Fusions of Description Logics and Abstract Description Systems

Franz Baader[1] Carsten Lutz[1] Holger Sturm[2] Frank Wolter[3]

[1] LuFG Theoretical Computer Science, RWTH Aachen,
Ahornstraße 55, 52074 Aachen, Germany

[2] Fachbereich Philosophie, Universität Konstanz,
78457 Konstanz, Germany

[3] Institut für Informatik, Universität Leipzig,
Augustus-Platz 10-11, 04109 Leipzig, Germany

emails: baader@cs.rwth-aachen.de, lutz@cs.rwth-aachen.de,
holger.sturm@uni-konstanz.de, wolter@informatik.uni-leipzig.de

Abstract

Fusions are a simple way of combining logics. For normal modal logics, fusions have been investigated in detail. In particular, it is known that, under certain conditions, decidability transfers from the component logics to their fusion. Though description logics are closely related to modal logics, they are not necessarily normal. In addition, ABox reasoning in description logics is not covered by the results from modal logics.

In this paper, we extend the decidability transfer results from normal modal logics to a large class of description logics. To cover different description logics in a uniform way, we introduce abstract description systems, which can be seen as a common generalization of description and modal logics, and show the transfer results in this general setting.

1 Introduction

Knowledge representation systems based on Description Logics (DL) can be used to represent the knowledge of an application domain in a structured and formally well-understood way [13, 7, 12, 50, 11]. In such systems, the important notions of the domain can be described by *concept descriptions*, i.e., expressions that are built from atomic concepts (unary predicates) and atomic roles (binary predicates) using the concept constructors provided by the Description Logic employed by the system. The atomic concepts and the concept descriptions represent sets of individuals, whereas roles represent binary relations between individuals. For example, using the atomic concepts Woman and Human, and the atomic role child, the concept of all *women having only daughters* (i.e., women such that all their children are again women) can be represented by the description $\mathsf{Woman} \sqcap \forall \mathsf{child}.\mathsf{Woman}$, and the concept of all *mothers* by the description $\mathsf{Woman} \sqcap \exists \mathsf{child}.\mathsf{Human}$. In this example, we have used the constructors concept conjunction ($\sqcap$), value restriction ($\forall R.C$), and existential restriction ($\exists R.C$). In the DL literature, also various other constructors have been considered. A prominent example are so-called number restrictions, which are available in almost all DL systems. For example, using number restrictions the concept of all *women having exactly two children* can be represented by the concept description

$$\mathsf{Woman} \sqcap (\leq 2\mathsf{child}) \sqcap (\geq 2\mathsf{child}).$$

The knowledge base of a DL system consists of a *terminological component* (TBox) and an *assertional component* (ABox). In its simplest form, the TBox consists of concept definitions, which assign names (abbreviations) to complex descriptions. More general TBox formalisms allow for so-called *general concept inclusion* axioms (GCIs) between complex descriptions. For example, the concept inclusion

$$\mathsf{Human} \sqcap (\geq 3 \mathsf{child}) \sqsubseteq \exists \mathsf{entitled}.\mathsf{Taxbreak}$$

states that people having at least three children are entitled to a tax break. The ABox formalism consists of concept assertions (stating that an individual belongs to a concept) and role assertions (stating that two individuals are related by a role). For example, the assertions $\mathsf{Woman}(\mathsf{Mary}), \mathsf{child}(\mathsf{Mary}, \mathsf{TOM}), \mathsf{Human}(\mathsf{TOM})$ state that Mary is a woman, who has a child, Tom, who is a human.

DL systems provide their users with various inference capabilities that allow them to deduce implicit knowledge from the explicitly represented knowledge. For instance, the *subsumption* algorithm allows one to determine subconcept-superconcept relationships: C is subsumed by D ($C \sqsubseteq D$) if, and only if, all instances of C are also instances of D, i.e., the first description is always interpreted as a subset of the second description. For example, the concept description Woman obviously subsumes the concept description $\mathsf{Woman} \sqcap \forall \mathsf{child}.\mathsf{Woman}$. In DLs allowing for conjunction and negation of concepts, subsumption can be reduced to (un)satisfiability: $C \sqsubseteq D$ iff $C \sqcap \neg D$ is unsatisfiable. The concept description C is *satisfiable* iff it is non-contradictory, i.e., it can be interpreted by a nonempty set. The *instance* algorithm allows one to decide whether a given individual is an instance of a given concept. For example, w.r.t. the assertions from above, Mary is an instance of the concept description $\mathsf{Woman} \sqcap \exists \mathsf{child}.\mathsf{Human}$. In DLs allowing for negation of concepts, the instance problem can be reduced to (in)consistency of ABoxes: i is an instance of C w.r.t. the ABox $\mathcal{A}$ iff $\mathcal{A} \cup \{\neg C(i)\}$ is inconsistent. The ABox $\mathcal{A}$ is *consistent* iff it is non-contradictory, i.e., it has a model.

In order to ensure a reasonable and predictable behaviour of a DL system, reasoning in the DL employed by the system should at least be decidable, and preferably of low complexity. Consequently, the expressive power of the DL in question must be restricted in an appropriate way. If the imposed restrictions are too severe, however, then the important notions of the application domain can no longer be expressed. Investigating this trade-off between the expressivity of DLs and the complexity of their inference problems has thus been one of the most important issues in DL research (see, e.g., [35, 39, 45, 46, 40, 19, 18, 17, 43, 20]).

This paper investigates an approach for extending the expressivity of DLs that (in many cases) guarantees that reasoning remains decidable: the fusion of DLs. In order to explain the difference between the usual union and the fusion of DLs, let us consider a simple example. Assume that the DL $\mathcal{D}_1$ is $\mathcal{ALC}$, i.e., it provides for the Boolean operators $\sqcap$, $\sqcup$, $\neg$ and the additional concept constructors value restriction $\forall R.C$ and existential restriction $\exists R.C$, and that the DL $\mathcal{D}_2$ provides for the Boolean operators and number restrictions $(\leq nR)$ and $(\geq nR)$. If an application requires concept constructors from both DLs for expressing its relevant concepts, then one would usually consider the *union* $\mathcal{D}_1 \cup \mathcal{D}_2$ of $\mathcal{D}_1$ and $\mathcal{D}_2$, which allows for the unrestricted use of all constructors. For example, the concept description $C_1 := (\exists R.A) \sqcap (\exists R.\neg A) \sqcap (\leq 1R)$ is a legal $\mathcal{D}_1 \cup \mathcal{D}_2$ description. Note that this description is unsatisfiable, due to the interaction between constructors of $\mathcal{D}_1$ and $\mathcal{D}_2$. The *fusion* $\mathcal{D}_1 \otimes \mathcal{D}_2$ of $\mathcal{D}_1$ and $\mathcal{D}_2$ prevents such interactions by imposing the following restriction: one assumes that the set of all role names is partitioned into two sets, one that can be used in constructors of $\mathcal{D}_1$, and another one that can be used in constructors of $\mathcal{D}_2$. Thus, the description C_1 from above is not a legal $\mathcal{D}_1 \otimes \mathcal{D}_2$ description since it uses the same role R both in the existential restrictions (which are $\mathcal{D}_1$-constructors) and in the number restriction (which is a $\mathcal{D}_2$-constructor). In contrast, the descriptions $(\exists R_1.A) \sqcap (\exists R_1.\neg A) \sqcap (\leq 1R_2)$ and $(\exists R_1.(\leq 1R_2))$ are admissible in $\mathcal{D}_1 \otimes \mathcal{D}_2$ since they employ different roles in the $\mathcal{D}_1$- and $\mathcal{D}_2$-constructors. If the concepts that must be expressed are such that they require both constructors from $\mathcal{D}_1$ and $\mathcal{D}_2$, but the ones from $\mathcal{D}_1$ for other roles than the ones from $\mathcal{D}_2$, then one does not really need the union of $\mathcal{D}_1$ and $\mathcal{D}_2$; the fusion would be sufficient.

What is the advantage of taking the fusion instead of the union? Basically, for the union of two DLs one must design new reasoning methods, whereas reasoning in the fusion can be reduced to reasoning in the component DLs. Indeed, reasoning in the union may even be undecidable whereas reasoning in the fusion is still decidable. As an example, we consider the DLs (i) $\mathcal{ALCF}$, which extends the basic DL $\mathcal{ALC}$ by functional roles (features) and the same-as constructor (agreement) on chains of functional roles [30, 5]; and (ii) $\mathcal{ALC}^{+,\circ,\sqcup}$, which extends $\mathcal{ALC}$ by transitive closure, composition, and union of roles [3, 44]. For both DLs, subsumption of concept descriptions is known to be decidable [30, 44, 3]. However, their *union* $\mathcal{ALCF}^{+,\circ,\sqcup}$ has an undecidable subsumption problem [5]. This undecidability result depends on the fact that, in $\mathcal{ALCF}^{+,\circ,\sqcup}$, the role constructors transitive closure, composition, and union can be applied to functional roles that also appear within the same-as constructor. This is not allowed in the fusion $\mathcal{ALCF} \otimes \mathcal{ALC}^{+,\circ,\sqcup}$. Of course, failure of a certain undecidability proof does not make the fusion decidable.

Why do we know that the fusion of decidable DLs is again decidable? Actually, in general we don't, and this was our main reason for writing this paper. The notion "fusion" was introduced and investigated in modal logic, basically to transfer results like finite axiomatizability, decidability, finite model property, etc. from uni-modal logics (with one pair of box and diamond operators) to multi-modal logics (with several such pairs, possibly satisfying different axioms). This has led to rather general transfer results (see, e.g., [49, 34, 21, 47, ?] for results that concern decidability), which are sometimes restricted to so-called *normal* modal logics [14]. Since there is a close relationship between modal logics and DLs [44], it is clear that these transfer results also apply to some DLs. The question is, however, to which DLs exactly and to which inference problems. First, some DLs allow for constructors that are not considered in modal logics (e.g., the same-as constructor mentioned above). Second, some DL constructors that have been considered in modal logics (like qualified number restrictions $(\leq nR.C)$, $(\geq nR.C)$ [29], which correspond to graded modalities [48]) can easily be shown to be non-normal. Third, the transfer results for decidability are concerned with the satisfiability problem (with or without GCIs). ABoxes and the related inference problems are not considered. ABoxes can be simulated in modal logics allowing for so-called nominals, i.e., names for individuals, within formulae [42, 23, 1]. However, as we will see below, the general transfer results do not apply to modal logics with nominals.

The purpose of this paper is to clarify for which DLs decidability of the component DLs transfers to their fusion. To this purpose, we introduce so-called *abstract description systems* (ADSs), which can be seen as a common generalization of description and modal logics. We define the fusion of ADSs, and state four theorems that say under which conditions decidability transfers from the component ADSs to their fusion. Two of these theorems are concerned with inference w.r.t. general concept inclusion axioms and two with inference without TBox axioms. In both cases, we first formulate and prove the results for the consistency problem of ABoxes (more precisely, the corresponding problem for ADSs) and then establish analogous results for the satisfiability problem of concepts.

From the DL point of view, the four theorems shown in this paper are concerned with the following two *decision problems*:

(i) decidability of consistency of ABoxes w.r.t. TBox axioms (Theorem 18);

(ii) decidability of satisfibility of concepts w.r.t. TBox axioms; (Corollary 23);

(iii) decidability of consistency of ABoxes without TBox axioms (Theorem 28); and

(iv) decidability of satisfiability of concepts without TBox axioms (Corollary 33).

These theorems imply that decidability of the consistency problem and the satisfiability problem transfers to the fusion for most DLs considered in the literature. The *main exceptions* (which do not satisfy the prerequisites of the theorems) are

(a) DLs that are not propositionally closed, i.e., do not contain all Boolean connectives;

(b) DLs allowing for individuals (called nominals in modal logic) in concept descriptions; and

(c) DLs explicitly allowing for the universal role or for negation of roles.

Results from modal logic for problem (iv) usually require the component modal logics to be normal. Our Theorem 28 is less restrictive, and thus also applies to DLs allowing for constructors like qualified number restrictions.

2 Description logics

Before defining abstract description systems in the next section, we introduce the main features of DLs that must be covered by this definition. To this purpose, we first introduce $\mathcal{ALC}$, the basic DL containing all Boolean connectives, and the relevant inference problems. Then, we consider different possibilities for extending $\mathcal{ALC}$ to more expressive DLs.

Definition 1 ($\mathcal{ALC}$ Syntax). Let N_C, N_R, and N_I be pairwise disjoint sets of concept, role, and individual names, respectively. The set of $\mathcal{ALC}$ concept descriptions is the smallest set such that

1. every concept name is a concept description,
2. if C and D are concept descriptions and R is a role name, then the following expressions are also concept descriptions:
   - $\neg C$ (negation), $C \sqcap D$ (conjunction), $C \sqcup D$ (disjunction),
   - $\exists R.C$ (existential restriction), and $\forall R.C$ (value restriction).

We use $\top$ as an abbreviation of $A \sqcup \neg A$ and $\bot$ as an abbreviation for $A \sqcup \neg A$ (where A is an arbitrary concept name).

Let C and D be concept descriptions. Then $C \sqsubseteq D$ is a *general concept inclusion axiom* (GCI). A finite set of such axioms if called a *TBox*.

Let C be a concept description, R a role name, and i, j individual names. Then $C(i)$ is a *concept assertion* and $R(i, j)$ a *role assertion*. A finite set of such assertions is called an ABox.

The meaning of $\mathcal{ALC}$-concept descriptions, TBoxes, and ABoxes can be defined with the help of a set-theoretic semantics.

Definition 2 ($\mathcal{ALC}$ Semantics). An $\mathcal{ALC}$-*interpretation* $\mathcal{I}$ is a pair $(\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where $\Delta^{\mathcal{I}}$ is a nonempty set, the *domain* of the interpretation, and $\cdot^{\mathcal{I}}$ is the *interpretation function*. The interpretation function maps

- each concept name A to a subset $A^{\mathcal{I}}$ of $\Delta^{\mathcal{I}}$,
- each role name R to a subset $R^{\mathcal{I}}$ of $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$,
- each individual name i to an element $i^{\mathcal{I}}$ of $\Delta^{\mathcal{I}}$ such that different names are mapped to different elements (unique name assumption).

For a role name R and an element $a \in \Delta^{\mathcal{I}}$ we define $R^{\mathcal{I}}(a) := \{b \mid (a, b) \in R^{\mathcal{I}}\}$. The interpretation function can inductively be extended to complex concepts as follows:

$$
\begin{aligned}
(\neg C)^{\mathcal{I}} &:= \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}} \\
(C \sqcap D)^{\mathcal{I}} &:= C^{\mathcal{I}} \cap D^{\mathcal{I}} \\
(C \sqcup D)^{\mathcal{I}} &:= C^{\mathcal{I}} \cup D^{\mathcal{I}} \\
(\exists R.C)^{\mathcal{I}} &:= \{a \in \Delta^{\mathcal{I}} \mid R^{\mathcal{I}}(a) \cap C^{\mathcal{I}} \neq \emptyset\} \\
(\forall R.C)^{\mathcal{I}} &:= \{a \in \Delta^{\mathcal{I}} \mid R^{\mathcal{I}}(a) \subseteq C^{\mathcal{I}}\}
\end{aligned}
$$

An interpretation $\mathcal{I}$ is a *model of the TBox* $\mathcal{T}$ iff it satisfies $C^\mathcal{I} \subseteq D^\mathcal{I}$ for all GCIs $C \sqsubseteq D$ in $\mathcal{T}$. It is a *model of the ABox* $\mathcal{A}$ iff it satisfies $i^\mathcal{I} \in C^\mathcal{I}$ for all concept assertions $C(i) \in \mathcal{A}$ and $(i,j)^\mathcal{I} \in R^\mathcal{I}$ for all role assertions $R(i,j) \in \mathcal{A}$. Finally, $\mathcal{I}$ is a *model of an ABox relative to a TBox* iff it is a model of both the ABox and the TBox.

Given this semantics, we can now formally define the relevant inference problems.

Definition 3 (Inferences). Let C and D be concept descriptions, i an individual name, $\mathcal{T}$ a TBox, and $\mathcal{A}$ an ABox. We say that C *subsumes* D *relative to the TBox* $\mathcal{T}$ ($D \sqsubseteq_\mathcal{T} C$) iff $D^\mathcal{I} \subseteq C^\mathcal{I}$ for all models $\mathcal{I}$ of $\mathcal{T}$. The concept description C is *satisfiable relative to the TBox* $\mathcal{T}$ iff there exists a model $\mathcal{I}$ of $\mathcal{T}$ such that $C^\mathcal{I} \neq \emptyset$. The individual i is an *instance* of C in the ABox $\mathcal{A}$ *relative to the TBox* $\mathcal{T}$ iff $i^\mathcal{I} \in C^\mathcal{I}$ for all models of $\mathcal{A}$ relative to $\mathcal{T}$. The ABox $\mathcal{A}$ is *consistent relative to the TBox* $\mathcal{T}$ iff there exists a model of $\mathcal{A}$ relative to $\mathcal{T}$.

These three inferences can also be considered without reference to a TBox: C *subsumes* D (C is satisfiable) iff C subsumes D (C is satisfiable) relative to the empty TBox, and i is an *instance* of C in $\mathcal{A}$ ($\mathcal{A}$ is *consistent*) iff i is an instance of C in $\mathcal{A}$ ($\mathcal{A}$ is consistent) relative to the empty TBox.

As mentioned in the introduction, subsumption can be reduced to (un)satisfiability since $C \sqsubseteq_\mathcal{T} D$ iff $C \sqcap \neg D$ is unsatisfiable relative to $\mathcal{T}$. Conversely, (un)satisfiability can be reduced to subsumption since C is unsatisfiable relative to $\mathcal{T}$ iff $C \sqsubseteq_\mathcal{T} \bot$. For this reason, it is irrelevat whether we consider the subsumption or the satisfiability problem in our transfer results.

Similarly, the instance problem can be reduced to the (in)consistency problem and vice versa: i is an instance of C in $\mathcal{A}$ relative to $\mathcal{T}$ iff $\mathcal{A} \cup \{\neg C(i)\}$ is inconsistent relative to $\mathcal{T}$; and $\mathcal{A}$ is inconsistent relative to $\mathcal{T}$ iff i is an instance of $\bot$ in $\mathcal{A}$ relative to $\mathcal{T}$, where i is an arbitrary individual name. Consequently, it is irrelevat whether we consider the instance problem or the consistency problem in our transfer results.

Finally, the satisfiability problem can be reduced to the consistency problem: C is satisfiable relative to $\mathcal{T}$ iff the ABox $\{C(i)\}$ is consistent, where i is an arbitrary individual name. However, the converse need not be true. It should be obvious that this implies that a transfer result for the satisfiability problem does not yield the corresponding transfer result for the consistency problem: from decidability of the consistency problem for the component DLs we can only deduce decidability of the satisfiability problem in their fusion. What might be less obvious is that a transfer result for the consistency problem need not imply the corresponding transfer result for the satisfiability problem: if the satisfiability problems in the component DLs are decidable, then the transfer result for the consistency problem can just not be applied (since the component DLs need not satisfy the stronger prerequisite of this transfer result, namely, decidability of the consistency problem). However, we will show that the method used to show the transfer result for the consistency problem also applies to the satisfiability problem.

2.1 More expressive DLs

There are several possibilities for extending $\mathcal{ALC}$ in order to obtain a more expressive DL. The three most prominent are adding additional concept constructors, adding role constructors, and formulating restrictions on role interpretations. We start with the third possibility, since we need to refer to restrictions on roles when defining certain concept constructors.

Restrictions on role interpretations

These restrictions enforce the interpretations of roles to satisfy certain properties, such as functionality, transitivity, etc. We consider three prominent examples:

1. **Functional roles.** Here one considers a subset N_F of the set of role names N_R, whose elements are called *features*. An interpretation must map features $f \in N_F$ to functional binary relations $f^\mathcal{I} \subseteq \Delta^\mathcal{I} \times \Delta^\mathcal{I}$, i.e., relations satisfying $\forall a,b,c. f^\mathcal{I}(a,b) \wedge f^\mathcal{I}(a,c) \rightarrow b = c$.

Name	Syntax	Semantics	Symbol
Unqualified number restrictions	$\geq nR$ $\leq nR$	$\{a \in \Delta^\mathcal{I} \mid \lvert R^\mathcal{I}(a)\rvert \geq n\}$ $\{a \in \Delta^\mathcal{I} \mid \lvert R^\mathcal{I}(a)\rvert \leq n\}$	$\mathcal{N}$
Qualified number restrictions	$\geq nR.C$ $\leq nR.C$	$\{a \in \Delta^\mathcal{I} \mid \lvert R^\mathcal{I}(a) \cap C^\mathcal{I}\rvert \geq n\}$ $\{a \in \Delta^\mathcal{I} \mid \lvert R^\mathcal{I}(a) \cap C^\mathcal{I}\rvert \leq n\}$	$\mathcal{Q}$
Nominals	I	$I^\mathcal{I} \subseteq \Delta^\mathcal{I}$ with $\lvert I^\mathcal{I}\rvert = 1$	$\mathcal{O}$
Feature agreement and disagreement	$u_1{\downarrow}u_2$ $u_1{\uparrow}u_2$	$\{a \in \Delta^\mathcal{I} \mid \exists b \in \Delta^\mathcal{I} : u_1^\mathcal{I}(a) = b = u_2^\mathcal{I}(a)\}$ $\{a \in \Delta^\mathcal{I} \mid \exists b_1, b_2 \in \Delta^\mathcal{I} : u_1^\mathcal{I}(a) = b_1 \neq b_2 = u_2^\mathcal{I}(b_1)\}$	$\mathcal{F}$

Figure 1: Some Description Logic concept constructors.

We will sometimes treat functional relations as partial functions, and write $f^\mathcal{I}(a) = b$ rather than $f^\mathcal{I}(a, b)$. $\mathcal{ALC}$ extended with features is denoted by $\mathcal{ALC}_f$.

2. **Transitive roles.** Here one considers a subset N_{R^+} of N_R. Role names $R \in N_{R^+}$ are called *transitive roles*. An interpretation must map transitive roles $R \in N_{R^+}$ to transitive binary relations $R^\mathcal{I} \subseteq \Delta^\mathcal{I} \times \Delta^\mathcal{I}$. $\mathcal{ALC}$ extended with transitive roles is denoted by $\mathcal{ALC}_{R^+}$.

