Loris Bozzato, Thomas Eiter, Luciano Serafini, **Enhancing context knowledge repositories with justifiable exceptions**, ARTIFICIAL INTELLIGENCE, Volume 257, April 2018, Pages 72-126
DOI: 10.1016/j.artint.2017.12.005

Enhancing Context Knowledge Repositories with Justifiable Exceptions[A]

Loris Bozzato[a,*], Thomas Eiter[b], Luciano Serafini[a]

[a] Fondazione Bruno Kessler, Via Sommarive 18, 38123 Trento, Italy
[b] Institut für Informationssysteme, Technische Universität Wien, Favoritenstraße 9-11, A-1040 Vienna, Austria

Abstract

Dealing with context dependent knowledge is a well-known area of study that roots in John McCarthy's seminal work. More recently, the Contextualized Knowledge Repository (CKR) framework has been conceived as a logic-based approach in which knowledge bases have a two layered structure, modeled by a global context and a set of local contexts. The global context not only contains the meta-knowledge defining the properties of local contexts, but also holds the global (context independent) object knowledge that is shared by all of the local contexts. In many practical cases, however, it is desirable to leave the possibility to "override" the global object knowledge at the local level: in other words, it is interesting to recognize the pieces of knowledge that can admit exceptional instances in the local contexts that do not need to satisfy the general axiom. To address this need, we present in this paper an extension of CKR in which defeasible axioms can be included in the global context. The latter are verified in the local contexts only for the instances for which no exception to overriding exists, where exceptions require a justification in terms of facts that are provable from the knowledge base. We formally define this semantics and study some semantic and computational properties, where we characterize the complexity of the major reasoning tasks, among them satisfiability testing, instance checking, and conjunctive query answering. Furthermore, we present a translation of extended CKRs with knowledge bases in the Description Logic $\mathcal{SROIQ}$-RL under the novel semantics to datalog programs under the stable model (answer set) semantics. We also present an implementation prototype and examine its scalability with respect to the size of the input CKR and the amount (level) of defeasibility in experiments. Finally, we compare our representation approach with some major formalisms for expressing defeasible knowledge in Description Logics and contextual knowledge representation. Our work adds to the body of results on using deductive database technology such as SQL and datalog in these areas, and provides an expressive formalism (in terms of intrinsic complexity) for exception handling by overriding.

Keywords: Knowledge representation, contextual reasoning, description logics, datalog, defeasible knowledge

[A] Part of this work has been previously presented in preliminary form in [1, 2, 3].

* Corresponding author at: Fondazione Bruno Kessler, Via Sommarive 18, 38123 Trento, Italy.

Email addresses: bozzato@fbk.eu (Loris Bozzato), eiter@kr.tuwien.ac.at (Thomas Eiter), serafini@fbk.eu (Luciano Serafini)

1. Introduction

In the field of Knowledge Representation and Reasoning, the problem of dealing with context dependent knowledge is a well-known area of study. Initial proposals for a formal definition of contextual knowledge and reasoning date back to the works of McCarthy [4], Lenat [5], and Giunchiglia et al. [6, 7]. In the era of the Semantic Web (SW), representation of context dependent knowledge has been recognized as an extremely relevant issue, due to the necessity to qualify each data set with meta-data to allow users and applications to interpret the data set contents in the right context. This interest has led to a number of logic based proposals, e.g. [8, 9, 10, 11, 12, 13, 14, 15]. In the current article we will extend one of the current formalisms, the *Contextualized Knowledge Repository (CKR) framework* [12, 16, 17], with its latest formulation in [1], with a new form of non monotonic reasoning based on justification.

A CKR knowledge base is a two-layer structure: the higher level consists of a global context; the lower level consists of a set of local contexts. For example, a CKR for a touristic recommendation system in the Trentino region,[1] is composed of a global context that describes all the locations, the venues and the events that are available in the region, and by a set of local contexts each of which describes the details of an event or the profile, interests and plans of a single user. The global context contains two types of knowledge: the former is composed by a context independent kernel of facts about the domain of discourse. The truth of these pieces of knowledge is assumed to be immutable; for instance, the fact that *Castello del Buonconsiglio* is located in *Trento*. This knowledge is accessible by all the local contexts. The second type of knowledge contained in the global context, is *meta-knowledge* specifying the properties of local contexts. Local contexts, on the other hand, contain knowledge that holds under specific circumstances or assumptions (e.g. during a certain period of time, or when a certain event occurs) and thus they represent different partial and perspective views of the domain. Knowledge in different contexts is not completely independent, as the global context independent knowledge is propagated from the global to the local contexts and it is used to constrain local knowledge in different contexts.

In many practical cases, however, it is desirable to leave the possibility to "override" the global object knowledge at the local level, by allowing axioms to admit *exceptions* in their local instantiations. For example, in the above scenario of the event recommendation system, we might want to assert at the global level that *"by default, all of the cheap events are interesting"*, but then override this implication for particular kind of events in the context of a specific participant. (e.g., a user might not be interested in sport events independently of their price). We might also want to express defeasibility on the propagation of information: for instance, in a CKR representing an organization, we might want to express that *"by default, all the employees of a year will be employees in the next year"* and override the axiom in the context of a specific year for employees that finished their working contract.[2] In other words, we want to allow certain global axioms to be *defeasible*, so that they admit exceptional instances in local contexts, while holding in the general case: this clearly requires to add a notion of *non-monotonicity* across the global and the local parts of a CKR.

The aim of this work is thus to extend the CKR framework in order to support the form of defeasibility for global object knowledge as described above, under some desiderata: (1) defeasibility should be used parsimoniously, in the sense that information is inherited as much as possible, such that the information loss in conclusions at the local level is as little as necessary; (2)

[1] cf. Examples 4 and 5 in Section 3.
[2] cf. Example 6 in Section 3.

overriding should be supported by clear evidence, in terms of facts that lead to a contradiction; and (3) reasoning with exceptions should be realized using deductive database technology, in particular SQL and datalog, that has been fostered for CKRs [1] in line with work around Description Logics [18, 19, 20, 21, 22, 23, 24].

To this end, we introduce defeasible axioms guided by the approach of inheritance logic programs in [25], extending the datalog representation of CKR semantics in [1]. In inheritance logic programs the idea is that special rules recognize exceptional facts at the local level and others propagate global facts only if they are not proved to be overridden at the local level, which happens if the opposite is derived; in the same vein, we consider instances of axioms that might be overridden at the local level. On the basis of this semantics, we develop a translation for CKRs on $\mathcal{SROIQ}$-RL (i.e. OWL-RL [26]) with defeasible axioms into datalog programs. Specifically, instance checking over a CKR reduces this way to (cautious) inference from such programs under the answer set semantics [27] (also known as stable model semantics [28]), which thus can be used to implement query answering for CKR with defeasibility.

The main contributions of this paper are briefly summarized as follows:

(1) We present a new syntax and semantics of an extension of CKR for *defeasible axioms* $D(\alpha)$ in the global context. Notably, this allow us to introduce for the first time a notion of non-monotonicity across contexts in CKR (Section 3). Intuitively, a global defeasible axiom $D(\alpha)$ means that, at the level of instantiations for individuals, α is inherited by local contexts unless it generates a contradiction in the local context knowledge base. Model based semantics of CKR needs thus to be extended in order to reason with exceptions for such axioms. Axiom instances $\alpha(\mathbf{e})$ representing local exceptions are called *clashing assumptions*: in the evaluation of α at a local context, its instantiation with $\mathbf{e}$ is not considered (i.e. α is "overridden" for $\mathbf{e}$). However, such assumptions of exceptions must be justified: the instance of α for $\mathbf{e}$ must be provable to be unsatisfiable at the local context. This is ensured if (atomic) assertions can be derived which prove this unsatisfiability; we call such assertions *clashing sets* (cf. Section 3.2). As such, CKR interpretations are thus extended with a set of the local clashing assumptions *CAS* and called *CAS*-interpretations: intuitively, *CAS*-interpretations interpret local axioms by disregarding exceptional instances in *CAS* (cf. Section 3.2.1). Then, CKR models can be defined as those *CAS*-models that are *justified*, i.e. that provide a reason for the presence of each clashing assumption in *CAS* by verifying a correspondent local clashing set.

(2) We characterize reasoning in CKR with defeasible axioms, where we consider entailment of axioms and conjunctive queries (CQs) (Section 4). In details, we derive helpful semantic characterizations of justified clashing assumptions; based on this, we study the computational complexity of major reasoning tasks. We show that justified *CAS*- and CKR-model checking are tractable, while satisfiabilty is NP-complete in general. Under data complexity, entailment of axioms is coNP-complete while answering conjunctive queries is Π^p_2-complete, with lower complexity for restricted inputs.

(3) We extend the datalog translation for $\mathcal{SROIQ}$-RL based CKR from [1] with rules for the translation of defeasible axioms and for considering local exceptions in the propagation of such knowledge (Section 5). We express non-monotonicity using answer set semantics, such that instance checking over a CKR with defeasible axioms reduces to cautious inference from all answer set of the translation, and likewise conjunctive query answering. In particular, we note that the proposed translation (based on positive datalog programs) is not trivial and need special attention for dealing with the *negative* knowledge inside clashing sets that needs to be derived for the justification of a clashing assumption (in particular in presence of negative disjunctive information, cf. Section 5.2). As a solution to this problem, we propose a translation in which reasoning over such negative knowledge is performed by encoding it through individual proofs by contradiction. In Section 5.3 we show that the proposed translation provides a sound and complete materialization calculus for instance checking and conjunctive query answering over CKRs in OWL-RL.

(4) We study scalability of our approach. In particular, the experiments confirm that scalability of the approach is influenced by the percentage of defeasible axioms in the initial CKR and the number of their exceptional instances. To this aim, we have developed a prototype implementation, called CKR*ew (CKR datalog Rewriter)* that compiles a CKR to a datalog program following the presented translation (Section 6). We present the prototype and we study its behavior with respect to different sizes of the input CKR and percentage of defeasible axioms. The prototype and test sets are freely distributed for use, replication of experiments and possible comparison with other similar implementations.

The contributions of this work are interesting in general for the area of (logic based) Knowledge Representation: our solution proposes an expressive means for combining reasoning with structured Description Logics knowledge bases (viz. contextualized Semantic Web knowledge bases) with a notion of axiom overriding. As such, our work can be compared not only with respect to methods for representation of defeasibility in contextualized logics (e.g. [9, 29]), but also to solutions for introducing non-monotonic reasoning in Description Logics (e.g. [30, 31]). In Section 7, we provide an extended comparison of our approach with some of the major non-monotonic formalisms for description logics and contextual knowledge representation mentioned above, highlighting commonalities and differences. In particular, our work differs from these formalisms with respect to some relevant aspects:

- our approach allows to reason with non-monotonic features in modular knowledge bases under an expressive language (cf. Sections 7.1 and 7.2);
- in case of conflicts across possible overridings, it does not request or elicit a preference on possible interpretations, but it presents—in line with the ASP paradigm—alternatives as different models, thus allowing to "reason by cases" on the conflicting solutions (cf. Sections 7.2 and 7.4);
- the definition of model is not defined by minimization, but through the idea of justification of exceptions which is based on semantic consequence (cf. Section 7.3). In particular, no "normal" members of a concept are defined, but instead single or tuples of individuals are regarded as "exceptional" w.r.t. defeasible axioms: this allows us to deal with inheritance of properties at the level of instances (cf. Section 7.4);
- we provide a translation to datalog that is a direct extension of the materialization calculus approach proposed for the monotonic case in [1] and shows how modular knowledge can be encoded for non-monotonic reasoning using existing tools.

To increase readability, some proofs of technical results have been moved to the Appendix. The prototype and test sets used in the experiments are available on-line at http://ckrew.fbk.eu/.

2. Preliminaries

In this section, we recall the relevant languages from description logics (DLs) and from logic programming that underlie the context knowledge repositories presented in the later sections.

More specifically, these are $\mathcal{SROIQ}$-RL, which is a fragment of $\mathcal{SROIQ}$ [32] corresponding to OWL-RL in [26], and datalog under answer set (i.e., stable model) semantics [27].

2.1. $\mathcal{SROIQ}$ *syntax and semantics*

In the following we assume the usual presentation of description logics [33] and we will consider the logic $\mathcal{SROIQ}$ [32]. For ease of reference, the detailed presentation of syntax and semantics for $\mathcal{SROIQ}$ constructors and axioms is presented in Table A.13 in the Appendix. We summarize in the following the basic definitions that we will use throughout the paper.

A *DL vocabulary* $\Sigma = \mathrm{NC} \cup \mathrm{NR} \cup \mathrm{NI}$ consists of three mutually disjoint countably infinite sets NC of *atomic concepts*, NR of *atomic roles*, and NI of *individual constants*. Complex *concepts* (complex *roles*) are recursively defined as the smallest sets containing all concepts and roles that can be inductively constructed using the usual concept constructors $\neg, \sqcap, \sqcup, \exists, \forall$ etc. and role constructors $^{-}, \circ$ etc. as usual (see Table A.13).

A $\mathcal{SROIQ}$ *knowledge base* $\mathcal{K} = (\mathcal{T}, \mathcal{R}, \mathcal{A})$ consists of a TBox $\mathcal{T}$ which contains *general concept inclusion (GCI)* axioms $C \sqsubseteq D$ and *concept equivalence* axioms $C \equiv D$, where C and D are concepts; an RBox $\mathcal{R}$ which contains *role inclusion (RIA)* axioms $S \sqsubseteq R$, reflexivity, and role disjointness axioms, where S and R are roles; and an ABox $\mathcal{A}$ which contains assertions of the forms $D(a)$, $R(a, b)$, $\neg R(a, b)$, $a = b$, and $a \neq b$, where a and b are any individual constants (see Table A.13)[3].

A *DL interpretation* is a pair $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$ where $\Delta^{\mathcal{I}}$ is a non-empty set called interpretation *domain* and $\cdot^{\mathcal{I}}$ is the *interpretation function* which provides denotations for individuals, concepts and roles: it assigns an element $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$ to each individual constant $a \in \mathrm{NI}$, a subset $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ to each concept $A \in \mathrm{NC}$, and a subset $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ to each role $R \in \mathrm{NR}$. Furthermore, $\cdot^{\mathcal{I}}$ extends to complex concepts and roles as described in Table A.13. An interpretation $\mathcal{I}$ *satisfies* an axiom (inclusion, assertion etc.) φ, denoted $\mathcal{I} \models_{\mathrm{DL}} \varphi$, if the respective semantic constraint in Table A.13 is fulfilled; $\mathcal{I}$ is a *model* of $\mathcal{K}$, denoted $\mathcal{I} \models_{\mathrm{DL}} \mathcal{K}$, if it satisfies all axioms of $\mathcal{K}$.

Furthermore, in the context of $\mathcal{K}$ each role in min-cardinality and self restrictions as well as in (ir)reflexivity, asymmetry, and disjointness axioms must be *simple* [32], which is defined as follows:

a) an atomic role R is simple, if it does not occur on the right-hand side of a RIA in $\mathcal{R}$;

b) an inverse role R^{-} is simple, if R is simple;

c) if R occurs on the right-hand side of a RIA in $\mathcal{R}$ and each such RIA is of the form $S \sqsubseteq R$ where S is simple, then also R is simple.

To preserve decidability, the RBox $\mathcal{R}$ in $\mathcal{SROIQ}$ knowledge bases is required to be *regular* [32]. Formally, a *regular order* is a strict partial order $\prec$ on roles such that, for any roles R, S, $R \prec S$ iff $R^{-} \prec S$. A RIA is $\prec$*-regular* if it is in one of the following forms: (i) $R \circ R \sqsubseteq R$; (ii) $R^{-} \sqsubseteq R$; (iii) $S_1 \circ \ldots \circ S_n \sqsubseteq R$ with $S_i \prec R$ for $i \in \{1, \ldots, n\}$; (iv) $R \circ S_1 \circ \ldots \circ S_n \sqsubseteq R$ with $S_i \prec R$ for $i \in \{1, \ldots, n\}$; (v) $S_1 \circ \ldots \circ S_n \circ R \sqsubseteq R$ with $S_i \prec R$ for $i \in \{1, \ldots, n\}$. An RBox $\mathcal{R}$ is *regular*, if there exists a regular order $\prec$ such that all role inclusions in $\mathcal{R}$ are $\prec$-regular.

For developing our approach, we use without loss of generality the *standard name assumption (SNA)* in the DL context, cf. [35, 36]: we have an infinite subset $\mathrm{NI}_S \subseteq \mathrm{NI}$ of individual constants,

[3] Note that the $Dis(C, D)$ axiom is not part of the original presentation of $\mathcal{SROIQ}$ [32] (while it is present as an operator in OWL 2 [34]). It can be easily expressed in terms of subsumption as $C \sqcap D \sqsubseteq \bot$ or $C \sqsubseteq \neg D$.

called *standard names* such that in every interpretation $\mathcal{I}$ we have (i) $\Delta^{\mathcal{I}} = \mathrm{NI}_S^{\mathcal{I}} = \{c^{\mathcal{I}} \mid c \in \mathrm{NI}_S\}$ and (ii) $c^{\mathcal{I}} \neq d^{\mathcal{I}}$, for every distinct $c, d \in \mathrm{NI}_S$; thus we may assume that $\Delta^{I} = \mathrm{NI}_S$ and $c^{\mathcal{I}} = c$ for each $c \in \mathrm{NI}_S$. Equality = is then in the FO-translation replaced by a predicate $\approx$ for which the axioms of a congruence relation are added, i.e., reflexivity, symmetry, transitivity, and $\forall \mathbf{x}.P(\mathbf{x}) \wedge \mathbf{x} \approx \mathbf{x}' \rightarrow P(\mathbf{x}')$, where $x = x_1, \ldots, x_n$, and $x' = x'_1, \ldots, x'_n$ and $\mathbf{x} \approx \mathbf{x}'$ stands for $\bigvee_{i=1}^{n} x_i \approx x'_i$. The standard names are supposed not to occur in the knowledge base, and allow us to access each element in an interpretation, apart from those elements that are "named" by individual constants occurring in a knowledge base (which are from $\mathrm{NI} \setminus \mathrm{NI}_S$). The *unique name assumption* can as usual be enforced by assertions $c \neq d$ for all individual constants in $\mathrm{NI} \setminus \mathrm{NI}_S$ resp. occurring in the knowledge base.

2.1.1. $\mathcal{SROIQ}$-*RL*

We base our framework on a restriction of the $\mathcal{SROIQ}$ syntax that corresponds to OWL-RL [26], which we refer to as $\mathcal{SROIQ}$-RL. To this end, we define the following grammars for a *left-side concept C* and a *right-side concept D* respectively:

$$C := A \mid \{a\} \mid C \sqcap C \mid C \sqcap C \mid \exists R.C \mid \exists R.\{a\} \mid \exists R.\top \tag{1}$$

$$D := A \mid \neg C \mid D \sqcap D \mid \exists R.\{a\} \mid \forall R.D \mid \leq nR.\top \tag{2}$$

where A is a concept name, R is role name and $n \in \{0, 1\}$. A *both-side concept* is a concept expression that is both a left- and right-side concept. Then, in $\mathcal{SROIQ}$-RL TBox axioms can only take the form $C \sqsubseteq D$, where C is a left-side and D is a right-side or $E \equiv F$, where E and F are both-side concepts. Next, in $\mathcal{SROIQ}$-RL the RBox can contain all role axioms of $\mathcal{SROIQ}$ except Ref(R). Finally, ABox concept assertions can only be of form $D(a)$, where D is a right-side concept; without loss of generality, we may also assume that D is atomic. For example, the following expressions are well-formed $\mathcal{SROIQ}$-RL axioms and assertions: $A \sqsubseteq \neg B$, $\{a\} \sqsubseteq \exists R.\{b\}$, $\exists R.\{b\} \equiv A$, $\neg B(a)$.

2.2. *Datalog programs and answer sets*

Following the approach in [20], we will express our rules in the language of *datalog*. However, while the rules in [20, 1] are positive, in order to capture defeasibility we need *(default) negation* not under the interpretation of answer sets semantics [27].

2.2.1. *Syntax*

A *signature* is a tuple $\langle \mathbf{C}, \mathbf{P} \rangle$ of a finite set $\mathbf{C}$ of *constants* and a finite set $\mathbf{P}$ of *predicates*. We assume a set $\mathbf{V}$ of *variables*; the elements of $\mathbf{C} \cup \mathbf{V}$ are *terms*. An *atom* is of the form $p(t_1, \ldots, t_n)$ where $p \in \mathbf{P}$ and $t_1, \ldots, t_n$, are terms.[4]

A (datalog) rule r is an expression of the form

$$a \leftarrow b_1, \ldots, b_k, \text{not } b_{k+1}, \ldots, \text{not } b_m. \tag{3}$$

where $a, b_1, \ldots, b_m$ are atoms and not is the negation as failure symbol (NAF). We denote with $Head(r)$ the head a of rule r and with $Body(r) = \{b_1, \ldots, b_k, \text{not } b_{k+1}, \ldots, \text{not } b_m\}$ the body of r, respectively. We allow that a is missing (*constraint*), viewing a as logical constant for falsity. A (datalog) *program P* is a finite set of rules.

