and such that for every type t realized in $\mathcal{I}_i$ there exists $(T_1, T_2) \in \mathcal{B}$ with $t \in T_i$. We extend the relations $R^{s,i}_{p,q}$ defined previously to inverse roles s in the obious way and select for any mosaic $p = (T_1, T_2)$ and any $\exists s.C \in t \in T_i$ such that there exists a Σ-role r with $\mathcal{O} \models s \sqsubseteq r$ a mosaic $q = (T_1', T_2')$ such that $(t, t') \in R^{s,i}_{p,q}$ and $C \in t'$ and denote the resulting set by $\mathcal{S}(p)$.

Form again the set $\mathcal{T}$ of sequences

$$\sigma = p_0 \cdots p_j = (T_1^0, T_2^0) \cdots (T_1^j, T_2^j)$$

with $j \leq k$, $p_0 \in \mathcal{B}$ and $p_{i+1} \in \mathcal{S}(p_i)$ for $i < j$. Let $\text{tail}(\sigma) = p_j$ and $\text{tail}_i(\sigma) = T_i^j$. We next define the domain of $\mathcal{J}_1$ and $\mathcal{J}_2$ as

$$\Delta^{\mathcal{J}_i} = \{(t, \sigma) \mid \sigma \in \mathcal{T}, t \in \text{tail}_i(\sigma)\}$$

We define interpretations $\mathcal{J}_1, \mathcal{J}_2$ in the expected way.

— For any concept name A, $(t, \sigma) \in A^{\mathcal{J}_i}$ iff $A \in t$;
— Let r be a role name. Then we let for $\sigma p \in \mathcal{T}$,
  – $((t, \sigma), (t', \sigma p)) \in r^{\mathcal{J}_i}$ if $(t, t') \in R^{r,i}_{\text{tail}(\sigma),p}$;
  – $((t', \sigma p), (t, \sigma)) \in r^{\mathcal{J}_i}$ if $(t, t') \in R^{r^-,i}_{\text{tail}(\sigma),p}$.
— We still have to take care of existential restrictions $\exists r.C$ with r a role that does not entail any Σ-role. If $\sigma \in \mathcal{T}$, $\exists r.C \in t \in T_i$ with $\text{tail}_i(\sigma) = T_i$ and $\mathcal{O} \not\models r \sqsubseteq s$ for any Σ-role s, we add $((t, \sigma), (t', p))$ to $r^{\mathcal{J}_i}$ (and all $s^{\mathcal{J}_i}$ with $\mathcal{O} \models r \sqsubseteq s$) for some $p = (T_1', T_2') \in \mathcal{B}$ and $t' \in T_i'$ with $C \in t'$ such that there are e, e' realizing t, t' in $\mathcal{I}_i$ and $(e, e') \in r^{\mathcal{I}_i}$.

The fact that $\mathcal{J}_i \models \mathcal{O}$, for $i \in \{1, 2\}$, is proved similarly to the case with nominals. One can also prove again by induction on the structure of C that for every $(t, \sigma) \in \Delta^{\mathcal{J}_i}$ and every $C \in \Xi$ of depth $\leq k - |\sigma|$,

$$(t, \sigma) \in C^{\mathcal{J}_i} \text{ iff } C \in t.$$

Next, we observe that the relation

$$S = \{((t,\sigma),(t',\sigma)) \in \Delta^{\mathcal{J}_1} \times \Delta^{\mathcal{J}_2} \mid \sigma \in \mathcal{T}\}$$

is an $\mathcal{ALCHI}(\Sigma)$-bisimulation. Indeed, it can be seen, similar to the case with nominals, that S satisfies [AtomC]. We now give a proof of [Forth]. We provide the proof for role names; the proof for inverse roles is similar.

[Forth] Let $((t,\sigma),(t',\sigma)) \in S$ and $((t,\sigma),(\hat{t},\hat{\sigma})) \in r^{\mathcal{J}_1}$. We distinguish two cases. Assume first that there exists a mosaic p with $\hat{\sigma} = \sigma p$. Then $(t,\hat{t}) \in R^{r,1}_{\text{tail}(\sigma),p}$. Thus, there exist $d, \hat{d}$ realizing $t, \hat{t}$, respectively, such that $(T_1(d), T_2(d)) = \text{tail}(\sigma)$, $(T_1(\hat{d}), T_2(\hat{d})) = p$, and $(d,\hat{d}) \in r^{\mathcal{I}_1}$. Since $((t,\sigma),(t',\sigma)) \in S$, there exists e realizing t' such that $\mathcal{I}_1, d \sim_{\mathcal{ALCHI},\Sigma} \mathcal{I}_2, e$. As d and e are bisimilar, we also have some $\hat{e} \in \Delta^{\mathcal{I}_2}$ such that $(e,\hat{e}) \in r^{\mathcal{I}_2}$ and $\mathcal{I}_1, \hat{d} \sim_{\mathcal{ALHI},\Sigma} \mathcal{I}_2, \hat{e}$, with $\hat{e}$ realizing some $\hat{t}'$. Hence, $(t',\hat{t}') \in R^{r,2}_{\text{tail}(\sigma),p}$, and it follows that $((t',\sigma),(\hat{t}',\sigma p)) \in r^{\mathcal{J}_2}$. Moreover, $((\hat{t},\sigma p),(\hat{t}',\sigma p)) \in S$.
Assume now that $\sigma = \hat{\sigma} p$ for some mosaic p. Then $(\hat{t}, t) \in R^{r^-,1}_{\text{tail}(\hat{\sigma}),p}$. Thus, there exist $\hat{d}, d$ realizing $\hat{t}, t$, respectively, such that $(T_1(\hat{d}), T_2(\hat{d})) = \text{tail}(\hat{\sigma})$, $(T_1(d), T_2(d)) = p$, and $(\hat{d}, d) \in (r^-)^{\mathcal{I}_1}$. Since $((t,\sigma),(t',\sigma)) \in S$, there exists e realizing t' such that $\mathcal{I}_1, d \sim_{\mathcal{ALCHI},\Sigma} \mathcal{I}_2, e$. As d and e are bisimilar, we also have some $\hat{e} \in \Delta^{\mathcal{I}_2}$ such that $(\hat{e}, e) \in (r^-)^{\mathcal{I}_2}$ and $\mathcal{I}_1, \hat{d} \sim_{\mathcal{ALHI},\Sigma} \mathcal{I}_2, \hat{e}$, with $\hat{e}$ realizing some $\hat{t}'$. Hence, $(\hat{t}', t') \in R^{r^-,2}_{\text{tail}(\hat{\sigma}),p}$, and it follows that $((t',\sigma),(\hat{t}',\hat{\sigma})) \in r^{\mathcal{J}_2}$. Moreover, $((\hat{t},\hat{\sigma}),(\hat{t}',\hat{\sigma})) \in S$.

Observe that again the models $\mathcal{J}_i$, $i = 1,2$, are of at most exponential size in the size of $\mathcal{O}, C_1, C_2$. We also have $(T_1(d_1), T_2(d_1)) \in \mathcal{B}$ and so $(\text{tp}_\Xi(\mathcal{I}_1, d_1), T_1(d_1)) \in C_1^{\mathcal{J}_1}$, $(\text{tp}_\Xi(\mathcal{I}_2, d_2), T_2(d_1)) \in C_2^{\mathcal{J}_2}$, and

$$((\text{tp}_\Xi(\mathcal{I}_1, d_1), T_1(d_1)), (\text{tp}_\Xi(\mathcal{I}_2, d_2), T_2(d_1)) \in S,$$

as required.

9 LOWER BOUND PROOFS WITHOUT ONTOLOGY

In this section, we first show (the hardness part of) Points 1 and 2 of Theorem 5.13 by a reduction of the case with ontologies, and then show the nondeterministic exponential time lower bounds for Points 3 and 4 of that Theorem. Points 1 and 2 of Theorem 5.13 are a direct consequence of the following lemma.

Lemma 9.1. *Let $\mathcal{L} \in DL_{nr}$ admit the universal role or both inverse roles and nominals. Then the following holds:*

(1) if $\mathcal{L}$ admits RIs, then projective $\mathcal{L}$-definition existence can be reduced in polynomial time to RI-ontology projective $\mathcal{L}$-definition existence;
(2) if $\mathcal{L}$ does not admit RIs, then projective $\mathcal{L}$-definition existence can be reduced in polynomial time to ontology-free projective $\mathcal{L}$-definition existence.

Proof. Assume $\mathcal{O}, C, C_0$, and Σ are given. We may assume that $\mathcal{O}$ takes the form $\{\top \sqsubseteq D\} \cup \mathcal{O}'$ with $\mathcal{O}'$ a set of RIs.

Assume first that $\mathcal{L}$ admits the universal role. Then one can easily show that there exists an explicit $\mathcal{L}(\Sigma)$-definition of C_0 under $\mathcal{O}$ and C iff there exists an explicit $\mathcal{L}(\Sigma)$-definition of C_0 under $\mathcal{O}'$ and $C \sqcap \forall u.D$.