3. **Role hierarchies.** A *role inclusion axiom* is an expression of the form $R \sqsubseteq S$ with $R, S \in N_R$. A finite set H of role inclusion axioms is called a *role hierarchy*. An interpretation must satisfy $R^\mathcal{I} \subseteq S^\mathcal{I}$ for all $R \sqsubseteq S \in H$. $\mathcal{ALC}$ extended with a role hierarchy H is denoted by $\mathcal{ALC}_{\mathcal{H}(H)}$. If H is clear from the context or irrelevant, we write $\mathcal{ALC}_\mathcal{H}$ instead of $\mathcal{ALC}_{\mathcal{H}(H)}$.

The above restrictions can also be combined with each other. For example, $\mathcal{ALC}_{\mathcal{H}R^+}$ is $\mathcal{ALC}$ with a role hierarchy and transitive roles.

A remark on role hierarchies is in order: in our definition, if H_1 and H_2 are different role hierarchies, then $\mathcal{ALC}_{\mathcal{H}(H_1)}$ and $\mathcal{ALC}_{\mathcal{H}(H_2)}$ are different DLs. In the DL literature, usually only one logic $\mathcal{ALC}_\mathcal{H}$ is defined and role hierarchies are treated like TBoxes, i.e., satisfiability and subsumption are defined relative to TBoxes *and* role hierarchies (see, e.g., [31]). For our purposes, however, it is more convenient to define one DL per role hierarchy since distinct role hierarchies impose distinct restrictions on the interpretation of roles. The advantages of this approach will become clear later on when frames and abstract description systems are introduced.

Concept constructors

Concept constructors take concept and/or role descriptions and transform them into more complex concept descriptions. In addition to the constructors available in $\mathcal{ALC}$, various other concept constructors are considered in the DL literature. A small collection of such constructors can be found in Figure 1, where $|S|$ denoted the cardinality of a set S. The symbols in the rightmost column indicate the naming scheme for the resulting DL. In contrast to the case of restrictions on role interpretations, the name modifiers for concept constructors are not written in subscript. For example, $\mathcal{ALC}_{\mathcal{H}R^+}$ extended with qualified number restrictions is called $\mathcal{ALCQ}_{\mathcal{H}R^+}$. The syntax of the extended DLs is as expected, i.e., the constructors may be arbitrarily combined. The semantics is obtained by augmenting the semantics of $\mathcal{ALC}$ with the appropriate conditions, which can be found in the third column in Figure 1. Nominals and feature (dis)agreements need some more explanation:

- **Nominals.** We consider a set N_O of (names for) *nominals*, which is pairwise disjoint to the sets N_C, N_R, and N_I. Elements from N_O are often denoted by I (possibly with index). An interpretation must map nominals to singleton subsets of $\Delta^\mathcal{I}$. The intension underlying nominals is that they stand for elements of Δ, just like individual names. However, since we want to use the nominal $I \in N_O$ as a (nullary) concept constructors, I

Name	Syntax	Semantics	Symbol
Role composition	$R_1 \circ R_2$	$\{(a,b) \in \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \mid \exists c \in \Delta^{\mathcal{I}} : (a,c) \in R_1^{\mathcal{I}} \wedge (c,b) \in R_2^{\mathcal{I}}\}$	$\circ$
Role complement	$\overline{R}$	$\{(a,b) \in \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \mid (a,b) \notin R^{\mathcal{I}}\}$	$\bar{\cdot}$
Role conjunction	$R_1 \sqcap R_2$	$\{(a,b) \in \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \mid (a,b) \in R_1^{\mathcal{I}} \wedge (a,b) \in R_2^{\mathcal{I}}\}$	$\sqcap$
Role disjunction	$R_1 \sqcup R_2$	$\{(a,b) \in \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \mid (a,b) \in R_1^{\mathcal{I}} \vee (a,b) \in R_2^{\mathcal{I}}\}$	$\sqcup$
Inverse roles	R^-	$\{(a,b) \in \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \mid (b,a) \in R^{\mathcal{I}}\}$	-1
Transitive closure	R^+	$\{(a,b) \in \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \mid (a,b) \in (R^{\mathcal{I}})^+\}$	$+$
Universal role	U	$\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$	U

For a binary relation R, R^+ denotes the transitive closure of r.

Figure 2: Some Description Logic role constructors.

must interpret them by a set, namely the singleton set consisting of the individual that I denotes.

- **Feature (dis)agreements.** $\mathcal{ALCF}$ is the extension of $\mathcal{ALC}_f$ with feature agreements and disagreements. Beside the additional concept constructors, $\mathcal{ALCF}$ uses feature chains as part of the (dis)agreement constructor. A *feature chain* is an expression of the form $u = f_1 \circ \cdots \circ f_n$. The interpretation $u^{\mathcal{I}}$ of such a feature chain is just the composition of the partial functions $f_1^{\mathcal{I}}, \ldots, f_n^{\mathcal{I}}$, where composition is to be read from left to right.

DLs including nominals or feature (dis)agreements *and* additional concept constructors or restrictions on role interpretations are defined (and named) in the obvious way.

Role constructors

Role constructors allow us to build complex role descriptions. A collection of role constructors can be found in Figure 2. Again, the rightmost column indicates the naming scheme, where name modifiers for role constructors are written in superscript and separated by commas. For example, $\mathcal{ALCQ}$ with inverse roles and transitive closure is called $\mathcal{ALCQ}^{+,-}$. In DLs admitting role constructors, the set of role descriptions is defined inductively, analogously to the set of concept descriptions. The semantics of role constructors is given in the third column of Figure 2. As with concept descriptions, it can be used to extend the interpretation function from role names to role descriptions.

In a DL with role constructors, role descriptions can be used wherever role names may be used in the corresponding DLs without role constructors. For example,

$$\exists(R_1 \sqcap R_3).C \sqcap \forall(R_2 \sqcup \overline{R}_2).\neg C$$

is an (unsatisfiable) $\mathcal{ALC}^{\bar{\cdot},\sqcap,\sqcup}$-concept description. Note that role descriptions can also be used within role assertions in an ABox.

2.2 Restricting the syntax

Until now, constructors could be combined arbitrarily. Sometimes it makes sense to restrict the interaction between constructors since reasoning in the restricted DL may be easier than reasoning in the unrestricted DL. We will consider DLs imposing certain restrictions on

1. which roles may be used inside certain concept constructors,
2. which roles may be used inside certain role constructors,
3. the combination of role constructors, and
4. the role constructors that may be used inside certain concept constructors.

As an example for the first case, consider the fragment of $\mathcal{ALCQ}_{R^+}$ in which transitive roles may be used in existential and universal restrictions, but not in number restrictions (see, e.g., [32]).

As the result of taking the fusion of two DLs, we will obtain DLs whose set of roles N_R is partitioned. For example, the fusion of $\mathcal{ALCQ}$ with $\mathcal{ALC}^-$ yields a fragment of $\mathcal{ALCQ}^-$ where N_R is partitioned in two sets, say N_{R_1} and N_{R_2}. In this fragment, the inverse role constructor and roles from N_{R_2} may not be used within qualified number restrictions, while roles from N_{R_1} may not be used inside the inverse role constructor.[1] Thus, this DL is an example for both the first, the second, and the fourth case.

Now consider the DL $\mathcal{ALCF}$ introduced above. $\mathcal{ALCF}$ does not only extend $\mathcal{ALC}_f$ with feature (dis)agreement as a concept constructor, but also provides the role composition constructor. However, the role chains built using composition have to be comprised exclusively of features and non-functional roles may not appear inside feature (dis)agreement. Hence, $\mathcal{ALCF}$ is an example for the first, second, and fourth case.

As an example for the third case, consider the fragment of $\mathcal{ALC}^{-,\sqcap}$ in which role conjunction may not be used inside the inverse role constructor.

For these restricted DLs, we do not introduce an explicit naming scheme. Note that we do *not* consider DLs in which the combinability of concept constructors with each other is restricted since these DLs would not fit into the framework of abstract description systems introduced in the next section. An example of such a DL would be one with atomic negation, i.e., where negation may only be applied to concept names.

3 Abstract description systems

In order to define the fusion of DLs and prove general results for fusions of DLs, one needs a formal definition of what are "Description Logics". Since there exists a wide variety of DLs with very different characteristics, we introduce a very general formalization, which should cover all of the DLs considered in the literature, but also includes logics that would usually not be subsumed under the name DL.

In this formalization, concept descriptions will be represented by terms that are built using an abstract description language. General inclusion axioms in DLs will be represented by term assertions and ABox assertions in DLs will be represented by object assertions.

Definition 4 (Abstract description language). An *abstract description language* (ADL) is determined by a countably infinite set V of set variables, a countably infinite set $\mathcal{X}$ of object variables, a (possibly infinite) sequence $(R_i)_{i\in\mathcal{R}}$ of relation symbols of arity two,[2] and a (possibly infinite) sequence $(f_i)_{i\in I}$ of functions symbols, which are equipped with arities $(n_i)_{i\in I}$.

The *terms* t_j of this ADL are built using the follow syntax rules:

$$t_j \quad\longrightarrow\quad x,\ \neg t_1,\ t_1\wedge t_2,\ t_1\vee t_2,\ f_i(t_1,\ldots,t_{n_i}),$$

where $x\in V$ and the Boolean operators $\neg,\wedge,\vee$ are different from all f_i. For a term t, we denote by $\mathit{var}(t)$ the set of set variables used in t. The symbol $\top$ is used as an abbreviation of $x\vee\neg x$ and $\bot$ as an abbreviation for $x\wedge\neg x$ (where x is a set variable).

The *term assertions* of this ADL are

- $t_1 \sqsubseteq t_2$, for all terms t_1, t_2,

and the *object assertions* are

- $R_i(a,b)$, for $a,b\in\mathcal{X}$ and $i\in\mathcal{R}$;

[1]This will become clearer once we have given a formal definition of the fusion.

[2]To keep things simpler, we restrict our attention to the case of binary predicates, i.e., roles in DL. However, the results can easily be extended to the general case.

- $(a : t)$, for $a \in \mathcal{X}$ and t a term.

The sets of term and object assertions together form the set of *assertions* of the ADL.

From the DL point of view, the set variables correspond to concept names, object variables to individual names, relation symbols to roles, and the Boolean operators as well as the function symbols correspond to concept constructors. Thus, terms correspond to concept descriptions. As an example, let us view concept descriptions of the DL $\mathcal{ALCN}^{\sqcap}$, i.e., $\mathcal{ALC}$ extended with number restrictions and conjunction of roles, as terms of an ADL. Value restrictions and existential restrictions can be seen as unary function symbols: for each role description R, we have the function symbols $f_{\forall R}$ and $f_{\exists R}$, which take a term t_C (corresponding to the concept description C) and transform it into the more complex terms $f_{\forall R}(t_C)$ and $f_{\exists R}(t_C)$ (corresponding to the concept descriptions $\forall R.C$ and $\exists R.C$). Similarly, number restrictions can be seen as nullary function symbols: for each role description R and each $n \in \mathbb{N}$, we have the function symbols $f_{\geq nR}$ and $f_{\leq nR}$. Hence, the $\mathcal{ALCN}^{\sqcap}$-concept description $A \sqcap \forall(R_1 \sqcap R_2).\neg(B \sqcap (\geq 2R_1))$ corresponds to the term $x_A \wedge f_{\forall(R_1 \sqcap R_2)}(\neg(x_B \wedge f_{(\geq 2R_1)}))$. We will analyze the connection between ADLs and DLs more formally later on.

The semantics of abstract description systems is defined based on abstract description models.

Definition 5. Let L be an ADL as in Definition 4. An *abstract description model* (ADM) for L is of the form

$$\mathfrak{W} = \left\langle W, F^{\mathfrak{W}} = (f_i^{\mathfrak{W}})_{i \in I}, R^{\mathfrak{W}} = (R_i^{\mathfrak{W}})_{i \in \mathcal{R}} \right\rangle,$$

where W is a nonempty set, the $f_i^{\mathfrak{W}}$ are functions mapping every sequence $\langle X_1, \ldots, X_{n_i} \rangle$ of subsets of W to a subset of W, and the $R_i^{\mathfrak{W}}$ are binary relations on W.

Since ADMs do not interpret variables, we need an assignment that assigns a subset of W to each set variable, before we can evaluate terms in an ADM. To evaluate object assertions, we need an additional assignment that assigns an element of W to each object variable.

Definition 6. Let $\mathcal{L}$ be an ADL and $\mathfrak{W} = \left\langle W, F^{\mathfrak{W}}, R^{\mathfrak{W}} \right\rangle$ be an ADM for $\mathcal{L}$. An *assignment* for $\mathfrak{W}$ is a pair $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ such that $\mathcal{A}_1$ is a mapping from the set of set variables V into 2^W, and $\mathcal{A}_2$ is an injective[3] mapping from the set of object variables $\mathcal{X}$ into W. Let $\mathfrak{W}$ be an ADM and $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ be an assignment for $\mathfrak{W}$. With each $\mathcal{L}$-term t, we inductively associate a value $t^{\mathfrak{W},\mathcal{A}}$ in 2^W as follows:

- $x^{\mathfrak{W},\mathcal{A}} := \mathcal{A}_1(x)$ for all variables $x \in V$,
- $(\neg t)^{\mathfrak{W},\mathcal{A}} := W \setminus (t)^{\mathfrak{W},\mathcal{A}}$, $(t_1 \wedge t_2)^{\mathfrak{W},\mathcal{A}} := t_1^{\mathfrak{W},\mathcal{A}} \cap t_2^{\mathfrak{W},\mathcal{A}}$, $(t_1 \vee t_2)^{\mathfrak{W},\mathcal{A}} := t_1^{\mathfrak{W},\mathcal{A}} \cup t_2^{\mathfrak{W},\mathcal{A}}$,
- $f_i(t_1, \ldots, t_k)^{\mathfrak{W},\mathcal{A}} := f_i^{\mathfrak{W}}(t_1^{\mathfrak{W},\mathcal{A}}, \ldots, t_k^{\mathfrak{W},\mathcal{A}})$.

If $x_1, \ldots, x_n$ are the set variables occurring in t, then we often write $t^{\mathfrak{W}}(X_1, \ldots, X_n)$ as shorthand for $t^{\mathfrak{W},\mathcal{A}}$, where $\mathcal{A}$ is an assignment with $x_i^{\mathcal{A}} = X_i$ for $1 \leq i \leq n$.

The truth-relation $\models$ between $\langle \mathfrak{W}, \mathcal{A} \rangle$ and assertions is defined as follows:

- $\langle \mathfrak{W}, \mathcal{A} \rangle \models R_i(a, b)$ iff $\mathcal{A}_2(a) R_i^{\mathfrak{W}} \mathcal{A}_2(b)$,
- $\langle \mathfrak{W}, \mathcal{A} \rangle \models a : t$ iff $\mathcal{A}_2(a) \in t^{\mathfrak{W},\mathcal{A}}$,
- $\langle \mathfrak{W}, \mathcal{A} \rangle \models t_1 \sqsubseteq t_2$ iff $t_1^{\mathfrak{W},\mathcal{A}} \subseteq t_2^{\mathfrak{W},\mathcal{A}}$.

In this case we say that the assertion is satisfied in $\langle \mathfrak{W}, \mathcal{A} \rangle$

[3] This corresponds to the unique name assumption.

There are two differences between ADMs and DL interpretations. First, in a DL interpretation, the interpretation of the role names fixes the interpretation of the function symbols corresponding to concept constructors that involve roles (like value restrictions, number restrictions, etc.). The interpretation of the concept names corresponds to an assignment. Thus, a DL model is an ADM together with an assignment, whereas an ADM alone corresponds to what is called frame in modal logics. Second, in DL the roles used in concept constructors may, of course, also occur in role assertions. In contrast, the definition of ADMs per se does not enforce any connection between the interpretation of the function symbols and the interpretation of the relation symbols. Such connections can, however, be enforced by restricting the attention to a subclass of all possible ADMs for the ADL.

Definition 7. An *abstract description system* (ADS) is a pair $(\mathcal{L}, \mathcal{M})$, where $\mathcal{L}$ is an ADL and $\mathcal{M}$ is a class of ADMs for $\mathcal{L}$ that is closed under isomorphic copies.

From the DL point of view, the choice of the class $\mathcal{M}$ defines the semantics of the concept and role constructors, and it allows us, e.g., to incorporate restrictions on role interpretations. As mentioned above, in a DL interpretation the interpretation of the function symbols is given by the interpretation of the role names. Thus one can, for example, restrict the class to ADMs that interpret a certain role as a transitive relation. Another restriction that can be realized by the choice of $\mathcal{M}$ is that nominals (corresponding to nullary function symbols) must be interpreted as singleton sets.

Let us now define reasoning problems for abstract description systems. We will introduce satisfiability of sets of assertions (with or without term assertions), which corresponds to consistency of ABoxes (with or without GCIs), and satisfiability of terms (with or without term assertions), which corresponds to satisfiability of concept descriptions (with or without GCIs).

Definition 8. Given an ADS $(\mathcal{L}, \mathcal{M})$, a finite set of assertions Γ is called *satisfiable* in $(\mathcal{L}, \mathcal{M})$ iff there exists an ADM $\mathfrak{W} \in \mathcal{M}$ and an assignment $\mathcal{A}$ such that $\langle \mathfrak{W}, \mathcal{A} \rangle$ satisfies all assertions in Γ. We say that the *satisfiability problem* for $(\mathcal{L}, \mathcal{M})$ is decidable iff there is an algorithm that, given a finite set of *object* assertions Γ, answers "yes" if Γ is satisfiable in $(\mathcal{L}, \mathcal{M})$, and "no" otherwise. The *term* t is called *satisfiable* iff $\{a : t\}$ is satisfiable, where a is an arbitrary object variable.

We say that the *relativized satisfiability problem* for $(\mathcal{L}, \mathcal{M})$ is decidable iff there is an algorithm that, given a finite set of assertions Γ, answers "yes" if Γ is satisfiable in $(\mathcal{L}, \mathcal{M})$, and "no" otherwise. Decidability of the (relativized) *term satisfiability problem* is definied accordingly.

In the next section, we will define the fusion of two ADSs, and show that (relativized) satisfiability is decidable in the fusion if (relativized) satisfiability in the component ADSs is decidable. For this transfer results to hold, we must restrict ourselves to so-called local ADSs.

Definition 9. Given a family $(\mathfrak{W}_p)_{p \in P}$ of ADMs $\mathfrak{W}_p = \left\langle W_p, F^{\mathfrak{W}_p}, R^{\mathfrak{W}_p} \right\rangle$ over pairwise disjoint domains W_p, we say that $\mathfrak{W} = \left\langle W, F^{\mathfrak{W}}, R^{\mathfrak{W}} \right\rangle$ is the *disjoint union* of $(\mathfrak{W}_p)_{p \in P}$ iff

- $W = \bigcup_{p \in P} W_p$,
- $f_i^{\mathfrak{W}}(X_1, \ldots, X_{n_i}) = \bigcup_{p \in P} f_i^{\mathfrak{W}_p}(X_1 \cap W_p, \ldots, X_{n_i} \cap W_p)$ for all $X_1, \ldots, X_{n_i} \subseteq W$ and $i \in I$,
- $R_i^{\mathfrak{W}} = \bigcup_{p \in P} R_i^{\mathfrak{W}_p}$ for all $i \in \mathcal{R}$.

An ADS $S = (\mathcal{L}, \mathcal{M})$ is called *local* iff $\mathcal{M}$ is closed under disjoint unions.

Correspondence to description logics

In the remainder of this section, we analyze the connection between ADSs and DLs in more detail. In particular, we will show that the DLs introduced in Section 2 correspond to ADSs. In order to do this, we first need to introduce frames, a notion well-known from modal logic. Let L be one of the DLs introduced in Section 2.

Definition 10 (Frames). An L-frame $\mathcal{F}$ is a pair $(\Delta^{\mathcal{F}}, \cdot^{\mathcal{F}})$, where $\Delta^{\mathcal{F}}$ is a nonempty set, called the *domain* of $\mathcal{F}$, and $\cdot^{\mathcal{F}}$ is the *interpretation function*, which maps

- each nominal I to a singleton subset $I^{\mathcal{F}}$ of $\Delta^{\mathcal{F}}$, and
- each role name R to a subset $R^{\mathcal{F}}$ of $\Delta^{\mathcal{F}} \times \Delta^{\mathcal{F}}$ such that the restrictions for role interpretations in L are satisfied. For example, in $\mathcal{ALC}_{R^+}$, each $R \in N_{R^+}$ is mapped to a transitive binary relation.

The interpretation function $\cdot^{\mathcal{F}}$ can inductively be extended to complex roles in the obvious way, i.e., by interpreting the role constructors in L according to their semantics as given in Figure 2.

An interpretation $\mathcal{I}$ is *based* on a frame $\mathcal{F}$ iff $\Delta^{\mathcal{I}} = \Delta^{\mathcal{F}}$, $R^{\mathcal{I}} = R^{\mathcal{F}}$ for all roles $R \in N_R$, and $I^{\mathcal{I}} = I^{\mathcal{F}}$ for all nominals $I \in N_O$.

A frame can be viewed as an interpretation that is partial in the sense that the interpretation of individual and concept names is not fixed. Note that (in contrast to the case of concept and individual names) the interpretation of nominals is already fixed in the frame. The reason for this is that, if we do not interpret nominals in the frame, then we have to treat them as set variables on the ADS side. This would, however, have to be variables to which only singleton sets may be assigned. Since such a restriction is not possible in the framework of ADSs as defined above, we interpret nominals in the frame. The consequence is that they correspond to functions of arity 0 on the ADS side.

Now, we define the abstract description system $S = (\mathcal{L}, \mathcal{M})$ corresponding to a DL L. It is straightforward to translate the syntax of L into an abstract description language $\mathcal{L}$.