[4] Note that we do not use strong ("classical") negation $\neg p$ over atoms p, i.e. only positive *literals* appear in our rules.

An atom (rule etc.) is *ground*, if no variables occur in it. A *ground substitution* σ for $\langle \mathbf{C}, \mathbf{P} \rangle$ is any function $\sigma : \mathbf{V} \to \mathbf{C}$; the *ground instance* of an atom (rule, etc.) χ from σ, denoted $\chi\sigma$, is obtained by replacing in χ each occurrence of variable $v \in \mathbf{V}$ with $\sigma(v)$. A *fact* H is a ground rule r with empty body (i.e., $m = 0$); we then omit $\leftarrow$. The *grounding* of a rule r, $grnd(r)$, is the set of all ground instances of r, and the *grounding* of a program P is $grnd(P) = \bigcup_{r \in P} grnd(r)$.

2.2.2. Semantics

Given a program P, the *(Herbrand) universe* U_P of P is the set of all constants occurring in P and the *(Herbrand) base* B_P of P is the set of all the ground atoms constructible from the predicates in P and the constants in U_P. An *interpretation* $I \subseteq B_P$ is any subset of B_P. An atom l is *true* in I, denoted $I \models l$, if $l \in I$.[5]

Given a rule $r \in grnd(P)$, we say that $Body(r)$ is true in I, denoted $I \models Body(r)$, if (i) $I \models b$ for each atom $b \in Body(r)$ and (ii) $I \not\models b$ for each atom not $b \in Body(r)$. A rule r is *satisfied* in I, denoted $I \models r$, if either $I \models Head(r)$ or $I \not\models Body(r)$. An interpretation I is a *model* of P, denoted $I \models P$, if $I \models r$ for each $r \in grnd(P)$; moreover, I is *minimal*, if $I' \not\models P$ for each subset $I' \subset I$.

Given an interpretation I for P, the (Gelfond-Lifschitz) *reduct* of P w.r.t. I, denoted by $G_I(P)$, is the set of rules obtained from $grnd(P)$ by (i) removing every rule r such that $I \models l$ for some not $l \in Body(r)$; and (ii) removing the NAF part from the bodies of the remaining rules. Then I is an *answer set* of P, if I is a minimal model of of $G_I(P)$; the minimal model is unique and exists iff $G_I(P)$ has some model. Furthermore, the minimal model is obtainable by fixpoint iteration. The following property is well-known.

Proposition 1. *If M is an answer set for P, then M is a minimal model of P.*

Using this interpretation for our programs, we say that an atom $a \in B_P$ is a *consequence* of P and we write $P \models a$ iff for every answer set M of P we have that $M \models a$.

3. Contextualized Knowledge Repositories with Defeasible Axioms

We now introduce CKRs and extend them with primitives to express defeasible axioms. We first present the syntax and then define a model-based semantics for the interpretation of defeasible inheritance from the upper contexts.

A *Contextualized Knowledge Repository (CKR)* is a two layered structure. The upper layer consists of a knowledge base G, which describes two types of knowledge:

(i). the structure and the properties of contexts of the CKR (called *meta-knowledge*), and

(ii). the knowledge that is context independent, i.e., that holds in every context (called *global knowledge*).

The lower layer is constituted by a set of (local) contexts; each contains (locally valid) facts and can also refer to what holds in other contexts. In order to support knowledge reuse, the knowledge of each context is organized in multiple knowledge modules that may be shared with other contexts. To model this, an association between contexts and modules is represented in the

[5]The semantics can be easily extended to negative literals $\neg p$: interpretations $I \subseteq B_P$ are required to be consistent (i.e., not contain *complementary* literals p and $\neg p$, viewing $\neg p$ as fresh predicate. This amounts to 3-valued interpretations, in which an atom p can be *true*, *false* ($\neg p$ is true), and *unknown* (neither p nor $\neg p$ is true).

meta-knowledge via a binary relation (mod), which can be either explicitly asserted or inferred from the meta-knowledge by reasoning; hence, each context can be associated with one or more knowledge modules which define its contents, while a knowledge module can be shared by one or more contexts. The mod association also allows the knowledge engineer to link modules to classes of contexts, and thus to describe the general knowledge that is valid for all contexts of the same kind. In principle, the knowledge in a CKR can be expressed using any DL language: we thus provide in the following a parametric definition for any DL language and we successively instantiate it to SROIQ-RL.

3.1. Syntax

Meta-Language. The meta-knowledge of a CKR is expressed in a DL language containing the elements to define the contextual structure[6].

Definition 1 (meta-vocabulary)**.** *A* meta-vocabulary *is a DL vocabulary* $\Gamma = NC_\Gamma \uplus NR_\Gamma \uplus NI_\Gamma$ *that consists of sets* NC_Γ *of atomic concepts, a set* NR_Γ *of atomic roles, and a set* NI_Γ *of individual constants that are mutually disjoint and contain the following sets of symbols:*

1. $\mathsf{N} \subseteq NI_\Gamma$ *of* context names*;*
2. $\mathsf{M} \subseteq NI_\Gamma$ *of* module names*;*
3. $\mathsf{C} \subseteq NC_\Gamma$ *of* context classes, *including* Ctx[7]*;*
4. $\mathsf{R} \subseteq NR_\Gamma$ *of* contextual relations.

We use the role $\mathsf{mod} \in NR_\Gamma$ defined on $\mathsf{N} \times \mathsf{M}$ to express associations between contexts and modules. Intuitively, modules represent pieces of knowledge specific to a context or context class.[8]

Definition 2 (meta-language)**.** *The* meta-language $\mathcal{L}_\Gamma$ *of a CKR is a DL language over* Γ *with the following syntactic conditions on the application of role restrictions: for every* $\bullet \in \{\forall, \exists, \leq n, \geq n\}$ *and concept C, if* $C = \bullet\,\mathsf{mod}.\mathsf{B}$*, then* $\mathsf{B} = \{\mathsf{m}\}$ *with* $\mathsf{m} \in \mathsf{M}$.

Object Language. The knowledge in contexts of a CKR is expressed via a DL language called *object-language* $\mathcal{L}_\Sigma$ over an object-vocabulary $\Sigma = NC_\Sigma \uplus NR_\Sigma \uplus NI_\Sigma$ akin to Γ. Expressions in $\mathcal{L}_\Sigma$ will be evaluated locally at each context, i.e., each context can interpret each symbol independently. However, sometimes one wants to constrain the meaning of a symbol in a context with the meaning of a symbol in some other context. For instance, if John likes all Indian restaurants in Trento, then the extension of the concept *GoodRestaurant* in the context of John preferences, contains the extension of *IndianRestaurant* in the context of tourism in Trento. To access the interpretation of expressions inside a specific context or context class, we extend the object language as follows.

Definition 3 (object language with *eval*)**.** *The language* $\mathcal{L}^e_\Sigma$ *extends* $\mathcal{L}_\Sigma$ *with* eval expressions

$$eval(X, \mathsf{C}), \tag{4}$$

where X is a concept or role expression of $\mathcal{L}_\Sigma$ *and* C *is a concept expression of* $\mathcal{L}_\Gamma$ *(with* $\mathsf{C} \sqsubseteq \mathsf{Ctx}$*).*

[6]To ease readability, we use sans-serif typeface for elements of the meta-vocabulary.

[7]Intuitively, Ctx will be used to denote the class of all contexts.

[8]Compared to CKRs in [1], for simplicity we do not deal here explicitly with contextual attributes and values. A possible way to reintroduce them would be to fix the interpretation of attribute values as sets of rigidly interpreted elements; then, semantically constrain contextual attributes to range only over their fixed sets of values.

The DL language L^e_Σ extends L_Σ with the set of eval-expressions in L_Σ. Intuitively, the expression *eval*$(C, \{c\})$ represents the extension of the concept C in the context c, and *eval*(C, C), with C a context class, represents the union of the extensions of C in each context c of type C. For *eval*(R, C) with R a role, this is similar.

Example 1. *The example above can be formalized by adding the following axiom to the context of John's preferences:* *eval*(*IndianRestaurant*, {trento_tourism}) ± *GoodRestaurant* Q

We note that nested *eval* expressions are not allowed: every expression occurring inside an *eval* must be an expression in L_Σ. Moreover, as for SROIQ-RL the occurrence of *eval* expressions in axioms and assertions will be syntactically restricted.

Defeasible Axioms. With respect to the initial definition of CKR in [1], we extend the types of axioms that can appear in the global object knowledge with defeasible axioms.

Definition 4 (defeasible axiom). *A* defeasible axiom *is any expression of the form* $\mathrm{D}(\alpha)$, *where* $\alpha \in L_\Sigma$.

Intuitively, $\mathrm{D}(\alpha)$ means that at the level of instantiations for individuals, α is inherited by local contexts unless it generates a contradiction there. In other words, a local exception to α for some individuals is tolerated.

Example 2. *A defeasible global axiom* D(*Concert* ± *Expensive*) *might be used to express that "in general concerts are expensive" and propagate this piece of knowledge to local contexts. At such a context, this might be contradicted by local assertions Concert*(*freeconcert*2016), ¬*Expensive*(*freeconcert*2016) *which "override" the global axiom for* *freeconcert*2016. Q

Example 3. *Note that we want that global defeasible axioms hold globally for all their local instances, but they allow exceptional instances in local contexts. For example, let us say that in the global context we assert that* D(*Horse* ± *Fly*). *However, in the local context greek_myths of Greek mythology, we can assert that this axiom does not hold for the particular instance of the flying horse Pegasus: Horse*(*pegasus*), *Fly*(*pegasus*). *On the other hand, for any other instance of Horse not explicitly violating the axiom, we want to be able to apply the global axiom: for example, if we consider Pedasos, one of Achilles' horses, and assert Horse*(*pedasos*), *we want to be able to derive* ¬*Fly*(*pedasos*). Q

Definition 5 (object language with defeasible axioms). *The DL language* L^{D}_Σ *extends* L_Σ *with the set of defeasible axioms in* L_Σ.

Equipped with the above languages, we are now ready to give our formal definition of Contextualized Knowledge Repository with defeasible axioms.

Definition 6 (contextualized knowledge repository, CKR). *A* contextualized knowledge repository (CKR) *over a meta-vocabulary* Γ *and an object vocabulary* Σ *is a structure*

$$\mathfrak{K} = (\mathfrak{G}, \{K_m\}_{m \in \mathbf{M}})$$

where:

– $\mathfrak{G}$ *is a DL knowledge base over* $L_\Gamma \cup L^{\mathrm{D}}_\Sigma$, *and*

– *every* K_m *is a DL knowledge base over* L^e_Σ *for each module name* $m \in \mathbf{M}$.

Furthermore, K *is a* $\mathcal{SROIQ}$*-RL CKR, if* G *and all* $\mathrm{K_m}$ *are knowledge bases over the extended language of* $\mathcal{SROIQ}$*-RL where eval-expressions can occur only in left-concepts and contain only left-concepts respectively roles.*

In the following, we tacitly focus on $\mathcal{SROIQ}$-RL CKRs.

Example 4. *We introduce an example from the tourism recommendation domain.*[9] *In this scenario, we use a CKR to implement a knowledge base* K_{tour} *that, after being populated with touristic events, locations, organizations, and tourist preferences and profiles, is capable of identifying events that are interesting for a particular tourist (or for a generic tourist class) starting from their preferences. A simplified version of the structure of* K_{tour} *and its contexts is shown in Figure 1. For our example, we focus on sportive events and in particular on volleyball matches.*

– *Intuitively, in the global context* G*, every sport event and tourist is modelled with a context; in the figure, these are depicted as black diamonds and we see some of the oflcial volleyball matches and a tourist. Contexts are grouped by types and organized in hierarchies by means of context classes; in the figure, they are depicted as boxes and we see distinct context class hierarchies for event types (e.g.* SportiveEvent, VolleyMatch*) and for tourists types (e.g.* SportiveTourist*).*

– *The meta-knowledge in* G *associates to contexts and context classes sets of knowledge modules, by axioms of the form* $\mathsf{Event} \sqsubseteq \exists \mathsf{mod}.\{\mathsf{m\_event}\}$ *and* $\{\mathsf{modena\_trento}\} \sqsubseteq \exists \mathsf{mod}.\{\mathsf{m\_match2}\}$*: in the figure these associations are represented by dotted lines to the gray empty diamonds depicting module name individuals.*

– *Knowledge bases associated with modules are depicted as the corresponding gray boxes in the lower part of Figure 1: for example, in* $\mathrm{K}_{\mathsf{v\_match}}$ *we have general axioms about the structure of volleyball matches, while in modules for specific matches as* $\mathrm{K}_{\mathsf{match1}}$ *we store assertions about the actual results of the match. Intuitively, the semantics will enforce a form of inheritance of modules via context class hierarchy.*

– *Contextual relations across events and tourists are depicted as bold arrows in the figure: the only relation* hasParentEvent *connects matches with the competition in which they occur.*

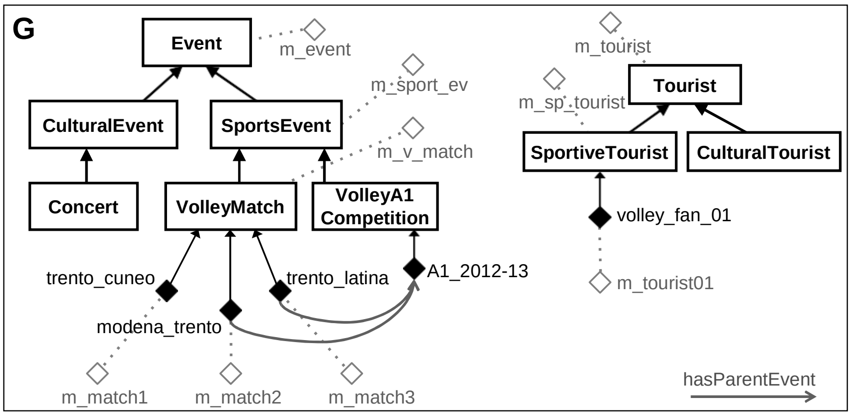
Note that in some of the knowledge modules, we use eval expressions to define references across contexts. For example, in $\mathrm{K}_{\mathsf{sport\_ev}}$ *we can state that "Winners of major volleyball matches are top teams" with:*

$$\mathit{eval}(\mathit{Winner}, \mathit{TopMatch}) \sqsubseteq \mathit{TopTeam}$$

where $\mathit{TopMatch} = \mathsf{VolleyMatch} \sqcap \exists\, \mathsf{hasParentEvent}.\mathsf{VolleyA1Competition}$. *Similarly, in* $\mathrm{K}_{\mathsf{sp\_tourist}}$ *we say that for each sportive tourist "top teams are preferred teams" with the reference axiom* $\mathit{eval}(\mathit{TopTeam}, \mathsf{SportsEvent}) \sqsubseteq \mathit{PreferredTeam}$. Q

Example 5. *We can extend this example with defeasible information: we can assert some general preferences that can be locally overriden by tourist specific assertions. In particular, we want to assert that, in general, all of the Cheap events are Interesting: we can do this using a defeasible axiom in the global context. Furthermore, we propose local markets (market) and football matches*

[9] This example is a simplified version of a real case scenario in which we apply CKR to a tourist event recommendation system in Trentino (see https://dkm.fbk.eu/projects/toolisse).



K_{v_match} *HomeTeam ⊑ Team, HostTeam ⊑ Team, Winner ⊑ Team, Loser ⊑ Team, ...*	K_{match1} *Winner(bre_banca_cuneo_volley), Loser(itas_trentino_volley), ...*
K_{sport_ev} *eval(Winner,* TopMatch*) ⊑ TopTeam*	K_{match2} *Winner(casa_modena_volley), Loser(itas_trentino_volley), ...*
$K_{sp\_tourist}$ *eval(TopTeam,* SportsEvent*) ⊑ PreferredTeam*	K_{match3} *Winner(itas_trentino_volley), Loser(andreoli_latina_volley), ...*

...

G = { CulturalEvent ⊑ Event, SportsEvent ⊑ Event,
Concert ⊑ CulturalEvent, VolleyMatch ⊑ SportsEvent,
VolleyA1Competition ⊑ SportsEvent,
SportiveTourist ⊑ Tourist, CulturalTourist ⊑ Tourist,
VolleyMatch(trento_cuneo), VolleyMatch(modena_trento), VolleyMatch(trento_latina),
VolleyA1Competition(A1_2012-13), hasParentEvent(modena_trento, A1_2012-13),
hasParentEvent(trento_latina, A1_2012-13), SportiveTourist(volley_fan_01),
Event ⊑ ∃mod.{m_event}, SportsEvent ⊑ ∃mod.{m_sport_ev},
VolleyMatch ⊑ ∃mod.{m_v_match},
{trento_cuneo} ⊑ ∃mod.{m_match1}, {modena_trento} ⊑ ∃mod.{m_match2},
{trento_latina} ⊑ ∃mod.{m_match3}, Tourist ⊑ ∃mod.{m_tourist},
SportiveTourist ⊑ ∃mod.{m_sp_tourist}, {volley_fan_01} ⊑ ∃mod.{m_tourist01} }

Figure 1: CKR knowledge base K_{tour} (*TopMatch* = VolleyMatch ⊓ ∃hasParentEvent.VolleyA1Competition))

(fbmatch) as examples of cheap events. However, we want to reflect that tourists interested in cultural events are not interested in a sports event like a football match: we express this by locally negating their interest in f bmatch. Thus, our example CKR K_{tour} *can be extended with the following axioms:*

$$G: \left(\begin{array}{l} \mathrm{D}(Cheap \sqsubseteq Interesting),\ Cheap(fbmatch),\ Cheap(market), \\ \mathrm{mod}(\mathrm{cultural\_tourist}, \mathrm{ctourist\_m}) \end{array} \right)$$

$$K_{\mathrm{ctourist\text{-}m}}: \{ \neg Interesting(fbmatch) \}$$

Note that the negative assertion in the local context represents an exception to the defeasible axiom: we want to recognize this "overriding" for the fbmatch instance, but still apply the defeasible inclusion for market. ◻

Example 6. *Our next example shows how we can represent a form of defeasible propagation of information across local contexts using* **eval** *expressions. We want to represent the information* *about an organization in a CKR, using contexts to represent its situation in different years. We express the rule that every employee working the years before (WorkingBefore) also works in the current year (WorkingNow) by a defeasible inclusion. In the module associated to 2015, we say that alice, bob and charlie were working last year. In the module for 2016, we say (using an* eval *expression) that all of the employees working in 2015 have to be considered in the set of employees working in the past years; moreover, we say that charlie no longer works for the organization. This can be encoded in the CKR* $\mathfrak{K}_{org} = (\mathfrak{G}, \{\mathrm{K}_{\mathsf{em2015\_m}}, \mathrm{K}_{\mathsf{em2016\_m}}\})$, *where*

$$\mathfrak{G}: \left\{ \begin{array}{l} \mathrm{D}(\mathit{WorkingBefore} \sqsubseteq \mathit{WorkingNow}), \\ \mathsf{mod}(\mathsf{employees2015}, \mathsf{em2015\_m}), \mathsf{mod}(\mathsf{employees2016}, \mathsf{em2016\_m}) \end{array} \right\}$$

$$\mathrm{K}_{\mathsf{em2015\_m}}: \{ \mathit{WorkingNow}(\mathit{alice}), \mathit{WorkingNow}(\mathit{bob}), \mathit{WorkingNow}(\mathit{charlie}) \}$$

$$\mathrm{K}_{\mathsf{em2016\_m}}: \left\{ \begin{array}{l} \mathsf{eval}(\mathit{WorkingNow}, \{\mathsf{employees2015}\}) \sqsubseteq \mathit{WorkingBefore}, \\ \neg \mathit{WorkingNow}(\mathit{charlie}) \end{array} \right\}$$

Intuitively, at the local context employees2016, *where* $\mathit{WorkingBefore}(\mathit{charlie})$ *can be derived, the negative assertion* $\neg\mathit{WorkingNow}(\mathit{charlie})$ *should override the instance of the inclusion axiom in the global context for charlie, as it would lead to the opposite, i.e.,* $\mathit{WorkingNow}(\mathit{charlie})$*; on the other hand, for alice and bob no overriding should happen and we can derive that they still work for the organization.* ◊

In the previous example, the overriding of the defeasible axiom is uncontroversial and leads to an intuitive set of conclusions. However, it may be the case that axioms lead to conflicting conclusions; while this results for strict (classical) axioms in inconsistency, for defeasible axioms we still may retain consistency but different sets of conclusions can be appealing, in line with a conflict resolutions

Example 7 (cont'd)**.** *Consider an extension of the CKR in Example 6, where the global knowledge contains a further defeasible axiom* $\mathrm{D}(\mathit{LotteryWinner} \sqsubseteq \neg\mathit{WorkingNow})$ *that states that who wins in the lottery usually does no longer work, and the module* $\mathrm{K}_{\mathsf{em2016\_m}}$ *an additional assertion* $\mathit{LotteryWinner}(\mathit{alice})$*. Then, at the local context* employees2016, *where* $\mathit{WorkingBefore}(\mathit{alice})$ *can be derived, the defeasible axioms in* $\mathfrak{G}$ *lead to the conflicting conclusions* $\mathit{WorkingNow}(\mathit{alice})$ *and* $\neg\mathit{WorkingNow}(\mathit{alice})$*; thus, at least one of the defeasible axiom instances for alice must be overridden if consistency should be maintained.* ◊

The readers familiar with nonmonotonic logics and formalisms will recognize that the situation emerging in the previous example amounts to the classic Nixon diamond scenario, which we shall discuss in more detail in Section 7.4. Accordingly, a solution is to override one of the two defeasible axioms such that we can conclude either $\mathit{WorkingNow}(\mathit{alice})$ or alternatively $\neg\mathit{WorkingNow}(\mathit{alice})$. The semantics of CKRs that we propose has this feature, where assumptions about overriding in models must be reasonably justified; informally, we obtain in Example 7 two classes of models for the CKR, in which $\mathit{WorkingNow}(\mathit{alice})$ resp. $\neg\mathit{WorkingNow}(\mathit{alice})$ is true.