Now assume that $\mathcal{L}$ admits inverse roles and nominals. We use the spy-point technique to encode the universal role [1]. Introduce a fresh individual a and a fresh role name r_0 and define U as the conjunction of the concepts

$$\{a\}, \quad \exists r_0.\{a\}, \quad \exists r_0.(\{b\} \sqcap \exists r_0.\{a\}), \quad \forall r_0^-.\forall s.\exists r_0.\{a\},$$

for all $s \in \{r, r^-\}$ with $r \in \mathrm{sig}(\mathcal{O}, C, C_0)$ and $b \in \mathrm{sig}(\mathcal{O}, C, C_0)$. Observe that if $d \in (U \sqcap \forall r_0^-.F)^{\mathcal{I}}$ for some interpretation $\mathcal{I}$ and concept F, then $e \in F^{\mathcal{I}}$ holds for all elements e in $\Delta^{\mathcal{I}}$ that can be reached in $\mathcal{I}$ from d or any $b^{\mathcal{I}}$ with $b \in \mathrm{sig}(\mathcal{O}, C, C_0)$ along roles in $\mathrm{sig}(\mathcal{O}, C, C_0)$. It follows that for any $\mathcal{L}(\Sigma)$-concept E, we have

$$\mathcal{O} \models C \sqsubseteq (C_0 \leftrightarrow E) \quad \text{iff} \quad \mathcal{O}' \models (C \sqcap U \sqcap \forall r_0^-.D) \sqsubseteq (C_0 \leftrightarrow E).$$

Hence there exists an explicit $\mathcal{L}(\Sigma)$-definition of C_0 under $\mathcal{O}$ and C iff there exists an explicit $\mathcal{L}(\Sigma)$-definition of C_0 under $\mathcal{O}'$ and $C \sqcap U \sqcap \forall r_0^-.D$. □

We show the lower bound for Theorem 5.13, Points 3 and 4, by proving NExpTime-hardness for the version of joint consistency formulated in Theorem 5.8. We reduce the exponential torus tiling problem. A *tiling system* is a triple $P = (T, H, V)$, where $T = \{0, \ldots, k\}$ is a finite set of *tile types* and $H, V \subseteq T \times T$ are the *horizontal* and *vertical* matching conditions, respectively. An *initial condition* for P takes the form $c = (c_0, \ldots, c_{n-1}) \in T^n$. A mapping $\tau : \{0, \ldots, 2^n - 1\} \times \{0, \ldots, 2^n - 1\} \to T$ is a *solution for P and c* if $\tau(i, 0) = c_i$ for all $i < n$, and for all $i, j < 2^n$, the following conditions hold (where $\oplus_k$ denotes addition modulo k):

— if $\tau(i, j) = t_1$ and $\tau(i \oplus_{2^n} 1, j) = t_2$, then $(t_1, t_2) \in H$;
— if $\tau(i, j) = t_1$ and $\tau(i, j \oplus_{2^n} 1) = t_2$, then $(t_1, t_2) \in V$.

It is well-known that the problem of deciding whether there is a solution for given P and c is NExpTime-hard [8, Section 5.2.2]. For the following constructions, assume a tiling system P and an initial condition c of length n.

For the reduction for $\mathcal{ALCO}$, we give concepts C, C_0 and a signature Σ such that here exist $\mathcal{I}_1, d_1 \sim_{\mathcal{ALCO},\Sigma} \mathcal{I}_2, d_2$ with $d_1 \in (C \sqcap C_0)^{\mathcal{I}_1}$ and $d_2 \in (C \sqcap \neg C_0)^{\mathcal{I}_2}$ iff P has a solution given c. We start with setting

$$C_0 = \exists r^{2n}.\{a\} \sqcap \forall r^{2n}.\{a\}$$

with $a \notin \Sigma$ and $r \in \Sigma$. In addition to r, Σ contains concept names $B_0, \ldots, B_{2n-1}$ that serve as bits in the binary representation of *grid positions* (i, j) with $0 \le i, j \le 2^n - 1$, where bits $B_0, \ldots, B_{n-1}$ represent the horizontal position i and $B_n, \ldots, B_{2n-1}$ the vertical position j, and concept names $T_0, \ldots, T_k$ representing tile types. We also use the following concept names that are not in Σ: another four sets of concepts names $A_0, \ldots, A_{2n-1}$ and $V_0, \ldots, V_{2n-1}$ with $V \in \{X, Y, Z\}$ that also serve as bits in the binary representation of grid position (i, j) with $0 \le i, j \le 2^n - 1$, and concept names $R_0, \ldots, R_{2n}$, M, M_1, and M_2. We now define the concept C as a conjunction of several concepts. The first conjunct is

$$\neg C_0 \sqcap \exists r^{2n}.\top \to R_0.$$

Intuitively, R_0 generates a binary r-tree of depth $2n$ with R_i true at level i for $0 \le i \le 2^{2n}$ and each leaf represents a grid position (i, j) using the concept names A_i. To achieve this let C contain the following conjuncts for generating the binary tree:

$$\bigsqcap_{0 \le i < 2n} \forall r^i.\big(R_i \to (\exists r.(A_i \sqcap R_{i+1}) \sqcap \exists r.(\neg A_i \sqcap R_{i+1}))\big)$$

$$\bigsqcap_{1 \le i < 2n} \bigsqcap_{0 \le j < i} \forall r^i.\big((A_j \to \forall r.A_j) \sqcap (\neg A_j \to \forall r.\neg A_j)\big)$$

As usual, $\forall r^i$ abbreviates a sequence of i times $\forall r$.

We next express using additional conjuncts of C that any leaf d representing (i, j) using A_i has the following properties (A)–(C):

(A) d has an r-successor representing (i, j) using B_i with a tile type $T_{(i,j)}$ true in it; moreover, no r-successor of d representing (i, j) satisfies a tile type different from $T_{(i,j)}$. This is achieved using the marker M which holds in exactly those r-successors of d that represent (i, j) using B_i. The latter condition is expressed using the counter X_i which represents (i, j) on all r-successors of d. In detail, we add the following conjuncts to C:

$$\forall r^{2n}.\exists r.M$$

$$\forall r^{2n}.\left(\prod_{i<2n}(A_i \rightarrow \forall r.X_i) \sqcap (\neg A_i \rightarrow \forall r.\neg X_i)\right)$$

$$\forall r^{2n+1}.\left(M \leftrightarrow \prod_{i<2n}(X_i \leftrightarrow B_i) \sqcap (\neg X_i \leftrightarrow \neg B_i)\right)$$

$$\forall r^{2n}.\left(\forall r.(M \rightarrow \bigsqcup_{i\leq k} T_i) \sqcap \prod_{i\leq k} \exists r.(M \sqcap T_i) \rightarrow \forall r.(M \rightarrow T_i)\right)$$

$$\forall r^{2n+1}.\prod_{i\neq j}\neg(T_i \sqcap T_j)$$

(B) d has an r-successor representing $(i \oplus_{2^n} 1, j)$ using B_i with a tile type $T^{\text{right}}_{(i,j)}$ true in it such that $(T_{(i,j)}, T^{\text{right}}_{(i,j)}) \in H$; moreover, no r-successor of d representing $(i \oplus_{2^n} 1, j)$ satisfies a tile type different from $T^{\text{right}}_{(i,j)}$. This is achieved in a similar way as (A) using the marker M_1 which holds in exactly those r-successors of d that represent $(i \oplus_{2^n} 1, j)$ using B_i. The latter condition is expressed using the counter Y_i which represents $(i \oplus_{2^n} 1, j)$ on all r-successors of d. The implementation of these conditions is similar to (A) and omitted.

(C) d has an r-successor representing $(i, j \oplus_{2^n} 1)$ using B_i with a tile type $T^{\text{up}}_{(i,j)}$ true in it such that $(T_{(i,j)}, T^{\text{up}}_{(i,j)}) \in V$; moreover, no r-successor of d representing $(i, j \oplus_{2^n} 1)$ satisfies a tile type different from $T^{\text{up}}_{(i,j)}$. This is achieved in a similar way as (A) using the marker M_2 which holds in exactly those r-successors of d that represent $(i, j \oplus_{2^n} 1)$ using B_i. The latter condition is expressed using the counter Z_i which represents $(i, j \oplus_{2^n} 1)$ on all r-successors of d. The implementation is again similar to (A) and omitted.

Finally, we ensure that the initial condition holds, that is $T_{(i,0)} = c_i$ for $i < n$. To this end, we add the conjuncts

$$\forall r^{2n}.(A = (i, 0) \rightarrow (\forall r.(M \rightarrow c_i)))$$

for $i < n$, where $A = (i, 0)$ stands for the representation of $(i, 0)$ using A_i; for instance, $A = (0, 0)$ stands for $\prod_{0\leq i<2n} \neg A_i$.

This finishes the definition of C, C_0 and we verify next that they are as required.

Claim. There exist $\mathcal{I}_1, d_1 \sim_{\mathcal{ALCO},\Sigma} \mathcal{I}_2, d_2$ with $d_1 \in (C \sqcap C_0)^{\mathcal{I}_1}$ and $d_2 \in (C \sqcap \neg C_0)^{\mathcal{I}_2}$ iff P has a solution given c.

Proof of the Claim. Observe that if $\mathcal{I}_1, d_1 \sim_{\mathcal{ALCO},\Sigma} \mathcal{I}_2, d_2$ with $d_1 \in (C\sqcap C_0)^{\mathcal{I}_1}$ and $d_2 \in (C\sqcap\neg C_0)^{\mathcal{I}_2}$, then there are elements $e_{(i,j)}$, $0 \leq i, j \leq 2^n - 1$ such that

$$\mathcal{I}_1, a^{\mathcal{I}_1} \sim_{\mathcal{ALCO},\Sigma} \mathcal{I}_2, e_{(i,j)}$$

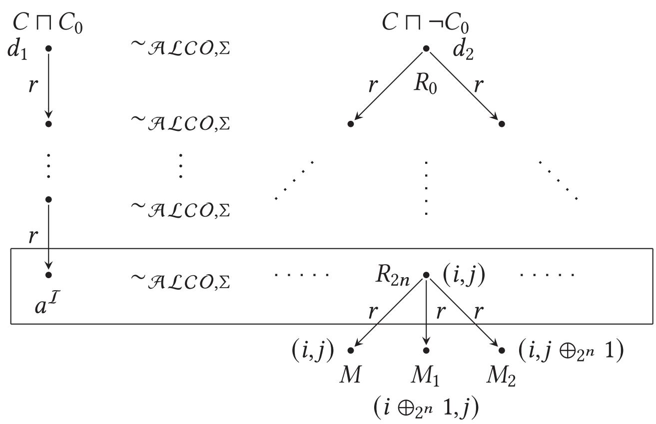

Fig. 10. Interpretation $\mathcal{I}$ with elements $d_1 \in (C \sqcap C_0)^{\mathcal{I}}$ and $d_2 \in (C \sqcap \neg C_0)^{\mathcal{I}}$ such that $\mathcal{I}, d_1 \sim_{\mathcal{ALCO},\Sigma} \mathcal{I}, d_2$.

and $e_{(i,j)}$ has (at least) three r-successors satisfying Conditions (A) to (C) and the initial condition. By Σ-bisimilarity and since $r \in \Sigma$, all $e_{(i,j)}$ have r-successors satisfying the same concept names in Σ. Hence, since the concept names B_i and T_i are in Σ, for every grid position (i, j) every $e_{(i',j')}$ has an r-successor representing (i, j) using B_i and all r-successors representing (i, j) using B_i satisfy the same tile type $T_{(i,j)}$. Moreover, $T_{(i \oplus_{2^n} 1, j)} = T^{\text{right}}_{(i,j)}$ and $T_{(i, j \oplus_{2^n} 1)} = T^{\text{up}}_{(i,j)}$. It follows that the mapping τ defined by setting $\tau(i, j) = T_{(i,j)}$ is a solution of P given c.