Definition 11 (Corresponding ADL). Let L be a DL with concept and role constructors as well as restrictions on role interpretations as introduced in Section 2. The corresponding ADL $\mathcal{L}$ is defined as follows. For every concept name A in L, there exists a set variable x_A in $\mathcal{L}$, and for every individual name i in L there exists an object variable a_i in $\mathcal{L}$. Let $\mathcal{R}$ be the set of (possibly complex) role descriptions of L. The set of relation symbols of $\mathcal{L}$ is $\mathcal{R}$, and the set of function symbols of $\mathcal{L}$ is the smallest set containing

1. for every role description $R \in \mathcal{R}$, unary function symbols $f_{\exists R}$ and $f_{\forall R}$,
2. if L provides unqualified number restrictions, then, for every $n \in \mathbb{N}$ and every role description $R \in \mathcal{R}$, function symbols $f_{\geq nR}$ and $f_{\leq nR}$ of arity 0,
3. if L provides qualified number restrictions, then, for every $n \in \mathbb{N}$ and every role $R \in \mathcal{R}$, unary function symbols $f_{\dot{\geq} nR}$ and $f_{\dot{\leq} nR}$,
4. if L provides nominals, then, for every $I \in N_O$, a function symbol f_I of arity 0,
5. if L provides feature agreement and disagreement, then, for every pair of feature chains (u_1, u_2), two function symbols $f_{u_1 \downarrow u_2}$ and $f_{u_1 \uparrow u_2}$ of arity 0.

For an L-concept description C, let t_C denote the representation of C as an $\mathcal{L}$-term, which is defined in the obvious way: concept names A are translated into set variables x_A, the concept constructors $\neg$, $\sqcap$, and $\sqcup$ are mapped to $\neg$, $\wedge$, and $\vee$, respectively, and all other concept constructors are translated to the corresponding function symbols.

Obviously, both the sets of function and relation symbols of $\mathcal{L}$ may be infinite. An example of the translation of concept descriptions into terms of an ADL was already given above (see remark below Definition 4).

We now define the set of abstract description models $\mathcal{M}$ corresponding to the DL L. For every L-frame, $\mathcal{M}$ contains a corresponding ADM.

Definition 12 (Corresponding ADMs). Let $\mathcal{F} = (\Delta^{\mathcal{F}}, \cdot^{\mathcal{F}})$ be a frame. The *corresponding abstract description model* $\mathfrak{W} = \langle W, F^{\mathfrak{W}}, R^{\mathfrak{W}} \rangle$ has domain $W = \Delta^{\mathcal{F}}$. The relation symbols of $\mathcal{L}$ are just the role descriptions of L, and thus they are interpreted in the frame $\mathcal{F}$. For each relation symbol $R_i \in \mathcal{R}$ we can thus define $R_i^{\mathfrak{W}} := R_i^{\mathcal{F}}$.

To define $F^{\mathfrak{W}}$, we need to define $f^{\mathfrak{W}}$ for every nullary function symbol f in $\mathcal{L}$, and $f^{\mathfrak{W}}(X)$ for every unary function symbol f in $\mathcal{L}$ and every $X \subseteq \Delta^{\mathcal{I}}$. Let A be an arbitrary concept name. For each $X \subseteq \Delta^{\mathcal{F}}$, let $\mathcal{I}_X$ be the interpretation based on $\mathcal{F}$ mapping the concept name A to X and every other concept name to $\emptyset$.[4] To define $f^{\mathfrak{W}}$, we make a case distinction according to the type of f:

1. $f^{\mathfrak{W}}_{\exists R}(X) := (\exists R.A)^{\mathcal{I}_X}, \quad f^{\mathfrak{W}}_{\forall R}(X) := (\forall R.A)^{\mathcal{I}_X}$,
2. $f^{\mathfrak{W}}_{\geq nR} := (\geq nR)^{\mathcal{I}_\emptyset}, \quad f^{\mathfrak{W}}_{\leq nR} := (\leq nR)^{\mathcal{I}_\emptyset}$,
3. $f^{\mathfrak{W}}_{\dot{\geq} nR}(X) := (\geq nR.A)^{\mathcal{I}_X}, \quad f^{\mathfrak{W}}_{\dot{\leq} nR}(X) := (\leq nR.A)^{\mathcal{I}_X}$,
4. $f^{\mathfrak{W}}_{I} := I^{\mathcal{I}_\emptyset}$,
5. $f^{\mathfrak{W}}_{u_1 \downarrow u_2} = (u_1 \downarrow u_2)^{\mathcal{I}_\emptyset}, \quad f^{\mathfrak{W}}_{u_1 \uparrow u_2} = (u_1 \uparrow u_2)^{\mathcal{I}_\emptyset}$.

The class of ADMs $\mathcal{M}$ thus obtained from a DL L is obviously closed under isomorphic copies since this also holds for the set of L-frames (independently of which DL L we consider). Hence, the tuple $S = (\mathcal{L}, \mathcal{M})$ corresponding to a DL L is indeed an ADS.

As an example, let us view the DL $\mathcal{ALCN}^{\sqcap}$ as an ADS. The ADL $\mathcal{L}$ corresponding to $\mathcal{ALCN}^{\sqcap}$ has already been discussed. Thus, we concentrate on the class of ADMs $\mathcal{M}$ induced by the frames of $\mathcal{ALCN}^{\sqcap}$. Assume that $\mathcal{F}$ is such a frame, i.e., $\mathcal{F}$ consists of a nonempty domain and interpretations $R^{\mathcal{F}}$ of the role names R. The ADM $\mathfrak{W} = \langle W, F^{\mathfrak{W}}, R^{\mathfrak{W}} \rangle$ induced by $\mathcal{F}$ is defined as follows. The set W is identical to the domain of $\mathcal{F}$. Each role description yields a relation symbol, which is interpreted in $\mathfrak{W}$ just as in the frame. For example, $(R_1 \sqcap R_2)^{\mathfrak{W}} = R_1^{\mathcal{F}} \cap R_2^{\mathcal{F}}$. It remains to define the interpretation of the function symbols. We illustrate this on two examples. First, consider the (unary) function symbol $f_{\forall(R_1 \sqcap R_2)}$. Given a subset X of W, the function $f^{\mathfrak{W}}_{\forall(R_1 \sqcap R_2)}$ maps X to

$$f^{\mathfrak{W}}_{\forall(R_1 \sqcap R_2)}(X) := \{w \in W \mid v \in X \text{ for all } v \text{ with } (w, v) \in R_1^{\mathcal{F}} \cap R_2^{\mathcal{F}}\},$$

i.e., the interpretation of the concept description $\forall(R_1 \sqcap R_2).A$ in the interpretations based on $\mathcal{F}$ interpreting A by X. Accordingly, the value of the constant symbol $f_{(\geq 2R)}$ in $\mathfrak{W}$ is given by the interpretation of $(\geq 2R)$ in the interpretations based on $\mathcal{F}$.

It is easy to show that the interpretation of concept descriptions in L coincides with the interpretation of the corresponding terms in $S = (\mathcal{L}, \mathcal{M})$.

Lemma 13. *Let $\mathcal{F}$ be a frame, $\mathfrak{W} = \langle W, F^{\mathfrak{W}} \rangle$ be the ADM corresponding to $\mathcal{F}$, $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ be an assignment for $\mathfrak{W}$, C be a concept description, and let the concept names used in C be among $A_1, \ldots, A_k$. For all interpretations $\mathcal{I}$ based on $\mathcal{F}$ with $A_i^{\mathcal{I}} = \mathcal{A}_1(x_{A_i})$ for all $1 \leq i \leq k$, we have that $t_C^{\mathfrak{W}, \mathcal{A}} = C^{\mathcal{I}}$.*

As an easy consequence of this lemma, there is a close connection between reasoning in a DL L and reasoning in the corresponding ADS. Given a TBox $\mathcal{T}$ and an ABox $\mathcal{A}$ of the DL L, we define the corresponding set $S(\mathcal{T}, \mathcal{A})$ of assertions of the corresponding ADL $(\mathcal{L}, \mathcal{M})$ in the obvious way, i.e., each GCI $C \sqsubseteq D$ in $\mathcal{T}$ yields a term assertion $t_C \sqsubseteq t_D$, each role assertion $R(i, j)$ in $\mathcal{A}$ yields an object assertion $R(a_i, a_j)$, and each concept assertion $C(i)$ yields an object assertion $a_i : t_C$.

Proposition 14. *The ABox $\mathcal{A}$ is consistent relative to the TBox $\mathcal{T}$ in L iff $S(\mathcal{T}, \mathcal{A})$ is satisfiable in the corresponding ADS.*

[4] Taking the empty set here is arbitrary.

We do not treat non-relativized consistency explicitly since it is the special case of relativized consistency where the TBox is empty.

As already mentioned above, our transfer results require the component ADSs to be local. We call a DL L *local* iff the ADS $(\mathcal{L}, \mathcal{M})$ corresponding to L is local. It turns out that not all DLs introduced in Section 2 are local.

Proposition 15. *Let L be one of the DLs introduced in Section 2. Then, L is local iff L does not include any of the following constructors: nominals, role complement, universal role.*

Proof. We start with the "only if" direction, which is more interesting since it shows *why* ADSs corresponding to DLs with nominals, role complement, or the universal role are not local. We make a case distinction according to which of these constructors L contains.

- **Nominals.** Consider the disjoint union $\mathfrak{W}$ of the ADMs $\mathfrak{W}_1$ and $\mathfrak{W}_2$, and assume that $\mathfrak{W}_1$ and $\mathfrak{W}_2$ correspond to frames of a DL with nominals. By definition of the disjoint union, we know that $\Delta^{\mathfrak{W}_1} \cap \Delta^{\mathfrak{W}_2} = \emptyset$. If $I \in N_O$ is a nominal, then the definition of the disjoint union implies that $f_I^{\mathfrak{W}} = f_I^{\mathfrak{W}_1} \cup f_I^{\mathfrak{W}_2}$. Since nominals are interpreted by singleton sets in $\mathfrak{W}_1$ and $\mathfrak{W}_2$, and since the domains of $\mathfrak{W}_1$ and $\mathfrak{W}_2$ are disjoint, this implies that $f_I^{\mathfrak{W}}$ is a set of cardinality 2. Consequently, $\mathfrak{W}$ cannot correspond to an ADM induced by a frame for a DL with nominals, since such frames interpret nominals by singleton sets.

- **Universal role.** Again, consider the disjoint union $\mathfrak{W}$ of the ADMs $\mathfrak{W}_1$ and $\mathfrak{W}_2$, and assume that $\mathfrak{W}_1$ and $\mathfrak{W}_2$ correspond to frames of a DL with the universal role. Let U denote the universal role, i.e., a role name for which the interpretation is restricted to the binary relation relating each pair of individuals of the domain. By the definition of the disjoint union, we have $U^{\mathfrak{W}} = U^{\mathfrak{W}_1} \cup U^{\mathfrak{W}_2} = \Delta^{\mathfrak{W}_1} \times \Delta^{\mathfrak{W}_1} \cup \Delta^{\mathfrak{W}_2} \times \Delta^{\mathfrak{W}_2} \neq \Delta^{\mathfrak{W}} \times \Delta^{\mathfrak{W}}$. Consequently, $\mathfrak{W}$ cannot correspond to an ADM induced by a frame for a DL with universal role, since such a frame would interpret U by $\Delta^{\mathfrak{W}} \times \Delta^{\mathfrak{W}}$.

- **Role complement.** Again, consider the disjoint union $\mathfrak{W}$ of the ADMs $\mathfrak{W}_1$ and $\mathfrak{W}_2$, and assume that $\mathfrak{W}_1$ and $\mathfrak{W}_2$ correspond to frames of a DL with role negation. For an arbitrary role name R, we have $\overline{R}^{\mathfrak{W}} = \overline{R}^{\mathfrak{W}_1} \cup \overline{R}^{\mathfrak{W}_2} = (\Delta^{\mathfrak{W}_1} \setminus R^{\mathfrak{W}_1}) \cup (\Delta^{\mathfrak{W}_2} \setminus R^{\mathfrak{W}_2}) \neq \Delta^{\mathfrak{W}} \setminus R^{\mathfrak{W}} = (\Delta^{\mathfrak{W}_1} \cup \Delta^{\mathfrak{W}_2}) \setminus (R^{\mathfrak{W}_1} \cup R^{\mathfrak{W}_2})$.

It remains to prove the "if" direction. Assume that L is one of the DLs introduced in Section 2 that does not allow for nominals, role complements, and the universal role. Let $(\mathcal{F}_p)_{p\in P}$ be a family of L-frames $\mathcal{F}_p = (\Delta^{\mathcal{F}_p}, \cdot^{\mathcal{F}_p})$ and let $\mathfrak{W}_p = \langle W_p, F^{\mathfrak{W}_p}, R^{\mathfrak{W}_p}\rangle$ be the ADMs corresponding to them. By definition, $\Delta^{\mathcal{F}_p} = W_p$ for all $p \in P$. Assume that the domains $(W_p)_{p\in P}$ are pairwise disjoint. We must show that the disjoint union of $(\mathfrak{W}_p)_{p\in P}$ also corresponds to an L-frame. To this purpose, we define the frame $\mathcal{F} = (\Delta^{\mathcal{F}}, \cdot^{\mathcal{F}})$ as follows:

- $\Delta^{\mathcal{F}} := \bigcup_{p\in P} \Delta^{\mathcal{F}_p}$ and
- $R^{\mathcal{F}} := \bigcup_{p\in P} R^{\mathcal{F}_p}$ for all $R \in N_R$.

Let $\mathfrak{W} = \langle W, F^{\mathfrak{W}}, R^{\mathfrak{W}}\rangle \in \mathcal{M}$ be the ADM corresponding to $\mathcal{F}$. By Definition 12 (corresponding ADM), we have $W = \bigcup_{p\in P} W_p$ and $R^{\mathfrak{W}} = \bigcup_{p\in P} R^{\mathfrak{W}_p}$ for all $R \in N_R$. By induction on the structure of complex roles, it is easy to show that this also holds for all $R \in \mathcal{R}$, i.e., all complex role descriptions. Since $R^{\mathfrak{W}_p} = R^{\mathcal{F}_p}$ for all $R \in \mathcal{R}$ and $p \in P$, we obtain the following fact:

Fact 16. *For all $p \in P$, $a \in \Delta^{\mathcal{F}_p}$, and role descriptions $R \in \mathcal{R}$, the following holds: $R^{\mathcal{F}}(a) = R^{\mathcal{F}_p}(a)$; in particular, $R^{\mathcal{F}}(a) \subseteq \Delta^{\mathcal{F}_p}$.*

It remains to show that, for all $n \geq 0$, all $X_1, \ldots, X_n \subseteq W$, and all function symbols f of arity n, we have

$$f^{\mathfrak{W}}(X_1, \ldots, X_n) = \bigcup_{p\in P} f^{\mathfrak{W}_p}(X_1 \cap W_p, \ldots, X_n \cap W_p).$$

This can be proved by making a case distinction according to the type of f. We treat two cases exemplarily.

- $f = f_{u_1\downarrow u_2}$. Obviously, $f^{\mathfrak{W}}_{u_1\downarrow u_2}$ is the disjoint union of the sets $f^{\mathfrak{W}}_{u_1\downarrow u_2} \cap W_p$ for $p \in P$. It remains to show that $f^{\mathfrak{W}}_{u_1\downarrow u_2} \cap W_p = f^{\mathfrak{W}_p}_{u_1\downarrow u_2}$ $(p \in P)$. By definition of $f^{\mathfrak{W}_p}_{u_1\downarrow u_2}$, we know that $a \in f^{\mathfrak{W}_p}_{u_1\downarrow u_2}$ iff $a \in \Delta^{\mathcal{F}_p}$, both $u_1^{\mathcal{F}_p}(a)$ and $u_2^{\mathcal{F}_p}(a)$ are defined, and $u_1^{\mathcal{F}_p}(a) = u_2^{\mathcal{F}_p}(a)$. By Fact 16, this is the case iff $a \in \Delta^{\mathcal{F}_p}$, both $u_1^{\mathcal{F}}(a)$ and $u_2^{\mathcal{F}}(a)$ are defined and $u_1^{\mathcal{F}}(a) = u_2^{\mathcal{F}}(a)$, which is equivalent to $a \in f^{\mathfrak{W}}_{u_1\downarrow u_2} \cap W_p$.

- $f = f_{\dot{\geq} nR}$. Obviously, $f^{\mathfrak{W}}_{\dot{\geq} nR}(X)$ is the disjoint union of the sets $f^{\mathfrak{W}}_{\dot{\geq} nR}(X) \cap W_p$ for $p \in P$. It remains to show that $f^{\mathfrak{W}}_{\dot{\geq} nR}(X) \cap W_p = f^{\mathfrak{W}_p}_{\dot{\geq} nR}(X \cap W_p)$ $(p \in P)$. By definition of $f^{\mathfrak{W}_p}_{\dot{\geq} nR}$, we know that $a \in f^{\mathfrak{W}_p}_{\dot{\geq} nR}(X \cap W_p)$ iff $a \in \Delta^{\mathcal{F}_p}$ and $|R^{\mathcal{F}_p}(a) \cap (X \cap W_p)| \geq n$. By Fact 16, this is the case iff $|R^{\mathcal{F}}(a) \cap (X \cap W_p)| \geq n$ iff $|R^{\mathcal{F}}(a) \cap X| \geq n$, and hence iff $a \in f^{\mathfrak{W}}_{\dot{\geq} nR}(X) \cap W_p$.

❑

It should be noted that arguments similar to the ones used in the proof of the "only if" direction show that, in the presence of the universal role or of role negation, function symbols (e.g., $f_{\forall U}$) may also violate the locality condition.

The transfer results for decidability that are developed in this paper only apply to fusions of local ADSs. Hence, the "if" direction of the proposition implies that our results are not applicable to fusions of ADSs corresponding to DLs that incorporate nominals, role complement, or the universal role.

4 Fusions of abstract description systems

In this section, we define the fusion of abstract description systems and prove two transfer theorems for decidability, one concerning satisfiability and the other one concerning relativized satisfiability.

Definition 17. The *fusion* $S_1 \otimes S_2 = (\mathcal{L}_1 \otimes \mathcal{L}_2, \mathcal{M}_1 \otimes \mathcal{M}_2)$ of two abstract description systems $S_1 = (\mathcal{L}_1, \mathcal{M}_1)$ and $S_2 = (\mathcal{L}_2, \mathcal{M}_2)$ over

- disjoint sets of function symbols $(f_i)_{i \in I}$ and $(g_j)_{j \in J}$,
- disjoint sets of relation symbols $(R_i)_{i \in \mathcal{R}}$ and $(Q_i)_{i \in \mathcal{Q}}$, and
- the same sets of set and object variables

is defined as follows: $\mathcal{L}_1 \otimes \mathcal{L}_2$ is the ADL based on the union $f_1, \ldots, g_1, \ldots$ of the function symbols of $\mathcal{L}_1$ and $\mathcal{L}_2$ and the union $R_1, \ldots, S_1, \ldots$ of the relation symbols of $\mathcal{L}_1$ and $\mathcal{L}_2$, and

$$\mathcal{M}_1 \otimes \mathcal{M}_2 := \{\langle W, (f_i^{\mathfrak{W}})_{i \in I} \cup (g_j^{\mathfrak{W}})_{j \in J}, (R_i)_{i \in \mathcal{R}} \cup (Q_i)_{i \in \mathcal{Q}} \rangle \mid \langle W, (f_i^{\mathfrak{W}})_{i \in I}, (R_i^{\mathfrak{W}})_{i \in \mathcal{R}} \rangle \in \mathcal{M}_1 \text{ and } \langle W, (g_j^{\mathfrak{W}})_{j \in J}, (Q_i^{\mathfrak{W}})_{i \in \mathcal{Q}} \rangle \in \mathcal{M}_2\}.$$

As an example, consider the ADSs S_1 and S_2 corresponding to the DLs $\mathcal{ALCF}$ and $\mathcal{ALC}^{+,\circ,\sqcup}$ introduced in Section 2. We concentrate on the function symbols provided by their fusion. The ADS S_1 is based on the following function symbols: (i) unary functions symbol $f_{\forall R}$ and $f_{\exists R}$ for every role name R, (ii) nullary functions symbols corresponding to the same-as constructor for every pair of chains of functional roles; and the ADS S_2 is based on the following function symbols: (iii) unary functions symbol $f_{\forall R}$ and $f_{\exists R}$ for every role description R built from role names using union, composition, and transitive closure. If we assume that the set of role names employed by $\mathcal{ALCF}$ and $\mathcal{ALC}^{+,\circ,\sqcup}$ are disjoint, then these sets of function symbols are disjoint. The union of these sets provides us both with the symbols for the same-as constructor and with the symbols for value and existential restrictions on role descriptions involving union, composition, and transitive closure. However, the role descriptions contain only role names from $\mathcal{ALC}^{+,\circ,\sqcup}$, and thus none of the functional roles of $\mathcal{ALCF}$ occurs in such descriptions. Thus, the fusion of $\mathcal{ALCF}$ and $\mathcal{ALC}^{+,\circ,\sqcup}$ yields a strict fragment of their union $\mathcal{ALCF}^{+,\circ,\sqcup}$.

4.1 Relativized satisfiability

Our first transfer result is concerned with the relativized satisfiability problem.

Theorem 18. *Let S_1 and S_2 be local ADSs, and suppose that the relativized satisfiability problems for S_1 and S_2 are decidable. Then the relativized satisfiability problem for $S_1 \otimes S_2$ is also decidable.*

For the proof of this theorem, we fix two local ADSs $S_i = (\mathcal{L}_i, \mathcal{M}_i)$, $i \in \{1, 2\}$, in which $\mathcal{L}_1$ is based on the set of function symbols $F = \{f_i \mid i \in I\}$ and relation symbols $R = \{R_i \mid i \in \mathcal{R}\}$, and $\mathcal{L}_2$ is based on $G = \{g_j \mid j \in J\}$ and $Q = \{Q_i \mid i \in \mathcal{Q}\}$. Let $\mathcal{L} = \mathcal{L}_1 \otimes \mathcal{L}_2$ and $\mathcal{M} = \mathcal{M}_1 \otimes \mathcal{M}_2$.

In what follows we use the following notation: for a set of assertions Γ, denote by $term(\Gamma)$ and $obj(\Gamma)$ the set of terms and object names in Γ, respectively.