3.2. Semantics

We now define a model-based semantics of CKRs with defeasible axioms, which extends the semantics of CKRs [1] in order to reason with exceptions and their justifications. Intuitively, we model local exceptions of axiom instances by pairs $\langle \alpha, \mathbf{e} \rangle$ of an axiom $\alpha \in \mathcal{L}_\Sigma$ and a tuple $\mathbf{e}$ of individuals in NI_Σ (called *clashing assumptions*): in the evaluation of α at a local context, its instantiation with $\mathbf{e}$ is not considered. However, such assumptions of exceptions must be justified: the instance of α for $\mathbf{e}$ must be unsatisfiable at the local context. This is ensured if assertions can be derived which prove this unsatisfiability; we call such assertions *clashing sets*.

Example 8. *If we consider the concert scenario with the defeasible axiom*

$$\mathrm{D}(Concert \sqsubseteq Expensive),$$

our clashing assumptions on the local context should contain $\langle Concert \sqsubseteq Expensive, freeconcert2016\rangle$; *this clashing assumption is in fact justified by the clashing set*

$$\{Concert(freeconcert2016),\ \neg Expensive(freeconcert2016)\}$$

Models of a CKR will be then CKR interpretations extended with clashing assumptions that are all justified. ◊

We start with a formal definition of CKR interpretations.

Definition 7 (CKR interpretation)**.** *A* CKR interpretation *for* (Γ, Σ) *is a structure* $\mathcal{I} = \langle \mathcal{M}, \mathcal{I} \rangle$ *where*

(i) $\mathcal{M}$ *is a DL interpretation of* $\Gamma \cup \Sigma$ *such that* $c^{\mathcal{M}} \in \mathsf{Ctx}^{\mathcal{M}}$*, for every* $c \in \mathbf{N}$*, and* $C^{\mathcal{M}} \subseteq \mathsf{Ctx}^{\mathcal{M}}$*, for every* $C \in \mathbf{C}$*;*

(ii) for every $x \in \mathsf{Ctx}^{\mathcal{M}}$*,* $\mathcal{I}(x)$ *is a DL interpretation over* Σ *s.t.,* $\Delta^{\mathcal{I}(x)} = \Delta^{\mathcal{M}}$ *and* $a^{\mathcal{I}(x)} = a^{\mathcal{M}}$*, for every* $a \in \mathrm{NI}_{\Sigma}$.

The interpretation of ordinary DL expressions on $\mathcal{M}$ and $\mathcal{I}(x)$ in $\mathcal{I} = \langle \mathcal{M}, \mathcal{I} \rangle$ is as usual (e.g. see Table A.13); *eval* expressions are interpreted as follows: for every $x \in \mathsf{Ctx}^{\mathcal{M}}$,

$$\mathit{eval}(X, \mathrm{C})^{\mathcal{I}(x)} = \bigcup_{e \in \mathrm{C}^{\mathcal{M}}} X^{\mathcal{I}(e)}$$

According to the previous definition, a CKR interpretation is composed by an interpretation for the "upper-layer" $\mathcal{M}$ (which includes the global knowledge and the meta-knowledge) and an interpretation $\mathcal{I}(x)$ of the object language for each instance x of type context (i.e., for all $x \in \mathsf{Ctx}^{\mathcal{M}}$), providing a semantics of the object-vocabulary in x.

We next aim to extend CKR interpretation with exceptions for defeasible axioms. To this end, we need some further notions.

First-order translation. As well-known, $\mathcal{SROIQ}$-RL knowledge bases can be expressed in first-order (FO) logic [26], where every axiom $\alpha \in \mathcal{L}_{\Sigma}$ is translated into an equivalent FO sentence $\forall \mathbf{x}.\varphi_\alpha(\mathbf{x})$ where $\mathbf{x}$ contains all free variables of φ_α depending on the type of the axiom (see below). A reference translation is given in Appendix A.2; notably, the resulting formulas $\varphi_\alpha(\mathbf{x})$ amount semantically to Horn formulas. In fact, each left-side concept C can be expressed by an existential positive FO-formula, and every right-side concept D by a conjunction of Horn clauses.

To contextualize DL-axioms for CKR knowledge bases, the translation is extended with a further argument x_c for the context, such that the formula $\forall \mathbf{x}.\varphi_\alpha(\mathbf{x}, x_c)$ expresses the axiom α within context x_c; in particular, for any context name c the sentence $\forall \mathbf{x}.\varphi_\alpha(\mathbf{x}, \mathrm{c})$ expressed α within c. Furthermore this translation is easily extended to $\mathcal{L}^e_\Sigma$ such that the Horn property is maintained for $\mathcal{SROIQ}$-RL, due to the restrictions on the form and occurrence of *eval* expressions; the presence of *eval* expressions requires the contextualized form. We note the following property.

Lemma 1. *For any DL knowledge base* $\mathcal{K}$ *over* $\mathcal{L}_\Gamma$ *resp.* $\mathcal{L}^e_\Sigma$ *its FO-translation (resp. its contextualized FO translation)*

$$\varphi_{\mathcal{K}} := \bigwedge_{\alpha \in \mathcal{K}} \forall \mathbf{x} \varphi_\alpha(\mathbf{x}) \quad (\textit{resp.} \quad \varphi_{\mathcal{K}, x_c} := \bigwedge_{\alpha \in \mathcal{K}} \forall \mathbf{x} \varphi_\alpha(\mathbf{x}, x_c)\ ) \tag{5}$$

is semantically equivalent to a conjunction of universal Horn clauses.

Notably this lemma remains valid under the SNA, as the axioms defining a congruence relation $\approx$ are Horn clauses.

We now formally introduce the notion of instantiation of an axiom. This notion forms the basis for defining exceptions to axioms in terms of clashing assumptions about violated instances, which have to be evidenced by clashing sets that, in our formalization, are provable from the knowledge base.

Definition 8 (axiom instantiation). *Given an axiom* $\alpha \in \mathcal{L}_\Sigma$ *with FO-translation* $\forall \mathbf{x}.\varphi_\alpha(\mathbf{x})$, *the instantiation of* α *with a tuple* $\mathbf{e}$ *of individuals in* NI_Σ, *written* $\alpha(\mathbf{e})$, *is the specialization of* α *to* $\mathbf{e}$, *i.e.*, $\varphi_\alpha(\mathbf{e})$, *depending on the type of* α.

In particular, $\mathbf{e}$ is (i) void for assertions α, (ii) a single element e for GCIs α, and (iii) a pair e_1, e_2 of elements for role axioms α.

Definition 9 (clashing assumptions and sets). *A* clashing assumption *is a pair* $\langle \alpha, \mathbf{e} \rangle$ *such that* $\alpha(\mathbf{e})$ *is an axiom instantiation for an axiom* $\alpha \in \mathcal{L}_\Sigma$. *A* clashing set *for a clashing assumption* $\langle \alpha, \mathbf{e} \rangle$ *is a satisfiable set S of ABox assertions over* $\mathcal{L}_\Sigma$ *such that* $S \cup \{\alpha(\mathbf{e})\}$ *is unsatisfiable.*

Intuitively, a clashing assumption $\langle \alpha, \mathbf{e} \rangle$ represents that $\alpha(\mathbf{e})$ is not (DL-)satisfiable, and a clashing set S provides an assertional "justification" for the assumption of local overriding of α on $\mathbf{e}$.

Example 9. *For example, the clashing assumption* $\langle A \pm B, a \rangle$ *has* $\{A(a), \neg B(a)\}$ *as a clashing set, and* $\langle A \sqcap B \pm C, a \rangle$ *has* $\{A(a), B(a), \neg C(a)\}$. *Furthermore,* $\langle A \pm \exists R.\{a\}, b \rangle$ *has the clashing set* $\{A(b), \neg R(b, a)\}$, *and* $\langle A \pm \leq 1 R.B, a \rangle$ *has* $\{A(a), R(a, a), R(a, b), B(a), B(b), \neg A(b)\}$, *for instance. In each case the clashing set S is minimal in that no proper subset $S' \subset S$ is a clashing set; multiple minimal clashing sets may exist (e.g.,* $\{A(a), R(a, a), R(a, b), B(a), B(b), C(a), \neg C(b)\}$ *would be another minimal clashing set for* $\langle A \pm \leq 1 R.B, a \rangle$*).* Q

We remark that this notion of "assertional justification" is directly connected with the datalog translation in Section 5: it corresponds to the provability of an assertional condition stating the inconsistency of the inherited axiom. By the Horn nature of $\mathcal{SROIQ}$-RL, such a "constructive" justification can always be found.

Proposition 2. *Let* $\langle \alpha, \mathbf{e} \rangle$ *be a clashing assumption where* α *is a* $\mathcal{SROIQ}$*-RL axiom. If* $\alpha(\mathbf{e})$ *is not valid (i.e.,* $\neg\varphi_\alpha(\mathbf{e})$ *is satisfiable), then a clashing set S for* $\langle \alpha, \mathbf{e} \rangle$ *exists and each concept assertion in S is of the form $A(a)$ resp. $\neg A(a)$, and* $A \in \mathrm{NC}$. *Furthermore, every non-redundant (i.e. $\subseteq$-minimal) such set S has size linear in the size of* α.

3.2.1. CAS-models

We then extend CKR interpretations to CAS-interpretations that take clashing assumptions into account as follows.

Definition 10 (CAS-interpretation). *A* CAS-interpretation *is a structure* $\mathfrak{I}_{CAS} = \langle \mathfrak{M}, \mathcal{I}, \chi \rangle$ *where* $\mathfrak{I} = \langle \mathfrak{M}, \mathcal{I} \rangle$ *is a CKR interpretation and* χ *maps every* $x \in \Delta^{\mathfrak{M}}$ *to a set* $\chi(x)$ *of clashing assumptions for x.*

Intuitively, a CAS-interpretation pairs a usual CKR interpretation with an exception set for each local context.

What remains then is to define satisfaction of axioms on CKR- resp. CAS-interpretations, and to single out appropriate models of a given CKR $\mathfrak{K}$. To achieve this, we extend the definition of CKR models from [1] by introducing the condition to disregard "exceptional elements" asserted by clashing assumptions in $\chi(x)$ in the local interpretation of their defeasible axioms, leading to CAS-models. However, such models have arbitrary exceptions, while we are interested in justifiable exceptions; this will be captured by the notion of *justified* CAS-models.

For convenience, we call two DL interpretations $\mathcal{I}_1$ and $\mathcal{I}_2$ (resp. CAS-interpretations $\mathcal{I}^i_{CAS} = (\mathfrak{M}_i, \mathcal{I}_i, \chi_i)$, $i \in \{1, 2\}$) NI-*congruent*, if $c^{\mathcal{I}_1} = c^{\mathcal{I}_2}$ (resp. $c^{\mathfrak{M}_1} = c^{\mathfrak{M}_2}$) holds for every $c \in \mathrm{NI}$.

Definition 11 (CAS-model). *Given a CKR* $\mathfrak{K} = (\mathfrak{G}, \{K_m\}_{m \in \mathbf{M}})$, *a CAS-interpretation* $\mathcal{I}_{CAS} = (\mathfrak{M}, \mathcal{I}, \chi)$ *is a CAS-model for* $\mathfrak{K}$ *(denoted* $\mathcal{I}_{CAS} \models \mathfrak{K}$*), if the following holds:*

(i) for every $\alpha \in \mathcal{L}_\Sigma \cup \mathcal{L}_\Gamma$ *in* $\mathfrak{G}$, $\mathfrak{M} \models \alpha$*;*

(ii) for every $\mathrm{D}(\alpha) \in \mathfrak{G}$ *(where* $\alpha \in \mathcal{L}_\Sigma$*),* $\mathfrak{M} \models \alpha$*;*

(iii) for every $(x, y) \in \mathsf{mod}^{\mathfrak{M}}$ *such that* $y = \mathsf{m}^{\mathfrak{M}}$, $\mathcal{I}(x) \models K_\mathsf{m}$*;*

(iv) for every $\alpha \in \mathfrak{G} \cap \mathcal{L}_\Sigma$ *and* $x \in \mathsf{Ctx}^{\mathfrak{M}}$, $\mathcal{I}(x) \models \alpha$, *and*

(v) for every $\mathrm{D}(\alpha) \in \mathfrak{G}$ *(where* $\alpha \in \mathcal{L}_\Sigma$*),* $x \in \mathsf{Ctx}^{\mathfrak{M}}$, *and* $|\mathbf{x}|$*-tuple* $\mathbf{d}$ *of elements in* NI_Σ *such that* $\mathbf{d} \notin \{\mathbf{e} \mid (\alpha, \mathbf{e}) \in \chi(x)\}$, *we have* $\mathcal{I}(x) \models \phi_\alpha(\mathbf{d})$.

In the previous definition, conditions (i) and (ii) verify that the global interpretation $\mathfrak{M}$ satisfies the (strict and defeasible) axioms in $\mathfrak{G}$. Condition (iii) states that each local interpretation $\mathcal{I}(x)$ satisfies all local modules K_m that are associated with context x. Moreover, by condition (iv), all strict axioms from global object knowledge in $\mathfrak{G}$ need to be satisfied in local interpretations. The local interpretation of defeasible axioms is defined by condition (v): for every global defeasible axiom $\mathrm{D}(\alpha)$ and instantiation $\alpha(\mathbf{d})$ of it $\mathcal{I}(x)$ must satisfy $\alpha(\mathbf{d})$ if $\langle \alpha, \mathbf{d} \rangle \notin \chi(x)$, i.e., $\alpha(\mathbf{d})$ is not an exceptional instantiation.

As for condition (iii), we note that contexts and module names are not necessarily interpreted as the same objects, and that a module can be shared by more contexts: the fact that the piece of knowledge identified by m should be included in the knowledge of context c is provided by the truth of the role assertion $\mathsf{mod}(\mathsf{c}, \mathsf{m})$. Thus, it is not the case that if a context c belongs to a context class C then its modules belong to C too: on the other hand, if we write $\mathsf{C} \sqsubseteq \exists \mathsf{mod}.\{\mathsf{m}\}$, we can state that all contexts in C share the same module m (e.g. defining the common features of elements in C).

We can express the CAS-models of a CKR $\mathfrak{K}$ using an extended FO translation, in which the clashing assumption χ is represented by predicates $app_\alpha(\mathbf{x}, x_c)$ which informally state that in context x_c, the axiom $\alpha = \forall \mathbf{x}.\phi_\alpha(\mathbf{x})$ has for $\mathbf{x}$ not an exception, i.e., α applies on $\mathbf{x}$ (cf. Appendix). More in detail, let us call

$$\phi^{\mathsf{Ctx}}_\alpha(\mathbf{x}, x_c) = Ctx(x_c) \rightarrow \phi_\alpha(\mathbf{x}, x_c), \qquad \alpha \in \mathcal{L}_\Sigma \tag{6}$$

$$\phi^{CAS}_\alpha(\mathbf{x}, x_c) = Ctx(x_c) \wedge app_\alpha(\mathbf{x}, x_c) \rightarrow \phi_\alpha(\mathbf{x}, x_c), \qquad \alpha \in \mathcal{L}^e_\Sigma, \tag{7}$$

the *context-constraint* resp. *clashing-constraint translation* of α; then the sentence

$$\phi_{\mathfrak{K}} = \begin{array}{l} \bigwedge_{\alpha \in \mathfrak{G}_{\Sigma,\Gamma}} \forall \mathbf{x}.\phi_\alpha(\mathbf{x}) \wedge \bigwedge_{\mathsf{m} \in \mathbf{M}} \forall x_c.\ \mathsf{mod}(x_c, \mathsf{m}) \rightarrow \phi_{K_\mathsf{m}, x_c}(x_c) \wedge \\ \bigwedge_{\alpha \in \mathfrak{G} \cap \mathcal{L}_\Sigma} \forall x_c.\phi^{\mathsf{Ctx}}_\alpha(\mathbf{x}, x_c) \wedge \bigwedge_{D(\alpha) \in \mathfrak{G}} \forall x_c.\phi^{CAS}_\alpha(\mathbf{x}, x_c) \end{array} \tag{8}$$

where $\mathfrak{G}_{\Sigma,\Gamma} = \mathfrak{G} \cap (\mathcal{L}_\Sigma \cup \mathcal{L}_\Gamma) \cup \{\alpha \mid D(\alpha) \in \mathfrak{G}\}$, expresses the CAS-models of $\mathfrak{K}$, i.e., those *CAS*-interpretations which are *CAS*-models relative to the represented clashing assumption. Clearly, $\phi_{\mathfrak{K}}$ amounts semantically to a Horn sentence.

Example 10. *We can now provide an example of CAS model for the CKR* $\mathfrak{K}_{tourD}$ *from Example 5.*

We can consider the model $\mathfrak{I}_{CAS_{t1}} = \langle \mathfrak{M}, \mathfrak{I}, \chi_{t1} \rangle$ *such that*

$$\chi_{t1}(\mathsf{cultural\text{-}tourist}^{\mathfrak{M}}) = \{\langle Cheap \sqsubseteq Interesting, \{fbmatch\}\rangle\}$$

In this case we have the intuitive interpretation where $\mathcal{I}(\mathsf{cultural\text{-}tourist}^{\mathfrak{M}}) \models Interesting(fbmatch)$. *However, it is not the only legitimate CAS model for* $\mathfrak{K}_{tourD}$*: we can also consider the model* $\mathfrak{I}_{CAS_{t2}} = \langle \mathfrak{M}, \mathfrak{I}, \chi_{t2} \rangle$ *where:*

$$\chi_{t2}(\mathsf{cultural\text{-}tourist}^{\mathfrak{M}}) = \{\langle Cheap \sqsubseteq Interesting, fbmatch\rangle, \langle Cheap \sqsubseteq Interesting, market\rangle\}$$

In this case, also the individual market is considered as exceptional and it holds that $\mathcal{I}(\mathsf{cultural\text{-}tourist}^{\mathfrak{M}}) \not\models Interesting(market)$. *We will see in the following how to limit the models we consider only to the exceptional cases that are actually motivated by the contents of the CKR at hand.* ◇

Justification. A clashing assumption allows us to dispense the application of an axiom in a context. However, for this to happen, we should have a good reason; an exception should be made only if needed. To reflect this, we say that a clashing assumption $\langle \alpha, \mathbf{e} \rangle \in \chi(x)$ is *justified* for a *CAS* model $\mathfrak{I}_{CAS} = \langle \mathfrak{M}, \mathfrak{I}, \chi \rangle$, if some clashing set $S = S_{\langle \alpha, \mathbf{e}\rangle, x}$ exists such that, for every CAS-model $\mathfrak{I}'_{CAS} = \langle \mathfrak{M}', \mathfrak{I}', \chi \rangle$ of $\mathfrak{K}$ that is NI-congruent with $\mathfrak{I}_{CAS}$, it holds that $\mathcal{I}'(x) \models S_{\langle \alpha, \mathbf{e}\rangle, x}$. Informally, justification requires that we have factual evidence that an instantiation of an axiom can not be satisfied, and this evidence is provable. This leads us to justified CAS-models, from which we obtain CKR-models of $\mathfrak{K}$ by stripping off the clashing part.