Conversely, assume that P and c have a solution τ. The definition of an interpretation $\mathcal{I}$ with elements d_1 and d_2 such that $\mathcal{I}, d_1 \sim_{\mathcal{ALCO},\Sigma} \mathcal{I}, d_2$ with $d_1 \in (C \sqcap C_0)^{\mathcal{I}}$ and $d_2 \in (C \sqcap \neg C_0)^{\mathcal{I}}$ is rather straightforward. An abstract version is depicted in Figure 10. We omit the counters, and note that $a^{\mathcal{I}}$ and *all* elements at level R_{2n} have, for all $0 \le i, j < 2^n$, an r-successor representing (using concept names B_i) grid position (i, j) which satisfies the concept name $T_{\tau(i,j)}$. We show only the three special successors from Conditions (A)–(C). This finishes the proof of the Claim and thus the reduction for $\mathcal{ALCO}$.

We come to the lower bound for $\mathcal{ALCH}$ and $\mathcal{ALCHI}$. Let

$$\mathcal{O} = \{r \sqsubseteq r_1, r \sqsubseteq r_2, r_1 \sqsubseteq v, r_2 \sqsubseteq v\},$$

and Σ contains r_1, r_2 but not r nor v. In addition to r_1 and r_2, Σ contains exactly the same concept names as in the $\mathcal{ALCO}$ proof and we also use the same concept names not in Σ. We aims to construct concepts C, C_0 such that there exist models $\mathcal{I}_1, \mathcal{I}_2$ of $\mathcal{O}$ and $d_1 \in (C \sqcap C_0)^{\mathcal{I}_1}$ and $d_2 \in (C \sqcap \neg C_0)^{\mathcal{I}_2}$ with $\mathcal{I}_1, d_1 \sim_{\mathcal{ALCH},\Sigma} \mathcal{I}_2, d_2$ iff P has a solution given c.

We set $C_0 = \exists r^{2n}.\top$. The concept C is again a conjunction of several concepts; we start in a similar way as for $\mathcal{ALCO}$ with

$$\neg C_0 \sqcap \exists v^{2n}.\top \to R_0$$

The concept name R_0 will enforce that

(∗∗) the end point of any r_1/r_2-path of length $2n$ starting in an element satisfying R_0 carries a pair of counter values (i, j) represented by concept names A_i which describe the path in a canonical way.[2]

[2]Notice the similarity with Property (∗) from the proof of Lemma 7.5.

To achieve this, we include the following conjuncts in C:

$$\bigsqcap_{0\le i<2n} \forall v^i.\big(R_i \to \forall r_1.(A_i \sqcap R_{i+1}) \sqcap \forall r_2.(\neg A_i \sqcap R_{i+1})\big)$$

$$\bigsqcap_{1\le i<2n}\ \bigsqcap_{0\le j<i} \forall v^i.\big((A_j \to \forall v.A_j) \sqcap (\neg A_j \to \forall v.\neg A_j)\big)$$

Note that we can use the role name v to address all elements reachable along r_1/r_2-paths of length i via $\forall v^i$. We continue the definition of C in exactly the same way as for $\mathcal{ALCO}$ except that we use $\forall v^{2n}$ to reach the end points of the paths mentioned in $(**)$ and r_1-successors of the leaves to encode a solution of the tiling problem. One can then easily prove the following.

Claim. There exist $\mathcal{I}_1, d_1 \sim_{\mathcal{ALCH},\Sigma} \mathcal{I}_2, d_2$ with $\mathcal{I}_1, \mathcal{I}_2$ models of $\mathcal{O}$, $d_1 \in (C \sqcap \exists r^{2n}.\top)^{\mathcal{I}_1}$, and $d_2 \in (C \sqcap \neg\exists r^{2n}.\top)^{\mathcal{I}_2}$ iff P has a solution given c.

Proof of the Claim. Observe that if $\mathcal{I}_1, d_1 \sim_{\mathcal{ALCH},\Sigma} \mathcal{I}_2, d_2$ with $\mathcal{I}_1, \mathcal{I}_2$ models of $\mathcal{O}$, $d_1 \in (C \sqcap \exists r^{2n}.\top)^{\mathcal{I}_1}$, and $d_2 \in (C \sqcap \neg\exists r^{2n}.\top)^{\mathcal{I}_2}$, then there exists an element e reachable from d_1 along an r-path of length $2n$ in $\mathcal{I}_1$. Since $d_2 \in R_0^{\mathcal{I}_2}$ and e is reachable via arbitrary r_1/r_2-paths of length $2n$ from d_1, Property $(**)$ implies that there are elements $e_{(i,j)}$, $0 \le i,j \le 2^n - 1$, reachable from d_2 along a v-path of length $2n$ in $\mathcal{I}_2$ such that $\mathcal{I}_1, e \sim_{\mathcal{ALCH},\Sigma} \mathcal{I}_2, e_{(i,j)}$ and $e_{i,j}$ represents the pair (i,j) using the concept names A_i. The remaining proof is now essentially the same as for $\mathcal{ALCO}$.

The converse direction is rather straightforward and similar to the proof for $\mathcal{ALCO}$. The difference is that the binary tree over role r in the right side of interpretation $\mathcal{I}$ depicted in Figure 10 is now a binary tree over roles r_1 (left successor) and r_2 (right successor). This finishes the proof of the Claim.

To prove the claim above for $\mathcal{ALCHI}$, we adapt the model construction in a similar way as in the case with ontologies (Section 7.3). More precisely, for each element e at level $\ell > 0$ in the binary tree below d_2, add $(d,e) \in r_1^{\mathcal{I}}$ and $(d,e) \in r_2^{\mathcal{I}}$, where d is the element in distance $\ell - 1$ from d_1. One can then verify that $\mathcal{I}$ is as required, that is, $\mathcal{I}, d_1 \sim_{\mathcal{ALCHI}} \mathcal{I}, d_2$, $d_1 \in (C \sqcap C_0)^{\mathcal{I}}$, and $d_2 \in (C \sqcap \neg C_0)^{\mathcal{I}}$.

10 COMPUTATION PROBLEM

In the previous sections, we have presented algorithms for *deciding* the existence of interpolants and explicit definitions, but these algorithms (and their correctness proofs) do not give immediately rise to a way of *computing* interpolants and explicit definitions in case they exist. Intuitively, this is due to the fact that compactness is used in the proof of the model-theoretic characterization of interpolant and explicit definition existence in terms of joint consistency modulo bisimulations which was provided in Theorems 5.6 and 5.8, respectively. In this section, we address the computation problem for logics in $\mathrm{DL_{nr}}$ that do not admit nominals, by showing that we can actually compute interpolants in case they exist. We use *DAG representation* for the interpolants; recall that in DAG representation common sub-formulas are stored only once, and that thus DAG representation is more succinct than formula representation. Our approach is inspired by a recent note on a type elimination based computation of interpolants in modal logic [87], which was originally provided for the GF [15].

Theorem 10.1. *Let $\mathcal{L} \in DL_{nr}$ not admit nominals, $\mathcal{O}$ be an $\mathcal{L}$-ontology, C_1, C_2 be $\mathcal{L}$-concepts, and Σ be a signature. Then, if there is an $\mathcal{L}(\Sigma)$-interpolant for $C_1 \sqsubseteq C_2$ under $\mathcal{O}$, we can compute the DAG representation of an $\mathcal{L}(\Sigma)$-interpolant in time $2^{2^{p(n)}}$ where p is a polynomial and $n = ||\mathcal{O}|| + ||C_1|| + ||C_2||$.*

Note that this implies that the DAG representation is also of double exponential size, and that a formula representation of the interpolant can be computed in triple exponential time. Moreover, this also allows us to compute explicit definitions since, given $\mathcal{O}$, C, and Σ, any $\mathcal{L}(\Sigma)$-interpolant for $C_\Sigma \sqsubseteq C$ under $\mathcal{O} \cup \mathcal{O}_\Sigma$ is an explicit $\mathcal{L}(\Sigma)$-definition of C under $\mathcal{O}$, where $\mathcal{O}_\Sigma$ and C_Σ are obtained from $\mathcal{O}$ and C by replacing all symbols not in Σ by fresh symbols. We conjecture that the triple exponential upper bound on formula size is actually tight, given the discussion on the explicit definitions that arise in the hardness proofs in Sections 7.2 and 7.3.

Let $\mathcal{L}$, $\mathcal{O}$, C_1, C_2, and Σ be as in Theorem 10.1. By Theorem 5.6, the existence of an $\mathcal{L}(\Sigma)$-interpolant for $C_1 \sqsubseteq C_2$ under $\mathcal{O}$ is equivalent to joint consistency of C_1 and $\neg C_2$ under $\mathcal{O}$ modulo $\mathcal{L}(\Sigma)$-bisimulations. Recall that we have provided before Lemma 6.4 in Section 6 a mosaic elimination procedure for deciding the latter. In fact, the computation of the $\mathcal{L}(\Sigma)$-interpolant relies on a finer analysis of that procedure. We need one more notion to formalize this analysis.

Let T be a set of Ξ-types. Let $\mathcal{I}$ be an interpretation and, for each $t \in T$, let d_t be a domain element of $\mathcal{I}$. We say that $\mathcal{I}$ and the elements $d_t, t \in T$ *jointly realize* T *modulo* $\mathcal{L}(\Sigma)$*-bisimulations* if, for all $t, t' \in T$, we have that $\mathrm{tp}_\Xi(\mathcal{I}, d_t) = t$ and $\mathcal{I}, d_t \sim_{\mathcal{L},\Sigma} \mathcal{I}, d_{t'}$. We call T *jointly realizable under* $\mathcal{O}$ *modulo* $\mathcal{L}(\Sigma)$*-bisimulations* if there is a model $\mathcal{I}$ of $\mathcal{O}$ and elements d_t for each $t \in T$ that jointly realize T modulo $\mathcal{L}(\Sigma)$-bisimulations. In contrast to the notion of joint consistency, we require here a single model $\mathcal{I}$ of $\mathcal{O}$. In what follows, let Real denote the set of all sets of types T which are jointly realizable under $\mathcal{O}$ modulo $\mathcal{L}(\Sigma)$-bisimulations. We can effectively determine Real since joint realizability of a set T can be decided in double exponential time, similar to joint consistency—we refrain from giving details.