For each $\mathcal{L}$-term t of the form $h(t_1, \dots, t_n)$, $h \in F \cup G$, we reserve a new variable x_t, which will be called the *surrogate* of t. We assume that the set of surrogate variables is disjoint to the original sets of variables. The main idea underlying the introduction of surrogate variables is that the decision procedure for S_1 (S_2) cannot deal with terms containing function symbols from G (F). Thus, these "alien" function symbols must be replaced before applying the procedure. To be more precise, we replace the whole alien subterm starting with the alien function symbol by its surrogate. For example, if the unary symbol f belongs to F, and the unary symbol g belongs to G, then $f(g(f(x)))$ is a "mixed" $\mathcal{L}$-term. To obtain a term of $\mathcal{L}_1$, we replace the subterm $g(f(x))$ by its surrogate, which yields $f(x_{g(f(x))})$. Analogously, to obtain a term of $\mathcal{L}_2$, we replace the whole term by its surrogate, which yields $x_{f(g(f(x)))}$. We now define this replacement process more formally.

Definition 19. For an $\mathcal{L}$-term t without surrogate variables, denote by $sur_1(t)$ the $\mathcal{L}_1$-term resulting from t when all occurrences of terms $g(t_1, \dots, t_n)$, $g \in G$, that are not within the scope of some $g' \in G$ are replaced by their surrogate variable $x_{g(t_1,\dots,t_n)}$. For a set Θ of terms, put $sur_1(\Theta) := \{sur_1(t) \mid t \in \Theta\}$ and define $sur_2(t)$ as well as $sur_2(\Theta)$ accordingly.

Denote by $sub(\Theta)$ the set of subterms of terms in Θ, and by $sub^1(\Theta)$ the set of variables in terms in Θ as well as all subterms of terms of the form $g(t_1, \dots t_n) \in sub(\Theta)$, $g \in G$. More formally, we can define

$$sub^1(\Theta) := sub\{t \mid x_t \in var(sur_1(\Theta))\} \cup var(\Theta).$$

Define $sub^2(\Theta)$ accordingly.

For example, let $f \in F$ be unary and $g \in G$ be binary. If $t = f(g(x, f(g(x, y))))$, then $sur_1(t) = f(x_{g(x,f(g(x,y)))})$. Note that the restriction "not within the scope of some $g' \in G$" is there to clarify that only the top-most alien subterms are to be replaced. For the term t of this example, we have $sub^1(\{t\}) = \{g(x, f(g(x, y))), f(g(x, y)), g(x, y), x, y\}$.

Note that the Boolean operators occurring in terms are "shared" function symbols in the sense that they are alien to neither $\mathcal{L}_1$ nor $\mathcal{L}_2$. Thus, $sur_1(f(x) \wedge g(x, y)) = f(x) \wedge x_{g(x,y)}$ and $sur_2(f(x) \wedge g(x, y)) = x_{f(x)} \wedge g(x, y)$.

Of course, when replacing whole terms by variables, some information is lost. For example, consider the (inconsistent) assertion $(\exists R_1.((\leq 1\, R_2) \sqcap (\geq 2\, R_2)))(i)$ and assume that R_1 is a role of one component of a fusion, and R_2 a role of the other component. Translated into abstract description language syntax, the concept description $\exists R_1.((\leq 1\, R_2) \sqcap (\geq 2\, R_2))$ yields the term $t := f_{\exists R_1}(f_{(\leq 1\, R_2)} \wedge f_{(\geq 2\, R_2)})$, where $f_{\exists R_1}$ is a function symbol of $\mathcal{L}_1$ and the other two function symbols belong to $\mathcal{L}_2$. Now, $sur_1(t) = f_{\exists R_1}(x \wedge y)$, where x is the surrogate for $f_{(\leq 1\, R_2)}$ and y

is the surrogate for $f_{(\geq 2\, R_2)}$. If the decision procedure for the first ADS only sees this term it has no way to know that the conjunction $x \wedge y$ is unsatisfiable. In fact, for this procedure x and y are arbitrary variables. Thus, we need some way to check satisfiability of certain Boolean combinations of surrogated subterms.

Definition 20. Given a finite set Θ of $\mathcal{L}$-terms, we define the *consistency set* $\mathcal{C}(\Theta)$ of Θ as $\mathcal{C}(\Theta) := \{t_c \mid c \subseteq \Theta\}$, where

$$t_c := \bigwedge\{\chi \mid \chi \in c\} \wedge \bigwedge\{\neg\chi \mid \chi \in \Theta \setminus c\}.$$

Given a finite set Γ of assertions in $\mathcal{L}$, we define $sub^i(\Gamma) := sub^i(term(\Gamma))$. We abbreviate $\mathcal{C}^i(\Gamma) := \mathcal{C}(sub^i(\Gamma))$, for $i \in \{1, 2\}$.

In the example above, we have

$$sub^1(f_{\exists R_1}(f_{(\leq 1\, R_2)} \wedge f_{(\geq 2\, R_2)})) = \{f_{(\leq 1\, R_2)}, f_{(\geq 2\, R_2)}\},$$

and thus $\mathcal{C}^1(\{a_i : f_{\exists R_1}(f_{(\leq 1\, R_2)} \wedge f_{(\geq 2\, R_2)})\})$ consists of the 4 terms

$$\begin{array}{rcl} f_{(\leq 1\, R_2)} & \wedge & f_{(\geq 2\, R_2)}, \\ f_{(\leq 1\, R_2)} & \wedge & \neg f_{(\geq 2\, R_2)}, \\ \neg f_{(\leq 1\, R_2)} & \wedge & f_{(\geq 2\, R_2)}, \text{ and} \\ \neg f_{(\leq 1\, R_2)} & \wedge & \neg f_{(\geq 2\, R_2)}. \end{array}$$

Given a set of terms Θ, an element t_c of its consistency set can be considered as the "type" of an element of the domain of an ADM w.r.t. Θ. Any such element belongs to the interpretations of some of the terms in Θ, and to the complements of the interpretations of the other terms. Thus, if c is the set of terms of Θ to which the element belongs, then it also belongs to the interpretation of t_c and it does not belong to the interpretation of any of the other terms in $\mathcal{C}(\Theta)$.

We are now ready to reduce the relativized satisfiability problem in a fusion of ADSs to relativized satisfiability in the component ADSs. From this Theorem 18 follows immediately.

Theorem 21. *Let Γ be a finite set of assertions from $\mathcal{L}$. Then the following are equivalent:*

1. *Γ is satisfiable in $(\mathcal{L}, \mathcal{M})$.*
2. *There exist*
   - *a set $D \subseteq \mathcal{C}^1(\Gamma)$,*
   - *for every term $t \in D$ an object variable $a_t \notin obj(\Gamma)$,*
   - *for every $a \in obj(\Gamma)$ a term $t_a \in D$,*

   such that the union Γ_1 of the following sets of assertions in $\mathcal{L}_1$ is satisfiable in $(\mathcal{L}_1, \mathcal{M}_1)$:

   - $\{aR_ib \mid aR_ib \in \Gamma, i \in \mathcal{R}\}$,
   - $\{a : sur_1(t_a) \mid a \in obj(\Gamma)\} \cup \{a : sur_1(s) \mid (a : s) \in \Gamma\}$,
   - $\{a_t : sur_1(t) \mid t \in D\}$,
   - $\{\top \sqsubseteq sur_1(\bigvee D)\}$,
   - $\{sur_1(t_1) \sqsubseteq sur_1(t_2) \mid t_1 \sqsubseteq t_2 \in \Gamma\}$*;*

   and the union Γ_2 of of the following sets of assertions in $\mathcal{L}_2$ is satisfiable in $(\mathcal{L}_2, \mathcal{M}_2)$:

   - $\{aQ_ib \mid aQ_ib \in \Gamma, i \in \mathcal{Q}\}$,
   - $\{a : sur_2(t_a) \mid a \in obj(\Gamma)\}$,

- $\{a_t : sur_2(t) \mid t \in D\}$,
- $\{\top \sqsubseteq sur_2(\bigvee D)\}$.

A proof of this theorem can be found in the appendix. Intuitively, the second condition "guesses" a set D of types (i.e., elements of the consistency set). The third set of assertions in Γ_1 and Γ_2 ensures that the guessed types are satisfiable in the respective ADSs. The fourth set of assertions states that the union of these types covers the whole domain, which means that the types not contained in D are unsatisfiable. This chosen set of types realizes the coordination between the separate satisfiability checks in the two ADSs. Basically, it makes sure that both algorithms base their decision on the same assumptions about interactions between the surrogated terms and their subterms.

The second condition is asymmetric in two respects. First, it guesses a subset of $\mathcal{C}^1(\Gamma)$ rather than a subset of $\mathcal{C}^2(\Gamma)$. Of course this is arbitrary, we could also have chosen index 2 instead of 1 here. Second, the set Γ_2 neither contains the assertions $\{sur_2(t_1) \sqsubseteq sur_2(t_2) \mid t_1 \sqsubseteq t_2 \in \Gamma\}$ nor $\{a : sur_2(s) \mid (a : s) \in \Gamma\}$. If we added these assertions, the theorem would still be true, but this would unnecessarily increase the amount of work to be done by the combined decision procedure.

By definition of the sets Γ_i and of the functions sur_i, the assertions in Γ_i are indeed assertions of $\mathcal{L}_i$, and thus the satisfiability algorithm for $(\mathcal{L}_i, \mathcal{M}_i)$ can be applied to Γ_i. This proves Theorem 18. Regarding the complexity of the obtained decision procedure, the costly step is guessing the right set D. Since the cardinality of the set $sub^1(\Gamma)$ is linear in the size of Γ, the cardinality of $\mathcal{C}^1(\Gamma)$ is exponential in the size of Γ (and each element of it has size quadratic in Γ). Thus, there are doubly exponentially many different subsets to be chosen from. Since the cardinality of the chosen set D may be exponential in the size of Γ, also the size of Γ_1 and Γ_2 may be exponential in Γ (because of the big disjunction over D). From this, the following corollary easily follows.

Corollary 22. *Let S_1 and S_2 be local ADSs, and suppose that the relativized satisfiability problems for S_1 and S_2 are decidable in* ExpTime *(*PSpace*). Then the relativized satisfiability problem for $S_1 \otimes S_2$ is decidable in* 2ExpTime *(*ExpSpace*).*

The relativized term satisfiability problem

The statement of Theorem 18 itself does not imply a transfer result for the relativized term satisfiability problem. The problem is that decidability of the relativized *term* satisfiability problem in S_1 and S_2 does not necessarily imply decidability of the relativized satisfiability problem in these ADSs, and thus the prerequisite for the theorem to apply is not satisfied. However, if we consider the statement of Theorem 21, then it is easy to see that this theorem also yields a transfer result for the relativized term satisfiability problem.

Corollary 23. *Let S_1 and S_2 be local ADSs, and suppose that the relativized term satisfiability problems for S_1 and S_2 are decidable. Then the relativized term satisfiability problem for $S_1 \otimes S_2$ is also decidable.*

Proof. Consider the satisfiability criterion in Theorem 21. If we are interested in relativized term satisfiability, then Γ is of the form $\{a : t\} \cup \Gamma'$, where Γ' is a set of term assertions. In this case, the sets of assertions Γ_1 and Γ_2 do not contain object assertions involving relations. Now, assume that Γ_i is of the form $\{a_1 : t_1, \ldots, a_n : t_n\} \cup \Gamma'_i$, where Γ'_i is a set of term assertions. Since two assertions of the form $b : s_1, b : s_2$ are equivalent to one assertion $b : s_1 \wedge s_2$, we may assume that the a_i are distinct from each other. Since S_i is local, it is easy to see that the following are equivalent:

1. $\{a_1 : t_1, \ldots, a_n : t_n\} \cup \Gamma'_i$ is satisfiable in S_i.
2. $\{a_j : t_j\} \cup \Gamma'_i$ is satisfiable in S_i for all $j = 1, \ldots, n$.

The second condition can now be checked by applying the term satisfiability test in S_i n times.
❑

Dealing with the universal role

As stated above (Proposition 15), ADSs corresponding to DLs with the universal role are not local, and thus Theorem 18 cannot be applied directly. Neverthless, in some cases this theorem can also be used to obtain a decidability result for fusions of DLs with the universal role, provided that both of them provide for a universal role. (We will comment on the usefulness of this approach in more detail in Section 5.4).

Definition 24. Given an ADS $S = (\mathcal{L}, \mathcal{M})$, we denote by S^U the ADS obtained from S by

1. extending $\mathcal{L}$ with two function symbols $f_{\exists U_S}$ and $f_{\forall U_S}$, and
2. extending every ADM $\mathfrak{W} = \langle W, F^{\mathfrak{W}}, R^{\mathfrak{W}} \rangle \in \mathcal{M}$ with the unary functions $f^{\mathfrak{W}}_{\exists U_S}$ and $f^{\mathfrak{W}}_{\forall U_S}(X)$, where
   - $f^{\mathfrak{W}}_{\exists U_S}(X) = \emptyset$ if $X = \emptyset$, and $f^{\mathfrak{W}}_{\exists U_S}(X) = W$ otherwise;
   - $f^{\mathfrak{W}}_{\forall U_S}(X) = W$ if $X = W$, and $f^{\mathfrak{W}}_{\forall U_S}(X) = \emptyset$ otherwise.

For ADSs S corresponding to a DL L, the ADS S^U corresponds to the extension of L with the universal role, where the universal role can only be used within value and existential restrictions.[5] Using simple extensions of results from [24], we obtain the following corollary to our first transfer theorem.

Corollary 25. *Let S_1, S_2 be local ADSs and assume that, for $i \in \{1, 2\}$, the relativized (term) satisfiability problem for S_i (or, equivalently, the non-relativized (term) satisfiability problem for S_i^U) is decidable. Then the relativized (term) satisfiability problem for $S_1^U \otimes S_2^U$ is decidable.*

Proof. By extending the results in [24] to the case of ADSs, one can show that, for every abstract description system S, the following conditions are equivalent:

- the relativized (term) satisfiability problem for S is decidable,
- the non-relativized (term) satisfiability problem for S^U is decidable,
- the relativized (term) satisfiability problem for S^U is decidable.

This explains the term "equivalently" used in the formulation of the theorem. Let S_1 and S_2 be as in the theorem. We know by Theorem 18 (Corollary 23) that the relativized (term) satisfiability problem for $S_1 \otimes S_2$ is decidable. Hence, the relativized (term) satisfiability problem for $(S_1 \otimes S_2)^U$ is decidable. But $S_1^U \otimes S_2^U$ is just a notational variant of $(S_1 \otimes S_2)^U$: the function symbols $f_{\exists U_{S_1}}$ and $f_{\exists U_{S_2}}$ can be replaced by $f_{\exists U_{S_1 \otimes S_2}}$ (and analogously for $f_{\forall U_{S_1 \otimes S_2}}$) since all three have identical semantics. ❑

4.2 Satisfiability

Note that Theorem 18 does not yield a transfer result for the unrelativized satisfiability problem. Of course, if the relativized satisfiability problems for S_1 and S_2 are decidable, then the theorem implies that the satisfiability problem for $S_1 \otimes S_2$ is also decidable (since it is a special case of the relativized satisfiability problem). However, to be able to apply the theorem to obtain decidability of the satisfiability problem in the fusion, the component ADSs must satisfy the

[5] Note that it is not necessary to add the universal role U to the set of relation symbols since an assertion of the form $U(a, b)$ is trivially true. However, the use of the universal role within (qualified) number restrictions is not covered by this extension.

stronger requirement that the relativized satisfiability problem is decidable. Indeed, the set Γ_i in Theorem 21 contains a term assertion (namely $\top \sqsubseteq sur_i(\bigvee D)$) even if Γ does not contain any term assertions.

There are cases where the relativized satisfiability problem is undecidable whereas the satisfiability problem is still decidable. For example, Theorem 18 cannot be applied for the fusion of $\mathcal{ALCF}$ and $\mathcal{ALC}^{+,\circ,\sqcup}$ since the relativized satisfiability problem for $\mathcal{ALCF}$ is already undecidable [5]. However, the satisfiability problem is decidable for both DLs. Before we can formulate a transfer result for the satisfiability problem, we need to introduce an additional notion, which generalizes the notion of a normal modal logic.

Definition 26 (Covering normal terms). Let $(\mathcal{L}, \mathcal{M})$ be an ADS and f be a function symbol of $\mathcal{L}$ of arity n. The term $t_f(x)$ (with one variable x) is a *covering normal term* for f iff the following holds for all $\mathfrak{W} \in \mathcal{M}$:

- $t_f^{\mathfrak{W}}(W) = W$
- for all $X, Y \subseteq W$, $t_f^{\mathfrak{W}}(X \cap Y) = t_f^{\mathfrak{W}}(X) \cap t_f^{\mathfrak{W}}(Y)$, and
- for all $X, X_1, \ldots, Y_n \subseteq W$: $X \cap X_i = X \cap Y_i$ for $1 \leq i \leq n$ implies

$$t_f^{\mathfrak{W}}(X) \cap f^{\mathfrak{W}}(X_1, \ldots, X_n) = t_f^{\mathfrak{W}}(X) \cap f^{\mathfrak{W}}(Y_1, \ldots, Y_n).$$

An ADS $(\mathcal{L}, \mathcal{M})$ is said to *have covering normal terms* iff one can effectively determine a covering normal term t_f for every function symbol f of $\mathcal{L}$.

For example, consider the term $f_{\forall R}(x)$, where $f_{\forall R}$ is the function symbol corresponding to the value restriction constructor for the role R. Then $f_{\forall R}(x)$ obviously satisfies the first two requirements for covering normal terms. In fact, it is easy to see that $f_{\forall R}(x)$ is a covering normal term for the function symbols corresponding to the value, existential, and (qualified) number restrictions on the role R.

Given covering normal terms t_f for the function symbols f of a finite set of funtion symbols E, one can construct a term t_E that is a covering normal term for all the elements of E.

Lemma 27. *Suppose the ADS $(\mathcal{L}, \mathcal{M})$ has covering normal terms and $\mathcal{L}$ is based on a set of function symbols F. Denote by t_f the covering normal term for the function symbol f, for all $f \in F$. Then, for every finite set $E \subseteq F$ of function symbols, the term*

$$t_E(x) := \bigwedge_{f \in E} t_f(x)$$

is a covering normal term for all $f \in E$.

Using covering normal terms, we can now formulate the second transfer theorem, which is concerned with the transfer of decidability of (non-relativized) satisfiability.

Theorem 28. *Let S_1 and S_2 be local ADSs having covering normal terms, and suppose that the satisfiability problems for S_1 and S_2 are decidable. Then the satisfiability problem for $S_1 \otimes S_2$ is also decidable.*

As in the proof of Theorem 18, we fix two local ADSs $S_i = (\mathcal{L}_i, \mathcal{M}_i)$, $i \in \{1, 2\}$, in which $\mathcal{L}_1$ is based on the set of function symbols $F = \{f_i \mid i \in I\}$ and relation symbols $R = \{R_i \mid i \in \mathcal{R}\}$, and $\mathcal{L}_2$ is based on $G = \{g_j \mid j \in J\}$ and $Q = \{Q_i \mid i \in \mathcal{Q}\}$. Let $\mathcal{L} = \mathcal{L}_1 \otimes \mathcal{L}_2$ and $\mathcal{M} = \mathcal{M}_1 \otimes \mathcal{M}_2$.

In the case of relativized satisfiability, term assertions of the form $\top \sqsubseteq sur_i(\bigvee D)$ were used to assert that all elements of the domain belong to $sur_i(\bigvee D)$. Now, we use covering normal terms to "propagate" $sur_i(\bigvee D)$ into terms up to a certain depth. For a set of function symbols E, define the E-depth $d_E(t)$ of a term t inductively:

$$\begin{array}{rcl} d_E(x_i) & = & 0 \\ d_E(\neg t) & = & d_E(t) \\ d_E(t_1 \wedge t_2) & = & max\{d_E(t_1), d_E(t_2)\} \\ d_E(f(t_1,\ldots,t_n)) & = & max\{d_E(t_1),\ldots,d_E(t_n)\} + 1 \ \text{ if } \ f \in E \\ d_E(f(t_1,\ldots,t_n)) & = & max\{d_E(t_1),\ldots,d_E(t_n)\} \ \text{ if } \ f \notin E \end{array}$$

If Γ is a finite set of assertions, then

$$d_E(\Gamma) := max\{d_E(t) \mid t \in term(\Gamma)\}.$$

Put, for a term $t(x)$ with one variable x, $t^0(x) := x$, $t^{m+1}(x) := t(t^m(x))$, $t^{\leq 0}(x) := x$, and $t^{\leq m+1}(x) := t^{m+1}(x) \wedge t^{\leq m}(x)$.

Let Γ be a set of object assertions in $\mathcal{L}$. For $i \in \{1,2\}$, denote by $\Sigma_i(\Gamma)$ the set of all $s \in \mathcal{C}^i(\Gamma)$ such that the term s is satisfiable in $(\mathcal{L},\mathcal{M})$.[6] Since in the following the set Γ will be fixed, we will mostly write Σ_i instead of $\Sigma_i(\Gamma)$. We are now in the position to formulate the result that provides us with a decision procedure for satisfiability in the fusion $(\mathcal{L},\mathcal{M})$.