Definition 12 (justified CAS model and CKR model)**.** *A CAS model* $\mathfrak{I}_{CAS} = \langle \mathfrak{M}, \mathfrak{I}, \chi \rangle$ *of a CKR* $\mathfrak{K}$ *is* justified, *if every* $\langle \alpha, \mathbf{e} \rangle \in \bigcup_{x \in \mathsf{Ctx}^{\mathfrak{M}}} \chi(x)$ *is justified. An interpretation* $\mathfrak{I} = \langle \mathfrak{M}, \mathfrak{I} \rangle$ *is a* CKR model *of* $\mathfrak{K}$ *(in symbols,* $\mathfrak{I} \models \mathfrak{K}$*), if* $\mathfrak{K}$ *has some justified CAS model* $\mathfrak{I}_{CAS} = \langle \mathfrak{M}, \mathfrak{I}, \chi \rangle$.

Example 11. *Let us reconsider the* $\mathfrak{K}_{tour}$ *CKR of Example 4 and provide a formal interpretation for it. Let* $\mathfrak{I} = \langle \mathfrak{M}, \mathfrak{I} \rangle$ *be the CKR model of* $\mathfrak{K}_{tour}$ *directly induced by its assertions. Note that, since* $\mathfrak{K}_{tour}$ *does not include defeasible axioms, the model can be defined independently from the choice of a clashing assumption map* χ*, as presented in the analogous example in [1]. By the definition of the model, we can now find the knowledge base associated to each context: for example, for the context of the match* modena_trento*, we have that*

$$\mathcal{I}(\mathsf{modena\text{-}trento}^{\mathfrak{M}}) \models K_{\mathsf{event}} \cup K_{\mathsf{sport\text{-}ev}} \cup K_{\mathsf{v\text{-}match}} \cup K_{\mathsf{match2}}$$

and similarly for the other matches. In particular, we have that the local interpretation satisfies the axiom $\mathit{eval}(Winner, \mathsf{TopMatch}) \sqsubseteq TopTeam$*, which is included in* $K_{\mathsf{sport\_ev}}$*. The formal reading of this axiom is as follows. We have that*

$$\begin{aligned} \mathit{eval}(Winner, \mathsf{TopMatch})^{\mathcal{I}(\mathsf{modena\text{-}trento}^{\mathfrak{M}})} &= \bigcup_{e \in \mathsf{TopMatch}^{\mathfrak{M}}} Winner^{\mathcal{I}(e)} \\ &= \bigcup_{e \in \{\mathsf{modena\text{-}trento},\ \mathsf{trento\text{-}latina}\}} Winner^{\mathcal{I}(e)}, \end{aligned}$$

where the last line follows from the assertions in the ABox of $\mathfrak{G}$*. Now, by assertions on Winner inside* K_{match2} *and* K_{match3}*, we obtain*

$$\{itas\text{-}trentino, casa\text{-}modena\} \subseteq TopTeam^{\mathcal{I}(\mathsf{modena\text{-}trento}^{\mathfrak{M}})}$$

For the context describing the tourist volley_fan_01, *we can reason similarly. We have that*

$$\mathcal{I}(\text{volley-fan-01}^{\mathfrak{M}}) \models K_{\text{tourist}} \cup K_{\text{sp-tourist}} \cup K_{\text{tourist01}}$$

Thus, the interpretation satisfies $\mathit{eval}(\mathit{TopTeam}, \text{SportsEvent}) \sqsubseteq \mathit{PreferredTeam}$ *from* $K_{\text{sp-tourist}}$. *As in the case above,* $\mathit{eval}(\mathit{TopTeam}, \text{SportsEvent})^{\mathcal{I}(\text{volley-fan-01}^{\mathfrak{M}})}$ *is interpreted as*

$$\bigcup_{e \in \text{SportsEvent}^{\mathfrak{M}}} \mathit{TopTeam}^{\mathcal{I}(e)} = \bigcup_{e \in \{\text{trento-cuneo, modena-trento, trento-latina}\}} \mathit{TopTeam}^{\mathcal{I}(e)},$$

by the assertions in $\mathfrak{G}$. *Finally, from the reference axiom above, we obtain*

$$\{\mathit{itas\text{-}trentino}, \mathit{casa\text{-}modena}\} \subseteq \mathit{PreferredTeam}^{\mathcal{I}(\text{volley-fan-01}^{\mathfrak{M}})}$$ ◊

Example 12. *We can now show an example of CKR models satisfying the CKRs presented in Examples 5 and 6. In the case of* $\mathfrak{K}_{tourD}$, *let us consider the model* $\mathfrak{I}_{CAS_{t1}} = \langle \mathfrak{M}, \mathcal{I}, \chi \rangle$ *of Example 10. We note that this interpretation is justified as it is easy to check that*

$$\mathcal{I}(\text{cultural-tourist}^{\mathfrak{M}}) \models \{\mathit{Cheap}(\mathit{fbmatch}), \neg\mathit{Interesting}(\mathit{fbmatch})\}$$

that represents a clashing set for the defeasible axiom. On the other hand, the CAS model $\mathfrak{I}_{CAS_{t2}}$ *is not justified: indeed, in the case of the individual market we can not find a clashing set for the respective clashing assumption, since* $\mathcal{I}(\text{cultural-tourist}^{\mathfrak{M}}) \models \neg\mathit{Interesting}(\mathit{market})$.

In the case of $\mathfrak{K}_{org}$, *we have that the model* $\mathfrak{I}_{CAS_{org}} = \langle \mathfrak{M}, \mathcal{I}, \chi_{org} \rangle$ *with*

$$\chi_{org}(\text{employees2016}^{\mathfrak{M}}) = \{\langle \mathit{WorkingBefore} \sqsubseteq \mathit{WorkingNow}, \{\mathit{charlie}\}\rangle\}$$

is a CKR model for the example CKR. For the interpretation of eval *expressions, in every interpretation of* K_{tour} *we have that* $\{\mathit{alice}^{\mathcal{I}(x)}, \mathit{bob}^{\mathcal{I}(x)}, \mathit{charlie}^{\mathcal{I}(x)}\} \subseteq \mathit{WorkingBefore}^{\mathcal{I}(x)}$, *where* x = employees2016. *Thus the justification of the model can be easily seen as*

$$\mathcal{I}(\text{employees2016}^{\mathfrak{M}}) \models S \ \text{ for } S = \{\mathit{WorkingBefore}(\mathit{charlie}), \neg\mathit{WorkingNow}(\mathit{charlie})\}$$

which represents a clashing set for the defeasible axiom on charlie. ◊

Different from arbitrary CAS-models, a characterization of justified *CAS*-models by a FO translation (even less into Horn formulas) is not straightforward; furthermore, no modular translation can exist, due to the inherent non-monotonicity of exceptions (see below).

3.3. Semantic properties

It appears that CAS-models and in particular justified CAS-models (thus CKR-models), have interesting properties.

Irrelevance of syntax. Straight from the definition is the property that the syntactic form of an axiom with exceptions is not important. That is,

Proposition 3 (irrelevance of syntax). *Suppose* $\mathfrak{K} = \langle \mathfrak{G}, \{K_m\}_{m \in \mathbf{M}} \rangle$ *has in* $\mathfrak{G}$ *a defeasible axiom* $D(\alpha)$. *If* $\beta \in \mathcal{L}_\Sigma$ *satisfies* $\varphi_\alpha(\mathbf{x}) \equiv \varphi_\beta(\mathbf{x})$ *(i.e.,* β *is of the same genus and logically equivalent to* α*), then* $\mathfrak{K}$ *and* $\mathfrak{K}' = \langle (\mathfrak{G} \setminus \alpha) \cup \{\beta\}, \{K_m\}_{m \in \mathbf{M}} \rangle$ *have the same CKR-models.*

Note that Proposition 3 does not hold for arbitrary CAS-models, as clashing assumptions are syntactically defined; however, the sets of CAS-models correspond under exchange of α and β there.

Nonmonotonicity. As expected, justified CAS-models behave nonmonotonically, in the following sense. Let us write $\mathfrak{K} \subseteq \mathfrak{K}'$ for $\mathfrak{K} = (\mathfrak{G}, \{K_m\}_{m \in \mathbf{M}})$ and $\mathfrak{K}' = (\mathfrak{G}', \{K'_m\}_{m \in \mathbf{M}})$, if $\mathfrak{G} \subseteq \mathfrak{G}'$ and $K'_m \subseteq K_m$, for all $m \in \mathbf{M}$.

Proposition 4 (non-monotonicity). *Suppose* $\mathbf{I}_{CAS} = (\mathcal{M}, \mathcal{I}, \chi)$ *is a justified CAS-model of a CKR* $\mathfrak{K}'$. *Then* $\mathbf{I}_{CAS}$ *is not necessarily a justified CAS-model of every* $\mathfrak{K} \subseteq \mathfrak{K}'$.

For example, if $\mathfrak{G}'$ consists of $D(A(c))$ and a context c with an associated module K'_m consisting of $\neg A(c)$, then $\neg A(c)$ is true at c in the justified CAS-model of $\mathfrak{K}'$, thanks to the justified clashing assumption $\langle A(c), \mathsf{s}\rangle$; if we remove $\neg A(c)$, then $\neg A(c)$ is false in the justified CAS-model of $\mathfrak{K}$, as the clashing assumption $\langle A(c), \mathsf{s}\rangle$ is no longer justified and must be dropped.

Context focus. A further simple property is that in CAS-models, only the clashing assumptions for contexts matter. Formally,

Proposition 5 (context focus). *Suppose* $\mathbf{I}_{CAS} = (\mathcal{M}, \mathcal{I}, \chi) \models \mathfrak{K}$ *for a CAS-interpretation of a CKR* $\mathfrak{K}$ *and that* χ' *coincides with* χ *on* $\mathsf{Ctx}^{\mathcal{M}}$. *Then* $\mathbf{I}'_{CAS} = (\mathcal{M}, \mathcal{I}, \chi') \models \mathfrak{K}$. *Furthermore, if* $\mathbf{I}_{CAS}$ *is justified, then also* $\mathbf{I}'_{CAS}$ *is justified.*

That is, if we consider a justified *CAS* model $\mathbf{I}_{CAS}$, any other *CAS*-interpretation $\mathbf{I}'_{CAS}$ that differs from *CAS* only on the clashing assumptions of elements *not* in $\mathsf{Ctx}^{\mathcal{M}}$ (i.e. non-context individuals) is also a justified model. Thus, clashing assumptions can be safely assumed to be void for non-context individuals.

Minimality of justification. In case of justified CAS-models, the clashing assumptions associated with the contexts are minimal in the sense that no assumption can be omitted. This follows from the property that the clashing assumptions must be setwise incomparable.

Proposition 6 (minimality of justification). *Suppose that* $\mathbf{I}_{CAS} = (\mathcal{M}, \mathcal{I}, \chi)$ *and* $\mathbf{I}'_{CAS} = (\mathcal{M}', \mathcal{I}', \chi')$ *are justified CAS-models of a CKR* $\mathfrak{K}$ *that are* NI*-congruent. Then,* $\mathsf{Ctx}^{\mathcal{M}} = \mathsf{Ctx}^{\mathcal{M}'}$ *and* $\chi'(x) \subseteq \chi(x)$ *for every* $x \in \mathsf{Ctx}^{\mathcal{M}}$ *implies* $\chi = \chi'$.

As a consequence, exceptions in CKR models are *minimally justified* in this sense; notably, this minimality condition is intrinsic and not explicitly part of the definition.

Intersection property and least model. Another property is that CAS-models of a $\mathcal{SROIQ}$-RL CKR enjoy an intersection property; this is due to the fact that the global and the local knowledge bases of a CKR amount to Horn theories, which as it is well-known have the intersection property.

Formally, for two NI-congruent DL interpretations $\mathcal{I}_1$ and $\mathcal{I}_2$, we denote by $\mathcal{I}_1 \cap \mathcal{I}_2$ the NI-congruent interpretation such that $C^{\mathcal{I}_1 \cap \mathcal{I}_2} = C_1^{\mathcal{I}} \cap C_2^{\mathcal{I}}$ and $R^{\mathcal{I}_1 \cap \mathcal{I}_2} = R_1^{\mathcal{I}} \cap R_2^{\mathcal{I}}$ for all $C \in$ NC and $R \in$ NR, respectively. Then:

Proposition 7 (intersection property). *Let* $\mathbf{I}^i_{CAS} = (\mathcal{M}_i, \mathcal{I}_i, \chi)$, $i \in \{1, 2\}$ *be* NI*-congruent CAS-models of a CKR* $\mathfrak{K}$. *Then* $\mathbf{I}_{CAS} = (\mathcal{M}, \mathcal{I}, \chi)$ *where* $\mathcal{M} = \mathcal{M}_1 \cap \mathcal{M}_2$ *and* $\mathcal{I} = \mathcal{I}_1 \cap \mathcal{I}_2$ *is the intersection of the* $\mathcal{M}_i$ *resp.* $\mathcal{I}_i$, *is also a CAS-model of* $\mathfrak{K}$. *Furthermore,* $\mathbf{I}_{CAS}$ *is justified if some* $\mathbf{I}^i_{CAS}$ *is justified,* $i \in \{1, 2\}$.

An immediate consequence of this result is that a least (justified) CAS-model exists. Technically, let a *name assignment* be any interpretation $\nu : \mathrm{NI} \to \Delta$ of the individual constants on the domain Δ (respecting SNA); the name assignment of a DL interpretation $\mathcal{I}$ (resp. CAS-interpretation $\mathbf{I}_{CAS} = (\mathcal{M}, \mathcal{I}, \chi)$) is the one induced by $\mathrm{NI}^{\mathcal{I}}$ (resp. $\mathrm{NI}^{\mathcal{M}}$). We call a clashing assumption *CAS* for a CKR $\mathfrak{K}$ *satisfiable* (resp., *justified*) for a name assignment ν, if $\mathfrak{K}$ has some CAS-model (resp., justified CAS-model) $\mathfrak{I}_{CAS}$ with name assignment ν. Then:

Corollary 1 (least model property). *If a clashing assumption χ for a CKR $\mathfrak{K}$ is satisfiable for name assignment ν, then $\mathfrak{K}$ has a least (unique minimal) CAS-model $\hat{\mathfrak{I}}_{\mathfrak{K}}(\chi,\nu) = (\hat{\mathfrak{M}}, \hat{\mathcal{I}}, \chi)$ w.r.t. inclusion $\mathfrak{M}' \subseteq \mathfrak{M}$ and $\mathcal{I}' \subseteq \mathcal{I}$ for ν. Furthermore, $\hat{\mathfrak{I}}_{\mathfrak{K}}(\chi,\nu)$ is justified if χ is justified.*

Named model focus. An important property concerns the scope of an interpretation. For $\mathcal{SROIQ}$-RL DL knowledge bases $\mathcal{K}$, and likewise for $\mathcal{SROIQ}$-RL CKRs $\mathfrak{K}$, we can focus on the *named* part of a DL interpretation $\mathcal{I}$ resp. a CAS-interpretation $\mathfrak{I}_{CAS} = (\mathfrak{M}, \mathcal{I}, \chi)$. We say $\mathcal{I}$ is *named* relative to $N \subseteq \mathrm{NI} \setminus \mathrm{NI}_S$, if $C^{\mathcal{I}} \subseteq N^{\mathcal{I}}$ and $R^{\mathcal{I}} \subseteq N^{\mathcal{I}} \times N^{\mathcal{I}}$ for each $C \in \mathrm{NC}$ and $R \in \mathrm{NR}$; if in addition $c^{\mathcal{I}} \neq d^{\mathcal{I}}$ for any distinct $c, d \in N$ and N includes all constants that occur in $\mathcal{K}$, we call $\mathcal{I}$ a *pseudo Herbrand interpretation* for $\mathcal{K}$ relative to N.[10] The following lemma is then not hard to establish. For convenience, let for any $N \subseteq \mathrm{NI} \setminus \mathrm{NI}_S$ be the *N-restriction of* $\mathcal{I}$, denoted by $\mathcal{I}^N$, the interpretation that results from $\mathcal{I}$ by restricting $C^{\mathcal{I}}$ to $N^{\mathcal{I}}$ for all $C \in \mathrm{NC}$ and $R^{\mathcal{I}}$ to $N^{\mathcal{I}} \times N^{\mathcal{I}}$ for every $R \in \mathrm{NR}$.

Lemma 2. *Suppose $\mathcal{I}$ is a model of a $\mathcal{SROIQ}$-RL knowledge base $\mathcal{K}$ and $N \subseteq \mathrm{NI} \setminus \mathrm{NI}_S$ includes all individuals occurring in $\mathcal{K}$. Then the N-restriction $\mathcal{I}^N$ is named w.r.t. N and a model of $\mathcal{K}$.*

In essence, we have model preservation under restriction to $N^{\mathcal{I}}$ (technically, because of the standard names we need to keep the whole domain).

This property extends to CAS-interpretations $\mathfrak{I}_{CAS}(\mathfrak{M}, \mathcal{I}, \chi)$ of CKRs $\mathfrak{K}$. Given that N includes each individual constant that occurs in $\mathfrak{K}$, a CAS-interpretation $\mathfrak{I}^N_{CAS}$ results from $\mathfrak{I}_{CAS}$ by (i) replacing $\mathfrak{M}$ and each $\mathcal{I}(x)$ with its N-restriction, (ii) removing each clashing assumption $\langle\alpha, \mathbf{d}\rangle$ from χ where $\mathbf{d}$ is not over $N \cup N^{\mathfrak{M}}$, and (iii) interpreting each constant symbol $c \in \mathrm{NI} \setminus (N \cup \mathrm{NI}_S)$ such that $c^{\mathfrak{M}} \in N^{\mathfrak{M}}$ (resp. $c^{\mathcal{I}(x)} \in N^{\mathcal{I}(x)}$) its interpretation $c^{\mathfrak{M}^N}$ (resp. $c^{\mathcal{I}(x)^N}$) by some arbitrary element not in $N^{\mathfrak{M}}$.[11] In particular, we write $N_{\mathfrak{K}}$ for N if the latter consists precisely of the individual constants that occur in $\mathfrak{K}$. Then we obtain:

Theorem 1 (named model focus). *Let $\mathfrak{I}_{CAS}$ be a CAS-model of $\mathfrak{K}$ and suppose $N_{\mathfrak{K}} \subseteq N \subseteq \mathrm{NI} \setminus \mathrm{NI}_S$. Then, also $\mathfrak{I}^N_{CAS}$, and in particular $\mathfrak{I}^{N_{\mathfrak{K}}}_{CAS}$, is a CAS-model for $\mathfrak{K}$. Furthermore, $\mathfrak{I}^N_{CAS}$ is justified if $\mathfrak{I}_{CAS}$ is justified, and every clashing assumption $\langle\alpha, \mathbf{e}\rangle$ in $\mathfrak{I}^N_{CAS}$ is justified by some clashing set S formulated with constants from N.*

Based on this, we can restrict query answering, which we will turn to next, to named CKR-models. This property is crucial for the datalog translation that we shall present in Section 5.

4. Reasoning and Complexity

In this section, we consider reasoning from CKRs: to this end, we first define entailment of axioms from a CKR, and then we proceed to define conjunctive queries over a CKR. After that, we characterize the computational complexity of elementary reasoning tasks and query answering from CKRs.

---

[10] Conceptually, we obtain a traditional Herbrand interpretation if we identify each $c \in N$ with the standard name $c^{\mathcal{I}}$ and dismiss all other standard names and restrict $\Delta^{\mathcal{I}}$ to $N^{\mathcal{I}}$. Technically, to stick with the infinitely many standard names NI_S, we leave the domain in $\mathcal{I}^N$ unchanged.

[11] Alternatively, if we are allowed to change NI we could simply remove all such c.

4.1. CKR Entailment

Based on CKR-models, we define notions of context and global entailment of axioms from a CKR as follows.

Definition 13 (c-entailment, global entailment). *Assume a CKR* K *over* (Γ, Σ) *and* $\mathsf{c} \in N_{\mathsf{K}}$*. An axiom* $\alpha \in \mathsf{L}^e_\Sigma$ *is* c*-entailed by* K*, denoted* $\mathsf{K} \models \mathsf{c} : \alpha$*, if* $\mathcal{I}(\mathsf{c}^{\mathsf{M}}) \models \alpha$ *for every CKR-model* $\mathfrak{I} = (\mathsf{M}, \mathcal{I})$ *of* K*. Furthermore, an axiom* α *is (globally) entailed by* K*, denoted* $\mathsf{K} \models \alpha$*, if*

$$\begin{cases} \mathsf{K} \models \mathsf{c} : \alpha \text{ for every } \mathsf{c} \in \mathbf{N}, & \text{if } \alpha \in \mathsf{L}^e_\Sigma, \\ \mathsf{M} \models \alpha \text{ for every CKR-model } \mathfrak{I} = (\mathsf{M}, \mathcal{I}) \text{ of } \mathsf{K}, & \text{if } \alpha \in \mathsf{L}_\Gamma. \end{cases}$$

Example 13. *Considering the CKR* K_{tourD} *and the model* $\mathfrak{I}_{CAS_{t1}}$ *from Example 12, we have that*

$$\mathsf{K}_{tourD} \models \mathsf{cultural\text{-}tourist} : \neg Interesting(fbmatch)$$

On the other hand, for the definition of satisfiability under the assumptions in CAS_{t1}*, we obtain that* $\mathcal{I}(\mathsf{cultural\ tourist}^{\mathsf{M}}) = Interesting(market)$.