In the (proof of the) following lemma, we show how to compute a concept differentiating between T_1 and T_2 when (T_1, T_2) is eliminated for $T_1, T_2 \in \mathsf{Real}$. In Lemma 10.3 below, we show how to assemble these differentiating concepts to an interpolant (in case it exists).

Lemma 10.2. *Let $T_1, T_2 \in \mathsf{Real}$. If (T_1, T_2) is eliminated in the mosaic elimination procedure, then we can compute an $\mathcal{L}(\Sigma)$-concept I_{T_1,T_2} such that*

(1) for all models $\mathcal{I}$ of $\mathcal{O}$ and elements d_t, for each $t \in T_1$, that jointly realize T_1 modulo $\mathcal{L}(\Sigma)$-bisimulations, $d_t \in I^{\mathcal{I}}_{T_1,T_2}$ for some (equivalently: all) $t \in T_1$;
(2) for all models $\mathcal{I}$ of $\mathcal{O}$ and elements d_t, for each $t \in T_2$, that jointly realize T_2 modulo $\mathcal{L}(\Sigma)$-bisimulations, $d_t \notin I^{\mathcal{I}}_{T_1,T_2}$ for some (equivalently: all) $t \in T_2$.

Moreover, a DAG representation of I_{T_1,T_2} can be computed in time $2^{2^{p(n)}}$ for some polynomial p and $n = ||\mathcal{O}|| + ||C_1|| + ||C_2||$.

Proof. We compute the I_{T_1,T_2} inductively in the order in which the (T_1, T_2) got eliminated in the elimination procedure. We distinguish cases why (T_1, T_2) got eliminated.

Suppose first that (T_1, T_2) was eliminated because of (failing) Σ-concept name coherence. Since T_1, T_2 are both jointly realizable, there are the following two cases.

(a) There is a concept name $A \in \Sigma$ such that $A \in t$ for all $t \in T_1$, but $A \notin t$, for all $t \in T_2$. Then $I_{T_1,T_2} = A$.
(b) There is a concept name $A \in \Sigma$ such that $A \in t$ for all $t \in T_2$, but $A \notin t$, for all $t \in T_1$. Then $I_{T_1,T_2} = \neg A$.

Clearly, in both cases, I_{T_1,T_2} satisfies Points (1) and (2) of Lemma 10.2.

Now, suppose that (T_1, T_2) was eliminated due to (failing) existential saturation from $\mathcal{S}_i$ during the elimination procedure. Since T_1, T_2 are both jointly realizable under $\mathcal{O}$, there are the following two cases.

(a) There exist $t \in T_1$, $\exists r.C \in t$, and a Σ-role s with $\mathcal{O} \models r \sqsubseteq s$, such that there is no $(T_1', T_2') \in \mathcal{S}_i$ such that *(i)* $(T_1, T_2) \rightsquigarrow_s (T_1', T_2')$ and *(ii)* there is $t' \in T_1'$ with $C \in t'$ and $t \rightsquigarrow_{r,\mathcal{O}} t'$. Then, take

$$I_{T_1,T_2} = \exists s. \left(\bigsqcup_{\substack{T_1' \in \mathsf{Real},\\ T_1 \rightsquigarrow_s T_1', t \rightsquigarrow_{r,\mathcal{O}} t', C \in t' \in T_1'}} \ \bigsqcap_{\substack{T_2' \in \mathsf{Real},\\ T_2 \rightsquigarrow_s T_2'}} I_{T_1',T_2'} \right)$$

(b) There exist $t \in T_2$, $\exists r.C \in t$, and a Σ-role s with $\mathcal{O} \models r \sqsubseteq s$, such that there is no $(T_1', T_2') \in \mathcal{S}$ such that *(i)* $(T_1, T_2) \rightsquigarrow_s (T_1', T_2')$ and *(ii)* there is $t' \in T_2'$ with $C \in t'$ and $t \rightsquigarrow_{r,\mathcal{O}} t'$. Then, take

$$I_{T_1,T_2} = \forall s. \left(\bigsqcup_{\substack{T_1' \in \mathsf{Real},\\ T_1 \rightsquigarrow_s T_1'}} \ \bigsqcap_{\substack{T_2' \in \mathsf{Real},\\ T_2 \rightsquigarrow_s T_2', t \rightsquigarrow_{r,\mathcal{O}} t', C \in t' \in T_2'}} I_{T_1',T_2'} \right)$$

We show Points (1) and (2) of the lemma for Case (a); Case (b) is dual. So suppose Case (a) applies and fix $t \in T_1$, $\exists r.C \in t$, and a Σ-role s witnessing that.

To show Point (1) of the lemma, let $\mathcal{I}$ be a model of $\mathcal{O}$ and fix d_{t_1} for each $t_1 \in T_1$ such that $\mathcal{I}$ and the d_{t_1} jointly realize T_1 modulo $\mathcal{L}(\Sigma)$-bisimulations. It suffices to show that $d_t \in I^{\mathcal{I}}_{T_1,T_2}$ for the type t that was fixed in the application of Case (a). Since d_t realizes t and $\exists r.C \in t$, there is some $e \in C^{\mathcal{I}}$ with $(d_t, e) \in r^{\mathcal{I}}$. Since $\mathcal{O} \models r \sqsubseteq s$, also $(d_t, e) \in s^{\mathcal{I}}$. Since the $d_{t_1}, t_1 \in T_1$ are mutually $\mathcal{L}(\Sigma)$-bisimilar and s is a Σ-role, we find elements $e_{t_1}, t_1 \in T_1$ such that:

— $e_{t_1}, t_1 \in T_1$ are mutually $\mathcal{L}(\Sigma)$-bisimilar,
— $(d_{t_1}, e_{t_1}) \in s^{\mathcal{I}}$, for all $t_1 \in T_1$,
— $e_t = e$.

Let

$$T_1' = \{\mathsf{tp}_\Xi(\mathcal{I}, e_{t_1}) \mid t_1 \in T_1\},$$

and let further $T_2' \in \mathsf{Real}$ be arbitrary with $T_2 \rightsquigarrow_s T_2'$. By definition of T_1', we have $T_1' \in \mathsf{Real}$ and $T_1 \rightsquigarrow_s T_1'$. Thus, (T_1', T_2') has been eliminated before (T_1, T_2): otherwise, Case (a) would not apply to the fixed $t, \exists r.C, s$. By induction, we can conclude that $e = e_t \in I^{\mathcal{I}}_{T_1',T_2'}$, and hence $d \in I^{\mathcal{I}}_{T_1,T_2}$.

To show Point (2) of the lemma, let $\mathcal{I}$ be a model of $\mathcal{O}$ and fix d_{t_2} for each $t_2 \in T_2$ such that $\mathcal{I}$ and the d_{t_2} jointly realize T_2 modulo $\mathcal{L}(\Sigma)$-bisimulations. Suppose, to the contrary of what has to be shown, that $d_{\widehat{t}} \in I^{\mathcal{I}}_{T_1,T_2}$ for some $\widehat{t} \in T_2$. Then, there is an e with $(d_{\widehat{t}}, e) \in s^{\mathcal{I}}$ and a $T_1' \in \mathsf{Real}$ with $T_1 \rightsquigarrow_s T_1'$ and a type $t_1' \in T_1$ with $t \rightsquigarrow_{r,\mathcal{O}} t_1'$ and $C \in t_1'$ such that

(∗) $e \in I^{\mathcal{I}}_{T_1',T}$ for all $T \in \mathsf{Real}$ with $T_2 \rightsquigarrow_s T$.

Since $\mathcal{I}$ and the elements $d_{t_2}, t_2 \in T_2$ jointly realize T_2 modulo $\mathcal{L}(\Sigma)$-bisimulations and s is a Σ-role, there are elements $e_{t_2}, t_2 \in T_2$ such that:

— $e_{t_2}, t_2 \in T_2$ are mutually $\mathcal{L}(\Sigma)$-bisimilar,
— $(d_{t_2}, e_{t_2}) \in s^{\mathcal{I}}$, for all $t_2 \in T_2$,
— $e_{\widehat{t}} = e$.

Let

$$T_2' = \{\mathsf{tp}_\Xi(\mathcal{I}, e_{t_2}) \mid t_2 \in T_2\}.$$

By definition of T_2', we have $T_2' \in \mathsf{Real}$ and $T_2 \rightsquigarrow_s T_2'$. Thus, (T_1', T_2') has been eliminated before (T_1, T_2): otherwise, Case (a) would not apply to the fixed $t, \exists r.C, s$. By induction, we obtain $e = e_{\widehat{t}} \notin I^{\mathcal{I}}_{T_1', T_2'}$, in contradiction to $(*)$.

For the analysis of the DAG representation, observe that we can use a single node for every I_{T_1,T_2}. Moreover, I_{T_1,T_2} looks as follows:

— If (T_1, T_2) was eliminated due to failing Σ-concept name coherence, I_{T_1,T_2} is a single concept name A or its negation $\neg A$.
— Otherwise, it is a node labeled with $\exists s$ (resp., $\forall s$), which has a single successor labeled with $\bigsqcup$. This successor has then at most double exponentially many successor nodes, each labeled with $\bigsqcap$ and each having at most double exponentially many successor nodes I_{T_1,T_2}.

Overall, we obtain double exponentially many nodes in the DAG and the DAG can be constructed in double exponential time (both in $p(||\mathcal{O}|| + ||C_1|| + ||C_2||)$). □

Lemma 10.3. *Suppose the result $\mathcal{S}^*$ of the mosaic elimination procedure does not contain a pair $(T_1, T_2) \in \mathsf{Real} \times \mathsf{Real}$ such that $C_1 \in t_1$ and $\neg C_2 \in t_2$ for some types $t_1 \in T_1$ and $t_2 \in T_2$. Then,*

$$C = \bigsqcup_{\substack{T_1 \in \mathsf{Real}: \\ \text{there is } t_1 \in T_1 \text{ with } C_1 \in t_1}} \quad \bigsqcap_{\substack{T_2 \in \mathsf{Real}: \\ \text{there is } t_2 \in T_2 \text{ with } \neg C_2 \in t_2}} I_{T_1,T_2}$$

is an $\mathcal{L}(\Sigma)$-interpolant for $C_1 \sqsubseteq C_2$ under $\mathcal{O}$. Moreover, a DAG representation of C can be computed in time $2^{2^{p(n)}}$, for some polynomial p and $n = ||\mathcal{O}|| + ||C_1|| + ||C_2||$.