Theorem 29. *Let Γ be a set of object assertions from $\mathcal{L}$. Put $m := d_F(\Gamma)$, $r := d_G(\Gamma)$, and let $c(x)$ ($d(x)$) be a covering normal term for all function symbols in Γ that are in F (G). Then the following three conditions are equivalent:*

1. *Γ is satisfiable in $(\mathcal{L},\mathcal{M})$.*
2. *There exist*
   - *for every $a \in obj(\Gamma)$ a term $t_a \in \Sigma_1$*
   - *for every $t \in \Sigma_1$ an object variable $a_t \notin obj(\Gamma)$*

   such that the union Γ_1 of the following sets of object assertions is satisfiable in $(\mathcal{L}_1,\mathcal{M}_1)$:
   - $\{aR_ib \mid aR_ib \in \Gamma, i \in \mathcal{R}\}$,
   - $\{a : sur_1(t_a \wedge c^{\leq m}(sur_1(\bigvee \Sigma_1)) \mid a \in obj(\Gamma)\} \cup \{a : sur_1(s) \mid (a : s) \in \Gamma\}$,
   - $\{a_t : sur_1(t \wedge c^{\leq m}(sur_1(\bigvee \Sigma_1)) \mid t \in \Sigma_1\}$;

   and the union Γ_2 of the following sets of object assertions is satisfiable in $(\mathcal{L}_2,\mathcal{M}_2)$:
   - $\{aQ_ib \mid aQ_ib \in \Gamma, i \in \mathcal{Q}\}$,
   - $\{a : sur_2(t_a \wedge d^{\leq r}(sur_2(\bigvee \Sigma_1)) \mid a \in obj(\Gamma)\}$,
   - $\{a_t : sur_2(t \wedge d^{\leq r}(sur_2(\bigvee \Sigma_1)) \mid t \in \Sigma_1\}$.
3. *The same condition as in (2) above, with Σ_1 replaced by Σ_2.*

The sets Γ_i in the above theorem are very similar to the ones in Theorem 21. The main difference is that the term assertion $\top \sqsubseteq sur_i(\bigvee D)$ is no longer there. Instead, the disjunction $sur_i(\bigvee \Sigma_1)$ is directly "inserted" into the terms using the covering normals terms. Another difference is that the set D, which had to be guessed in Theorem 21, is replaced by the set Σ_1 in (2) and Σ_2 in (3). Actually, guessing the set D is no longer possible in this case. In the proof of Theorem 29 we need to know that $\top \sqsubseteq sur_i(\bigvee D)$ is at least satisfiable in S_i. But we have no way to check this effectively since we don't have an algorithm for relativized satisfiability in S_i. Taking the set Σ_i ensures that this property is satisfied (see the proof in the appendix for details).

By definition, $\Sigma_i = \Sigma_i(\Gamma)$ is the set of all $s \in \mathcal{C}^i(\Gamma)$ such that the term s is satisfiable in $(\mathcal{L},\mathcal{M})$. Computing this set actually needs a recursive call to the decision procedure for satisfiability in $(\mathcal{L},\mathcal{M})$. This recursion is well-founded since the alternation depth decreases.

[6] Recall that this is the case iff $\{a : s\}$ is satisfiable in $(\mathcal{L},\mathcal{M})$ for an arbitrary object variable a.

Definition 30. For a term s of $\mathcal{L}$, denote by $a_1(s)$ and $a_2(s)$ the 1-alternation and the 2-alternation depth of s, respectively. That is to say, $a_1(s)$ is the length of the longest sequence of the form $(g_{i_1}, f_{i_2}, g_{i_3}, \ldots)$ such that

$$g_{i_1}(\ldots(f_{i_2}\ldots(g_{i_3}\ldots)))$$

with $g_{i_j} \in G$ and $f_{i_j} \in F$ appears in s. The 2-alternation depth $a_2(s)$ is defined by exchanging the roles of F and G. Put $a(s) := a_1(s) + a_2(s)$, and call this the *alternation depth*. For a finite set Θ of terms, $a(\Theta)$ is the maximum of all $a(s)$ with $s \in \Theta$.

Thus, $a_1(s)$ counts the maximal number of changes between symbols from the first and the second ADS, starting with the first symbol from S_2 (i.e., symbols from S_1 above this S_2-symbol are not counted here). The 2-alternation depth is defined accordingly. The alternation depth sums up the 1- and the 2-alternation depth.

Lemma 31. *If $a(term(\Gamma)) > 0$, then $a(\mathcal{C}^1(\Gamma)) < a(term(\Gamma))$ or $a(\mathcal{C}^2(\Gamma)) < a(term(\Gamma))$.*

If $a((term(\Gamma)) = 0$, then Γ consists of Boolean combinations of set variables. In this case, $\mathcal{C}^i(\Gamma)$ consists of set variables, and $\Sigma_i, i = 1, 2$, can be computed using Boolean reasoning. If $a(term(\Gamma)) > 0$, then we can assume by induction that either Σ_1 or Σ_2 can effectively be computed. Consequently, either Condition (2) or Condition (3) can effectively be checked using the decision procedure for S_1 or S_2. This proves Theorem 28.

Regarding the complexity, we must in principle also consider the complexity of computing covering normal terms and the size of these terms. In the examples from DL, these terms are just value restrictions, and thus their size and the complexity of computing them is linear. Here, we assume a polynomial bound on both. Under this assumption, we obtain the same complexity results as for the case of relativized satisfiability. In fact, the complexity of testing Condition (2) and (3) of Theorem 29 agrees with the complexity of testing Condition (2) of Theorem 21: it adds one exponential to the complexity of the decision procedure for the single ADSs. In order to compute Σ_i, we need exponentially many recursive calls to the procedure. Since the recursion depth is linear in the size of Γ, we end up with at most exponentially many tests of Condition (2) and (3).

Corollary 32. *Let S_1 and S_2 be local ADSs having covering normal terms, and assume that these covering normal terms can be computed in polynomial time. If the satisfiability problems for S_1 and S_2 are decidable in* ExpTime *(*PSpace*), then the satisfiability problem for $S_1 \otimes S_2$ is decidable in* 2ExpTime *(*ExpSpace*).*

With the same argument as in the case of relativized satisfiability, we can extend the transfer result also to term satisfiability.

Corollary 33. *Let S_1 and S_2 be local ADSs having covering normal terms, and suppose that the term satisfiability problems for S_1 and S_2 are decidable. Then the term satisfiability problem for $S_1 \otimes S_2$ is also decidable.*

5 Fusions of description logics

Given two DLs L_1 and L_2, their fusion is defined as follows. We translate them into the corresponding ADSs S_1 and S_2, and then build the fusion $S_1 \otimes S_2$. The fusion $L_1 \otimes L_2$ of L_1 and L_2 is the DL that corresponds to $S_1 \otimes S_2$. Since the definition of the fusion of ADSs requires their sets of function symbols to be disjoint, we must ensure that the ADSs corresponding to L_1 and L_2 are built over disjoint sets of function symbols. For the DLs introduced in Section 2, this can be achieved by assuming that the sets of role names of L_1 and L_2 are disjoint and the sets of nominals of L_1 and L_2 are disjoint. The DL $L_1 \otimes L_2$ then allows the use of the concept and role constructors of both DLs, but in a restricted way. Role descriptions are either role descriptions of L_1 or of L_2. There are no role descriptions involving constructors or names of both DLs. Concept descriptions may contain concept constructors of both DLs; however, a constructor of L_i may only use a role description of L_i ($i = 1, 2$).

Let us illustrate these restrictions by two simple examples. The fusion $\mathcal{ALC}^{+} \otimes \mathcal{ALC}^{-}$ of the two DLs $\mathcal{ALC}^{+}$ and $\mathcal{ALC}^{-}$ is the fragment of $\mathcal{ALC}^{+,-}$ whose set of role names is partitioned into two sets N_{R_1} and N_{R_2} such that

- the transitive closure operator may only be applied to names from N_{R_1};
- the inverse operator may only be applied to names from N_{R_2}.

Note that, although the two source DLs have disjoint sets of role names, in $\mathcal{ALC}^{+} \otimes \mathcal{ALC}^{-}$ role names from both sets may be used inside existential and value restrictions since these concept constructors are available in both DLs.

The fusion $\mathcal{ALCQ} \otimes \mathcal{ALC}_f$ of the two DLs $\mathcal{ALCQ}$ and $\mathcal{ALC}_f$ is the fragment of $\mathcal{ALCQ}_f$ whose set of role names N_R (with feature names $N_F \subseteq N_R$) is partitioned into two sets N_{R_1} and N_{R_2} with $N_F \subseteq N_{R_2}$ such that, inside qualifying number restrictions, only role names from N_{R_1} may be used. In particular, this means that feature names cannot occur within qualified number restrictions.

In the following, we give examples that illustrate the usefulness of the transfer results proved in the previous section. First, we will give an example for the case of satisfiability and then for relativized satisfiability. Subsequently, we will consider a more complex example involving so-called concrete domains. Here, our general transfer result can be used to prove a decidability result that has only recently been proved by designing a specialized algorithm for the fusion. Finally, we will give an example that demonstrates that the restriction to local ADSs is really necessary.

5.1 Decidability transfer for satisfiability

In this subsection, we will give an example for an application of Theorem 28 where the decidability result could not be obtained using Theorem 18.

Theorem 28 requires the ADSs to have covering normal terms. This is, however, satisfied by all the DLs that yield local ADSs.

Proposition 34. *Let L be one of the DLs introduced in Section 2, and let the corresponding ADS $S = (\mathcal{L}, \mathcal{M})$ be local. Then S has covering normal terms, and these terms can be computed in linear time.*

Proof. For all function symbols f in $\mathcal{L}$, the term t_f has the form $f_{\forall R}(x)$ for some role description R. The semantics of value restrictions implies that terms of this form satisfy the first two properties of Definition 26. This completes the proof for all function symbols f of arity 0 since for these the third condition of Definition 26 is trivially satisfied. Thus, for nullary function symbols, $f_{\forall R}(x)$ for an arbitrary role name R does the job.

It remains to show that, for every unary function symbol f of the form $f_{\exists R}$, $f_{\forall R}$, $f_{\dot{\geq} nR}$, and $f_{\dot{\leq} nR}$, the term $f_{\forall R}(x)$ also satisfies the third property. Again, this is an easy consequence of the semantics of these concept constructors. ❑

In the following, we consider the two description logics $\mathcal{ALCF}$ and $\mathcal{ALC}^{+,\circ,\sqcup}$. In [30], it is shown that satisfiability of $\mathcal{ALCF}$-concept descriptions is decidable. The same is true for consistency of $\mathcal{ALCF}$-ABoxes [36]. Note, however, that relativized satisfiability of $\mathcal{ALCF}$-concept descriptions and thus also relativized ABox consistency in $\mathcal{ALCF}$ is undecidable [5]. For $\mathcal{ALC}^{+,\circ,\sqcup}$, decidability of satisfiability is shown in [3, 44].[7] Decidability of ABox consistency in $\mathcal{ALC}^{+,\circ,\sqcup}$ follows from the results in [16].

[7] Note that $\mathcal{ALC}^{+,\circ,\sqcup}$ is a notational variant of test-free propositional dynamic logic (PDL) [22].

The unrestricted combination $\mathcal{ALCF}^{+,\circ,\sqcup}$ of the two DLs is undecidable. To be more precise, the results in [5] imply that satisfiability of $\mathcal{ALCF}^{+,\circ,\sqcup}$-concept descriptions (and thus also consistency of $\mathcal{ALCF}^{+,\circ,\sqcup}$-ABoxes) is undecidable. In contrast, Theorem 28 immediately implies that satisfiability of concept descriptions in the fusion of $\mathcal{ALCF}$ and $\mathcal{ALC}^{+,\circ,\sqcup}$ is decidable.

Theorem 35. *Satisfiability of concept descriptions and consistency of ABoxes is decidable in $\mathcal{ALCF} \otimes \mathcal{ALC}^{+,\circ,\sqcup}$, whereas satisfiability of $\mathcal{ALCF}^{+,\circ,\sqcup}$-concept descriptions is already undecidable.*

Taking the fusion thus yields a decidable combination of two DLs whose unrestricted combination is undecidable. The price one has to pay is that the fusion offers less expressivity than the unrestricted combination. The concept $f_1{\downarrow}f_2 \sqcap \forall f_1^+.C$ is an example of a concept description of $\mathcal{ALCF}^{+,\circ,\sqcup}$ that is not allowed in the fusion $\mathcal{ALCF} \otimes \mathcal{ALC}^{+,\circ,\sqcup}$.

5.2 Decidability transfer for relativized satisfiability

As an example for the application of Corollary 23 (and thus of Theorem 18), we consider the DL $\mathcal{ALC}_f^{+,\circ,\sqcap,\sqcup}$. For this DL, satisfiability of concept descriptions is undecidable. However, an expressive fragment with a decidable relativized satisfiability problem can be obtained by building the fusion of the two sublanguages $\mathcal{ALC}_f^{+,\circ,\sqcup}$ and $\mathcal{ALC}^{+,\circ,\sqcup,\sqcap}$.

Theorem 36. *Satisfiability of $\mathcal{ALC}_f^{+,\circ,\sqcap,\sqcup}$-concept descriptions is undecidable.*

Undecidability can easily be shown by a reduction of the domino problem [10, 33] (see, e.g., [8] for undecidability proofs of DLs using such a reduction). The main tasks to solve in such a reduction is that one can express the $\mathbb{N} \times \mathbb{N}$ grid and that one can access all points on the grid. One square of the grid can be expressed by a description of the form $\exists(x \circ y \sqcap y \circ x).\top$, where x, y are features. In fact, this description expresses that the "points" belonging to it have both an $x \circ y$ and a $y \circ x$ successor, and that these two successors coincide. Accessing all point on the grid can then be achieved by using the role description $(x \sqcup y)^+$.

Note that this undecidability result is also closely related to the known undecidability of IDPDL, i.e., deterministic propositional dynamic logic with intersection [28]. However, the undecidability proof for IDPDL in [28] uses the test construct, which is not available in $\mathcal{ALC}_f^{+,\circ,\sqcap,\sqcup}$.

Next, we show that relativized satisfiability in two expressive sublanguages of $\mathcal{ALC}_f^{+,\circ,\sqcap,\sqcup}$ is decidable.

Theorem 37. *Relativized satisfiability of concept descriptions is decidable in $\mathcal{ALC}_f^{+,\circ,\sqcup}$ and $\mathcal{ALC}^{+,\circ,\sqcup,\sqcap}$.*

Proof sketch. In both cases, TBoxes can be internalized as described in [44, 5]. Thus, it is sufficient to show decidability of (unrelativized) satisfiability.

For $\mathcal{ALC}_f^{+,\circ,\sqcup}$, this follows from decidability of DPDL [9], the known correspondence between PDL and $\mathcal{ALC}^{+,\circ,\sqcup}$ [44], and the fact that non-functional roles can be simulated by functional ones in the presence of composition and transitive closure [41].

For $\mathcal{ALC}^{+,\circ,\sqcup,\sqcap}$, decidability of satisfiability follows from decidability of IPDL, i.e., PDL with intersection [15]. ❑

Given this theorem, Corollary 23 now yields the following decidability result.

Corollary 38. *Relativized satisfiability of concept descriptions is decidable in $\mathcal{ALC}_f^{+,\circ,\sqcup} \otimes \mathcal{ALC}^{+,\circ,\sqcup,\sqcap}$.*

5.3 A "concrete" example

Description logics with concrete domains were introduced in [6] in order to allow for the reference to concrete objects like numbers, time intervals, spatial regions, etc. when defining concepts. To be more precise, Baader and Hanschke [6] define the extension $\mathcal{ALC}(\mathcal{D})$ of $\mathcal{ALC}$, where $\mathcal{D}$ is a concrete domain (see below). Under suitable assumptions on $\mathcal{D}$ they show that satisfiability in $\mathcal{ALC}(\mathcal{D})$ is decidable. One of the main problems with this extension of DLs is that relativized satisfiability (and satisfiability in DLs where TBoxes can be internalized) is usually undecidable [2] (though there are exceptions [37]). For this reason, Haarslev et al. [27] introduce a restricted way of extending DLs by concrete domains, and show that the corresponding extension of $\mathcal{ALCN}_{\mathcal{HR}^+}$ has a decidable relativized satisfiability problem.[8] In the following, we show that this result can also be obtained as an easy consequence of our Theorem 18. Moreover, $\mathcal{ALCN}_{\mathcal{HR}^+}$ can be replaced by an arbitrary local DL with a decidable relativized satisfiability problem.

Definition 39 (Concrete Domain). A *concrete domain* $\mathcal{D}$ is a pair $(\Delta_\mathcal{D}, \Phi_\mathcal{D})$, where $\Delta_\mathcal{D}$ is a nonempty set called the domain, and $\Phi_\mathcal{D}$ is a set of predicate names. Each predicate name $P \in \Phi_\mathcal{D}$ is associated with an arity n and an n-ary predicate $P^\mathcal{D} \subseteq \Delta_\mathcal{D}^n$. A concrete domain $\mathcal{D}$ is called *admissible* iff (1) the set of its predicate names is closed under negation and contains a name $\top_\mathcal{D}$ for $\Delta_\mathcal{D}$, and (2) the satisfiability problem for finite conjunctions of predicates is decidable.

Given a concrete domain $\mathcal{D}$ and one of the predicates $P \in \Phi_\mathcal{D}$ (of arity n), one can define a new concept constructor $\exists f_1, \ldots, f_n.P$ (predicate restriction), where $f_1, \ldots, f_n$ are concrete features.[9] In contrast to the abstract features considered until now, concrete features are interpreted by partial functions from the abstract domain $\Delta^\mathcal{I}$ into the concrete domain $\Delta_\mathcal{D}$. We consider the basic DL that allows for Boolean operators and these new concept constructors only.

Definition 40 ($\mathcal{B}(\mathcal{D})$). Let N_C be a set of concept names and N_{F_c} be a set of names for concrete features disjoint from N_C, and let $\mathcal{D}$ be an admissible concrete domain. Concepts descriptions of $\mathcal{B}(\mathcal{D})$ are Boolean combinations of concept names and *predicate restrictions*, i.e., expressions of the form $\exists f_1, \ldots, f_n.P$ where P is an n-ary predicate in $\Phi_\mathcal{D}$ and $f_1, \ldots, f_n \in N_{F_c}$.

The semantics of $\mathcal{B}(\mathcal{D})$ is defined as follows. We consider an interpretation $\mathcal{I}$, which has a nonempty domain $\Delta^\mathcal{I}$, and interprets concept names as subsets of $\Delta^\mathcal{I}$ and concrete features as partial functions from $\Delta^\mathcal{I}$ into $\Delta_\mathcal{D}$. The Boolean operators are interpreted as usual, and

$$\begin{aligned}(\exists u_1, \ldots, u_n.P)^\mathcal{I} = \{a \in \Delta^\mathcal{I} \mid \exists x_1, \ldots, x_n \in \Delta_\mathcal{D}.\\ f_i^\mathcal{I}(a) = x_i \text{ for all } 1 \leq i \leq n \text{ and } (x_1, \ldots, x_n) \in P^\mathcal{D}\}\end{aligned}$$

Note that concept descriptions are interpreted as subsets of $\Delta^\mathcal{I}$ and not of $\Delta^\mathcal{I} \cup \Delta_\mathcal{D}$. Thus, if we go to the ADS corresponding to $\mathcal{B}(\mathcal{D})$, the concrete domain is not an explicit part of the corresponding ADMs. It is hidden in the interpretation of the function symbols corresponding to predicate restrictions. The predicate restriction constructor is translated into a function symbol $f_{\exists u_1, \ldots, u_n.P}$ of arity 0, and, for an ADM $\mathfrak{W}$ corresponding to a frame $\mathcal{F}$, $f^\mathfrak{W}_{\exists u_1, \ldots, u_n.P}$ is defined as $(\exists u_1, \ldots, u_n.P)^{\mathcal{I}_\emptyset}$, where $\mathcal{I}_\emptyset$ is the interpretation based on $\mathcal{F}$ that maps all concept names to the empty set.

Theorem 41. *Let $\mathcal{D}$ be an admissible concrete domain. Then, $\mathcal{B}(\mathcal{D})$ is local and the relativized satisfiability problem for $\mathcal{B}(\mathcal{D})$-concept descriptions is decidable.*

[8] To be more precise, they even show that relativized ABox consistency is decidable in their restricted extension of $\mathcal{ALCN}_{\mathcal{HR}^+}$ by concrete domains. Here, we restrict ourself to satisfiability of concepts since the ABoxes introduced in [27] also allow for the use of concrete individuals and for predicate assertions on these individuals, which is not covered by the object assertions for ADSs introduced in the present paper.

[9] Note that the general framework introduced in [6] allows for feature chains in predicate restrictions. Considering only feature chains of length one is the main restriction introduced in [27].

Proof. Given the family $(\mathfrak{W}_p)_{p\in P}$ of ADMs $\mathfrak{W}_p$ corresponding to the frames $\mathcal{F}_p$ over pairwise disjoint domains $\Delta^{\mathcal{F}_p}$ $(p \in P)$, we first build the union $\mathcal{F}$ of the frames: the domain of $\mathcal{F}$ is $\bigcup_{p\in P} \Delta^{\mathcal{F}_p}$ and it interprets the concrete features in the obvious way, i.e., $f^{\mathcal{F}}(x) := f^{\mathcal{F}_p}(x)$ if $x \in \Delta^{\mathcal{F}_p}$. Let $\mathfrak{W}$ be the ADM induced by $\mathcal{F}$. It is easy to show that $\mathfrak{W}$ is in fact the disjoint union of $(\mathfrak{W}_p)_{p\in P}$.

Decidability of the unrelativized satisfiability problem is an immediate consequence of the decidability results for $\mathcal{ALC}(\mathcal{D})$ given in [6]. Since $\mathcal{B}(\mathcal{D})$ is a very simple DL that does not contain any concept constructors requiring the generation of abstract individuals, it is easy to see that a $\mathcal{B}(\mathcal{D})$-concept description C_0 is satisfiable relative to the TBox $C_1 \sqsubseteq D_1, \ldots, C_n \sqsubseteq D_n$ iff it is satisfiable in a one-element interpretation. But then the TBox can be internalized in a very simple way: C_0 is satisfiable relative to the TBox $C_1 \sqsubseteq D_1, \ldots, C_n \sqsubseteq D_n$ iff $C_0 \sqcap (\neg C_1 \sqcup D_1) \sqcap \ldots \sqcap (\neg C_n \sqcup D_n)$ is satisfiable. ❑

Given this theorem, Corollary 23 now yields the following transfer result, which shows that concrete domains with the restricted form of predicate restrictions introduced above can be integrated into any local DL with a decidable relativized satisfiability problem without losing decidability.

Corollary 42. *Let $\mathcal{D}$ be an admissible concrete domain and L be a local DL for which relativized satisfiability of concept descriptions is decidable. Then, relativized satisfiability of concept descriptions in $\mathcal{B}(\mathcal{D}) \otimes L$ is also decidable.*

5.4 Non-local DLs

By Proposition 15, DLs allowing for nominals, the universal role, or role negation are not local. It follows that the decidability transfer theorems are not applicable to fusions of such DLs. In the following, we try to clarify the reasons for this restricted applicability of the theorems.