For K_{org} *and the model* $\mathfrak{I}_{CAS_{org}}$*, similarly, we have:*

$$\mathsf{K}_{org} \models \mathsf{employees2016} : \neg WorkingNow(charlie)$$

However, for the satisfiability under the assumptions in CAS_{org}*, we obtain that* $\mathcal{I}(\mathsf{employees}^{\mathsf{M}}) \models WorkingNow(alice)$ *and* $\mathcal{I}(\mathsf{employees}^{\mathsf{M}}) \models WorkingNow(bob)$. ◊

In order to decide entailment of an axiom, it is helpful to know when a clashing assumption is justified. The following theorem provides such a characterization, which resorts to the least model $\hat{\mathfrak{I}}_{\mathsf{K}}(\chi, \mathsf{v})$ for a clashing assumption χ and a name assignment v.

Theorem 2 (justified CAS characterization). *Let* χ *be a satisfiable clashing assumption for CKR* K *and name assignment* v*. Then* χ *is justified iff* $(\alpha, \mathbf{e}) \in \chi(x)$ *implies some clashing set* $S = S_{(\alpha,\mathbf{e}),x}$ *exists such that*

(i) $\hat{I}(x) \models \beta$*, for each positive* $\beta \in S$*, where* $\hat{\mathfrak{I}}_{\mathsf{K}}(\chi, \mathsf{v}) = (\hat{\mathsf{M}}, \hat{\mathcal{I}}, \chi)$*, and*

(ii) no CAS-model $\mathfrak{I}_{CAS} = (\mathsf{M}, \mathcal{I}, \chi)$ *with name assignment* v *exists such that* $\mathcal{I}(x) \models \beta$ *for some* $\neg\beta \in S$.

Proof. Given that χ is a satisfiable clashing assumption for K and v, by Corollary 1 the least CAS-model $\hat{\mathfrak{I}}_{\mathsf{K}}(\chi, \mathsf{v}) = (\hat{\mathsf{M}}, \hat{\mathcal{I}}, \chi)$ w.r.t. inclusion $\mathsf{M}^J \subseteq \mathsf{M}$ and $I^J \subseteq \mathcal{I}$ for v exists.
(⇒) If χ is justified, then $\hat{\mathfrak{I}}_{\mathsf{K}}(\chi, \mathsf{v})$ is justified and hence by definition for every $(\alpha, \mathbf{e}) \in \chi(x)$ some clashing set $S = S_{(\alpha,\mathbf{e}),x}$ exists such that for each CAS-model $\mathfrak{I}^J_{CAS} = (\mathsf{M}^J, \mathcal{I}^J, \chi)$ of K that is NI-congruent with $\hat{\mathfrak{I}}_{\mathsf{K}}(\chi, \mathsf{v})$, i.e., with name assignment v, it holds that $\mathcal{I}^J(x) \models S$; hence, (ii) clearly holds. Furthermore, as $\mathfrak{I}^J_{CAS}$ is NI-congruent with itself, also (i) holds.
(⇐) Suppose every $(\alpha, \mathbf{e}) \in \chi(x)$ has some clashing set $S = S_{(\alpha,\mathbf{e}),x}$ such that items (i) and (ii) hold. Let $\mathfrak{I}^J_{CAS} = (\mathsf{M}^J, \mathcal{I}^J, \chi)$ be any CAS-model of K that is NI-congruent with $\hat{\mathfrak{I}}_{\mathsf{K}}(\chi, \mathsf{v})$. As $\hat{\mathcal{I}} \subseteq \mathcal{I}^J$, it follows from (i) that $\mathcal{I}^J(x) \models \beta$ for each positive $\beta \in S$; furthermore, from (ii) it follows that $\mathcal{I}^J(x) \models \neg\beta$ for each $\neg\beta \in S$. Hence, $\mathcal{I}^J(x) \models S$, and thus $(\alpha, \mathbf{e})$ is justified for $\hat{\mathfrak{I}}_{\mathsf{K}}(\chi, \mathsf{v})$. Consequently, $\hat{\mathfrak{I}}_{\mathsf{K}}(\chi, \mathsf{v})$ is justified. □

As for testing (ii), we can add to $\mathfrak{K}$ a module $\mathfrak{K}_{m_\beta} = \{\beta\}$ and the global assertion $\mathsf{mod}(c, m_\beta)$ where $c^{\mathcal{I}} = x$, and test whether the resulting CKR $\mathfrak{K}_{\beta,c}$ has no CAS-model $\mathfrak{I}'_{CAS} = \langle \mathfrak{M}', \mathcal{I}', \chi \rangle$ with naming ν; in other words, that the clashing assumption χ is not satisfiable for $\mathfrak{K}_\beta$ (equivalently, that $\hat{\mathfrak{I}}_{\mathfrak{K}_{\beta,c}}(\chi, \nu)$ does not exist).

With the characterization of justified clashing assumptions at hand, we can devise a refutation algorithm for $\mathfrak{K} \models c : \alpha$ resp. $\mathfrak{K} \models \alpha$ that finds a justified CAS-model $\mathfrak{I}_{CAS}$ of $\mathfrak{K}$ in which the query does not hold. As the axiom α amounts to a universal sentence $\forall \mathbf{x}.\varphi_\alpha(\mathbf{x})$ resp. $\forall \mathbf{x}.\varphi_\alpha(\mathbf{x}, c)$, it is sufficient to consider named models relative to the individual constants in $\mathfrak{K}$ and fresh (Skolem) constants for the negated query. This naturally leads to a non-deterministic algorithm. As we show in the next subsection, the refutation is feasible in nondeterministic polynomial time; this is worst-case optimal, as the entailment problem is coNP-hard.

Specifically, for positive assertions α, Corollary 1 implies that entailment (resp. $\mathfrak{K} \models \alpha$) is equivalent to truth of α at context c in (resp. at the global part of) the least model $\hat{\mathfrak{I}}_{\mathfrak{K}}(\chi, \nu)$, for every justifiable χ of $\mathfrak{K}$ and name assignment ν. For negative assertions $\alpha = \neg\beta$ entailment $\mathfrak{K} \models c : \alpha$ reduces similarly as in Theorem 2 to the nonexistence of the least model $\hat{\mathfrak{I}}(\chi, \nu)$ for $\mathfrak{K}_{\beta,c}$, for all justified χ and ν; for global entailment, this is analogous.

4.2. Conjunctive Queries

We can easily extend these results to manage conjunctive queries over different contexts. Formally, a *(general) conjunctive query (CQ)* is a formula $Q(\mathbf{x}) = \exists \mathbf{y}\gamma(\mathbf{x}, \mathbf{y})$ where $\mathbf{x}$, $\mathbf{y}$ are disjoint lists of different variables and $\gamma(\mathbf{x}, \mathbf{y}) = \gamma_1 \wedge \ldots \wedge \gamma_m$ is a conjunction of atoms γ_i of the form $c_i : \alpha_i(\mathbf{t}_i)$ resp. $\alpha_i(\mathbf{t}_i)$, $1 \leq i \leq m$ where c_i is a context name and α_i is either a concept name or a role name from the object vocabulary Σ or the meta-vocabulary Γ, and $\mathbf{t}_i$ is a tuple of variables from $\mathbf{x} \cup \mathbf{y}$ and individual constants that matches the arity of α_i. The CQ is *Boolean*, if $\mathbf{x}$ is empty.

Example 14. *Given the knowledge base $\mathfrak{K}_{org}$ in previous employees examples, a simple general CQ is to retrieve all employees that are currently working and also worked in the past years:*

$$Q_1(x) = \mathsf{employees2016} : WorkingNow(x) \wedge \mathsf{employees2016} : WorkingBefore(x)$$

We can obtain a Boolean query by instantiating Q_1 on one of the individuals in $\mathfrak{K}_{org}$, for example:

$$Q_2 = \mathsf{employees2016} : WorkingNow(alice) \wedge \mathsf{employees2016} : WorkingBefore(alice)$$

$\Diamond$

A CKR interpretation $\mathfrak{I} = \langle \mathfrak{M}, \mathcal{I} \rangle$ satisfies a Boolean CQ Q, denoted $\mathfrak{I} \models Q$, if for some substitution $\vartheta : \mathbf{y} \to \mathrm{NI}_s$ it holds that $\mathcal{I}(c_i^{\mathfrak{M}}) \models \alpha_i(\mathbf{t}_i\vartheta)$ resp. $\mathfrak{M} \models \alpha_i(\mathbf{t}_i\vartheta)$ for all $1 \leq i \leq m$. A CKR $\mathfrak{K}$ entails Q, denoted $\mathfrak{K} \models Q$, if every CKR model of $\mathfrak{K}$ satisfies Q. Based on this, the *(certain) answers* for a general CQ $Q(\mathbf{x})$ are defined as usual, i.e., as the tuples $\mathbf{c}$ of individual constants such that $\mathfrak{K} \models Q'$ where Q' is the boolean query $Q(\mathbf{c})$.

Example 15. *Considering the models for $\mathfrak{K}_{org}$ introduced in previous examples, it clearly holds that $\mathfrak{K}_{org} \models Q_2$ since we have that:*

$$\mathfrak{K}_{org} \models \mathsf{employees2016} : WorkingNow(alice),$$
$$\mathfrak{K}_{org} \models \mathsf{employees2016} : WorkingBefore(alice).$$

Moreover, the general query Q_1 then has as certain answers $\mathbf{c} \in \{alice, bob\}$, since we can verify that $\mathfrak{K}_{org} \models Q_1(alice) = Q_2$ and $\mathfrak{K}_{org} \models Q_1(bob)$. $\Diamond$

Conjunctive queries basically allow us to generalize the CKR-entailment to joins of one or more atomic facts. As such, from the results presented on CKR-entailment and the definition of CQ entailment, we directly obtain that also the evaluation of conjunctive queries can be restricted to named CKR models. Moreover, since Boolean CQs are basically conjunctions of positive atomic assertions, from Corollary 1 we have that for a Boolean conjunctive query Q, $\mathsf{K} \models Q$ iff $\hat{\mathfrak{I}}_{\mathsf{K}}(\chi, \nu) \models Q$, for every justifiable χ of K and name assignment ν.

4.3. Computational Complexity

We conduct in this section an analysis of the computational complexity of some major reasoning tasks for CKRs. In particular, we consider model checking, the entailment problem and conjunctive query answering. As for model checking, we assume throughout that interpretations are named and that constants not mapped to the named part are omitted; thus the named part and set the constants mapped to it are finite.

4.3.1. Model Checking

To begin with, we first note that satisfiability testing and model checking in (extended) $\mathcal{SROIQ}$-RL is tractable.

Lemma 3. *Given a $\mathcal{SROIQ}$-RL knowledge base K over $\mathcal{L}_\Sigma$ or $\mathcal{L}_\Sigma^e$ (resp. $\mathcal{L}_\Gamma$), one can decide in polynomial time (i) whether a given DL interpretation $\mathcal{I}$ of Σ (resp., Γ) satisfies K, and (ii) whether for a given CKR interpretation $\mathfrak{I} = \langle \mathcal{M}, \mathcal{I} \rangle$ and $x \in \mathsf{Ctx}^{\mathcal{M}}$, it holds that $\mathcal{I}(x) \models \mathsf{K}$ (resp. $\mathcal{M} \models \mathsf{K}$).*

Indeed, for each concept expression E and role expression R, we can compute $E^{\mathcal{I}}$ and $S^{\mathcal{I}}$, as well as for any $x \in \mathsf{Ctx}^{\mathcal{M}}$ also $E^{\mathcal{I}(x)}$ and $S^{\mathcal{I}(x)}$, inductively along its structure in polynomial time; note that each E^I is a unary relation, and each S^I is a binary relation. Based on this, we can easily check whether every axiom in K is satisfied in polynomial time.

For model checking of CAS-semantics, we then obtain the following result.

Proposition 8. *Given a CKR $\mathsf{K} = \langle \mathsf{G}, \{\mathsf{K}_{\mathsf{m}}\}_{\mathsf{m} \in \mathbf{M}} \rangle$ and a CAS-interpretation $\mathfrak{I}_{CAS} = \langle \mathcal{M}, \mathcal{I}, \chi \rangle$, deciding whether $\mathfrak{I}_{CAS} \models \mathsf{K}$ holds is feasible in polynomial time.*

Proof. From Lemma 3, it is immediate that the items (i)–(iv) of $\mathfrak{I}_{CAS} \models \mathsf{K}$ in Definition 11 can be checked in polynomial time. As for item (v), we must test the axiom α for all tuples $\mathbf{d} \notin \{\mathbf{e} \mid (\alpha, \mathbf{e}) \in \chi(x)\}$ in $\mathcal{I}(x)$. To this end, it is sufficient consider $\mathbf{d}$ over $N \cup \{c_1, \ldots, c_{|\mathbf{e}|}\}$ where the c_i are distinct standard names not in $N^{\mathcal{M}}$. There are polynomially many such $\mathbf{d}$, and for each the test is by Lemma 3), t feasible polynomial time. Overall, it follows that deciding $\mathfrak{I}_{CAS} \models \mathsf{K}$ is feasible in polynomial time. □

For justified CAS-model checking, in addition to $\mathfrak{I}_{CAS} \models \mathsf{K}$ we must verify that *CAS* is justified for the name assignment given by $\mathfrak{I}_{CAS}$. We can exploit Proposition 2 and Theorem 2, given the fact that the least model $\hat{\mathfrak{I}}_{\mathsf{K}}(\chi, \nu)$ can be efficiently constructed.

Lemma 4. *Given a CKR K, a clashing assumption χ for K and a name assignment ν,[12] one can compute $\hat{\mathfrak{I}}_{\mathsf{K}}(\chi, \nu)$ in polynomial time resp. recognize that $\hat{\mathfrak{I}}_{\mathsf{K}}(\chi, \nu)$ does not exist.*

[12]Technically, we assume ν is restricted to a finite given set N, $N_{\mathsf{K}} \subseteq N \subseteq \mathrm{NI} \setminus \mathrm{NI}_S$,

Proof. (Sketch) This can be done by computing the least model of the Horn sentence $\varphi_{\mathfrak{K}}$ in (8) for χ and ν represented as facts (and tacitly including the congruence axioms), which is possible using a standard fixpoint iteration of a one-step consequence operator $T_{\varphi_{\mathfrak{K}}}$. To avoid an exponential blowup of the naive translation, occurrences of disjunction $C_1 \sqcap C_2$ are eliminated using auxiliary predicates $P_{C_1 \sqcap C_2}$ (see Appendix). In each iteration, we must evaluate Horn implications of the form (A.1) where the antecedent $p_1(\mathbf{x}, \mathbf{x}_i, y_1) \wedge \cdots \wedge p_k(\mathbf{x}, \mathbf{x}_i, y_k)$ forms an acyclic conjunctive query. Matching acyclic queries against a relational interpretation is well-known to be feasible in polynomial time (cf. [37]). As all predicate arities are bounded by a constant, the number of iterations is polynomially bounded. If $\perp$ is derived, then $\hat{\mathfrak{I}}_{\mathfrak{K}}(\chi, \nu)$ does not exist, otherwise it is easily extracted from the computed least fixpoint. □

Proposition 9. *Given a CKR $\mathfrak{K} = \langle \mathfrak{G}, \{\mathrm{K}_m\}_{m \in \mathfrak{M}} \rangle$ and a CAS-interpretation $\mathfrak{I}_{CAS} = \langle \mathfrak{M}, \mathfrak{I}, \chi \rangle$, deciding whether $\mathfrak{I}$ is a justified CAS-model of $\mathfrak{K}$ is feasible in polynomial time.*

Proof. By Proposition 8, we can check whether $\mathfrak{I}_{CAS} \models \mathfrak{K}$ in polynomial time. By Theorem 2, it thus remains to check whether for every $\langle \alpha, \mathbf{e} \rangle$, x such that the conditions (i) and (ii) of the theorem are satisfied some clashing set $S_{\langle \alpha, \mathbf{e} \rangle, x}$ exists. To this end, we let S consist of (a) all positive atomic assertions β such that $\hat{\mathcal{I}}(x) \models \beta$ where $\hat{\mathfrak{I}}_{\mathfrak{K}}(\chi, \nu) = \langle \hat{\mathfrak{M}}, \hat{\mathcal{I}}, \chi \rangle$, and β is over N, where N are the constants of the named part,[13] and (b) all negative atomic assertions $\neg\beta$ over N such that $\hat{\mathfrak{I}}_{\mathfrak{K}_{\beta,\mathbf{c}}}(\chi, \nu)$ does not exist (i.e., no CAS-model $\mathfrak{I}_{CAS} = \langle \mathfrak{M}', \mathcal{I}', \chi \rangle$ with name assignment ν exists such that $\mathcal{I}'(x) \models \beta$).

It is easy to see that S is satisfiable, and it follows from the proof of Theorem 1 and Theorem 2 that S is moreover a clashing set for $\langle \alpha, \mathbf{e} \rangle$ iff some clashing set $S_{\langle \alpha, \mathbf{e} \rangle}$ for $\langle \alpha, \mathbf{e} \rangle$ exists, and that S includes some irredundant (minimal) clashing set S' of size linear in the size of α; thus we can restrict the candidates β resp. $\neg\beta$ for S' to axiom instances over a small (linear) extension of the individual constants $N_{\mathfrak{K}}$.

Testing whether $\{\varphi_\alpha(\mathbf{e})\} \cup S'$ is unsatisfiable can be done in polynomial time. As for each candidate β in (a) (resp. $\neg\beta$ in (b)) the test for inclusion in S is by Lemma 4 feasible in polynomial time, S' can for each $\langle \alpha, \mathbf{e} \rangle$, x be constructed in polynomial time; furthermore, the number of x is linear in the input. Hence overall, the test is feasible in polynomial time. □

As a corollary, we obtain that also CKR model checking is tractable.

Corollary 2. *Given a CKR $\mathfrak{K} = \langle \mathfrak{G}, \{\mathrm{K}_m\}_{m \in \mathfrak{M}} \rangle$ and a CKR-interpretation $\mathfrak{I} = \langle \mathfrak{M}, \mathcal{I} \rangle$, deciding whether $\mathfrak{I}$ is a CKR-model of $\mathfrak{K}$ is feasible in polynomial time.*

Proof. Indeed, we can by Proposition 5 and item (v) in Definition 11 construct a unique clashing assumption χ in which we collect at each context $x \in \mathsf{Ctx}^{\mathcal{M}}$ all instances of defeasible axioms $D(\alpha)$ in $\mathfrak{G}$ that are violated by $\mathfrak{I}$, and set $\chi(x) = \emptyset$ for each $x \in \Delta^{\mathcal{M}} \setminus \mathsf{Ctx}^{\mathcal{M}}$; then we test whether $\mathfrak{I}_{CAS}(\langle \mathfrak{M}, \mathcal{I}, \chi \rangle)$ is a justified CAS-model. By the form of $\varphi_\alpha(\mathbf{x})$, the number of instances $\langle \alpha, \mathbf{e} \rangle$ is polynomial and by Lemma 3 each test is polynomial. Furthermore, the test for $\mathfrak{I}_{CAS}\langle \mathfrak{M}, \mathcal{I}, \chi \rangle$ is polynomial by Proposition 9; this proves the result. □

4.3.2. Satisfiability

Based on the results above, we can characterize the complexity of satisfiability testing for CKRs. In general, defeasible axioms can lead to inconsistency that leaves one with a choice for

[13] In fact, a subset $N_0 \subseteq N$ modulo equality would suffice.

exceptions; e.g., if we had $D(A(a))$ and $D(\neg A(a))$ in the global knowledge. It is thus no surprise that the problem is intractable in general.

Theorem 3. *Given a CKR* $\mathfrak{K} = (\mathfrak{G}, \{\mathrm{K_m}\}_{\mathrm{m}\in\mathbf{M}})$*, deciding whether* $\mathfrak{K}$ *has some justified CAS-model resp. some CKR-model is NP-complete. The NP-hardness holds even if the module structure is fixed and only the assertions in the modules* $\mathrm{K_m}$ *vary (i.e., under data complexity).*

Proof. (Sketch) As for membership in NP, we can guess a justified CAS-model $\mathcal{I}_{CAS} = \langle \mathfrak{M}, \mathcal{I}, \chi \rangle$ over the (pseudo) Herbrand domain and verify $\mathcal{I}_{CAS}$ in polynomial time by Proposition 9. The hardness part is shown by a reduction from 3SAT: further details are provided in Appendix A.4. □

In the absence of defeasible axioms, satisfiability is tractable, as clashing assumptions play no role.