Proof. We have to show that $\mathcal{O} \models C_1 \sqsubseteq C$ and $\mathcal{O} \models C \sqsubseteq C_2$.

For $\mathcal{O} \models C_1 \sqsubseteq C$, let $\mathcal{I}$ be a model of $\mathcal{O}$ and suppose $d \in C_1^{\mathcal{I}}$. Let $T_1 = \{\mathrm{tp}_{\Xi}(\mathcal{I}, d)\}$ consist of the single type of d. Clearly, $T_1 \in \mathsf{Real}$. Let $T_2 \in \mathsf{Real}$ be arbitrary such that $\neg C_2 \in t$, for some $t \in T_2$. By assumption of Lemma 10.3, (T_1, T_2) got eliminated in the elimination procedure. Point (1) of Lemma 10.2 implies $d \in I^{\mathcal{I}}_{T_1,T_2}$. Hence, $d \in C^{\mathcal{I}}$.

For $\mathcal{O} \models C \sqsubseteq C_2$, let $\mathcal{I}$ be a model of $\mathcal{O}$ and let $d \in (\neg C_2)^{\mathcal{I}}$. Now, let $T_1 \in \mathsf{Real}$ be arbitrary such that $C_1 \in t$ for some $t \in T_1$, and set $T_2 = \{\mathrm{tp}_{\Xi}(\mathcal{I}, d)\}$. Clearly, $T_2 \in \mathsf{Real}$. By assumption of Lemma 10.3, (T_1, T_2) got eliminated in the elimination procedure. Point (2) of Lemma 10.2 implies $d \notin I^{\mathcal{I}}_{T_1,T_2}$. Hence, $d \notin C^{\mathcal{I}}$.

For the analysis of the DAG representation of C, it suffices to recall that the DAG representations of the I_{T_1,T_2} provided in Lemma 10.2 can be computed in time $2^{2^{p(n)}}$, and to observe that C adds only one $\bigsqcup$ node and at most double exponentially many $\bigsqcap$-nodes. □

To conclude the section, we give some intuition as to why the proof of Theorem 10.1 cannot be easily adapted to logics from $\mathrm{DL}_{\mathrm{nr}}$ that admit nominals. Recall that in any two interpretations $\mathcal{I}_1, \mathcal{I}_2$, every nominal a is realized (modulo bisimulation) in exactly one mosaic. We addressed this by starting the elimination procedure for all possible choices of mosaics realizing the nominals. More specifically, in the proof of Lemma 6.6, we showed there is an interpolant for $C_1 \sqsubseteq C_2$ under $\mathcal{O}$ iff, for all maximal sets $\mathcal{U}$ of mosaics that are good for nominals, the mosaic elimination procedure started with $\mathcal{U}$ leads to an $\mathcal{S}^*$ which does not satisfy Condition 2 of Lemma 6.5, which is akin to the precondition of Lemma 10.3 above. It is, however, unclear how to combine these different runs of the elimination procedure in proving analogs of Lemmas 10.2 and 10.3. An alternative approach might be to derive the interpolants from a suitably constrained proof of $\mathcal{O} \models C \sqsubseteq D$ in an appropriate proof system [81].

11 SOME CONSEQUENCES FOR MODAL AND HYBRID LOGICS

In this section, we formulate a few consequences of our results in terms of modal and hybrid logics. We focus on interpolant existence and do not discuss the transfer of results on explicit definition existence as they can be obtained in a similar way. We consider the local consequence relation and formulate results for standard hybrid modal languages without the backward modality but with any combination of nominals, the @-operator, and the universal modal modality. We also briefly discuss the reformulation of description logics with RIs into modal logic with inclusion conditions on the accessibility relations. For detailed introductions to (hybrid) modal logics, we refer the reader to [2, 4].

Let $\mathrm{ML}^u_@$ denote the modal hybrid language constructed using the rule

$$\varphi, \psi \quad := \quad p \mid \top \mid i \mid \neg\varphi \mid \varphi \wedge \psi \mid \Box\varphi \mid @_i\varphi \mid \Box_u\varphi,$$

where p ranges over a countably infinite set of *propositional variables*, i ranges over a countably infinite set of *nominals*, $\Box$ ranges over an infinite set of modal operators $\Box_0, \ldots,$ and $\Box_u$ denotes the *universal modality*. The fragment of $\mathrm{ML}^u_@$ without the universal modality is denoted $\mathrm{ML}_@$, the fragment of $\mathrm{ML}_@$ without the operators $@_i$ is denoted ML_n, and the fragment of ML_n without nominals is the standard language of polymodal logic and denoted ML. By ML^u_n we denote the fragment of $\mathrm{ML}^u_@$ without the operators $@_i$ and by ML^u the extension of ML with the universal modality.

The signature $\mathrm{sig}(\varphi)$ of a formula φ is the set of propositional variables, nominals, and modal operators (without the universal role) occurring in it.

The language $ML^u_@$ and its fragments are interpreted in *Kripke models* $\mathfrak{M} = (W, (R_i)_{i<\omega}, V)$ with W a nonempty set of *worlds*, $R_i \subseteq W \times W$ accessibility relations, and V a valuation such that $V(p) \subseteq W$ for every propositional variable p, and $V(i) \subseteq W$ a singleton for every nominal i. Then the truth relation $\mathfrak{M}, w \models \varphi$ between *pointed models* $\mathfrak{M}, w$ with $w \in W$ and formulas φ is defined inductively as follows:

$$\begin{array}{lll}
\mathfrak{M}, w \models \top, & & \\
\mathfrak{M}, w \models p & \text{iff} & w \in V(p), \\
\mathfrak{M}, w \models i & \text{iff} & V(i) = \{w\}, \\
\mathfrak{M}, w \models \neg\psi & \text{iff} & \mathfrak{M}, w \not\models \psi, \\
\mathfrak{M}, w \models \psi \wedge \chi & \text{iff} & \mathfrak{M}, w \models \psi \text{ and } \mathfrak{M}, w \models \chi, \\
\mathfrak{M}, w \models \Box_n\psi & \text{iff} & \mathfrak{M}, v \models \psi, \text{ for every } v \in W \text{ such that } (w, v) \in R_n, \\
\mathfrak{M}, w \models @_i\psi & \text{iff} & \mathfrak{M}, v \models \psi, \text{ for the unique element } v \in V(i), \\
\mathfrak{M}, w \models \Box_u\psi & \text{iff} & \mathfrak{M}, v \models \psi, \text{ for every } v \in W.
\end{array}$$

We set $\mathfrak{M} \models \varphi$ if $\mathfrak{M}, w \models \varphi$ for all $w \in W$. Observe that the @-operator can be defined using the universal modality as $@_i\varphi = \Box_u(i \rightarrow \varphi)$ and so ML^u_n and $\mathrm{ML}^u_@$ have the same expressive power.

There are two natural notions of consequence studied in modal and hybrid logics, local and global entailment, which also give rise to different notions of interpolants. We focus here on local entailment and briefly discuss global entailment at the end of this section. We say that φ *locally entails* ψ, in symbols $\varphi \models_{loc} \psi$, if for all pointed models $\mathfrak{M}, w$, if $\mathfrak{M}, w \models \varphi$ then $\mathfrak{M}, w \models \psi$. We note that deciding $\models_{loc}$ is PSPACE-complete for any of the languages introduced above without the universal modality and EXPTIME-complete for any of the languages introduced above with the universal modality [4].

We formulate the interpolant existence problems for hybrid modal logics in the expected way. Call a formula χ an *interpolant for* φ, ψ if $\mathrm{sig}(\chi) \subseteq \mathrm{sig}(\varphi) \cap \mathrm{sig}(\psi)$, $\varphi \models_{loc} \chi$ and $\chi \models_{loc} \psi$.

Definition 11.1. Let $\mathcal{L}$ be any of the languages introduced above. Then the *interpolant existence problem for* $\mathcal{L}$ is the problem to decide for any $\varphi, \psi \in \mathcal{L}$ whether there exists an interpolant for φ, ψ in $\mathcal{L}$.

Observe that since ML and ML^u enjoy the CIP (if $\varphi \models_{loc} \psi$ then an interpolant for φ, ψ exists [37]), the interpolant existence problem reduces to checking $\varphi \models_{loc} \psi$ and is PSpace-complete for ML and ExpTime-complete for ML^u. The following tight complexity bounds for their extensions with nominals and the @-operator are the main result of this section.

Theorem 11.2. *(1) Let $\mathcal{L} \in \{ML_n, ML_@\}$. Then the interpolant existence problem for $\mathcal{L}$ is* coNExpTime*-complete.*
(2) The interpolant existence problem for ML_n^u is 2ExpTime*-complete.*
These results also hold if one considers the language with a single modal operator only.

Proof. (1) Let $\cdot^m$ be the obvious bijection between $\mathcal{ALCO}$-concepts and ML_n-formulas and denote by $\cdot^d$ its inverse. Then $\models C \sqsubseteq D$ iff $C^m \models_{loc} D^m$ for any $\mathcal{ALCO}$-concepts C, D. Hence the following conditions are equivalent, for all formulas $\varphi, \psi \in \text{ML}_n$:

— there exists an interpolant for φ, ψ in ML_n;
— there exists an $\mathcal{ALCO}(\Sigma)$-interpolant for φ^d, ψ^d, where $\Sigma = \text{sig}(\varphi^d) \cap \text{sig}(\psi^d)$.