First, we show that there are DLs with decidable satisfiability problem such that their fusion has an undecidable satisfiability problem. The culprit in this case is the universal role (or role negation).

Theorem 43. *Satisfiability of concept descriptions is decidable in $\mathcal{ALC}^U$ and $\mathcal{ALCF}$, but undecidable in their fusion $\mathcal{ALC}^U \otimes \mathcal{ALCF}$.*

Proof. Decidability of $\mathcal{ALCF}$ was shown in [30] and of $\mathcal{ALC}^U$ in [4, 24]. Undecidability of $\mathcal{ALC}^U \otimes \mathcal{ALCF}$ (which is identical to $\mathcal{ALCF}^U$) follows from the results in [5] and the well-known fact that the universal role can be used to simulate TBoxes. ❑

Since role negation can be used to simulate the universal role, and decidability of $\mathcal{ALC}^{\neg}$ can, for example, be obtained from the known decidability of L_2 (i.e., predicate logic with two variables) [38, 25], the theorem also holds if we replace $\mathcal{ALC}^U$ by $\mathcal{ALC}^{\neg}$.

It should be noted that the example given in the above theorem depends on the fact that one of the two DLs allows for the universal role and the other becomes undecidable if the universal role is added. In fact, Corollary 25 shows that decidability does transfer if both DLs already provide for the universal role.

Concerning nominals, we do not have a counterexample to the transfer of decidability in their presence. However, we think that it is very unlikely that there can be a general transfer result in this case. In fact, note that for each DL L without nominals introduced in Section 2, its fusion with $\mathcal{ALCO}$ is identical to L extended with nominals. Since (relativized) satisfiability in $\mathcal{ALCO}$ is decidable, a general transfer result in this case would imply that this extension is decidable provided that L is decidable. Consequently, this would yield a general transfer result for adding nominals.

6 Related and future work

The work that is most closely related to the one presented here is [49]. There, analoga of our Theorems 21 and 29 are proved for normal modal logics within an algebraic framework. The present results extend the ones from [49] in two directions. First, we have added object assertions, and thus can also prove transfer results for ABox reasoning. Second, we can show transfer results for satisfiability in non-normal modal logics as long as we have covering normal terms. This allows us to handle non-normal concept constructors like qualified number restrictions (graded modalities) in our framework.

We also think that the introduction of abstract description systems (ADSs) is a contribution in its own right. ADSs abstract from the internal structure of concept constructors and thus allow us to treat a vast range of such constructors in a uniform way. Nevertheless, the model theoretic semantics provided by ADSs is less abstract than the algebraic semantics employed in [49]. It is closer to the usual semantics of DLs, and thus easier to comprehend for people used to this semantics. The results in this paper show that ADSs in fact yield a good level of abstraction for proving general results on description logics. Interestingly, the same notion also turned out to be very useful in the context of combining tableau algorithms for modal and description logics [**?**].

Regarding complexity, our transfer results yield only upper bounds. Basically, they show that the complexity of the algorithm for the fusion is one exponent higher than of the ones for the components. We do not have matching lower bounds, i.e., we have no example where this increase in the complexity is unavoidable since the complexity of the satisfiability problem in the fusion is in fact one exponent higher. However, we believe that in this general setting one cannot get a better upper bound. Spaan's results [47] on the transfer of NP and PSPACE decidability are restricted to normal modal logics, and they make additional assumptions on the algorithms used to solve the satisfiability problem in the component logics.

Anyway, from a practical point of view it may be more interesting to find good optimizations of the combined decision procedures. However, it is likely that such optimizations will also depend on additional requirements on the component algorithms: optimizations may require these algorithm to yield more information than just "decidable" or "undecidable."

References

[1] Carlos Areces, Patrick Blackburn, and Maarten Marx. The computational complexity of hybrid temporal logics. *Logic Journal of the IGPL*, 8(5):653–679, 2000.

[2] F. Baader and P. Hanschke. Extensions of concept languages for a mechanical engineering application. In *Proceedings of the 16th German AI-Conference, GWAI-92*, volume 671 of *Lecture Notes in Computer Science*, pages 132–143, Bonn (Germany), 1993. Springer–Verlag.

[3] Franz Baader. Augmenting concept languages by transitive closure of roles: An alternative to terminological cycles. In *Proc. of the 12th Int. Joint Conf. on Artificial Intelligence (IJCAI'91)*, Sydney, Australia, 1991.

[4] Franz Baader, Hans-Jürgen Bürckert, Bernhard Hollunder, Werner Nutt, and Jörg H. Siekmann. Concept logics. In John W. Lloyd, editor, *Computational Logics, Symposium Proceedings*, pages 177–201. Springer-Verlag, 1990.

[5] Franz Baader, Hans-Jürgen Bürckert, Bernhard Nebel, Werner Nutt, and Gert Smolka. On the expressivity of feature logics with negation, functional uncertainty, and sort equations. *J. of Logic, Language and Information*, 2:1–18, 1993.

[6] Franz Baader and Philipp Hanschke. A schema for integrating concrete domains into concept languages. In *Proc. of the 12th Int. Joint Conf. on Artificial Intelligence (IJCAI'91)*, pages 452–457, Sydney, 1991.

[7] Franz Baader and Bernhard Hollunder. A terminological knowledge representation system with complete inference algorithm. In *Proc. of the Workshop on Processing Declarative Knowledge (PDK'91)*, volume 567 of *Lecture Notes In Artificial Intelligence*, pages 67–86. Springer-Verlag, 1991.

[8] Franz Baader and Ulrike Sattler. Expressive number restrictions in description logics. *J. of Logic and Computation*, 9(3):319–350, 1999.

[9] Mordechai Ben-Ari, Joseph Y. Halpern, and Amir Pnueli. Deterministic propositional dynamic logic: Finite models, complexity, and completeness. *J. of Computer and System Sciences*, 25:402–417, 1982.

[10] R. Berger. The undecidability of the dominoe problem. *Mem. Amer. Math. Soc.*, 66:1–72, 1966.

[11] Alexander Borgida. Description logics in data management. *IEEE Trans. on Knowledge and Data Engineering*, 7(5):671–682, 1995.

[12] Ronald J. Brachman, Deborah L. McGuinness, Peter F. Patel-Schneider, Lori Alperin Resnick, and Alexander Borgida. Living with CLASSIC: When and how to use a KL-ONE-like language. In John F. Sowa, editor, *Principles of Semantic Networks*, pages 401–456. Morgan Kaufmann, Los Altos, 1991.

[13] Ronald J. Brachman and James G. Schmolze. An overview of the KL-ONE knowledge representation system. *Cognitive Science*, 9(2):171–216, 1985.

[14] Brian F. Chellas. *Modal logic.* Cambridge University Press, Cambridge, UK, 1980.

[15] Ryszard Danecki. Nondeterministic Propositional Dynamic Logic with intersection is decidable. In *Proc. of the 5th Sym. on Computation Theory*, volume 208 of *Lecture Notes in Computer Science*, pages 34–53. Springer-Verlag, 1984.

[16] Giuseppe De Giacomo. *Decidability of Class-Based Knowledge Representation Formalisms.* PhD thesis, Dipartimento di Informatica e Sistemistica, Università di Roma "La Sapienza", 1995.

[17] Francesco M. Donini, Bernhard Hollunder, Maurizio Lenzerini, Alberto Marchetti Spaccamela, Daniele Nardi, and Werner Nutt. The complexity of existential quantification in concept languages. *Artificial Intelligence*, 2–3:309–327, 1992.

[18] Francesco M. Donini, Maurizio Lenzerini, Daniele Nardi, and Werner Nutt. The complexity of concept languages. In James Allen, Richard Fikes, and Erik Sandewall, editors, *Proc. of the 2nd Int. Conf. on the Principles of Knowledge Representation and Reasoning (KR'91)*, pages 151–162. Morgan Kaufmann, Los Altos, 1991.

[19] Francesco M. Donini, Maurizio Lenzerini, Daniele Nardi, and Werner Nutt. Tractable concept languages. In *Proc. of the 12th Int. Joint Conf. on Artificial Intelligence (IJCAI'91)*, pages 458–463, Sydney, 1991.

[20] Francesco M. Donini, Maurizio Lenzerini, Daniele Nardi, and Andrea Schaerf. Deduction in concept languages: From subsumption to instance checking. *J. of Logic and Computation*, 4(4):423–452, 1994.

[21] Kit Fine and Gerhard Schurz. Transfer theorems for stratified modal logics. In J. Copeland, editor, *Logic and Reality: Essays in Pure and Applied Logic. In Memory of Arthur Prior*, pages 169–213. Oxford University Press, 1996.

[22] Michael J. Fischer and Richard E. Ladner. Propositional dynamic logic of regular programs. *J. of Computer and System Sciences*, 18:194–211, 1979.

[23] George Gargov and Valentin Goranko. Modal logic with names. *J. of Philosophical Logic*, 22:607–636, 1993.

[24] Valentin Goranko and Solomon Passy. Using the universal modality: Gains and questions. *Journal of Logic and Computation*, 2(1):5–30, March 1992.

[25] E. Grädel, P. Kolaitis, and M. Vardi. On the Decision Problem for Two-Variable First-Order Logic. *Bulletin of Symbolic Logic*, 3:53–69, 1997.

[26] George Grätzer. *Universal Algebra.* Springer-Verlag, New York, 1979.

[27] Volker Haarslev, Ralf Mller, and Michael Wessel. The description logic $\dashv lcnh_{R^+}$ extended with concrete domains: A practically motivated approach. In *Proceedings of the International Joint Conference on Automated Reasoning IJCAR'01*, Lecture Notes In Artificial Intelligence. Springer-Verlag, 2001.

[28] David Harel. Dynamic logic. In *Handbook of Philosophical Logic*, volume 2, pages 497–640. D. Reidel, Dordrecht, Holland, 1984.

[29] Bernhard Hollunder and Franz Baader. Qualifying number restrictions in concept languages. Technical Report RR-91-03, Deutsches Forschungszentrum für Künstliche Intelligenz (DFKI), Kaiserslautern, Germany, 1991. An abridged version appeared in *Proc. of the 2nd Int. Conf. on the Principles of Knowledge Representation and Reasoning (KR'91).*

[30] Bernhard Hollunder and Werner Nutt. Subsumption algorithms for concept languages. Technical Report RR-90-04, Deutsches Forschungszentrum für Künstliche Intelligenz (DFKI), Kaiserslautern, Germany, 1990.

[31] Ian Horrocks. Using an expressive description logic: FaCT or fiction? In *Proc. of the 6th Int. Conf. on the Principles of Knowledge Representation and Reasoning (KR'98)*, pages 636–647, 1998.

[32] Ian Horrocks, Ulrike Sattler, and Stephan Tobies. Practical reasoning for expressive description logics. In Harald Ganzinger, David McAllester, and Andrei Voronkov, editors, *Proceedings of the 6th International Conference on Logic for Programming and Automated Reasoning (LPAR'99)*, number 1705 in Lecture Notes in Artificial Intelligence, pages 161–180. Springer-Verlag, September 1999.

[33] Donald E. Knuth. *The Art of Computer Programming*, volume 3. Addison-Wesley, Mass., 1973.

[34] Marcus Kracht and Frank Wolter. Properties of independently axiomatizable bimodal logics. *The Journal of Symbolic Logic*, 56(4):1469–1485, December 1991.

[35] Hector J. Levesque and Ron J. Brachman. Expressiveness and tractability in knowledge representation and reasoning. *Computational Intelligence*, 3:78–93, 1987.

[36] C. Lutz. Reasoning with concrete domains. In Thomas Dean, editor, *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence IJCAI-99*, pages 90–95, Stockholm, Sweden, July 31 – August 6, 1999. Morgan-Kaufmann Publishers.

[37] C. Lutz. Interval-based temporal reasoning with general tboxes. In *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence*, Seattle, 2001.

[38] Michael Mortimer. On languages with two variables. *Zeitschrift für Mathematische Logik und Grundlagen der Mathematik*, 21:135–140, 1975.

[39] Bernhard Nebel. Computational complexity of terminological reasoning in BACK. *Artificial Intelligence*, 34(3):371–383, 1988.

[40] Bernhard Nebel. Terminological reasoning is inherently intractable. *Artificial Intelligence*, 43:235–249, 1990.

[41] Rohit Parikh. Propositional logics of programs: Systems, models and complexity. In *Proc. of the 7th ACM SIGACT-SIGPLAN Sym. on Principles of Programming Languages (POPL'80)*, pages 186–192, Las Vegas (USA), 1980.

[42] A. N. Prior. *Past, Present and Future*. Oxford University Press, Oxford, UK, 1967.

[43] Andrea Schaerf. On the complexity of the instance checking problem in concept languages with existential quantification. *J. of Intelligent Information Systems*, 2:265–278, 1993.

[44] Klaus Schild. A correspondence theory for terminological logics: Preliminary report. In *Proc. of the 12th Int. Joint Conf. on Artificial Intelligence (IJCAI'91)*, pages 466–471, Sydney, Australia, 1991.

[45] Manfred Schmidt-Schauß. Subsumption in KL-ONE is undecidable. In Ron J. Brachman, Hector J. Levesque, and Ray Reiter, editors, *Proc. of the 1st Int. Conf. on the Principles of Knowledge Representation and Reasoning (KR'89)*, pages 421–431. Morgan Kaufmann, Los Altos, 1989.

[46] Manfred Schmidt-Schauß and Gert Smolka. Attributive concept descriptions with complements. *Artificial Intelligence*, 48(1):1–26, 1991.

[47] E. Spaan. *Complexity of Modal Logics*. PhD thesis, Department of Mathematics and Computer Science, University of Amsterdam, The Netherlands, 1993.

[48] Wiebe Van der Hoek and Maarten De Rijke. Counting objects. *J. of Logic and Computation*, 5(3):325–345, 1995.

[49] Frank Wolter. Fusions of modal logics revisited. In M. Kracht, M. de Rijke, H. Wansing, and M. Zakharyaschev, editors, *Advances in Modal Logic*, pages 361–379. CSLI Publications, 1998.

[50] William A. Woods and James G. Schmolze. The KL-ONE family. In F. W. Lehmann, editor, *Semantic Networks in Artificial Intelligence*, pages 133–178. Pergamon Press, 1992. Published as a special issue of *Computers & Mathematics with Applications*, Volume 23, Number 2–9.

A Appendix: Proofs

In this appendix, we give detailed proofs of criteria for (relativized) satisfiability in the fusion of local ADSs. Recall that, from these criteria, the transfer theorems for decidability easily follow. We have deferred the proofs of these theorems to the appendix since they are rather technical.

A.1 Proof of Theorem 21

Before we can prove this theorem, we need a technical lemma.

Lemma 44. *Let $(\mathcal{L}, \mathcal{M})$ be a local ADS and Γ a set of assertions satisfiable in $(\mathcal{L}, \mathcal{M})$. Then there exists a cardinal κ such that, for all cardinals $\kappa' \geq \kappa$, there exists a model $\mathfrak{W} = \langle W, F^{\mathfrak{W}}, R^{\mathfrak{W}} \rangle \in \mathcal{M}$ with $|W| = \kappa'$ and an assignment $\mathcal{A}$ with $\langle \mathfrak{W}, \mathcal{A} \rangle \models \Gamma$ and $|s^{\mathfrak{W},\mathcal{A}}| \in \{0, \kappa'\}$ for all terms s.*

Proof. By assumption, there exists an ADM $\mathfrak{V} = \langle V, F^{\mathfrak{V}}, R^{\mathfrak{V}} \rangle \in \mathcal{M}$ and an assignment $\mathcal{B} = \langle \mathcal{B}_1, \mathcal{B}_2 \rangle$ in it such that $\langle \mathfrak{V}, \mathcal{B} \rangle \models \Gamma$. Since $\mathcal{M}$ is closed under disjoint unions, we can assume without loss of generality that $\kappa = |V|$ is infinite. We show that κ is as required. Let $\kappa' \geq \kappa$. Take κ' disjoint isomorphic copies $\langle \mathfrak{W}_\rho, \mathcal{B}_1^\rho \rangle$, $\mathfrak{W}_\rho = \langle W_\rho, F^{\mathfrak{W}_\rho}, R^{\mathfrak{W}_\rho} \rangle$, $\rho < \kappa'$, of $\langle \mathfrak{V}, \mathcal{B}_1 \rangle$. Assume, for simplicity, that $W_0 = V$. Let $\mathfrak{W}$ be the disjoint union of the $\mathfrak{W}_\rho$, $\rho < \kappa'$, and define $\langle \mathfrak{W}, \mathcal{A} = \langle \mathcal{A}_1, \mathcal{A}_2 \rangle \rangle$ by putting $\mathcal{A}_2(a) = \mathcal{B}_2(a)$, for all $a \in \mathcal{X}$, and

$$\mathcal{A}_1(x) = \bigcup_{\rho < \kappa'} \mathcal{B}_1^\rho(x),$$

for all $x \in V$ (note that all object variables are interpreted in W_0.) Using the definitions of term semantics, disjoint unions, and basic cardinal arithmetic, it is straightforward to prove that $\langle \mathfrak{W}, \mathcal{A} \rangle$ is as required.[10] ❑

The following definition will also be needed in the proof of the theorem.

Definition 45. Let κ be a cardinal. A set $\{X_1, \dots, X_n\}$ is called a κ-partition of a set W iff

1. $|X_i| = \kappa$, for all $1 \leq i \leq n$,
2. $X_i \cap X_j = \emptyset$ whenever $i \neq j$, and
3. $W = \bigcup_{1 \leq i \leq n} X_i$.

Before proving Theorem 21, we repeat its formulation. We fix two local ADSs $S_i = (\mathcal{L}_i, \mathcal{M}_i)$, $i \in \{1, 2\}$, in which $\mathcal{L}_1$ is based on the set of function symbols $F = \{f_i \mid i \in I\}$ and relation symbols $R = \{R_i \mid i \in \mathcal{R}\}$, and $\mathcal{L}_2$ is based on $G = \{g_j \mid j \in J\}$ and $Q = \{Q_i \mid i \in \mathcal{Q}\}$. Let $\mathcal{L} = \mathcal{L}_1 \otimes \mathcal{L}_2$ and $\mathcal{M} = \mathcal{M}_1 \otimes \mathcal{M}_2$.

Theorem 21. *Let Γ be a finite set of assertions from $\mathcal{L}$. Then the following are equivalent:*

1. *Γ is satisfiable in $(\mathcal{L}, \mathcal{M})$.*
2. *There exist*
   - *a set $D \subseteq \mathcal{C}^1(\Gamma)$,*
   - *for every term $t \in D$ an object variable $a_t \notin obj(\Gamma)$,*
   - *for every $a \in obj(\Gamma)$ a term $t_a \in D$,*

   such that the union Γ_1 of the following sets of assertions in $\mathcal{L}_1$ is satisfiable in $(\mathcal{L}_1, \mathcal{M}_1)$:

[10] See, e.g., [26] on rules for calculating with cardinals.

- $\{aR_ib \mid aR_ib \in \Gamma, i \in \mathcal{R}\}$,
- $\{a : sur_1(t_a) \mid a \in obj(\Gamma)\} \cup \{a : sur_1(s) \mid (a : s) \in \Gamma\}$,
- $\{a_t : sur_1(t) \mid t \in D\}$,
- $\{\top \sqsubseteq sur_1(\bigvee D)\}$,
- $\{sur_1(t_1) \sqsubseteq sur_1(t_2) \mid t_1 \sqsubseteq t_2 \in \Gamma\}$;

and the union Γ_2 of of the following sets of assertions in $\mathcal{L}_2$ is satisfiable in $(\mathcal{L}_2, \mathcal{M}_2)$:

- $\{aQ_ib \mid aQ_ib \in \Gamma, i \in \mathcal{Q}\}$,
- $\{a : sur_2(t_a) \mid a \in obj(\Gamma)\}$,
- $\{a_t : sur_2(t) \mid t \in D\}$,
- $\{\top \sqsubseteq sur_2(\bigvee D)\}$.

Proof. We start with the direction from (2) to (1). Take a set $D \subseteq \mathcal{C}^1(\Gamma)$ satisfying the properties listed in the theorem. Take cardinals κ_i, $i \in \{1,2\}$ as in Lemma 44 for $(\mathcal{L}_i, \mathcal{M}_i)$, put $\kappa = max\{\kappa_1, \kappa_2\}$, and take $\langle \mathfrak{W}_1, \mathcal{A}^1 = \langle \mathcal{A}_1^1, \mathcal{A}_2^1 \rangle \rangle$ and $\langle \mathfrak{W}_2, \mathcal{A}^2 = \langle \mathcal{A}_1^2, \mathcal{A}_2^2 \rangle \rangle$ with $\mathfrak{W}_i \in \mathcal{M}_i$ such that $\langle \mathfrak{W}_i, \mathcal{A}^i \rangle \models \Gamma_i$ for $i \in \{1,2\}$. By Lemma 44, we can assume $|W_i| = \kappa$ and, for all $s \in D$, $|sur_i(s)^{\mathfrak{W}_i, \mathcal{A}^i}| \in \{0, \kappa\}$ for $i \in \{0,1\}$.

The sets $\{sur_i(s)^{\mathfrak{W}_i, \mathcal{A}^i} : s \in D\}$ are κ-partitions of W_i for $i \in \{0,1\}$ since (i) for each $s \in D$, we have $(a_s : sur_i(s)) \in \Gamma_i$, (ii) $\langle \mathfrak{W}_i, \mathcal{A}^i \rangle \models \top \sqsubseteq sur_i(\bigvee D)$, and (iii) $s, s' \in D$ and $s \neq s'$ implies $sur_i(s)^{\mathfrak{W}_i, \mathcal{A}^i} \cap sur_i(s')^{\mathfrak{W}_i, \mathcal{A}^i}$ by definition of D and $\mathcal{C}^1$. Moreover, $obj(\Gamma_1) = obj(\Gamma_2)$ and, for all $a \in obj(\Gamma_1)$ and $s \in D$, we have $\mathcal{A}_2^1(a) \in sur_1(s)^{\mathfrak{W}_1, \mathcal{A}^1}$ iff $\mathcal{A}_2^2(a) \in sur_2(s)^{\mathfrak{W}_2, \mathcal{A}^2}$.