Proposition 10. *Given a CKR* $\mathfrak{K} = \langle \mathfrak{G}, \{\mathrm{K_m}\}_{\mathrm{m}\in\mathbf{M}}\rangle$ *where* $\mathfrak{G} \subseteq \mathcal{L}_\Gamma \cup \mathcal{L}_\Sigma$*, deciding whether* $\mathfrak{K}$ *has some justified CAS-model resp. some CKR-model is feasible in polynomial time.*

Proof. The reason is that the semantics of $\mathfrak{K}$ over any CAS-interpretation $\mathcal{I}_{CAS}\langle \mathfrak{M}, \mathcal{I}, \chi \rangle$ is independent of *CAS*; thus we can assume χ is void. We then can simplify $\varphi_{\mathfrak{K}}$ to the sentence

$$\varphi'_{\mathfrak{K}} = \bigwedge_{\alpha\in\mathfrak{G}_{\Sigma,\Gamma}} \forall \mathbf{x}.\varphi_\alpha(\mathbf{x}) \wedge \bigwedge_{\mathrm{m}\in\mathbf{M}} \forall x_c.\ \mathrm{mod}(x_c, \mathrm{m}) \to \varphi_{\mathrm{K_m},x_c}(x_c) \wedge \bigwedge_{\alpha\in\mathfrak{G}\cap\mathcal{L}_\Sigma} \forall x_c.\varphi_\alpha^{Ctx}(\mathbf{x}, x_c).$$

As for satisfiability, we can as discussed in the proof of Lemma 4 eliminate disjunctions $C_1 \sqcap C_2$ from $\mathfrak{K}$ in polynomial time to avoid an exponential blowup, and arrive at a formula $\varphi''_{\mathfrak{K}}$.

By a standard fixpoint-iteration, we can compute the least (pseudo) Herbrand model of $\varphi''_{\mathfrak{K}}$ for the universe $N_{\mathfrak{K}}$ in polynomially many steps (as only polynomially many ground atoms exist), or find out that no model exists. The rule matching in each iteration is polynomial, as the Horn clause bodies form acyclic conjunctive queries; thus the total computation is polynomial. □

4.3.3. CKR Entailment

For context and global entailment from a CKR, the complexity is dual to the one of satisfiability, as expected.

Theorem 4. *Given a CKR* $\mathfrak{K} = (\mathfrak{G}, \{\mathrm{K_m}\}_{\mathrm{m}\in\mathbf{M}})$*, a context name* c *and an axiom* α*, deciding whether* $\mathfrak{K} \models \mathrm{c} : \alpha$ *resp.* $\mathfrak{K} \models \alpha$ *holds is coNP-complete, and coNP-hardness holds under data complexity and assertional queries* α.

Proof. (Sketch) As for membership of $\mathfrak{K} \models \mathrm{c} : \alpha$ in coNP, since evaluating α at context c amounts to evaluating a universal FO sentence $\forall \mathbf{x}, x_c.\varphi_\alpha^{CAS}(\mathbf{x}, x_c)$, in order to refute $\mathfrak{K} \models \mathrm{c} : \alpha$ we can by Theorem 1 guess a CKR-interpretation $\mathfrak{I} = \langle \mathfrak{M}, \mathcal{I} \rangle$ of $\mathfrak{K}$ that is named relative to N, $N_{\mathfrak{K}} \subseteq N \subseteq \mathrm{NI} \setminus \mathrm{NI_S}$, such that (a) $\mathfrak{I} \models \mathfrak{K}$ and (b) $\mathcal{I}(\mathrm{c}^{\mathfrak{I}}) \not\models \alpha$, where N includes all constants that occur in α. The test (a) is feasible in polynomial time by Corollary 2, and the test (b) by Lemma 3.

The coNP-hardness under the given restrictions follows from the reduction of 3SAT to CKR-model existence in the proof of Theorem 3: the 3SAT instance E is unsatisfiable iff $\mathfrak{K} \models \mathrm{c} : V(c_1)$ resp. $\mathfrak{K} \models V(c_1)$, say, as this is equivalent to $\mathfrak{K}$ not having a CKR-model. □

As in the case of satisfiability, entailment is tractable if no defeasible axioms are present.

Proposition 11. *Given a CKR* $\mathfrak{K} = (\mathfrak{G}, \{\mathrm{K_m}\}_{\mathrm{m}\in\mathbf{M}})$ *where* $\mathfrak{G} \subseteq \mathcal{L}_\Gamma \cup \mathcal{L}_\Sigma$*, a context name* c *and an axiom* α*, deciding whether* $\mathfrak{K} \models \mathrm{c} : \alpha$ *resp.* $\mathfrak{K} \models \alpha$ *holds is feasible in polynomial time.*

Proof. Extending the argument in the proof of Proposition 10, to decide $\mathsf{K} \models \mathsf{c} : \alpha$ we can test whether $\varphi^{\cup}_{\mathsf{K}} \models \psi_\alpha$ holds, or equivalently whether $\varphi^{\cup}_{\mathsf{K}} \wedge \neg\psi_\alpha$ is unsatisfiable, where ψ_α is the Horn variant of $\forall\mathbf{x}.\varphi^{\mathsf{Ctx}}_\alpha(\mathbf{x}, \mathsf{c})$ that avoids exponential blowup. As ψ_α can be written as a conjunction of (polynomially many and linear size) Horn clauses as in (A.1). this reduces to a polynomial number of unsatisfiability tests for $\varphi_i = \varphi^{\cup}_{\mathsf{K}} \wedge p_1(\mathbf{e}_1) \wedge \cdots \wedge p_k(\mathbf{e}_k) \wedge \neg\, p_0(\mathbf{e}_0)$, where the $\mathbf{e}_j$ are fresh (Skolem) constants.

By a standard fixpoint-iteration, we can compute the least (pseudo) Herbrand model of φ_i (where equality is replaced by congruence $\approx$) respectively detect that no model exists. As the bodies of the Horn implications are acyclic and all predicate arities are bounded by a constant, the fixpoint iteration can be done in polynomial time (cf. proof of Lemma 4); as there are polynomially many φ_i, the test $\varphi^{\cup}_{\mathsf{K}} \models \psi_\alpha$ is feasible in polynomial time.

The proof for global entailment $\mathsf{K} \models \alpha$ is similar; this proves the result. □

We conclude with a remark that under suitably limited use of default axioms, satisfiability and CKR entailment would still be tractable, depending on the structure of the knowledge base K. For example, if the global knowledge base contains few defeasible assertions, and contexts do not access other contexts, i.e., *eval* does not occur. A detailed complexity study is beyond the scope of this paper, however.

4.3.4. Conjunctive Queries

From well-known results in database theory [38, 37], it follows that deciding given a CKR-interpretation $\mathfrak{I}$ and a conjunctive query Q, deciding whether $\mathfrak{I} \models Q$ is NP-complete. As a CKR K can have multiple (even exponentially many) named CKR-models, as expected CQ answering from CKRs is lifted to the second level of the polynomial hierarchy.

Theorem 5. *Given a CKR $\mathsf{K} = \langle \mathsf{G}, \{\mathsf{K_m}\}_{\mathsf{m} \in \mathsf{M}} \rangle$ and a Boolean CQ Q, deciding whether $\mathsf{K} \models Q$ is Π^p_2-complete. The problem remains Π^p_2-hard even if the module structure is fixed and only the assertions in the modules $\mathsf{K_m}$ vary. If in addition also the query Q is fixed (i.e., under data complexity), then the problem is coNP-complete.*

Proof. (Sketch) As for membership in Π^p_2, to refute Q we can guess a justified CAS-model $\mathfrak{I}^N_{CAS} = (\mathfrak{M}, \mathfrak{I}, \chi)$ such that $N = N_{\mathsf{K}}$ and $\mathfrak{I}^N = (\mathfrak{M}, \mathfrak{I}) \not\models Q$. Indeed, if $\mathfrak{I}_{CAS} = (\mathfrak{M}, \mathfrak{I}, \chi)$ is an arbitrary justified CAS-model such that $(\mathfrak{M}, \mathfrak{I}) \not\models Q$, then by Theorem 1 its named restriction can not satisfy Q either. We can verify that $\mathfrak{I}^N_{CAS}$ is a named justified CAS model in polynomial time by Proposition 8, and check that $\mathfrak{I}^N \not\models Q$ using an NP oracle in polynomial time; for fixed Q, the latter test is feasible in polynomial time. The Π^p_2-hardness is shown by a reduction from evaluating quantified Boolean formulas (QBF) Φ of the form $\forall X \exists Y E$ (see Appendix A.4 for the complete proof). □

We note that Π^p_2-hardness can also be shown if alternatively the module structure and the set of assertions (the data) are fixed. Furthermore, the complexity drops to coNP for acyclic CQs, and to NP for CKRs without defeasible axioms; the combined restrictions yield tractability. Indeed, answering acyclic CQs over a relational database is feasible in polynomial time (cf. [37]), and thus the check $\mathfrak{I}^N \not\models Q$ in the refutation algorithm in the proof sketch is feasible in polynomial time; this yields coNP membership. On the other hand, if the global knowledge G contains no defeasible axioms, then the guess for a justified CAS-model $\mathfrak{I}^N_{CAS}$ in which the query Q has no match in the proof sketch can be eliminated, and following the arguments in the proof of Proposition 10 a single such candidate $\mathfrak{I}^N_{CAS}$ can be constructed in polynomial time. Clearly then, the combination of the restrictions yields a query refutation algorithm that runs in polynomial time.

5. Datalog Translation for CKR in $\mathcal{SROIQ}$-RL

In this section, we present a translation of reasoning from $\mathcal{SROIQ}$-RL CKRs with defeasible axioms into Datalog. It extends a translation for CKRs without defeasible axioms into Datalog presented in [1] with rules for the detection of axiom overriding (i.e., making exceptions) and defeasible propagation of global knowledge; this requires the use of nonmonotonic negation.

In particular, we focus on positive instance queries under c- resp. global entailment (negative instance queries can be handled as described at the end of Section 4.1), and on conjunctive queries. For such queries, we provide an implementation considering a core fragment of $\mathcal{SROIQ}$-RL for expressing defeasible axioms that we call $\mathcal{SROIQ}$-RLD. Formally, we denote with $\mathcal{SROIQ}$-RLD the fragment of $\mathcal{SROIQ}$-RL in which (i) $D \sqcap D$ can not appear as a right-side concept, and (ii) each right-side concept $\forall R.D$ has $D \in$ NC. We confine here to CKRs $\mathfrak{K}$ in which defeasible axioms are of the form $\mathrm{D}(\alpha)$ where α is in $\mathcal{SROIQ}$-RLD, and denote by $\mathcal{SROIQ}$-RLD the class of such $\mathfrak{K}$. While this restriction is a slight limitation from the view of modeling, as we will illustrate by Example 16 in the following, it allows us to formulate an easier characterization for the datalog translation.

For developing a generic datalog encoding, we first introduce a useful normal form for the axioms of $\mathcal{SROIQ}$-RL. After that, we present the translation and argue about its correctness.

5.1. Normal form

In this section, we introduce a normal form for axioms that allows us to represent a CKR $\mathfrak{K}$ conveniently as facts of a datalog program, as it bounds the number of concept and role constructors to a single application in each axiom.

Definition 14. *A CKR $\mathfrak{K} = \langle \mathfrak{G}, \{\mathrm{K_m}\}_{m \in \mathbf{M}} \rangle$ is in* normal form, *if every non-defeasible axiom in $\mathfrak{G}$ and $\mathrm{K_m}$ matches a form in Table 1, and every defeasible axiom in $\mathfrak{G}$ is of the form $\mathrm{D}(\alpha)$ where α is of the form (I) in Table 1.*

In Table 1 and elsewhere, we assume that in $\mathbf{C}$ resp. NC_Σ the empty concept $\bot$ is available (which is easily expressed by $\bot \sqsubseteq \neg\top$). In Table 2 we present a set of rules that can be used to transform any $\mathcal{SROIQ}$-RLD CKR into an "equivalent" CKR in normal form. As in [20], we assume that rule chain axioms in the input are already decomposed in binary role chains.

It can be seen that for named interpretations, i.e., of the form $\mathfrak{I}_{CAS}^{N\mathfrak{K}}$, every CKR can be rewritten into an equivalent one in normal form (using new symbols).

Lemma 5. *For every $\mathcal{SROIQ}$-RLD CKR $\mathfrak{K} = \langle \mathfrak{G}, \{\mathrm{K_m}\}_{m \in \mathbf{M}} \rangle$ over meta and object vocabularies (Γ, Σ), a CKR $\mathfrak{K}' = \langle \mathfrak{G}', \{\mathrm{K}'_m\}_{m \in \mathbf{M}} \rangle$ over extended vocabularies (Γ', Σ') can be computed such that*

- *(a) all axioms in $\mathfrak{K}'$ are in normal form;*
- *(b) the size of $\mathfrak{K}'$ is linear in the size of $\mathfrak{K}$;*
- *(c) for every axiom α on $\Gamma \cup \Sigma$:*
  - *(i). for every justified named CAS-model $\mathfrak{I}_{CAS}^{N\mathfrak{K}}$ for $\mathfrak{K}$ such that $\mathfrak{I}_{CAS}^{N\mathfrak{K}} \models \alpha$, there exists some justified CAS-model $\mathfrak{I}'^{N\mathfrak{K}}_{CAS}$ for $\mathfrak{K}'$ such that $\mathfrak{I}'^{N\mathfrak{K}}_{CAS} \models \alpha$;*

Table 1: Normal form for $\mathfrak{G}$ axioms from $\mathcal{L}_r \cup \mathcal{L}_\Sigma$ (I) and $\mathcal{L}_\Sigma$ (II), and for $\mathfrak{K}_m$ axioms from $\mathcal{L}_\Sigma$ (I) and $\mathcal{L}^e_\Sigma \setminus \mathcal{L}_\Sigma$ (III)

(I) for $A, B, C \in \mathbf{C}$ (resp., $\in \mathrm{NC}_\Sigma$), $R, S, T \in \mathbf{R}$ (resp., $\in \mathrm{NR}_\Sigma$), $a, b \in \mathbf{N}$ (resp., $\in \mathrm{NI}_\Sigma$) :					
$A(a)$	$R(a,b)$	$\neg A(b)$	$\neg R(a,b)$	$a = b$	$a \neq b$
$A \sqsubseteq B$	$\{a\} \sqsubseteq B$	$A \sqcap B \sqsubseteq C$			
$\exists R.A \sqsubseteq B$	$A \sqsubseteq \exists R.\{a\}$	$A \sqsubseteq \forall R.B$	$A \sqsubseteq \leq 1 R.\top$		
$R \sqsubseteq T$	$R \circ S \sqsubseteq T$	$\mathrm{Dis}(R,S)$	$\mathrm{Inv}(R,S)$	$\mathrm{Irr}(R)$	
(II) for $C \in \mathbf{C}$, $m \in \mathbf{M}$:					
$C \sqsubseteq \exists \mathrm{mod}.\{m\}$					
(III) for $A, B \in \mathrm{NC}_\Sigma$, $R, T \in \mathrm{NR}_\Sigma$ and $C \in \mathbf{C}$:					
$\mathit{eval}(A, C) \sqsubseteq B$	$\mathit{eval}(R, C) \sqsubseteq T$				

(ii). for every justified named CAS-model $\mathcal{I}'^{N_\mathfrak{K}}_{CAS}$ *for* $\mathfrak{K}'$ *such that* $\mathcal{I}'^{N_\mathfrak{K}}_{CAS} \models \alpha$, *there exists some justified CAS-model* $\mathcal{I}^{N_\mathfrak{K}}_{CAS}$ *for* $\mathfrak{K}$ *such that* $\mathcal{I}^{N_\mathfrak{K}}_{CAS} \models \alpha$. □

In the following, we also refer with *explicit negated assertions* to any normal form ABox assertions of the kind $\neg A(b)$, $\neg R(a,b)$, $a \neq b$ that explicitly appear in the input CKR $\mathfrak{K}$.

Example 16. *We show how enabling the normal form translation to full* $\mathcal{SROIQ}$*-RL (i.e. considering also right-hand* $D \sqcap D$ *and* $\forall R.D$ *with* $D \notin \mathrm{NC}$*) can cause problems in the interpretation of justifications. Consider the following CKR* $\mathfrak{K}_{nf} = \langle \mathfrak{G}, \{\mathfrak{K}_m\} \rangle$ *where:*

$$\mathfrak{G} : \quad \{\, \mathrm{D}(A \sqsubseteq B_1 \sqcap B_2), \mathrm{mod}(c, m) \,\},$$
$$\mathfrak{K}_m : \quad \{\, B_1 \sqcap B_2 \sqsubseteq \bot, A(a) \,\}.$$

Note that the clashing assumption $\langle A \sqsubseteq B_1 \sqcap B_2, a \rangle$ *admits two minimal clashing sets* $S_1 = \{A(a), \neg B_1(a)\}$ *and* $S_2 = \{A(a), \neg B_2(a)\}$*. However, neither of them can be proved in* $\mathfrak{K}_{nf}$*, thus no justified model exists. On the other hand, if we normalize the right-hand conjunction defeasible axiom with the simple rule:*

$$\mathrm{D}(A \sqsubseteq D_1 \sqcap D_2) \mapsto \{\mathrm{D}(A \sqsubseteq X), X \sqsubseteq D_1, X \sqsubseteq D_2\}$$

we obtain the CKR $\mathfrak{K}'_{nf} = \langle \mathfrak{G}', \{\mathfrak{K}_m\} \rangle$ *where:*

$$\mathfrak{G} : \quad \{\, \mathrm{D}(A \sqsubseteq X), X \sqsubseteq B_1, X \sqsubseteq B_2, \mathrm{mod}(c, m) \,\},$$
$$\mathfrak{K}_m : \quad \{\, B_1 \sqcap B_2 \sqsubseteq \bot, A(a) \,\}.$$

In this case, the only clashing assumption would be $\langle A \sqsubseteq X, a \rangle$*, which admits only one clashing set* $S_1 = \{A(a), \neg X(a)\}$*. Differently from the non-normalized case,* S_1 *can be proved from* $\mathfrak{K}_{nf}$*, thus a justified model exists.*

Similarly, this can be shown for right-hand $\forall R.D$ *with* $D \notin \mathrm{NC}$*: that is, the simple translation provided for strict* $\mathcal{SROIQ}$*-RL axioms can not be applied naively to such defeasible axioms as it might not preserve their justification.* $\diamond$

Table 2: Normal form transformation ($\perp$ is the empty concept)

(1) strict axioms:

$$
\begin{array}{rclrcl}
D(a) & \mapsto & \{X(a), X \pm D\} & A \pm \neg C & \mapsto & \{A \sqcap C \pm \perp\} \\
C \pm D & \mapsto & \{C \pm X, X \pm D\} & C \sqcap A \pm B & \mapsto & \{C \pm X, X \sqcap A \pm B\} \\
A \pm \top & \mapsto & \emptyset & A \pm D_1 \sqcap D_2 & \mapsto & \{A \pm D_1, A \pm D_2\} \\
\perp \pm A & \mapsto & \emptyset & C_1 \sqcap C_2 \pm B & \mapsto & \{C_1 \pm B, C_2 \pm B\}
\end{array}
$$

$$
\begin{array}{rclrcl}
\exists R.C \pm A & \mapsto & \{C \pm X, \exists R.X \pm A\} & \mathrm{Sym}(P) & \mapsto & \{P \pm W, \mathrm{Inv}(P, W)\} \\
A \pm \forall R.D & \mapsto & \{A \pm \forall R.X, X \pm D\} & \mathrm{Trans}(P) & \mapsto & \{P \circ P \pm P\} \\
A \pm \leq 0 R.\top & \mapsto & \{A \pm \forall R.\perp\} & \mathrm{Asym}(P) & \mapsto & \{\mathrm{Dis}(P, W), \mathrm{Inv}(P, W)\}
\end{array}
$$

$$
\begin{array}{rcl}
\mathit{eval}(C_1, \mathrm{C}) \pm B \in \mathrm{K_m} & \mapsto & \{\mathit{eval}(X, Y) \pm B \in \mathrm{K_m}, \\
& & C_1 \pm X \in \mathrm{K_{mx}}, \mathrm{C} \pm Y \in \mathrm{G}, Y \pm \exists \mathrm{mod}.\{\mathrm{mx}\} \in \mathrm{G}\} \\
\mathit{eval}(R, \mathrm{C}) \pm T \in \mathrm{K_m} & \mapsto & \{\mathit{eval}(R, Y) \pm T \in \mathrm{K_m}, \mathrm{C} \pm Y \in \mathrm{G}\} \\
\exists \mathit{eval}(R, \mathrm{C}).A \pm B \in \mathrm{K_m} & \mapsto & \{\exists W.A \pm B \in \mathrm{K_m}, \mathit{eval}(R, \mathrm{C}) \pm W \in \mathrm{K_m}\} \\
\mathit{eval}(R, \mathrm{C}) \circ S \pm T \in \mathrm{K_m} & \mapsto & \{\mathit{eval}(R, \mathrm{C}) \pm W \in \mathrm{K_m}, W \circ S \pm T \in \mathrm{K_m}\} \\
\mathrm{Dis}(\mathit{eval}(R, \mathrm{C}), S) \in \mathrm{K_m} & \mapsto & \{\mathit{eval}(R, \mathrm{C}) \pm W \in \mathrm{K_m}, \mathrm{Dis}(W, S) \in \mathrm{K_m}\}
\end{array}
$$

(2) defeasible axioms:

$$
\begin{array}{rclrcl}
\mathrm{D}(C(a)) & \mapsto & \{X(a), \mathrm{D}(X \pm C)\} & \mathrm{D}(\mathrm{Sym}(P)) & \mapsto & \{P \pm W, \mathrm{D}(\mathrm{Inv}(P, W))\} \\
\mathrm{D}(C_1 \pm C_2) & \mapsto & \{C_1 \pm X, \mathrm{D}(X \pm C_2)\} & \mathrm{D}(\mathrm{Trans}(P)) & \mapsto & \{\mathrm{D}(P \circ P \pm P)\} \\
\mathrm{D}(A \pm \neg C) & \mapsto & \{\mathrm{D}(A \sqcap C \pm \perp)\} & \mathrm{D}(\mathrm{Asym}(P)) & \mapsto & \{\mathrm{D}(\mathrm{Dis}(P, W)), \mathrm{Inv}(P, W)\} \\
\mathrm{D}(A \pm \leq 0 R.\top) & \mapsto & \{\mathrm{D}(A \pm \forall R.\perp)\} & & &
\end{array}
$$

$a \in \mathrm{NI}$, $A, B \in \mathrm{NC}$, $R, S, T, P \in \mathrm{NR}$, $X, Y \in \mathrm{NC}$ are fresh concept names, $W \in \mathrm{NR}$ is a fresh role name, mx is a fresh module name and $\mathrm{K_{mx}}$ its associated knowledge base, $C, C_i, D, D_i, \mathrm{C}$ are (possibly complex) concept expressions.