The coNExpTime-completeness for interpolant existence for ML_n now follows from Point 3 of Theorem 5.13. We now come to $\text{ML}_@$. We did not consider the operator @ for DLs as it does not play a large role in description logic research.[3] Note, however, that $\mathcal{ALCO}$ can be extended to the DL $\mathcal{ALCO}_@$ with @ in a straightforward way by setting $@_a C := \forall u.(\{a\} \rightarrow C)$. The expressive power of $\mathcal{ALCO}_@$-concepts is characterized by $\mathcal{ALCO}_@(\Sigma)$*-bisimulations*, where an $\mathcal{ALCO}(\Sigma)$-bisimulation S between interpretations $\mathcal{I}$ and $\mathcal{J}$ is an $\mathcal{ALCO}_@(\Sigma)$-bisimulation if $(a^\mathcal{I}, a^\mathcal{J}) \in S$ for any $a \in \Sigma$. Then one can prove Lemma 3.1 also for $\mathcal{ALCO}_@$. Next one can prove the characterization (Theorem 5.6) for $\mathcal{ALCO}_@$ in exactly the same way as for $\mathcal{ALCO}$, and finally one can extend the NExpTime-upper bound proof for joint consistency modulo $\mathcal{ALCO}(\Sigma)$-bisimulations to joint consistency modulo $\mathcal{ALCO}_@(\Sigma)$-bisimulations (Lemma 8.1) by observing that for all nominal generated mosaics $(T_1(d), T_2(d))$ we now have that $T_i(d) \neq \emptyset$ for $i = 1, 2$. Hence $(a^\mathcal{I}, a^\mathcal{J}) \in S$ for any $a \in \Sigma$, for the bisimulation S constructed in the proof of Lemma 8.1.

The lower bound proof for Theorem 5.13, Point 3, provided in Section 9 still goes through as it does not use any nominal in the shared signature and so using @ does not make any difference. Note, moreover, that it uses only a single role name r which corresponds to using a single modal operator.

(2) can be proved in the same way as (1) by observing that there is a bijection $\cdot^m$ between $\mathcal{ALCO}^u$-concepts and ML_n^u-formulas, that $\models C \sqsubseteq D$ iff $C^m \models_{loc} D^m$ for any $\mathcal{ALCO}^u$-concepts C, D, and then applying Point 1 of Theorem 5.13. Note that the lower bound holds for a single role, see Lemma 7.4, which again translates to a single modal operator (and the universal modality). □

DLs with RIs correspond to modal logics determined by Kripke models satisfying inclusions $R_i \subseteq R_j$ between accessibility relations R_i and R_j. For any finite set I of pairs (i, j) let $\mathcal{M}_I$ denote the class of Kripke models satisfying $R_i \subseteq R_j$ for all $(i, j) \in I$. Define the consequence relation $\models_{loc}^I$ in the usual way by setting $\varphi \models_{loc}^I \psi$ if for all pointed models $\mathfrak{M}, w$ with $\mathfrak{M} \in \mathcal{M}_I$, if $\mathfrak{M}, w \models \varphi$ then $\mathfrak{M}, w \models \psi$. We then obtain the following complexity result directly from Points 4 and 2 of Theorem 5.13, respectively.

[3] An exception is the investigation of updates for description logic knowledge bases where the expressive power of the @-operator plays a significant role [66].

Theorem 11.3. *For all finite I, the interpolant existence problem for $\models^I_{loc}$ in ML is in* coNExpTime. *There exists a finite I such that the interpolant existence problem for $\models^I_{loc}$ in ML is* coNExpTime*-hard.*

For all finite I, the interpolant existence problem for $\models^I_{loc}$ in ML^u is in 2ExpTime. *There exists a finite I such that the interpolant existence problem for $\models^I_{loc}$ in ML^u is* 2ExpTime*-hard.*

We close this section with a brief discussion of interpolant existence for the global consequence relation. We say that φ *globally entails* ψ, in symbols $\varphi \models_{glo} \psi$, if for all models $\mathfrak{M}$ from $\mathfrak{M} \models \varphi$ it follows that $\mathfrak{M} \models \psi$. Call a formula χ a *global interpolant for* φ, ψ if $\mathrm{sig}(\chi) \subseteq \mathrm{sig}(\varphi) \cap \mathrm{sig}(\psi)$, $\varphi \models_{glo} \chi$ and $\chi \models_{glo} \psi$. The *global interpolant existence problem for* $\mathcal{L}$ is the problem to decide for any $\varphi, \psi \in \mathcal{L}$ whether there exists a global interpolant for φ, ψ in $\mathcal{L}$. It is straightforward to show that global interpolant existence corresponds to CI-interpolant existence in DLs in the same way as interpolant existence for the local consequence relation corresponds to ontology-free interpolant existence in DLs. We, therefore, obtain 2ExpTime-completeness of global interpolant existence for the language ML^u_n from Theorem 5.16. We conjecture that the same result holds for global interpolant existence for ML_n and $\mathrm{ML}_@$ but leave the proofs for future work.

12 CONCLUSION

We have investigated the problem of deciding the existence of interpolants and explicit definitions for description and modal logics with nominals and RIs, and we also presented an algorithm computing them for logics with RIs. There are many challenging problems left for future work, for instance, an algorithm computing interpolants for logics with nominals and the design and implementation of practical algorithms that could be applied in supervised concept learning and referring expression generation. From a theoretical viewpoint it would be of interest to gain a better understanding of when the existence of interpolants is computationally harder than entailment, for logics that do not enjoy the CIP. Logics to consider include more expressive DLs with nominals such as those also admitting qualified number restrictions and/or transitive roles and extensions of the two-variable fragment of FO with counting and/or further constraints on relations [53]. Another class of interest are decidable fragments of first-order modal logics and products of modal logics which both often do not enjoy the CIP [32, 73]. Here, it would be of interest to consider logics such as the one-variable or monodic fragments of K and S4 for which the complexity of interpolant existence was left open [61]. Finally, is it possible to prove general transfer results (for example, for families of normal modal logics) stating that decidable entailment implies decidability of interpolant existence?

ACKNOWLEDGMENTS

We thank Agi Kurucz, Michael Zakharyaschev, and two referees for helpful comments.

REFERENCES

[1] Carlos Areces, Patrick Blackburn, and Maarten Marx. 1999. A road-map on complexity for hybrid logics. In *Proceedings of the 8th Annual Conference of the European Association for Computer Science Logic, CSL 1999*. Springer, 307–321. DOI: https://doi.org/10.1007/3-540-48168-0_22

[2] Carlos Areces, Patrick Blackburn, and Maarten Marx. 2001. Hybrid logics: Characterization, interpolation and complexity. *Journal of Symbolic Logic* 66, 3 (2001), 977–1010. DOI: https://doi.org/10.2307/2695090

[3] Carlos Areces, Alexander Koller, and Kristina Striegnitz. 2008. Referring expressions as formulas of description logic. In *Proceedings of the 5th International Natural Language Generation Conference, INLG 2008*. The Association for Computer Linguistics. Retrieved from https://aclanthology.org/W08-1107/.

[4] Carlos Areces and Balder ten Cate. 2007. Hybrid logics. In *Handbook of Modal Logic*. Vol. 3. Elsevier, 821–868.

[5] Alessandro Artale, Jean Christoph Jung, Andrea Mazzullo, Ana Ozaki, and Frank Wolter. 2020. Living without beth and craig: Explicit definitions and interpolants in description logics with nominals (extended abstract). In *Proceedings of the 33rd International Workshop on Description Logics (DL 2020) Co-Located with the 17th International Conference*

on Principles of Knowledge Representation and Reasoning (KR 2020), Online Event [Rhodes, Greece], September 12th to 14th, 2020. Retrieved from https://ceur-ws.org/Vol-2663/abstract-1.pdf.

[6] Alessandro Artale, Jean Christoph Jung, Andrea Mazzullo, Ana Ozaki, and Frank Wolter. 2021. Living without beth and craig: Definitions and interpolants in description logics with nominals and role inclusions. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence, AAAI 2021.* AAAI Press, 6193–6201. Retrieved from https://ojs.aaai.org/index.php/AAAI/article/view/16770.

[7] Alessandro Artale, Andrea Mazzullo, Ana Ozaki, and Frank Wolter. 2021. On free description logics with definite descriptions. In *Proceedings of the 18th International Conference on Principles of Knowledge Representation and Reasoning, KR 2021.* 63–73. Retrieved from https://doi.org/10.24963/kr.2021/7

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

[9] Franz Baader, Diego Calvanese, Deborah L. McGuinness, Daniele Nardi, and Peter F. Patel-Schneider (Eds.). 2003. *The Description Logic Handbook: Theory, Implementation, and Applications.* Cambridge University Press.

[10] Vince Bárány, Michael Benedikt, and Balder ten Cate. 2013. Rewriting guarded negation queries. In *Proceedings of the 38th International Symposium on Mathematical Foundations of Computer Science, MFCS 2013.* Springer, 98–110. DOI: https://doi.org/10.1007/978-3-642-40313-2_11

[11] Vince Bárány, Michael Benedikt, and Balder ten Cate. 2018. Some model theory of guarded negation. *Journal of Symbolic Logic* 83, 4 (2018), 1307–1344.

[12] Michael Benedikt, Pierre Bourhis, and Michael Vanden Boom. 2019. Definability and interpolation within decidable fixpoint logics. *Logical Methods in Computer Science* 15, 3 (2019), 29:1–29:53. DOI: https://doi.org/10.23638/LMCS-15(3:29)2019

[13] Michael Benedikt, Julien Leblay, Balder ten Cate, and Efthymia Tsamoura. 2016. *Generating Plans from Proofs: The Interpolation-Based Approach to Query Reformulation.* Morgan & Claypool Publishers.

[14] Michael Benedikt, Balder ten Cate, and Michael Vanden Boom. 2015. Interpolation with decidable fixpoint logics. In *Proceedings of the 30th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2015.* IEEE Computer Society, 378–389.

[15] Michael Benedikt, Balder ten Cate, and Michael Vanden Boom. 2016. Effective interpolation and preservation in guarded logics. *ACM Transactions on Computational Logic (TOCL)* 17, 2 (2016), 8:1–8:46.

[16] Evert Willem Beth. 1956. On padoa's method in the theory of definition. *Journal of Symbolic Logic* 21, 2 (1956), 194–195.

[17] Alexander Borgida, David Toman, and Grant E. Weddell. 2016. On referring expressions in query answering over first order knowledge bases. In *Proceedings of the 15th International Conference on Principles of Knowledge Representation and Reasoning, KR 2016.* AAAI Press, 319–328. Retrieved from http://www.aaai.org/ocs/index.php/KR/KR16/paper/view/12860.