Together with the fact that $\mathcal{A}_2^1$ and $\mathcal{A}_2^2$ are injective, this implies the existence of a bijection b from W_1 onto W_2 such that

$$\{b(w) : w \in sur_1(t)^{\mathfrak{W}_1, \mathcal{A}^1}\} = sur_2(t)^{\mathfrak{W}_2, \mathcal{A}^2},$$

for all $t \in D$, and

$$b(\mathcal{A}_2^1(a)) = \mathcal{A}_2^2(a),$$

for all $a \in obj(\Gamma_1)$. Define a model $\mathfrak{W} = \langle W, (F \cup G)^{\mathfrak{W}}, (R \cup Q)^{\mathfrak{W}} \rangle \in \mathcal{M}$ by putting

- $W = W_1$,
- $f^{\mathfrak{W}} = f^{\mathfrak{W}_1}$, for $f \in F$,
- for all $g \in G$ of arity n and all $Z_1, \ldots, Z_n \subseteq W$,

  $$g^{\mathfrak{W}}(Z_1, \ldots, Z_n) = b^{-1}(g^{\mathfrak{W}_2}(b[Z_1], \ldots, b[Z_n])),$$

  where $b[Z] = \{b(z) : z \in Z\}$,
- $R_i^{\mathfrak{W}} = R_i^{\mathfrak{W}_1}$,
- $Q_i^{\mathfrak{W}}(x,y)$ iff $Q_i^{\mathfrak{W}_2}(b(x), b(y))$.

Obviously $\mathfrak{W} \in \mathcal{M}_1 \otimes \mathcal{M}_2$. Let $\mathcal{A} = \mathcal{A}^1$. To prove the implication from (2) to (1) of the theorem it remains to show that $\langle \mathfrak{W}, \mathcal{A} \rangle \models \Gamma$. To this end it suffices to prove the following claim:

Claim. For all terms $t \in sub^1(\Gamma)$, we have

$$t^{\mathfrak{W}, \mathcal{A}} = sur_1(t)^{\mathfrak{W}_1, \mathcal{A}^1} = b^{-1}(sur_2(t)^{\mathfrak{W}_2, \mathcal{A}^2}).$$

Before we prove this claim, let us show that it implies $\langle \mathfrak{W}, \mathcal{A} \rangle \models \Gamma$. First note that, from the claim, we obtain

$$t^{\mathfrak{W}, \mathcal{A}} = sur_1(t)^{\mathfrak{W}_1, \mathcal{A}^1} \text{ for all } t \in term(\Gamma). \tag{1}$$

This may be proved by induction on the construction of $t \in \mathit{term}(\Gamma)$ from terms in $\mathit{sub}^1(\Gamma)$ using the booleans and function symbols from $\mathcal{L}_1$, only. The basis of induction (i.e., the equality for members of $\mathit{sub}^1(\Gamma)$) is stated in the claim and the induction step is straightforward.

Let us now show that $\langle \mathfrak{W}, \mathcal{A}\rangle \models \Gamma$ is a consequence of (1). Suppose $aR_ib \in \Gamma$. Then $aR_ib \in \Gamma_1$ and thus $\langle \mathfrak{W}, \mathcal{A}\rangle \models aR_ib$. Similarly, $aQ_ib \in \Gamma$ implies $aQ_ib \in \Gamma_2$ and $\langle \mathfrak{W}, \mathcal{A}\rangle \models aQ_ib$. Suppose $(a : t) \in \Gamma$. Then $(a : \mathit{sur}_1(t)) \in \Gamma_1$ and so $\mathcal{A}_2^1(a) \in \mathit{sur}_1(t)^{\mathfrak{W}_1, \mathcal{A}^1}$ which implies, by (1), $\mathcal{A}_2^1(a) \in t^{\mathfrak{W}, \mathcal{A}}$. Hence $\langle \mathfrak{W}, \mathcal{A}\rangle \models (a : t)$. If $t_1 \sqsubseteq t_2 \in \Gamma$, then $\mathit{sur}_1(t_1) \sqsubseteq \mathit{sur}_1(t_2) \in \Gamma_1$ and so, by (1), $t_1^{\mathfrak{W}, \mathcal{A}} \subseteq t_2^{\mathfrak{W}, \mathcal{A}}$. Hence $\langle \mathfrak{W}, \mathcal{A}\rangle \models t_1 \sqsubseteq t_2$.

We come to the proof of the claim. It is proved by induction on the structure of t. Due to the following equalities holding for all $t \in \mathit{sub}^1(\Gamma)$, it suffices to show that $t^{\mathfrak{W}, \mathcal{A}} = \mathit{sur}_1(t)^{\mathfrak{W}_1, \mathcal{A}^1}$

$$\begin{aligned}
\mathit{sur}_1(t)^{\mathfrak{W}_1, \mathcal{A}^1} &= \bigcup\{\mathit{sur}_1(s)^{\mathfrak{W}_1, \mathcal{A}^1} : s \in D, t \text{ is a conjunct of } s\} \\
&= \bigcup\{b^{-1}(\mathit{sur}_2(s)^{\mathfrak{W}_2, \mathcal{A}^2}) : s \in D, t \text{ is a conjunct of } s\} \\
&= b^{-1}(\mathit{sur}_2(t)^{\mathfrak{W}_2, \mathcal{A}^2})
\end{aligned}$$

The first equality holds since $\mathit{sur}_1(\bigvee D)^{\mathfrak{W}_1, \mathcal{A}^1} = W_1$ and, for all $s \in D$, either t or $\neg t$ is a conjunct of s. The second equality is true by definition of b and the validity of the third equality can be seen analogously to the validity of the first one by considering that $\mathit{sur}_2(\bigvee D)^{\mathfrak{W}_2, \mathcal{A}^2} = W_2$.

Hence let us show $t^{\mathfrak{W}, \mathcal{A}} = \mathit{sur}_1(t)^{\mathfrak{W}_1, \mathcal{A}^1}$. For the induction start, let t be a variable. The equation $t^{\mathfrak{W}, \mathcal{A}} = \mathit{sur}_1(t)^{\mathfrak{W}_1, \mathcal{A}^1}$ is an immediate consequence of the fact that $\mathcal{A} = \mathcal{A}^1$. For the induction step, we distinguish several cases:

- $t = \neg t_1$. By induction hypothesis, $t_1^{\mathfrak{W}, \mathcal{A}} = \mathit{sur}_1(t_1)^{\mathfrak{W}_1, \mathcal{A}_1}$. Hence, $t^{\mathfrak{W}, \mathcal{A}} = W \setminus t_1^{\mathfrak{W}, \mathcal{A}} = W \setminus \mathit{sur}_1(t_1)^{\mathfrak{W}_1, \mathcal{A}^1} = \mathit{sur}_1(t)^{\mathfrak{W}_1, \mathcal{A}^1}$ (since $W = W_1$).
- $t = t_1 \wedge t_2$. By induction hypothesis, $t_i^{\mathfrak{W}, \mathcal{A}} = \mathit{sur}_1(t_i)^{\mathfrak{W}_1, \mathcal{A}_1}$ for $i \in \{1, 2\}$. Hence, $t^{\mathfrak{W}, \mathcal{A}} = t_1^{\mathfrak{W}, \mathcal{A}} \cap t_2^{\mathfrak{W}, \mathcal{A}} = \mathit{sur}_1(t_1)^{\mathfrak{W}_1, \mathcal{A}^1} \cap \mathit{sur}_1(t_2)^{\mathfrak{W}_1, \mathcal{A}^1} = \mathit{sur}_1(t)^{\mathfrak{W}_1, \mathcal{A}^1}$.
- $t = t_1 \vee t_2$. Similar to the above case.
- $t = f(t_1, \ldots, t_n)$. By induction hypothesis, $t_i^{\mathfrak{W}, \mathcal{A}} = \mathit{sur}_1(t_i)^{\mathfrak{W}_1, \mathcal{A}^1}$ for $1 \leq i \leq n$. Hence, $t^{\mathfrak{W}, \mathcal{A}} = f^{\mathfrak{W}}(t_1^{\mathfrak{W}, \mathcal{A}}, \ldots, t_n^{\mathfrak{W}, \mathcal{A}}) = f^{\mathfrak{W}}(\mathit{sur}_1(t_1)^{\mathfrak{W}_1, \mathcal{A}^1}, \ldots, \mathit{sur}_1(t_n)^{\mathfrak{W}_1, \mathcal{A}^1}) = \mathit{sur}_1(t)^{\mathfrak{W}_1, \mathcal{A}^1}$ (since $f^{\mathfrak{W}} = f^{\mathfrak{W}_1}$).
- $t = g(t_1, \ldots, t_n)$. In this case, $t^{\mathfrak{W}, \mathcal{A}} = b^{-1}(g^{\mathfrak{W}_2}(b(t_1^{\mathfrak{W}, \mathcal{A}}), \ldots, b(t_n^{\mathfrak{W}, \mathcal{A}})))$. Since, by the above equalities, $\mathit{sur}_1(t)^{\mathfrak{W}_1, \mathcal{A}^1} = b^{-1}(\mathit{sur}_2(t)^{\mathfrak{W}_2, \mathcal{A}^2})$, it remains to show that $\mathit{sur}_2(t)^{\mathfrak{W}_2, \mathcal{A}^2} = g^{\mathfrak{W}_2}(b(t_1^{\mathfrak{W}, \mathcal{A}}), \ldots, b(t_n^{\mathfrak{W}, \mathcal{A}}))$. Since $\mathit{sur}_2(t)^{\mathfrak{W}_2, \mathcal{A}^2} = g^{\mathfrak{W}_2}(\mathit{sur}_2(t_1)^{\mathfrak{W}_2, \mathcal{A}^2}, \ldots, \mathit{sur}_2(t_n)^{\mathfrak{W}_2, \mathcal{A}^2})$, this amounts to showing that $b(t_i^{\mathfrak{W}, \mathcal{A}}) = \mathit{sur}_2(t_i)^{\mathfrak{W}_2, \mathcal{A}^2}$ for $1 \leq i \leq n$. This, however, follows by induction hypothesis together with the above equations.

This concludes the proof of the direction from (2) to (1).

It remains to prove the direction from (1) to (2). Suppose $\langle \mathfrak{W}, \mathcal{A}\rangle \models \Gamma$, for some $\mathfrak{W} \in \mathcal{M}$ and $\mathcal{A} = \langle \mathcal{A}_1, \mathcal{A}_2\rangle$. Put

$$D = \{s \in \mathcal{C}^1(\Gamma) : s^{\mathfrak{W}, \mathcal{A}} \neq \emptyset\}.$$

We may assume that the sets $s^{\mathfrak{W}, \mathcal{A}}$ are infinite since the fusion of two local ADSs is again local. Take a new object name $a_s \notin \mathit{obj}(\Gamma)$ for every $s \in D$ and let, for $a \in \mathit{obj}(\Gamma)$,

$$t_a = \bigwedge\{t \in \mathit{sub}^1(\Gamma) : \mathcal{A}_2(a) \in t^{\mathfrak{W}, \mathcal{A}}\} \wedge \bigwedge\{\neg t : t \in \mathit{sub}^1(\Gamma), \mathcal{A}_2(a) \notin t^{\mathfrak{W}, \mathcal{A}}\}.$$

We prove that set of assertions Γ_1 based on D, t_a, $a \in \mathit{obj}(\Gamma)$, and a_s, $s \in D$, is satisfiable in $(\mathcal{L}_1, \mathcal{M}_1)$.

Let $F^{\mathfrak{W}}$ denote the restriction of $(F \cup G)^{\mathfrak{W}}$ to the symbols in F. Similarly, $R^{\mathfrak{W}}$ is the restriction of $(R \cup Q)^{\mathfrak{W}}$ to the symbols in R. Set $\mathfrak{W}_1 = \langle W, F^{\mathfrak{W}}, R^{\mathfrak{W}} \rangle \in \mathcal{M}_1$, $\mathcal{A}^1 = \langle \mathcal{A}_1^1, \mathcal{A}_2^1 \rangle$, where

$$\mathcal{A}_1^1 = \mathcal{A}_1 \cup \{x_t \mapsto t^{\mathfrak{W},\mathcal{A}} : t = g(t_1, \ldots, t_k) \in sub^1(\Gamma)\},$$

$\mathcal{A}_2^1(a) = \mathcal{A}_2(a)$, for $a \in obj(\Gamma)$, and $\mathcal{A}_2^1(a_s) \in s^{\mathfrak{W},\mathcal{A}}$, for all $s \in D$.

It is straightforward to show by induction that

$$sur_1(t)^{\mathfrak{W}_1,\mathcal{A}_1} = t^{\mathfrak{W},\mathcal{A}} \text{ for all } t \in term(\Gamma).$$

From this we immediately obtain $\langle \mathfrak{W}_1, \mathcal{A}^1 \rangle \models \Gamma_1$.

The construction of a model in $\mathcal{M}_2$ satisfying Γ_2 is similar and left to the reader.

❑

A.2 Proof of Theorem 29

As in the proof of Theorem 18, we fix two local ADSs $S_i = (\mathcal{L}_i, \mathcal{M}_i)$, $i \in \{1, 2\}$, in which $\mathcal{L}_1$ is based on the set of function symbols $F = \{f_i \mid i \in I\}$ and relation symbols $R = \{R_i \mid i \in \mathcal{R}\}$, and $\mathcal{L}_2$ is based on $G = \{g_j \mid j \in J\}$ and $Q = \{Q_i \mid i \in \mathcal{Q}\}$. Let $\mathcal{L} = \mathcal{L}_1 \otimes \mathcal{L}_2$ and $\mathcal{M} = \mathcal{M}_1 \otimes \mathcal{M}_2$. We assume that S_1 and S_2 have covering normal terms.

We need a lemma which describes the essential properties of $\Sigma_1 = \Sigma_1(\Gamma)$ (the description for $\Sigma_2(\Gamma)$ is dual and omitted).

Lemma 46. *Take a cardinal κ satisfying the conditions of Lemma 44 for $(\mathcal{L}, \mathcal{M})$ and $\Gamma = \bigvee \Sigma_1 = \top$. Let $\kappa' \geq \kappa$.*

(1) There exists $\mathfrak{W} \in \mathcal{M}_1$ and an assignment $\mathcal{A}$ such that

$$\{sur_1(s)^{\mathfrak{W},\mathcal{A}} \mid s \in \Sigma_1\}$$

is a κ'-partition of $\mathfrak{W}$.

(2) There exists $\mathfrak{W} \in \mathcal{M}_2$ and an assignment $\mathcal{A}$ such that

$$\{sur_2(s)^{\mathfrak{W},\mathcal{A}} \mid s \in \Sigma_1\}$$

is a κ'-partition of $\mathfrak{W}$.

Proof. (1) By the definition of Σ_1 and Lemma 44 we find $\mathfrak{W}' = \left\langle W', (F \cup G)^{\mathfrak{W}'}, (R \cup Q)^{\mathfrak{W}'} \right\rangle \in \mathcal{M}$ and an assignment $\mathcal{A}'$ such that

$$\{s^{\mathfrak{W}',\mathcal{A}'} \mid s \in \Sigma_1\}$$

is a κ'-partition of W'. Now let $\mathfrak{W}$ denote the restriction of $\mathfrak{W}'$ to $\mathcal{L}_1$ and define

$$\mathcal{A}_1 = \mathcal{A}_1' \cup \{x_t \mapsto t^{\mathfrak{W}',\mathcal{A}'} \mid t = g(t_1, \ldots, t_k) \in sub^1(\Gamma)\}.$$

Then $\langle \mathfrak{W}, \mathcal{A} \rangle$ is as required. (2) is similar and left to the reader. ❑

We repeat the formulation of the theorem to be proved.

Theorem 29. *Let Γ be a set of object assertions from $\mathcal{L}$. Put $m := d_F(\Gamma)$, $r := d_G(\Gamma)$, and let $c(x)$ ($d(x)$) be a covering normal term for all function symbols in Γ that are in F (G). Then the following three conditions are equivalent:*

1. *Γ is satisfiable in $(\mathcal{L}, \mathcal{M})$.*
2. *There exist*

- *for every $a \in obj(\Gamma)$ a term $t_a \in \Sigma_1$*
- *for every $t \in \Sigma_1$ an object variable $a_t \notin obj(\Gamma)$*

such that the union Γ_1 of the following sets of object assertions is satisfiable in $(\mathcal{L}_1, \mathcal{M}_1)$:

- $\{aR_ib \mid aR_ib \in \Gamma, i \in \mathcal{R}\}$,
- $\{a : sur_1(t_a \wedge c^{\leq m}(sur_1(\bigvee \Sigma_1)) \mid a \in obj(\Gamma)\} \cup \{a : sur_1(s) \mid (a : s) \in \Gamma\}$,
- $\{a_t : sur_1(t \wedge c^{\leq m}(sur_1(\bigvee \Sigma_1)) \mid t \in \Sigma_1\}$;

and the union Γ_2 of the following sets of object assertions is satisfiable in $(\mathcal{L}_2, \mathcal{M}_2)$:

- $\{aQ_ib \mid aQ_ib \in \Gamma, i \in \mathcal{Q}\}$,
- $\{a : sur_2(t_a \wedge d^{\leq r}(sur_2(\bigvee \Sigma_1)) \mid a \in obj(\Gamma)\}$,
- $\{a_t : sur_2(t \wedge d^{\leq r}(sur_2(\bigvee \Sigma_1)) \mid t \in \Sigma_1\}$.

3. *The same condition as in (2) above, with Σ_1 replaced by Σ_2.*

We start the proof with the direction from (1) to (2) and (1) to (3). The proofs are dual to each other, so we only give a proof for (1) $\Rightarrow$ (2). Suppose $\langle \mathfrak{W}, \mathcal{A} \rangle \models \Gamma$, where $\mathfrak{W} = \langle W, (F \cup G)^{\mathfrak{W}}, (R \cup Q)^{\mathfrak{W}} \rangle$. We can assume (because of closure under disjoint unions) that, for every $t \in \Sigma_1$, $|t^{\mathfrak{W},\mathcal{A}}|$ is infinite.

Take a new object name $a_s \notin obj(\Gamma)$ for every $s \in \Sigma_1$ and let, for $a \in obj(\Gamma)$,

$$t_a = \bigwedge\{t \in sub^1(\Gamma) : \mathcal{A}_2(a) \in t^{\mathfrak{W},\mathcal{A}}\} \wedge \bigwedge\{\neg t : t \in sub^1(\Gamma), \mathcal{A}_2(a) \notin t^{\mathfrak{W},\mathcal{A}}\}.$$

We prove that the set of assertions Γ_1 based on t_a, $a \in obj(\Gamma)$, and a_s, $s \in \Sigma_1$, is satisfiable in $(\mathcal{L}_1, \mathcal{M}_1)$.

Let $F^{\mathfrak{W}}$ (resp. $G^{\mathfrak{W}}$) denote the restriction of $(F \cup G)^{\mathfrak{W}}$ to the symbols in F (resp. G). Similarly, $R^{\mathfrak{W}}$ and $Q^{\mathfrak{W}}$ are the restrictions of $(R \cup Q)^{\mathfrak{W}}$ to the symbols in R and Q, respectively. Set $\mathfrak{W}_1 = \langle W, F^{\mathfrak{W}}, R^{\mathfrak{W}} \rangle \in \mathcal{M}_1$, $\mathcal{A}^1 = \langle \mathcal{A}_1^1, \mathcal{A}_2^1 \rangle$, where

$$\mathcal{A}_1^1 = \mathcal{A}_1 \cup \{x_t \mapsto t^{\mathfrak{W},\mathcal{A}} \mid t = g(t_1, \ldots, t_k) \in sub^1(\Gamma)\},$$

$\mathcal{A}_2^1(a) = \mathcal{A}_2(a)$, for $a \in obj(\Gamma)$, and $\mathcal{A}_2^1(a_t) \in t^{\mathfrak{W},\mathcal{A}}$, for all $t \in \Sigma_1$.

It is straightforward to show by induction that

$$sur_1(t)^{\mathfrak{W}_1,\mathcal{A}_1} = t^{\mathfrak{W},\mathcal{A}} \text{ for all } t \in term(\Gamma).$$

From this and the observation that $\langle \mathfrak{W}, \mathcal{A} \rangle \models \bigvee \Sigma_1 = \top$ we immediately obtain $\langle \mathfrak{W}_1, \mathcal{A}^1 \rangle \models \Gamma_1$.

The construction of a model in $\mathcal{M}_2$ satisfying Γ_2 is similar and left to the reader.

It remains to show the implications (2) $\Rightarrow$ (1) and (3) $\Rightarrow$ (1). They are similar, so we concentrate on the first. We require a rather technical lemma first. As in the proof of Theorem 18, take cardinals κ_i, $i \in \{1, 2\}$ as in Lemma 44 for $(\mathcal{L}_i, \mathcal{M}_i)$ and put $\kappa = max\{\kappa_1, \kappa_2\}$.

Lemma 47. *Suppose (2) holds.*

(a) There exist $\mathfrak{W}_1 = \langle W_1, F^{\mathfrak{W}}, R^{\mathfrak{W}} \rangle \in \mathcal{M}_1$, an assignment $\mathcal{A} = \langle \mathcal{A}_1, \mathcal{A}_2 \rangle$ into $\mathfrak{W}_1$, and a sequence $X_0, \ldots, X_m$ of subsets of W_1 such that

[a1] $\mathcal{A}_2(a) \in X_m$, for all $a \in obj(\Gamma_1)$,

[a2] $\langle \mathfrak{W}_1, \mathcal{A} \rangle \models \Gamma_1$,

[a3] $X_{n+1} \subseteq X_n \cap c^{\mathfrak{W}_1}(X_n)$, for all $0 \leq n < m$,

[a4] The set $\{sur_1(s)^{\mathfrak{W}_1,\mathcal{A}} \cap X_m : s \in \Sigma_1\}$ is a κ-partition of X_m,

[a5] The sets

$$\{sur_1(s)^{\mathfrak{W}_1,\mathcal{A}} \cap (X_n - X_{n+1}) : s \in \Sigma_1\}$$

are κ-partitions of $X_n - X_{n+1}$, for $0 \leq n < m$.