5.2. *Translation overview*

We are now prepared to present our translation of entailment reasoning into Datalog with non-monotonic negation. To ease the development, we adopt for it the *unique name assumption* in any interpretation; this allows us to omit dealing with equality in models, which else can be done by emulating a congruence relation via a standard guess-and-check approach.

As mentioned above, it extends a translation of CKR without defeasible axioms into Datalog presented in [1]. That translation was inspired by the materialization calculus K_{inst} of Krötzsch [20] for instance checking in the description logic $\mathcal{SROEL}(\sqcap, \times)$ (in essence, OWL-EL). Briefly, Krötzsch showed that a calculus for that problem can be encoded in a Datalog program, where the rules mimick inference rules. Furthermore, he presented – exploiting a normal form of axioms – a fixed datalog program that gives a universal encoding of the proof systems associated with concrete DL knowledge bases $\mathcal{K}$ where $\mathcal{K}$ and the instance query are represented by a set of facts. This technique was carried over to $\mathcal{SROIQ}$-RL and extended for CKRs in [1].

In the sequel, we present an extension of this translation for CKRs with defeasible axioms. This extension is non-trivial, as (i) it requires us to deal with exceptions for axioms, via clashing assumptions, which requires the use of non-monotonic negation; and in connection with this, (ii) it requires us to deal with strong negation, as provable falsity of atoms (resp. assertions) is crucial for clashing sets. As for (i), we can take advantage of a mechanism for inheritance axioms from

[39], while for (ii), we extend the materialization calculus to conclude negative literals. However, the latter requires us to deal with negative disjunctive information; e.g. from $A \sqcap B \sqsubseteq C$ and $\neg C(a)$ we can infer $\neg(A \sqcap B)(a)$, but neither $\neg A(a)$ nor $\neg B(a)$; this can not be readily expressed with datalog rules, even in the presence of strong negation; viewing $A \sqcap B \sqsubseteq C$ as $A \sqcap \neg C \sqsubseteq \neg B$, $B \sqcap \neg C \sqsubseteq \neg A$ and mimicking respective inference rules as datalog rules would not work, as the calculus would be incomplete. For this reason, we encode inference of negative literals through individual proofs by contradiction, which will be indicated by presence of a designated atom $\mathsf{unsat}(\cdot)$ for the literal in the answer set; notably, from the absence of $\mathsf{unsat}(\cdot)$ we can conclude that the literal is not derivable. Overall, this leads to a linear number of contradiction tests for the literals, which are encoded using designated test rules.[14]

The whole translation is rather involved and contains a number of rules and facts that serve different purposes. From a high level structural perspective, the translation has three components:

(1) the *input translations* I_{glob}, I_{loc}, $I_{\mathcal{D}}$, I_{rl}, where given an axiom or signature symbol α and $\mathsf{c} \in \mathbf{N}$, each $I(\alpha, \mathsf{c})$ is a (possibly empty) set of datalog facts or rules: intuitively, they encode the contents of the global and the local DL knowledge bases as datalog facts and rules. These input translations I are extended to knowledge bases (sets of axioms) S with their signature Σ, by $I(S, \mathsf{c}) = \bigcup_{\alpha \in S} I(\alpha, \mathsf{c}) \cup \bigcup_{s \in \Sigma} I(s, \mathsf{c})$.

(2) the *deduction rules* P_{loc}, $P_{\mathcal{D}}$, P_{rl}, which are sets of datalog rules: they represent the inference rules for the instance-level reasoning over the translated axioms; and

(3) the *output translation* O, where given an axiom α and $\mathsf{c} \in \mathbf{N}$, $O(\alpha, \mathsf{c})$ is either empty or a single datalog fact: O encodes the ABox assertion α that we want to prove to be entailed by the input CKR (in the context c) as a datalog fact.

We will describe next these components, the translation process as such and we will then consider an example.

5.2.1. Translation rule sets

The components of the translation comprise in turn groups of rules that serve different purposes: we show here some example rules for each group, while the complete rule sets are given in Tables 3–8 below.

(i). $\mathcal{SROIQ}$-*RL input translation*: Rules in $I_{rl}(S, c)$ translate to datalog facts $\mathcal{SROIQ}$-RL axioms and signature (in a context c). E.g., we translate atomic concept inclusions with the rule $A \sqsubseteq B \mapsto \{\mathsf{subClass}(A, B, c)\}$. Note that, for instance level predicates, we distinguish between the asserted (i.e. insta, $\mathsf{triplea}$) and derived (i.e. instd, $\mathsf{tripled}$) atoms: this distinction is needed in the rules for the (defeasible) propagation of knowledge, where we want to recognize which facts are part of the asserted "content" of the global context that might be propagated to lower contexts.

(ii). $\mathcal{SROIQ}$-*RL deduction rules*: The rules in P_{rl} are the deduction rules corresponding to axioms in $\mathcal{SROIQ}$-RL: e.g., for atomic concept inclusions, we have

$$\mathsf{instd}(x, z, c, t) \leftarrow \mathsf{subClass}(y, z, c), \mathsf{instd}(x, y, c, t).$$

[14] While the set of all positive literals entailed by a propositional Horn formula is computable in linear time, to the best of our knowledge it is unknown whether this holds for all negative literals; common algorithms run in quadratic time.

The rules of I_{rl} and P_{rl} are listed in Table 3. Note that the translation produces purely positive programs: possible derived inconsistencies are represented by the unsat predicate and constrained by the rule (prl-sat). The last argument in the instance level predicates (instd, tripled, eq) keeps track of the hypothesis used in the proofs for contradiction for negative literals (as mentioned in the translation overview): in the translation of instance level assertions in I_{rl}, this parameter is initialized to the constant main. The predicate unsat(t) denotes that the proof relative to the hypothesis t leads to an inconsistency: as noted above, in the case of main this is limited by the constraint (prl-sat). As we adopted the unique name assumption, reasoning on equality can be omitted; an explicit equality assertion raises an inconsistency by the rule (prl-eq), while inequality assertions – assuming that assertions $a \neq a$ are not admissible – are simply discarded.

(iii). *Global and local translations*: Global input rules of I_{glob} encode the interpretation of Ctx in the global context (i.e. conditions from Definition 7). Similarly, local input rules I_{loc} and local deduction rules P_{loc} provide the translation and rules for elements of the local object language. In particular for *eval* expressions in concept inclusions, we have the input rule *eval*(A, C) ± B ↦ {subEval(A, C, B, c)} and the corresponding positive deduction rule:

instd(x, b, c, t) ← subEval(a, c_1, b, c, t), instd(c', c_1, gm, t), instd(x, a, c', t).

The rules of I_{glob}, I_{loc} and P_{loc} are presented in Table 4.

(iv). *Defeasible axioms input translation*: Input rules in I_{D} provide the translation of defeasible axioms D(α) in the global context: I_{D}(D(α), gk) adds to the program (in the module gk for the global object knowledge) an atom specifying that the asserted axiom is defeasible. For example, D(A ± B) translates to def_subclass(A, B).

(v). *Overriding rules*: The inheritance and overriding of defeasible axioms is encoded by different sets of deduction rules in P_{D}, inspired by [39]. Overriding rules provide rules defining when an axiom of a certain form is locally overridden. Intuitively, such rules encode the proof of existence for a clashing set for an instance of such axiom. For example, for axioms of the form D($A \sqsubseteq B$), the following overriding rule is added to the local programs:

ovr(subClass, x, y, z, c) ← def_subclass(y, z), prec(c, g), instd(x, y, c, main),
not test_fails(nlit(x, z, c)).

Intuitively, this rule states that, if $y = A$ is included in $z = B$ by a defeasible global axiom (def_subclass(y, z)) and in context c we can prove for $x = e$ that $A(e)$ (i.e., instd(x, y, c, main)) but ¬ $B(e)$ (not test_fails(nlit(x, z, c))), then there is an overriding for this axiom with respect to e in context c (ovr(subClass, x, y, z, c)). Here prec(c, g) expresses that context c is more specific than context g, which represents the global context. The condition on the negative part ¬$B(e)$ of the clashing set {$A(e)$, ¬$B(e)$} for $A \sqsubseteq B$, e is verified, exploiting Theorem 2 and the remark after it, by a proof by contradiction: if this "test" *does not fail*,[15] i.e., after adding the positive version of the literal (in the example $B(e)$) inconsistency can be derived, then the clashing assumption is justified and we can derive the overriding. In the example rule above, such proof is performed on the term nlit(x, z, c), which represents the negative literal ¬instd(x, z, c, main). The complete list

[15]Note that we use a double negation in order to avoid cyclic dependencies across overriding and test rules.

Table 3: SROIQ-RL input and deduction rules

SROIQ-RL input translation $I_{rl}(S, c)$

(irl-nom)	$a \in \mathrm{NI} \mapsto \{\mathsf{nom}(a, c)\}$
(irl-cls)	$A \in \mathrm{NC} \mapsto \{\mathsf{cls}(A, c)\}$
(irl-rol)	$R \in \mathrm{NR} \mapsto \{\mathsf{rol}(R, c)\}$
(irl-inst1)	$A(a) \mapsto \{\mathsf{insta}(a, A, c, \mathsf{main})\}$
(irl-inst2)	$\neg A(a) \mapsto \{\mathsf{ninsta}(a, A, c)\}$
(irl-triple)	$R(a, b) \mapsto \{\mathsf{triplea}(a, R, b, c, \mathsf{main})\}$
(irl-ntriple)	$\neg R(a, b) \mapsto \{\mathsf{ntriplea}(a, R, b, c)\}$
(irl-eq)	$a = b \mapsto \{\mathsf{eq}(a, b, c, \mathsf{main})\}$
(irl-neq)	$a \neq b \mapsto \emptyset$
(irl-inst3)	$\{a\} \sqsubseteq B \mapsto \{\mathsf{insta}(a, B, c, \mathsf{main})\}$
(irl-subc)	$A \sqsubseteq B \mapsto \{\mathsf{subClass}(A, B, c)\}$
(irl-top)	$\top(a) \mapsto \{\mathsf{insta}(a, \mathsf{top}, c)\}$
(irl-bot)	$\bot(a) \mapsto \{\mathsf{insta}(a, \mathsf{bot}, c)\}$
(irl-subcnj)	$A_1 \sqcap A_2 \sqsubseteq B \mapsto \{\mathsf{subConj}(A_1, A_2, B, c)\}$
(irl-subex)	$\exists R.A \sqsubseteq B \mapsto \{\mathsf{subEx}(R, A, B, c)\}$
(irl-supex)	$A \sqsubseteq \exists R.\{a\} \mapsto \{\mathsf{supEx}(A, R, a, c)\}$
(irl-forall)	$A \sqsubseteq \forall R.B \mapsto \{\mathsf{supForall}(A, R, B, c)\}$
(irl-leqone)	$A \sqsubseteq \leq 1 R.\top \mapsto \{\mathsf{supLeqOne}(A, R, c)\}$
(irl-subr)	$R \sqsubseteq S \mapsto \{\mathsf{subRole}(R, S, c)\}$
(irl-subrc)	$R \circ S \sqsubseteq T \mapsto \{\mathsf{subRChain}(R, S, T, c)\}$
(irl-dis)	$\mathrm{Dis}(R, S) \mapsto \{\mathsf{dis}(R, S, c)\}$
(irl-inv)	$\mathrm{Inv}(R, S) \mapsto \{\mathsf{inv}(R, S, c)\}$
(irl-irr)	$\mathrm{Irr}(R) \mapsto \{\mathsf{irr}(R, c)\}$

SROIQ-RL deduction rules P_{rl}

(prl-instd)	$\mathsf{instd}(x, z, c, t)$	$\leftarrow$	$\mathsf{insta}(x, z, c, t)$.
(prl-tripled)	$\mathsf{tripled}(x, r, y, c, t)$	$\leftarrow$	$\mathsf{triplea}(x, r, y, c, t)$.
(prl-ninstd)	$\mathsf{unsat}(t)$	$\leftarrow$	$\mathsf{ninsta}(x, z, c)$, $\mathsf{instd}(x, z, c, t)$.
(prl-ntripled)	$\mathsf{unsat}(t)$	$\leftarrow$	$\mathsf{ntriplea}(x, r, y, c)$, $\mathsf{tripled}(x, r, y, c, t)$.
(prl-eq)	$\mathsf{unsat}(t)$	$\leftarrow$	$\mathsf{eq}(x, y, c, t)$.
(prl-top)	$\mathsf{instd}(x, \mathsf{top}, c, \mathsf{main})$	$\leftarrow$	$\mathsf{nom}(x, c)$.
(prl-bot)	$\mathsf{unsat}(t)$	$\leftarrow$	$\mathsf{instd}(x, \mathsf{bot}, c, t)$.
(prl-subc)	$\mathsf{instd}(x, z, c, t)$	$\leftarrow$	$\mathsf{subClass}(y, z, c)$, $\mathsf{instd}(x, y, c, t)$.
(prl-subcnj)	$\mathsf{instd}(x, z, c, t)$	$\leftarrow$	$\mathsf{subConj}(y_1, y_2, z, c)$, $\mathsf{instd}(x, y_1, c, t)$, $\mathsf{instd}(x, y_2, c, t)$.
(prl-subex)	$\mathsf{instd}(x, z, c, t)$	$\leftarrow$	$\mathsf{subEx}(v, y, z, c)$, $\mathsf{tripled}(x, v, x', c, t)$, $\mathsf{instd}(x', y, c, t)$.
(prl-supex)	$\mathsf{tripled}(x, r, x', c, t)$	$\leftarrow$	$\mathsf{supEx}(y, r, x', c)$, $\mathsf{instd}(x, y, c, t)$.
(prl-supforall)	$\mathsf{instd}(y, z', c, t)$	$\leftarrow$	$\mathsf{supForall}(z, r, z', c)$, $\mathsf{instd}(x, z, c, t)$, $\mathsf{tripled}(x, r, y, c, t)$.
(prl-leqone)	$\mathsf{unsat}(t)$	$\leftarrow$	$\mathsf{supLeqOne}(z, r, c)$, $\mathsf{instd}(x, z, c, t)$, $\mathsf{tripled}(x, r, x_1, c, t)$, $\mathsf{tripled}(x, r, x_2, c, t)$.
(prl-subr)	$\mathsf{tripled}(x, w, x', c, t)$	$\leftarrow$	$\mathsf{subRole}(v, w, c)$, $\mathsf{tripled}(x, v, x', c, t)$.
(prl-subrc)	$\mathsf{tripled}(x, w, z, c, t)$	$\leftarrow$	$\mathsf{subRChain}(u, v, w, c)$, $\mathsf{tripled}(x, u, y, c, t)$, $\mathsf{tripled}(y, v, z, c, t)$.
(prl-dis)	$\mathsf{unsat}(t)$	$\leftarrow$	$\mathsf{dis}(u, v, c)$, $\mathsf{tripled}(x, u, y, c, t)$, $\mathsf{tripled}(x, v, y, c, t)$.
(prl-inv1)	$\mathsf{tripled}(y, v, x, c, t)$	$\leftarrow$	$\mathsf{inv}(u, v, c)$, $\mathsf{tripled}(x, u, y, c, t)$.
(prl-inv2)	$\mathsf{tripled}(y, u, x, c, t)$	$\leftarrow$	$\mathsf{inv}(u, v, c)$, $\mathsf{tripled}(x, v, y, c, t)$.
(prl-irr)	$\mathsf{unsat}(t)$	$\leftarrow$	$\mathsf{irr}(u, c)$, $\mathsf{tripled}(x, u, x, c, t)$.
(prl-sat)		$\leftarrow$	$\mathsf{unsat}(\mathsf{main})$.

Table 4: Global, local and output rules

Global input rules $I_{glob}(\mathrm{G})$

(igl-subctx1) C ∈ **C** ›→ {subClass(C, Ctx, gm)}

(igl-subctx2) c ∈ **N** ›→ {insta(c, Ctx, gm, main)}

Local input rules $I_{loc}(\mathrm{K_m}, \mathrm{c})$

(ilc-subevalat) *eval*$(A, \mathrm{C}) \pm B$ ›→ {subEval$(A, \mathrm{C}, B, \mathrm{c})$}

(ilc-subevalr) *eval*$(R, \mathrm{C}) \pm T$ ›→ {subEvalR$(R, \mathrm{C}, T, \mathrm{c})$}

Local deduction rules P_{loc}

(plc-subevalat) instd(x, b, c, t) ← subEval(a, c_1, b, c), instd$(c', c_1, \mathrm{gm}, t)$, instd$(x, a, c', t)$.

(plc-subevalr) tripled(x, s, y, c, t) ← subEvalR(r, c_1, s, c), instd$(c', c_1, \mathrm{gm}, t)$, tripled$(x, r, y, c', t)$.

Output translation $O(\alpha, \mathrm{c})$

(o-concept) $A(a)$ ›→ {instd$(a, A, \mathrm{c}, \mathrm{main})$}

(o-role) $R(a, b)$ ›→ {tripled$(a, R, b, \mathrm{c}, \mathrm{main})$}

Table 5: Input rules $I_{\mathrm{D}}(S)$ for defeasible axioms

(id-inst)	$\mathrm{D}(A(a))$ ›→ { def insta(A, a). }	(id-forall)	$\mathrm{D}(A \pm \forall R.B)$ ›→ { def supforall(A, R, B). }
(id-triple)	$\mathrm{D}(R(a, b))$ ›→ { def triplea(R, a, b). }	(id-leqone)	$\mathrm{D}(A \pm \leq 1R.\top)$ ›→ { def supleqone(A, R). }
(id-ninst)	$\mathrm{D}(\neg A(a))$ ›→ { def ninsta(A, a). }	(id-subr)	$\mathrm{D}(R \pm S)$ ›→ { def subr(R, S). }
(id-ntriple)	$\mathrm{D}(\neg R(a, b))$ ›→ { def ntriplea(R, a, b). }		
(id-subc)	$\mathrm{D}(A \pm B)$ ›→ { def subclass(A, B). }	(id-subrc)	$\mathrm{D}(R \circ S \pm T)$ ›→ { def subrc(A_1, A_2, B). }
(id-subcnj)	$\mathrm{D}(A_1 \sqcap A_2 \pm B)$ ›→ { def subcnj(A_1, A_2, B). }	(id-dis)	$\mathrm{D}(\mathrm{Dis}(R, S))$ ›→ { def dis(R, S). }
(id-subex)	$\mathrm{D}(\exists R.A \pm B)$ ›→ { def subex(R, A, B). }	(id-inv)	$\mathrm{D}(\mathrm{Inv}(R, S))$ ›→ { def inv(R, S). }
(id-supex)	$\mathrm{D}(A \pm \exists R.\{a\})$ ›→ { def supex(A, R, a). }	(id-irr)	$\mathrm{D}(\mathrm{Irr}(R))$ ›→ { def irr(R). }

of overriding rules in P_{D} is shown in Table 6; they incorporate sufficient clashing sets for the clashing assumptions that are made in overriding (cf. Table A.17 in the Appendix).