[18] Alexander Borgida, David Toman, and Grant E. Weddell. 2017. Concerning referring expressions in query answers. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence, IJCAI 2017.* ijcai.org, 4791–4795. DOI: https://doi.org/10.24963/ijcai.2017/668

[19] Lorenz Bühmann, Jens Lehmann, and Patrick Westphal. 2016. DL-learner - a framework for inductive learning on the semantic web. *Journal of Web Semantics* 39 (2016), 15–24. https://www.sciencedirect.com/journal/journal-of-web-semantics/vol/39/suppl/C

[20] Lorenz Bühmann, Jens Lehmann, Patrick Westphal, and Simon Bin. 2018. DL-learner structured machine learning on semantic web data. In *Proceedings of the Web Conference Companion 2018, WWW 2018.* ACM, 467–471. DOI: https://doi.org/10.1145/3184558.3186235

[21] Diego Calvanese, Silvio Ghilardi, Alessandro Gianola, Marco Montali, and Andrey Rivkin. 2020. Combined covers and beth definability. In *Proceedings of the 10th International Joint Conference on Automated Reasoning, Part I, IJCAR 2020.* Springer, 181–200. DOI: https://doi.org/10.1007/978-3-030-51074-9_11

[22] Diego Calvanese, Silvio Ghilardi, Alessandro Gianola, Marco Montali, and Andrey Rivkin. 2022. Combination of uniform interpolants via beth definability. *Journal of Automated Reasoning* 66, 3 (2022), 409–435. DOI: https://doi.org/10.1007/s10817-022-09627-1

[23] Ashok K. Chandra, Dexter C. Kozen, and Larry J. Stockmeyer. 1981. Alternation. *Journal of the ACM* 28, 1 (1981), 114–133. DOI: https://doi.org/10.1145/322234.322243

[24] C. C. Chang and H. Jerome Keisler. 1998. *Model Theory.* Elsevier.

[25] Alessandro Cimatti, Alberto Griggio, and Roberto Sebastiani. 2009. Interpolant generation for UTVPI. In *Proceedings of the 22nd International Conference on Automated Deduction, CADE 2022.* Springer, 167–182. DOI: https://doi.org/10.1007/978-3-642-02959-2_15

[26] S. D. Comer. 1969. Classes without the amalgamation property. *Pacific Journal of Mathematics* 28, 2 (1969), 309–318.

[27] William Craig. 1957. Three uses of the Herbrand-Gentzen theorem in relating model theory and proof theory. *Journal of Symbolic Logic* 22, 3 (1957), 269–285. DOI: https://doi.org/10.2307/2963594

[28] Giovanna D'Agostino and Marco Hollenberg. 1996. Uniform interpolation, automata and the modal μ-Calculus. In *Proceedings of the 1st Workshop on Advances in Modal Logic, AiML-1*. CSLI Publications, 73–84.

[29] Razvan Diaconescu. 1993. Logical support for modularisation. *Logical Environments* 83 (1993), 130. https://dl.acm.org/doi/10.5555/185881.185899

[30] Nicola Fanizzi, Claudia d'Amato, and Floriana Esposito. 2008. DL-FOIL concept learning in description logics. In *Proceedings of the 18th International Conference on Inductive Logic Programming, ILP 2008*. Springer, 107–121. DOI: https://doi.org/10.1007/978-3-540-85928-4_12

[31] Nicola Fanizzi, Giuseppe Rizzo, Claudia d'Amato, and Floriana Esposito. 2018. DLFoil: Class expression learning revisited. In *Proceedings of the 21st International Conference on Knowledge Engineering and Knowledge Management, EKAW 2018*. Springer, 98–113. DOI: https://doi.org/10.1007/978-3-030-03667-6_7

[32] Kit Fine. 1979. Failures of the interpolation lemma in quantified modal logic. *Journal of Symbolic Logic* 44, 2 (1979), 201–206. DOI: https://doi.org/10.2307/2273727

[33] Marie Fortin, Boris Konev, and Frank Wolter. 2022. Interpolants and explicit definitions in extensions of the description logic $\mathcal{EL}$. In *Proceedings of the 19th International Conference on Principles of Knowledge Representation and Reasoning, KR 2022*.

[34] Enrico Franconi and Volha Kerhet. 2019. Effective query answering with ontologies and DBoxes. In *Description Logic, Theory Combination, and All That—Essays Dedicated to Franz Baader on the Occasion of His 60th Birthday*. Springer, 301–328. DOI: https://doi.org/10.1007/978-3-030-22102-7_14

[35] Enrico Franconi, Volha Kerhet, and Nhung Ngo. 2013. Exact query reformulation over databases with first-order and description logics ontologies. *Journal of Artificial Intelligence Research* 48 (2013), 885–922. https://www.jair.org/index.php/jair/article/view/10854

[36] Maurice Funk, Jean Christoph Jung, Carsten Lutz, Hadrien Pulcini, and Frank Wolter. 2019. Learning description logic concepts: When can positive and negative examples be separated?. In *Proceedings of the 28h International Joint Conference on Artificial Intelligence, IJCAI 2019*. 1682–1688. DOI: https://doi.org/10.24963/ijcai.2019/233

[37] Dov M. Gabbay. 1972. Craig's interpolation theorem for modal logics. In *Proceedings of the Conference in Mathematical Logic—London 1970*. Springer, 111–127.

[38] Dov M. Gabbay, Agi Kurucz, Frank Wolter, and Michael Zakharyaschev. 2003. Many-dimensional Modal Logics: Theory and Applications. North Holland Publishing Company.

[39] Amit Goel, Sava Krstic, and Cesare Tinelli. 2009. Ground interpolation for combined theories. In *Proceedings of the 22nd International Conference on Automated Deduction, CADE 2022*. Springer, 183–198. DOI: https://doi.org/10.1007/978-3-642-02959-2_16

[40] Valentin Goranko and Martin Otto. 2007. Model theory of modal logic. In *Handbook of Modal Logic*. Elsevier, 249–329.

[41] K. Henkell. 1988. Pointlike sets: The finest aperiodic cover of a finite semigroup. *Journal of Pure and Applied Algebra* 55, 1-2 (1988), 85–126.

[42] K. Henkell, J. Rhodes, and B. Steinberg. 2010. Aperiodic pointlikes and beyond. *International Journal of Algebra and Computation* 20, 2 (2010), 287–305.

[43] Eva Hoogland and Maarten Marx. 2002. Interpolation and definability in guarded fragments. *Studia Logica* 70, 3 (2002), 373–409.

[44] Eva Hoogland, Maarten Marx, and Martin Otto. 1999. Beth definability for the guarded fragment. In *Proceedings of the 6th International Conference on Logic Programming and Automated Reasoning, LPAR 1999*. Springer, 273–285. DOI: https://doi.org/10.1007/3-540-48242-3_17

[45] Luigi Iannone, Ignazio Palmisano, and Nicola Fanizzi. 2007. An algorithm based on counterfactuals for concept learning in the Semantic Web. *Applied Intelligence* 26, 2 (2007), 139–159.

[46] Rosalie Iemhoff. 2019. Uniform interpolation and the existence of sequent calculi. *Annals of Pure and Applied Logic* 170, 11 (2019), 102711.

[47] Ernesto Jiménez-Ruiz, Terry R. Payne, Alessandro Solimando, and Valentina A. M. Tamma. 2016. Limiting logical violations in ontology alignnment through negotiation. In *Proceedings of the 15th International Conference on Principles of Knowledge Representation and Reasoning, KR 2016*. 217–226. Retrieved from http://www.aaai.org/ocs/index.php/KR/KR16/paper/view/12893.

[48] Jean Christoph Jung, Carsten Lutz, Hadrien Pulcini, and Frank Wolter. 2020. Logical separability of incomplete data under ontologies. In *Proceedings of the 17th International Conference on Principles of Knowledge Representation and Reasoning, KR 2020*. 517–528. DOI: https://doi.org/10.24963/kr.2020/52

[49] Jean Christoph Jung, Carsten Lutz, Hadrien Pulcini, and Frank Wolter. 2021. Separating data examples by description logic concepts with restricted signatures. In *Proceedings of the 18th International Conference on Principles of Knowledge Representation and Reasoning, KR 2021*. 390–399. DOI: https://doi.org/10.24963/kr.2021/37

[50] Jean Christoph Jung, Carsten Lutz, Hadrien Pulcini, and Frank Wolter. 2022. Logical separability of labeled data examples under ontologies. *Artificial Intelligence* 313 (2022), 103785. https://www.sciencedirect.com/journal/artificial-intelligence/vol/313/suppl/C

[51] Jean Christoph Jung, Andrea Mazzullo, and Frank Wolter. 2022. More on interpolants and explicit definitions for description logics with nominals and/or role inclusions. In *Proceedings of the 35th International Workshop on Description Logics, DL 2022.*

[52] Jean Christoph Jung and Frank Wolter. 2021. Living without Beth and Craig: Definitions and interpolants in the guarded and two-variable fragments. In *Proceedings of the 36th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2021.* IEEE, 1–14. DOI: https://doi.org/10.1109/LICS52264.2021.9470585

[53] Emanuel Kieronski, Ian Pratt-Hartmann, and Lidia Tendera. 2018. Two-variable logics with counting and semantic constraints. *ACM SIGLOG News* 5, 3 (2018), 22–43. DOI: https://doi.org/10.1145/3242953.3242958

[54] Boris Konev, Carsten Lutz, Denis K. Ponomaryov, and Frank Wolter. 2010. Decomposing description logic ontologies. In *Proceedings of the 12th International Conference on Principles of Knowledge Representation and Reasoning, KR 2010.* AAAI Press.

[55] Boris Konev, Carsten Lutz, Dirk Walther, and Frank Wolter. 2009. Formal properties of modularisation. In *Proceedings of the Modular Ontologies: Concepts, Theories and Techniques for Knowledge Modularization.* Springer, 25–66. DOI: https://doi.org/10.1007/978-3-642-01907-4_3

[56] Boris Konev, Dirk Walther, and Frank Wolter. 2009. Forgetting and uniform interpolation in large-scale description logic terminologies. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence, IJCAI 2009.* 830–835. Retrieved from http://ijcai.org/Proceedings/09/Papers/142.pdf.