[a6] $|W_1 - X_0| = \kappa$.

(b) There exist $\mathfrak{W}_2 = \langle W_2, G^{\mathfrak{W}}, Q^{\mathfrak{W}} \rangle \in \mathcal{M}_2$, an assignment $\mathcal{B} = \langle \mathcal{B}_1, \mathcal{B}_2 \rangle$, and a sequence $Y_0, \ldots, Y_r$ of subsets of W_2 such that

[b1] $\mathcal{B}_2(a) \in Y_r$, for all $a \in obj(\Gamma_1)$,

[b2] $\langle \mathfrak{W}_2, \mathcal{B} \rangle \models \Gamma_2$,

[b3] $Y_{n+1} \subseteq Y_n \cap d^{\mathfrak{W}_2}(Y_n)$, for all $0 \leq n < r$,

[b4] The set $\{sur_2(s)^{\mathfrak{M},\mathcal{A}} \cap Y_r : s \in \Sigma_1\}$ is a κ-partition of Y_r,

[b5] The sets

$$\{sur_2(s)^{\mathfrak{M},\mathcal{A}} \cap (Y_n - Y_{n+1}) : s \in \Sigma_1\}$$

are κ-partitions of $Y_n - Y_{n+1}$, for $0 \leq n < r$.

[b6] $|W_2 - Y_0| = \kappa$.

Proof. We prove (a). Claim (b) is proved similarly and left to the reader. By assumption and Lemma 44, we find an ADM $\mathfrak{W}_a = \langle W_a, F^{\mathfrak{W}_a}, R^{\mathfrak{W}_a} \rangle \in \mathcal{M}_1$ with $|W_a| = \kappa$ and an assignment $\mathcal{A}^a = \langle \mathcal{A}_1^a, \mathcal{A}_2^a \rangle$ such that $\langle \mathfrak{W}_a, \mathcal{A}^a \rangle \models \Gamma_1$.

Let

$$Z_n = (c^{\leq n}(sur_1(\bigvee \Sigma_1)))^{\mathfrak{W}_a,\mathcal{A}_a},$$

for $0 \leq n \leq m$. By Lemma 46 (1) we can take for every n with $-1 \leq n \leq m$ an ADM $\mathfrak{W}_n = \langle W_n, F^{\mathfrak{W}_n}, R^{\mathfrak{W}_n} \rangle \in \mathcal{M}_1$ and assignments $\mathcal{A}^n$ such that

$$\{sur_1(s)^{\mathfrak{W}_n,\mathcal{A}^n} : s \in \Sigma_1\}$$

are κ-partitions of $\mathfrak{W}_n$.

Take the disjoint union $\mathfrak{W}$ (with $\mathfrak{W} = \langle W, F^{\mathfrak{W}}, R^{\mathfrak{W}} \rangle$) of the $\mathfrak{W}_n$, $-1 \leq n \leq m$, and $\mathfrak{W}_a$. Define $\mathcal{A} = \langle \mathcal{A}_1, \mathcal{A}_2 \rangle$ in $\mathfrak{W}$ by putting

$$\mathcal{A}_1(x) = \mathcal{A}_1^a(x) \cup \bigcup_{-1 \leq i \leq m} \mathcal{A}_1^i(x),$$

for all set variables x and $\mathcal{A}_2(b) = \mathcal{A}_2^a(b)$, for all object variables b. Let, for $0 \leq n \leq m$,

$$X_n = Z_n \cup \bigcup_{n \leq i \leq m} W_i.$$

We show that $\langle \mathfrak{W}, \mathcal{A} \rangle$ and the sets X_n, $0 \leq n \leq m$, are as required.

[a1] We have $\langle \mathfrak{W}_a, \mathcal{A}^a \rangle \models \Gamma_1$ and so $\mathcal{A}_2(b) = \mathcal{A}_2^a(b) \in Z_m$ for all $b \in ob(\Gamma_1)$. Hence $\mathcal{A}_2(b) \in X_m = Z_m \cup W_m$ for all $b \in ob(\Gamma_1)$.

[a2] By the definition of disjoint unions and because $\langle \mathfrak{W}_a, \mathcal{A}^a \rangle \models \Gamma_1$.

[a3] Firstly, we have, by the definition of $c^{\leq n}t$ and since $c^{\mathfrak{W}}$ is monotone (it distributes over intersections),

$$Z_{n+1} \subseteq Z_n \cap c^{\mathfrak{W}}(Z_n) \subseteq X_n \cap c^{\mathfrak{W}}(X_n). \tag{2}$$

Secondly, by the definition of disjoint unions, the first property of covering normal terms, and since $c^{\mathfrak{W}}$ is monotone

$$\bigcup_{n+1 \leq i \leq m} W_i \subseteq \bigcup_{n \leq i \leq m} W_i \subseteq \bigcup_{n \leq i \leq m} W_i \cap c^{\mathfrak{W}}(\bigcup_{n \leq i \leq m} W_i) \subseteq X_n \cap c^{\mathfrak{W}} X_n. \tag{3}$$

From (2) and (3) we obtain

$$X_{n+1} = Z_{n+1} \cup \bigcup_{n+1 \leq i \leq m} W_i \subseteq X_n \cap c^{\mathfrak{W}} X_n. \tag{4}$$

[a4] We show that the three properties from Definition 45 are satisfied. Since

$$\{sur_1(s)^{\mathfrak{W}_m, \mathcal{A}_m} : s \in \Sigma_1\}$$

is a κ-partition of W_m, we have $|sur_1(s)^{\mathfrak{W}_m, \mathcal{A}_m}| = \kappa$ for all $s \in \Sigma_1$. This implies Property 1 since $sur_1(s)^{\mathfrak{W}, \mathcal{A}} \cap W_m = sur_1(s)^{\mathfrak{W}_m, \mathcal{A}_m}$, $W_m \subseteq X_m$, and $|X_m| \leq \kappa$.

Property 2 is an immediate consequence of the definition of Σ_1. As for Property 3, we show that, for all $w \in X_m$, we have $w \in s^{\mathfrak{W}, \mathcal{A}}$ for an $s \in \Sigma_1$. Fix a $w \in X_m$. We distinguish two cases: firstly, assume $w \in W_m$. Then, by the fact that $\{sur_1(s)^{\mathfrak{W}_m, \mathcal{A}_m} : s \in \Sigma_1\}$ is a κ-partition of W_m, it is clear that there exists an $s \in \Sigma_1$ as required. Secondly, assume $w \in Z_m = (c^{\leq m}(sur_1(\bigvee \Sigma_1)))^{\mathfrak{W}_a, \mathcal{A}_a}$. By definition of $c^{\leq m}t$, we have $w \in (sur_1(\bigvee \Sigma_1))^{\mathfrak{W}_a, \mathcal{A}_a}$ and so again $w \in sur_1(s)^{\mathfrak{W}, \mathcal{A}}$ for some $s \in \Sigma_1$.

[a5] The proof is similar to that of Property [a4].

[a6] By definition.

❑

Suppose now that we have

$$\mathfrak{W}_1 = \langle W_1, F^{\mathfrak{W}_1}, R^{\mathfrak{W}_1} \rangle,\ \mathcal{A},\ X_m, \ldots, X_0 \text{ and } \mathfrak{W}_2 = \langle W_2, G^{\mathfrak{W}_2}, Q^{\mathfrak{W}_2} \rangle,\ \mathcal{B},\ Y_r, \ldots, Y_0$$

satisfying the properties listed in the lemma above. We may assume that

$$(W_1 - X_m) \cap (W_2 - Y_r) = \emptyset.$$

Using an appropriate bijection b from X_m onto Y_r we may also assume that $X_m = Y_r$, $\mathcal{A}_2(a) = \mathcal{B}_2(a)$ for all object variables $a \in obj(\Gamma_1)$, and

$$sur_1(s)^{\mathfrak{W}_1, \mathcal{A}} \cap X_m = sur_2(s)^{\mathfrak{W}_2, \mathcal{B}} \cap X_m \text{ for all } s \in \Sigma_1. \tag{5}$$

This follows from the fact that all object variables are mapped by $\mathcal{A}_2$ and $\mathcal{B}_2$ into X_m and Y_r ([a1], [b1]), respectively, the injectivity of the mappings $\mathcal{A}_2$ and $\mathcal{B}_2$, and the conditions [a4] and [b4] which state that $\{sur_1(s)^{\mathfrak{W}_1, \mathcal{A}} \cap X_m : s \in \Sigma_1\}$ and $\{sur_2(s)^{\mathfrak{W}_2, \mathcal{B}} \cap Y_r : s \in \Sigma_1\}$ both form κ-partitions of $X_m = Y_r$. Some abbreviations are useful: set

- $A_i = X_i - X_{i+1}$, for $0 \leq i < m$,
- $B_i = Y_i - Y_{i+1}$, for $0 \leq i < r$,
- $A_{-1} = W_1 - X_0$, $B_{-1} = W_2 - Y_0$.

We find models $\mathfrak{W}^i = \left\langle A_i, G^{\mathfrak{W}^i}, Q^{\mathfrak{W}^i} \right\rangle \in \mathcal{M}_2$ with assignments $\mathcal{B}^i = \langle \mathcal{B}_1^i, \mathcal{B}_2^i \rangle$, $-1 \le i \le m-1$, such that, for $0 \le i \le m-1$,

$$sur_2(s)^{\mathfrak{W}^i, \mathcal{B}^i} = sur_1(s)^{\mathfrak{W}_1, \mathcal{A}} \cap A_i \text{ for all } s \in \Sigma_1. \tag{6}$$

This follows from [a5], [a6], and Lemma 46 (2).

And we find, this time using [b5], [b6], and Lemma 46 (1), models $\mathfrak{V}^i = \left\langle B_i, F^{\mathfrak{V}^i}, R^{\mathfrak{V}^i} \right\rangle \in \mathcal{M}_1$ with assignments $\mathcal{A}^i = \langle \mathcal{A}_1^i, \mathcal{A}_2^i \rangle$, $-1 \le i \le r-1$, such that, for $0 \le i \le r-1$,

$$sur_1(s)^{\mathfrak{V}^i, \mathcal{A}^i} = sur_2(s)^{\mathfrak{W}_2, \mathcal{B}} \cap B_i \text{ for all } s \in \Sigma_1. \tag{7}$$

Let

$$\mathfrak{W}_1' = \left\langle W_1 \cup (W_2 - Y_r), F^{\mathfrak{W}_1'}, R^{\mathfrak{W}_1'} \right\rangle \in \mathcal{M}_1$$

be the disjoint union of the $\mathfrak{V}^i$, $-1 \le i < r$, and $\mathfrak{W}_1$, and let

$$\mathfrak{W}_2' = \left\langle W_2 \cup (W_1 - X_m), G^{\mathfrak{W}_2'}, Q^{\mathfrak{W}_2'} \right\rangle \in \mathcal{M}_2$$

be the disjoint union of the $\mathfrak{W}^i$, $-1 \le i < m$, and $\mathfrak{W}_2$. We assume $X_m = Y_r$ and so the domain of both ADMs is $W_1 \cup W_2$.

Define a model $\mathfrak{W} = \langle W, (F \cup G)^{\mathfrak{W}}, (R \cup Q)^{\mathfrak{W}} \rangle \in \mathcal{M}$ based on $W = W_1 \cup W_2$ by putting

- $R^{\mathfrak{W}} = R^{\mathfrak{W}_1'}$,
- $F^{\mathfrak{W}} = F^{\mathfrak{W}_1'}$,
- $Q^{\mathfrak{W}} = Q^{\mathfrak{W}_2'}$,
- $G^{\mathfrak{W}} = G^{\mathfrak{W}_2'}$.

Define an assignment $\mathcal{C} = \langle \mathcal{C}_1, \mathcal{C}_2 \rangle$ in $\mathfrak{W}$ by putting

- $\mathcal{C}_2(a) = \mathcal{A}_2(a) (= \mathcal{B}_2(a))$, for all $a \in obj(\Gamma_1)$.
- $\mathcal{C}_1(x) = \mathcal{A}_1(x) \cup \bigcup_{-1 \le i < r} \mathcal{A}_1^i(x)$, for all set variables x in $term(\Gamma)$.

  Notice that $\mathcal{C}_1(x) = \mathcal{B}_1(x) \cup \bigcup_{-1 \le i < m} \mathcal{B}_1^i(x)$, for all set variables $x \in term(\Gamma)$.
- $\mathcal{C}_1(x_t) = \mathcal{A}_1(x_t) \cup \bigcup_{-1 \le i < r} \mathcal{A}_1^i(x_t)$, for all $t = g(t_1, \dots, t_k) \in sub^1(\Gamma)$.
- $\mathcal{C}_1(x_t) = \mathcal{B}_1(x_t) \cup \bigcup_{-1 \le i < m} \mathcal{B}_1^i(x_t)$, for all $t = f(t_1, \dots, t_k) \in sub^1(\Gamma)$.

We will show that $\langle \mathfrak{W}, \mathcal{C} \rangle \models \Gamma$. Firstly, however, we make a list of the relevant properties of $\langle \mathfrak{W}, \mathcal{C} \rangle$:

Claim 1.

[c1] $\mathcal{C}_2(a) \in X_m = Y_r$, for all $a \in obj(\Gamma)$;

[c2] $\langle \mathfrak{W}, \mathcal{C} \rangle \models \Gamma_1 \cup \Gamma_2$;

[c3] $sur_1(t)^{\mathfrak{W}, \mathcal{C}} \cap (X_0 \cup Y_0) = sur_2(t)^{\mathfrak{W}, \mathcal{C}} \cap (X_0 \cup Y_0)$, for all $t \in \Sigma_1$;

[c4] $sur_1(s)^{\mathfrak{W}, \mathcal{C}} \cap (X_0 \cup Y_0) = sur_2(s)^{\mathfrak{W}, \mathcal{C}} \cap (X_0 \cup Y_0)$, for all $s \in sub^1(\Gamma)$;

[c5] $X_{n+1} \subseteq X_n \cap c^{\mathfrak{W}}(X_n)$, for all $0 \le n < m$;

[c6] $Y_{n+1} \subseteq Y_n \cap d^{\mathfrak{W}}(Y_n)$, for all $0 \le n < r$;

[c7] for all $g \in G$ of arity l, $0 \leq n < m$, and all $C_1, \ldots, C_l \subseteq W$:

$$g^{\mathfrak{W}}(C_1, \ldots, C_l) \cap X_n = g^{\mathfrak{W}}(C_1 \cap X_n, \ldots, C_l \cap X_n) \cap X_n;$$

[c8] for all $f \in F$ of arity l, $0 \leq n < r$, and all $C_1, \ldots, C_l \subseteq W$:

$$f^{\mathfrak{W}}(C_1, \ldots, C_l) \cap Y_n = f^{\mathfrak{W}}(C_1 \cap Y_n, \ldots, C_l \cap Y_n) \cap Y_n.$$

Proof of the Claim 1. [c1] follows from [a1] and [b1] and the construction of $\langle \mathfrak{W}, \mathcal{C} \rangle$. [c2] follows from [a2] and [b2]. [c3] follows from the construction of $\langle \mathfrak{W}, \mathcal{C} \rangle$ and equations (5), (6), and (7). [c4] follows from [c3]. [c5] and [c6] follow from [a3] and [b3], respectively. It remains to prove [c7] and [c8]. But [c7] follows from the fact that $\langle \mathfrak{W}, G^{\mathfrak{W}} \rangle$ is the disjoint union of structures based on X_n and $W - X_n$, for $0 \leq n < m$, and [c8] is dual to [c7]. Claim 1 is proved.

We now show $\langle \mathfrak{W}, \mathcal{C} \rangle \models \Gamma$. To this end we first show the following

Claim 2. For all k_1, k_2 with $0 \leq k_1 \leq m$ and $0 \leq k_2 \leq r$ and all $s \in sub^1(\Gamma)$ with $d_F(s) \leq k_1$ and $d_G(s) \leq k_2$ we have, for $Z \in \{X_{k_1}, Y_{k_2}\}$,

$$Z \cap s^{\mathfrak{M}, \mathcal{C}} = Z \cap sur_1(s)^{\mathfrak{M}, \mathcal{C}} = Z \cap sur_2(s)^{\mathfrak{M}, \mathcal{C}}.$$

Proof of Claim 2. By [c4] it suffices to prove the first equation. The proof is by induction on the cardinal $k_1 + k_2$. The induction base $k_1 = k_2 = 0$ follows from $sur_1(s) = sur_2(s)$ for $d_F(s) = d_G(s) = 0$.

Suppose the claim is proved for all $X_k, Y_{k'}$ with $k \leq m$, $k' \leq r$ and $k + k' < k_1 + k_2$. We prove the claim for X_{k_1}, Y_{k_2}. The proof is by induction on the construction of terms s with $d_F(s) \leq k_1$ and $d_G(s) \leq k_2$. The boolean cases are trivial.

Suppose $s = f(s_1, \ldots, s_l)$ with $d_F(s) \leq k_1$ and $d_G(s) \leq k_2$. We have to show the following two statements:

(1) $X_{k_1} \cap s^{\mathfrak{W}, \mathcal{C}} = X_{k_1} \cap sur_1(s)^{\mathfrak{M}, \mathcal{C}}$.

(2) $Y_{k_2} \cap s^{\mathfrak{W}, \mathcal{C}} = Y_{k_2} \cap sur_1(s)^{\mathfrak{M}, \mathcal{C}}$.

Consider (1) first. By induction hypothesis

$$X_{k_1 - 1} \cap s_i^{\mathfrak{W}, \mathcal{C}} = X_{k_1 - 1} \cap sur_1(s_i)^{\mathfrak{W}, \mathcal{C}},$$

for $1 \leq i \leq l$. We have

$$\begin{aligned} X_{k_1-1} \cap c^{\mathfrak{W}}(X_{k_1-1}) \cap s^{\mathfrak{W},\mathcal{C}} &= X_{k_1-1} \cap c^{\mathfrak{W}}(X_{k_1-1}) \cap f^{\mathfrak{W}}(s_1^{\mathfrak{W},\mathcal{C}}, \ldots, s_l^{\mathfrak{W},\mathcal{C}}) \\ &= X_{k_1-1} \cap c^{\mathfrak{W}}(X_{k_1-1}) \cap f^{\mathfrak{W}}(sur_1(s_1)^{\mathfrak{W},\mathcal{C}}, \ldots, sur_1(s_l)^{\mathfrak{W},\mathcal{C}}) \\ &= X_{k_1-1} \cap c^{\mathfrak{W}}(X_{k_1-1}) \cap sur_1(s)^{\mathfrak{W},\mathcal{C}}. \end{aligned}$$

The second equation is an immediate consequence of the third property of covering normal terms as given in Definition 26. Now the equality follows from [c5], i.e. $X_{k_1} \subseteq X_{k_1-1} \cap c^{\mathfrak{W}}(X_{k_1-1})$. (1) is proved.

(2) Suppose first that $k_2 = r$. Then $Y_{k_2} = X_m$ and the claim can be proved as above since $X_m \subseteq X_{k_1}$ and, by induction hypothesis, $X_{k_1-1} \cap s_i^{\mathfrak{W},\mathcal{C}} = X_{k_1-1} \cap sur_1(s_i)^{\mathfrak{W},\mathcal{C}}$, for $1 \leq i \leq l$.

Asssume now that $k_2 < r$. By induction hypothesis,

$$Y_{k_2} \cap s_i^{\mathfrak{W},\mathcal{C}} = Y_{k_2} \cap sur_2(s_i)^{\mathfrak{W},\mathcal{C}},$$

for $1 \leq i \leq l$. Hence

$$f^{\mathfrak{W}}(Y_{k_2} \cap s_1^{\mathfrak{W},\mathcal{C}}, \ldots, Y_{k_2} \cap s_l^{\mathfrak{W},\mathcal{C}}) = f^{\mathfrak{W}}(Y_{k_2} \cap sur_2(s_1)^{\mathfrak{W},\mathcal{C}}, \ldots, Y_{k_2} \cap sur_2(s_l)^{\mathfrak{W},\mathcal{C}}).$$

We intersect both sides of the equality with Y_{k_2} and derive with the help of [c8]:

$$Y_{k_2} \cap f^{\mathfrak{W}}(s_1^{\mathfrak{W},\mathcal{C}}, \ldots, s_l^{\mathfrak{W},\mathcal{C}}) = Y_{k_2} \cap f^{\mathfrak{W}}(sur_2(s_1)^{\mathfrak{W},\mathcal{C}}, \ldots, sur_2(s_l)^{\mathfrak{W},\mathcal{C}}).$$

This means $Y_{k_2} \cap s^{\mathfrak{W},\mathcal{C}} = Y_{k_2} \cap sur_2(s)^{\mathfrak{W},\mathcal{C}}$, and the equation follows. The statements are proved.

The case $s = g(s_1, \ldots, s_l)$ is dual and left to the reader. We have proved claim 2.

By induction (compare in the proof of Theorem 21 the proof of (1) from the corresponding claim) be obtain from Claim 2:

$$X_m \cap s^{\mathfrak{W},\mathcal{C}} = X_m \cap sur_1(s)^{\mathfrak{M},\mathcal{C}} \text{ for all } s \in term(\Gamma). \tag{8}$$

Let us see how $\langle \mathfrak{W}, \mathcal{A} \rangle \models \Gamma$ follows from (8). We distinguish three cases: Suppose $aR_ib \in \Gamma$. Then $aR_ib \in \Gamma_1$ and therefore $\langle \mathfrak{W}, \mathcal{C} \rangle \models aR_ib$. Similarly, $aQ_ib \in \Gamma$ implies $aQ_ib \in \Gamma_2$ and $\langle \mathfrak{W}, \mathcal{C} \rangle \models aQ_ib$. Suppose $(a : t) \in \Gamma$. Then $(a : sur_1(t)) \in \Gamma_1$ and so, by [c2], $\mathcal{C}_2(a) \in sur_1(t)^{\mathfrak{W},\mathcal{C}}$ which implies, by (8), $\mathcal{C}_2(a) \in t^{\mathfrak{W},\mathcal{C}}$. Hence $\langle \mathfrak{W}, \mathcal{C} \rangle \models (a : t)$. This finishes the proof of Theorem 29.