[57] Patrick Koopmann and Renate A. Schmidt. 2014. Count and forget: Uniform interpolation of $\mathcal{SHQ}$-ontologies. In *Proceedings of the 7th International Joint Conference on Automated Reasoning, IJCAR 2014.* Springer, 434–448. DOI: https://doi.org/10.1007/978-3-319-08587-6_34

[58] Patrick Koopmann and Renate A. Schmidt. 2015. Uniform interpolation and forgetting for $\mathcal{ALC}$ ontologies with ABoxes. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence, AAAI 2015.* AAAI Press, 175–181. Retrieved from http://www.aaai.org/ocs/index.php/AAAI/AAAI15/paper/view/9981.

[59] Tomasz Kowalski and George Metcalfe. 2019. Uniform interpolation and coherence. *Annals of Pure and Applied Logic* 170, 7 (2019), 825–841.

[60] Emiel Krahmer and Kees van Deemter. 2012. Computational generation of referring expressions: A survey. *Computational Linguistics* 38, 1 (2012), 173–218.

[61] Agi Kurucz, Frank Wolter, and Michael Zakharyaschev. 2023. Definitions and (uniform) interpolants in first-order modal logic. In *Proceedings of the 20th International Conference on Principles of Knowledge Representation and Reasoning.* 417–428. DOI: https://doi.org/10.24963/kr.2023/41

[62] Jens Lehmann and Christoph Haase. 2009. Ideal downward refinement in the $\mathcal{EL}$ description logic. In *Proceedings of the 19th International Conference on Inductive Logic Programming, ILP 2009.* Springer, 73–87. DOI: https://doi.org/10.1007/978-3-642-13840-9_8

[63] Jens Lehmann and Pascal Hitzler. 2010. Concept learning in description logics using refinement operators. *Machine Learning* 78 (2010), 203–250. https://link.springer.com/article/10.1007/s10994-009-5146-2

[64] Francesca A. Lisi. 2012. A formal characterization of concept learning in description logics. In *Proceedings of the 25th International Workshop on Description Logics, DL 2012.* CEUR-WS.org. Retrieved from http://ceur-ws.org/Vol-846/paper_66.pdf.

[65] Francesca A. Lisi and Umberto Straccia. 2015. Learning in description logics with fuzzy concrete domains. *Fundamenta Informaticae* 140, 3-4 (2015), 373–391. DOI: https://doi.org/10.3233/FI-2015-1259

[66] Hongkai Liu, Carsten Lutz, Maja Milicic, and Frank Wolter. 2011. Foundations of instance level updates in expressive description logics. *Artificial Intelligence* 175, 18 (2011), 2170–2197. DOI: https://doi.org/10.1016/j.artint.2011.08.003

[67] Carsten Lutz, Robert Piro, and Frank Wolter. 2011. Description logic TBoxes: Model-theoretic characterizations and rewritability. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence, IJCAI 2011.* IJCAI/AAAI, 983–988. DOI: https://doi.org/10.5591/978-1-57735-516-8/IJCAI11-169

[68] Carsten Lutz, Inanç Seylan, and Frank Wolter. 2012. An automata-theoretic approach to uniform interpolation and approximation in the description logic EL. In *Proceedings of the 13th International Conference on Principles of Knowledge Representation and Reasoning, KR 2012.* AAAI Press. Retrieved from http://www.aaai.org/ocs/index.php/KR/KR12/paper/view/4511.

[69] Carsten Lutz, Inanç Seylan, and Frank Wolter. 2019. The data complexity of ontology-mediated queries with closed predicates. *Logical Methods in Computer Science* 15, 3 (2019), 23:1–23:54.

[70] Carsten Lutz and Frank Wolter. 2011. Foundations for uniform interpolation and forgetting in expressive description logics. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence, IJCAI 2011.* IJCAI/AAAI, 989–995. DOI: https://doi.org/10.5591/978-1-57735-516-8/IJCAI11-170

[71] Larisa Maksimova and Dov Gabbay. 2005. *Interpolation and Definability in Modal and Intuitionistic Logics.* Clarendon Press.

[72] Maarten Marx. 1998. Interpolation in modal logic. In *Proceedings of the 7th International Conference on Algebraic Methodology and Software Technology, AMAST 1998*. Springer, 154–163. DOI: https://doi.org/10.1007/3-540-49253-4_13

[73] Maarten Marx and Carlos Areces. 1998. Failure of interpolation in combined modal logics. *Notre Dame Journal of Formal Logic* 39, 2 (1998), 253–273.

[74] Kenneth L. McMillan. 2003. Interpolation and SAT-based model checking. In *Proceedings of the 15th International Conference on Computer Aided Verification, CAV 2003*. Springer, 1–13. DOI: https://doi.org/10.1007/978-3-540-45069-6_1

[75] Nadeschda Nikitina and Sebastian Rudolph. 2014. (Non-)Succinctness of uniform interpolants of general terminologies in the description logic $\mathcal{EL}$. *Artificial Intelligence* 215 (2014), 120–140. https://www.sciencedirect.com/science/article/pii/S0004370214000824?via%3Dihub

[76] D. Pigozzi. 1971. Amalgamation, congruence-extension, and interpolation properties in algebras. *Algebra Universalis* 1 (1971), 269–349. https://link.springer.com/article/10.1007/BF02944991

[77] Andrew M. Pitts. 1992. On an interpretation of second order quantification in first order intuitionistic propositional logic. *Journal of Symbolic Logic* 57, 1 (1992), 33–52. DOI: https://doi.org/10.2307/2275175

[78] Thomas Place. 2018. Separating regular languages with two quantifier alternations. *Logical Methods in Computer Science* 14, 4:16 (2018), 1–58.

[79] Thomas Place and Marc Zeitoun. 2016. Separating regular languages with first-order logic. *Logical Methods in Computer Science* 12, 1:5 (2016), 1–30.

[80] Thomas Place and Marc Zeitoun. 2020. Adding successor: A transfer theorem for separation and covering. *ACM Transactions on Computational Logic* 21, 2 (2020), 9:1–9:45. DOI: https://doi.org/10.1145/3356339

[81] Wolfgang Rautenberg. 1983. Modal tableau calculi and interpolation. *Journal of Philosophical Logic* 12, 4 (1983), 403–423.

[82] Giuseppe Rizzo, Nicola Fanizzi, and Claudia d'Amato. 2020. Class expression induction as concept space exploration: From DL-Foil to DL-Focl. *Future Generation Computer Systems* 108 (2020), 256–272. https://www.sciencedirect.com/science/article/abs/pii/S0167739X19303991

[83] Giuseppe Rizzo, Nicola Fanizzi, Claudia d'Amato, and Floriana Esposito. 2018. A framework for tackling myopia in concept learning on the web of data. In *Proceedings of the 21st International Conference on Knowledge Engineering and Knowledge Management, EKAW 2018*. Springer, 338–354. DOI: https://doi.org/10.1007/978-3-030-03667-6_22

[84] Md. Kamruzzaman Sarker and Pascal Hitzler. 2019. Efficient concept induction for description logics. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence, AAAI 2019*. AAAI Press, 3036–3043. DOI: https://doi.org/10.1609/aaai.v33i01.33013036

[85] Inanç Seylan, Enrico Franconi, and Jos de Bruijn. 2009. Effective query rewriting with ontologies over DBoxes. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence, IJCAI 2009*. 923–925. Retrieved from http://ijcai.org/Proceedings/09/Papers/157.pdf.

[86] Balder ten Cate. 2005. Interpolation for extended modal languages. *Journal of Symbolic Logic* 70, 1 (2005), 223–234.

[87] Balder ten Cate. 2022. *Lyndon Interpolation for Modal Logic via Type Elimination Sequences*. Technical Report. ILLC, Amsterdam.

[88] Balder ten Cate, Willem Conradie, Maarten Marx, and Yde Venema. 2006. Definitorially complete description logics. In *Proceedings of the 10th International Conference on Principles of Knowledge Representation and Reasoning, KR 2006*. AAAI Press, 79–89. Retrieved from http://www.aaai.org/Library/KR/2006/kr06-011.php.

[89] Balder ten Cate, Enrico Franconi, and Inanç Seylan. 2013. Beth definability in expressive description logics. *Journal of Artificial Intelligence Research* 48 (2013), 347–414. https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahUKEwjMqYzH0fqAAxUy1QIHHcVnAVoQFnoECBAQAQ&url=https%3A%2F%2Fjair.org%2Findex.php%2Fjair%2Farticle%2Fview%2F10842%2F25876&usg=AOvVaw1p4oavuaUl58jYzXtEhInH&opi=89978449

[90] David Toman and Grant E. Weddell. 2011. *Fundamentals of Physical Design and Query Compilation*. Morgan & Claypool Publishers.

[91] David Toman and Grant E. Weddell. 2020. First order rewritability for ontology mediated querying in horn-DLFD. In *Proceedings of the 33rd International Workshop on Description Logics, DL 2020*. CEUR-WS.org.

[92] David Toman and Grant E. Weddell. 2021. FO rewritability for OMQ using beth definability and interpolation. In *Proceedings of the 34th International Workshop on Description Logics, DL 2021*. CEUR-WS.org. Retrieved from http://ceur-ws.org/Vol-2954/paper-29.pdf.

[93] An C. Tran, Jens Dietrich, Hans W. Guesgen, and Stephen Marsland. 2017. Parallel symmetric class expression learning. *Journal of Machine Learning Research* 18 (2017), 64:1–64:34. https://jmlr.csail.mit.edu/preview/14-317.pdf

[94] Johan van Benthem. 2008. The many faces of interpolation. *Synth.* 164, 3 (2008), 451–460. DOI: https://doi.org/10.1007/s11229-008-9351-5

[95] Albert Visser. 1996. Uniform interpolation and layered bisimulation. In *Gödel'96: Logical Foundations of Mathematics, Computer Science and Physics—Kurt Gödel's Legacy*. Association for Symbolic Logic. 139–164. https://www.routledge.com/Godel-96-Logical-Foundations-of-Mathematics-Computer-Science-and-Physics/Hajek/p/book/9781568811536

[96] Yizheng Zhao and Renate A. Schmidt. 2016. Forgetting concept and role symbols in $\mathcal{ALCOIH}$-Ontologies. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence, IJCAI 2016*. IJCAI/AAAI Press, 1345–1353. Retrieved from http://www.ijcai.org/Abstract/16/194.

Received 25 August 2022; accepted 4 April 2023