(vi). *Inheritance rules*: P_{D} provides the rules for defeasible inheritance of axioms from the global context to the local contexts. E.g., the following rule propagates an atomic concept inclusion axiom: if the (possibly defeasible) axiom is in the program of the global context and applicable to a local instance, it is applied unless the latter is recognized as an exception.

$$\mathrm{instd}(x, z, c, t) \leftarrow \mathrm{subClass}(y, z, g), \mathrm{instd}(x, y, c, t),$$
$$\mathrm{prec}(c, g), \mathrm{not}\ \mathrm{ovr}(\mathrm{subClass}, x, y, z, c).$$

The inheritance rules in P_{D} are shown in Table 7. Similar as for the rules P_{rl} above, the predicate unsat is used to indicate inconsistency. Note that such rules are applied both to the defeasible and strict axioms: in the latter case, the axioms are always inherited by the lower contexts, as no exception can arise.

(vii). *Test rules*: the last kind of rules in P_{D} are the test rules, which are used to instantiate and define the "environments" for the tests for negative literals in overriding rules. Intuitively, the asserted instance knowledge from the input CKR is stated to belong to the main test environment (note, e.g., the input translation rules for $A(b)$ and $R(a, b)$). Additional test environments are generated when a proof for contradiction of a negative literal is needed

Table 6: Deduction rules P_D for defeasible axioms: overriding rules

(ovr-inst)	$\text{ovr}(\text{insta}, x, y, c) \leftarrow$	$\text{def\_insta}(x, y), \text{prec}(c, g), \text{not test\_fails}(\text{nlit}(x, y, c)).$
(ovr-triple)	$\text{ovr}(\text{triplea}, x, r, y, c) \leftarrow$	$\text{def\_triplea}(x, r, y), \text{prec}(c, g), \text{not test\_fails}(\text{nrel}(x, r, y, c)).$
(ovr-ninst)	$\text{ovr}(\text{ninsta}, x, y, c) \leftarrow$	$\text{def\_ninsta}(x, y), \text{prec}(c, g), \text{instd}(x, z, c, \text{main}).$
(ovr-ntriple)	$\text{ovr}(\text{ntriplea}, x, r, y, c) \leftarrow$	$\text{def\_ntriplea}(x, r, y), \text{prec}(c, g), \text{tripled}(x, r, y, c, \text{main}).$
(ovr-subc)	$\text{ovr}(\text{subClass}, x, y, z, c) \leftarrow$	$\text{def\_subclass}(y, z), \text{prec}(c, g), \text{instd}(x, y, c, \text{main}),$ $\text{not test\_fails}(\text{nlit}(x, z, c)).$
(ovr-cnj)	$\text{ovr}(\text{subConj}, x, y_1, y_2, z, c) \leftarrow$	$\text{def\_subcnj}(y_1, y_2, z), \text{prec}(c, g), \text{instd}(x, y_1, c, \text{main}),$ $\text{instd}(x, y_2, c, \text{main}), \text{not test\_fails}(\text{nlit}(x, z, c)).$
(ovr-subex)	$\text{ovr}(\text{subEx}, x, r, y, z, c) \leftarrow$	$\text{def\_subex}(r, y, z), \text{prec}(c, g), \text{tripled}(x, r, w, c, \text{main}),$ $\text{instd}(w, y, c, \text{main}), \text{not test\_fails}(\text{nlit}(x, z, c)).$
(ovr-supex)	$\text{ovr}(\text{supEx}, x, y, r, w, c) \leftarrow$	$\text{def\_supex}(y, r, w), \text{prec}(c, g),$ $\text{instd}(x, y, c, \text{main}), \text{not test\_fails}(\text{nrel}(x, r, w, c)).$
(ovr-forall)	$\text{ovr}(\text{supForall}, x, y, z, r, w, c) \leftarrow$	$\text{def\_supforall}(z, r, w), \text{prec}(c, g), \text{instd}(x, z, c, \text{main}),$ $\text{tripled}(x, r, y, c, \text{main}), \text{not test\_fails}(\text{nlit}(y, w, c)).$
(ovr-leqone)	$\text{ovr}(\text{supLeqOne}, x, x_1, x_2, z, r, c) \leftarrow$	$\text{def\_supleqone}(z, r), \text{prec}(c, g), \text{instd}(x, z, c, \text{main}),$ $\text{tripled}(x, r, x_1, c, \text{main}), \text{tripled}(x, r, x_2, c, \text{main}),$
(ovr-subr)	$\text{ovr}(\text{subRole}, x, y, r, s, c) \leftarrow$	$\text{def\_subr}(r, s), \text{prec}(c, g), \text{tripled}(x, r, y, c, \text{main}),$ $\text{not test\_fails}(\text{nrel}(x, s, y, c)).$
(ovr-subrc)	$\text{ovr}(\text{subRChain}, x, y, z, r, s, t, c) \leftarrow$	$\text{def\_subrc}(r, s, t), \text{prec}(c, g), \text{tripled}(x, r, y, c, \text{main}),$ $\text{tripled}(y, s, z, c, \text{main}), \text{not test\_fails}(\text{nrel}(x, t, z, c)).$
(ovr-dis)	$\text{ovr}(\text{dis}, x, y, r, s, c) \leftarrow$	$\text{def\_dis}(r, s), \text{prec}(c, g), \text{tripled}(x, r, y, c, \text{main}),$ $\text{tripled}(x, s, y, c, \text{main}).$
(ovr-inv1)	$\text{ovr}(\text{inv}, x, y, r, s, c) \leftarrow$	$\text{def\_inv}(r, s), \text{prec}(c, g), \text{tripled}(x, r, y, c, \text{main}),$ $\text{not test\_fails}(\text{nrel}(x, s, y, c)).$
(ovr-inv2)	$\text{ovr}(\text{inv}, x, y, r, s, c) \leftarrow$	$\text{def\_inv}(r, s), \text{prec}(c, g), \text{tripled}(y, s, x, c, \text{main}),$ $\text{not test\_fails}(\text{nrel}(x, r, y, c)).$
(ovr-irr)	$\text{ovr}(\text{irr}, x, R, c) \leftarrow$	$\text{def\_irr}(r), \text{prec}(c, g), \text{tripled}(x, r, x, c, \text{main}).$

(cf. nlit(x, z, c) in the previous overriding rule example): the environment consists of a copy of the original program to which a positive version of the literal is added to the context in which the overriding is tested. If an inconsistency is found, then the test is successful, otherwise the test fails.

A first set of rules is used to instantiate the tests on the base of the form of defeasible axioms. For example, for atomic inclusions, the rule reads as:

$$\text{test}(\text{nlit}(x, z, c)) \leftarrow \text{def\_subclass}(y, z), \text{instd}(x, y, c, \text{main}), \text{prec}(c, g).$$

Similarly, a set of constraints makes sure that, if the test fails, no overriding can take place. For example, for the subClass overriding, we have:

$$\leftarrow \text{test\_fails}(\text{nlit}(x, z, c)), \text{ovr}(\text{subClass}, x, y, z, c).$$

A test fails if no clashes (i.e. instances of the predicate unsat) can be found. This is expressed by the rule:

$$\text{test\_fails}(\text{nlit}(x, z, c)) \leftarrow \text{instd}(x, z, c, \text{nlit}(x, z, c)), \text{not unsat}(\text{nlit}(x, z, c)).$$

Table 7: Deduction rules P_D for defeasible axioms: inheritance rules

(prop-inst)	$\text{instd}(x, z, c, t)$	$\leftarrow \text{insta}(x, z, g, t), \text{prec}(c, g), \text{not ovr}(\text{insta}, x, z, c).$
(prop-triple)	$\text{tripled}(x, r, y, c, t)$	$\leftarrow \text{triplea}(x, r, y, g, t), \text{prec}(c, g), \text{not ovr}(\text{triplea}, x, r, y, c).$
(prop-ninst)	$\text{unsat}(t)$	$\leftarrow \text{ninsta}(x, z, g, t), \text{instd}(x, z, c, t),$ $\text{prec}(c, g), \text{not ovr}(\text{ninsta}, x, z, c).$
(prop-ntriple)	$\text{unsat}(t)$	$\leftarrow \text{ntriplea}(x, r, y, g, t), \text{tripled}(x, r, y, c, t),$ $\text{prec}(c, g), \text{not ovr}(\text{ntriplea}, x, r, y, c).$
(prop-subc)	$\text{instd}(x, z, c, t)$	$\leftarrow \text{subClass}(y, z, g), \text{instd}(x, y, c, t),$ $\text{prec}(c, g), \text{not ovr}(\text{subClass}, x, y, z, c).$
(prop-cnj)	$\text{instd}(x, z, c, t)$	$\leftarrow \text{subConj}(y_1, y_2, z, g), \text{instd}(x, y_1, c, t), \text{instd}(x, y_2, c, t),$ $\text{prec}(c, g), \text{not ovr}(\text{subConj}, x, y_1, y_2, z, c).$
(prop-subex)	$\text{instd}(x, z, c, t)$	$\leftarrow \text{subEx}(v, y, z, g), \text{tripled}(x, v, x', c, t), \text{instd}(x', y, c, t),$ $\text{prec}(c, g), \text{not ovr}(\text{subEx}, x, v, y, z, c).$
(prop-supex)	$\text{tripled}(x, r, x', c, t)$	$\leftarrow \text{supEx}(y, r, x', g), \text{instd}(x, y, c, t),$ $\text{prec}(c, g), \text{not ovr}(\text{supEx}, x, y, r, x', c).$
(prop-forall)	$\text{instd}(y, z', c, t)$	$\leftarrow \text{supForall}(z, r, z', g), \text{instd}(x, z, c, t), \text{tripled}(x, r, y, c, t),$ $\text{prec}(c, g), \text{not ovr}(\text{supForall}, x, y, z, r, z', c).$ $\text{tripled}(x, r, x_1, c, t), \text{tripled}(x, r, x_2, c, t),$ $\text{prec}(c, g), \text{not ovr}(\text{supLeqOne}, x, x_1, x_2, z, r, c).$
(prop-subr)	$\text{tripled}(x, w, x', c, t)$	$\leftarrow \text{subRole}(v, w, g), \text{tripled}(x, v, x', c, t),$ $\text{prec}(c, g), \text{not ovr}(\text{subRole}, x, y, v, w, c).$
(prop-subrc)	$\text{tripled}(x, w, z, c, t)$	$\leftarrow \text{subRChain}(u, v, w, g), \text{tripled}(x, u, y, c, t), \text{tripled}(y, v, z, c, t),$ $\text{prec}(c, g), \text{not ovr}(\text{subRChain}, x, y, z, u, v, w, c).$
(prop-dis)	$\text{unsat}(t)$	$\leftarrow \text{dis}(u, v, g), \text{tripled}(x, u, y, c, t), \text{tripled}(x, v, y, c, t),$ $\text{prec}(c, g), \text{not ovr}(\text{dis}, x, y, u, v, c).$
(prop-inv1)	$\text{tripled}(y, v, x, c, t)$	$\leftarrow \text{inv}(u, v, g), \text{tripled}(x, u, y, c, t),$ $\text{prec}(c, g), \text{not ovr}(\text{inv}, x, y, u, v, c).$
(prop-inv2)	$\text{tripled}(x, u, y, c, t)$	$\leftarrow \text{inv}(u, v, g), \text{tripled}(y, v, x, c, t),$ $\text{prec}(c, g), \text{not ovr}(\text{inv}, x, y, u, v, c).$
(prop-irr)	$\text{unsat}(t)$	$\leftarrow \text{irr}(u, g), \text{tripled}(x, u, x, c, t),$ $\text{prec}(c, g), \text{not ovr}(\text{irr}, x, u, c).$

Finally, a set of rules generates the test environment by copying the instance knowledge from main and adding the complement of the tested literal. E.g., the following two rules copy all the (class assertion) instance data from main and add the complement for nlit:

$$\text{instd}(x_1, y_1, c, t) \leftarrow \text{instd}(x_1, y_1, c, \text{main}), \text{test}(t).$$
$$\text{instd}(x, z, c, \text{nlit}(x, z, c)) \leftarrow \text{test}(\text{nlit}(x, z, c)).$$

The set of test rules in P_D is shown in Table 8.

(viii). *Output rules*: Finally, the rules in $O(\alpha, \mathrm{c})$ translate ABox assertions that can be verified to hold in context c by applying the rules of the final program. For example, assertions $A(a)$ in a given context c are translated by $A(a) \mapsto \{\text{instd}(a, A, \mathrm{c}, \text{main})\}$ The rules in O are listed in Table 4.

Table 8: Deduction rules P_D for defeasible axioms: test rules

(test-inst)	test(nlit(x, y, c))	← def_insta(x, y), prec(c, g).
(constr-inst)		← test_fails(nlit(x, y, c)), ovr(insta, x, y, c).
(test-triple)	test(nrel(x, r, y, c))	← def_triplea(x, r, y), prec(c, g).
(constr-triple)		← test_fails(nrel(x, r, y, c)), ovr(triplea, x, r, y, c).
(test-subc)	test(nlit(x, z, c))	← def_subclass(y, z), instd(x, y, c, main), prec(c, g).
(constr-subc)		← test_fails(nlit(x, z, c)), ovr(subClass, x, y, z, c).
(test-subcnj)	test(nlit(x, z, c))	← def_subcnj(y_1, y_2, z), instd(x, y_1, c, main), instd(x, y_2, c, main), prec(c, g).
(constr-subcnj)		← test_fails(nlit(x, z, c)), ovr(subConj, x, y_1, y_2, z, c).
(test-subex)	test(nlit(x, z, c))	← def_subex(r, y, z), tripled(x, r, w, c, main), instd(w, y, c, main), prec(c, g).
(constr-subex)		← test_fails(nlit(x, z, c)), ovr(subEx, x, r, y, z, c).
(test-supex)	test(nrel(x, r, w, c))	← def_supex(y, r, w), instd(x, y, c, main), prec(c, g).
(constr-supex)		← test_fails(nrel(x, r, w, c)), ovr(supEx, x, r, y, w, c).
(test-supforall)	test(nlit(y, w, c))	← def_supforall(z, r, w), instd(x, z, c, main), tripled(x, r, y, c, main), prec(c, g).
(constr-supforall)		← test_fails(nlit(y, w, c)), ovr(supForall, x, y, z, r, w, c).
(test-subr)	test(nrel(x, s, y, c))	← def_subr(r, s), tripled(x, r, y, c, main), prec(c, g).
(constr-subr)		← test_fails(nrel(x, s, y, c)), ovr(subRole, x, r, y, s, c).
(test-subrc)	test(nrel(x, t, z, c))	← def_subrc(r, s, t), tripled(x, r, y, c, main), tripled(y, s, z, c, main), prec(c, g).
(constr-subrc)		← test_fails(nrel(x, t, z, c)), ovr(subRChain, x, y, z, r, s, t, c).
(test-inv1)	test(nrel(x, s, y, c))	← def_inv(r, s), tripled(x, r, y, c, main), prec(c, g).
(test-inv2)	test(nrel(y, r, x, c))	← def_inv(r, s), tripled(x, s, y, c, main), prec(c, g).
(constr-inv1)		← not test_fails(nrel(x, s, y, c)), ovr(inv, x, y, r, s, c).
(constr-inv2)		← not test_fails(nrel(y, r, x, c)), ovr(inv, x, y, r, s, c).
(test-fails1)	test_fails(nlit(x, z, c))	← instd(x, z, c, nlit(x, z, c)), not unsat(nlit(x, z, c)).
(test-fails2)	test_fails(nrel(x, r, y, c))	← tripled(x, r, y, c, nrel(x, r, y, c)), not unsat(nrel(x, r, y, c)).
(test-add1)	instd(x, z, c, nlit(x, z, c))	← test(nlit(x, z, c)).
(test-add2)	tripled(x, r, y, c, nrel(x, r, y, c))	← test(nrel(x, r, y, c)).
(test-copy1)	instd(x_1, y_1, c, t)	← instd(x_1, y_1, c, main), test(t).
(test-copy2)	tripled(x_1, r, y_1, c, t)	← tripled(x_1, r, y_1, c, main), test(t).

We remark that the translation parts as presented above include all rules that are structurally expected. Logical optimization by eliminating some rules or constraints is possible (e.g. (constr-subc) can be omitted as (ovr-subc) is the single rule defining subclass overriding), but we refrain from this here.

5.2.2. Translation process

We describe in the following the "translation process" to produce, given a CKR $\mathsf{K} = \langle \mathsf{G}, \{\mathsf{K_m}\}_{\mathsf{m} \in \mathsf{M}} \rangle$ in $\mathcal{SROIQ}$-RLD normal form, a program $PK(\mathsf{K})$ that encodes query answering from the CKR-models of K:

1. the *global program* for G is constructed as (where gm, gk are new context names):

$$PG(\mathsf{G}) = P_{rl} \cup I_{glob}(\mathsf{G_r}) \cup I_{\mathrm{D}}(\mathsf{G}_\Sigma) \cup I_{rl}(\mathsf{G_r}, \mathsf{gm}) \cup I_{rl}(\mathsf{G}_\Sigma \cup \mathsf{G}^{\mathrm{D}}_\Sigma, \mathsf{gk})$$

   where $\mathsf{G_r} = \mathsf{G} \cap \mathsf{L_r}$, $\mathsf{G}_\Sigma = \mathsf{G} \cap \mathsf{L}^{\mathrm{D}}_\Sigma$ and $\mathsf{G}^{\mathrm{D}}_\Sigma = \{\alpha \in \mathsf{L}_\Sigma \mid \mathrm{D}(\alpha) \in \mathsf{G}_\Sigma\}$. Intuitively, $PG(\mathsf{G})$ encodes all metaknowledge information in facts with a context parameter gm, and it encodes the global knowledge (including defeasible axioms)[16] in facts with a parameter gk. Notably, $PG(\mathsf{G})$ is a datalog program without negation, and hence it has a unique answer set (which is its least model), if it has a model.

2. We define the set of contexts $\mathbf{N}_{\mathsf{G}}$ as

$$\mathbf{N}_{\mathsf{G}} = \{\mathsf{c} \in \mathbf{N} \mid PG(\mathsf{G}) \models \mathsf{instd}(\mathsf{c}, \mathsf{Ctx}, \mathsf{gm}, \mathsf{main})\},$$

   and for every $\mathsf{c} \in \mathbf{N}_{\mathsf{G}}$ its associated knowledge base K_{c} as

$$\mathrm{K}_{\mathsf{c}} = \bigcup \{\mathrm{K_m} \in \mathsf{K} \mid PG(\mathsf{G}) \models \mathsf{tripled}(\mathsf{c}, \mathsf{mod}, \mathsf{m}, \mathsf{gm}, \mathsf{main})\}.$$

3. We define for each $\mathsf{c} \in \mathbf{N}_{\mathsf{G}}$ the each *local program* $PC(\mathsf{c}, \mathsf{K})$ as

$$PC(\mathsf{c}, \mathsf{K}) := P_{rl} \cup P_{loc} \cup P_{\mathrm{D}} \cup I_{loc}(\mathrm{K}_{\mathsf{c}}, \mathsf{c}) \cup I_{rl}(\mathrm{K}_{\mathsf{c}}, \mathsf{c}) \cup \{\mathsf{prec}(\mathsf{c}, \mathsf{gk})\};$$

   that is, local programs encode the object knowledge in all modules associated with the context c as datalog facts and include $\mathcal{SROIQ}$-RL deduction rules P_{rl}, local deduction rules P_{loc} and propagation rules P_{D} for defeasible axioms.

4. Finally, the *CKR program* $PK(\mathsf{K})$ is defined as follow:

$$PK(\mathsf{K}) = PG(\mathsf{G}) \cup \bigcup_{\mathsf{c} \in \mathbf{N}_{\mathsf{G}}} PC(\mathsf{c}, \mathsf{K}) \tag{9}$$

Intuitively, the knowledge from the global program $PG(\mathsf{G})$, which is Horn, is passed on to the local programs $PC(\mathsf{c}, \mathsf{K})$. The contexts in $\mathbf{N}_{\mathsf{G}}$ are those relevant for CKR-inference, and we can focus on them.[17] At the local contexts c, clashing assumptions $\langle \alpha, \mathbf{e} \rangle$ are reflected by literals

[16] Note that defeasible axioms are added both in their translation I_{D} and as any other global knowledge axiom by $I_{rl}(\mathsf{G}^{\mathrm{D}}_\Sigma, \mathsf{gk})$.

[17] Technically, we could move Step 2 (construction of $\mathbf{N}_{\mathsf{G}}$) into the program $PK(\mathsf{K})$ itself, and use a generic local program $P(x)$ where the concrete contexts c and modules $\mathrm{K_m}$ for K_{c} are singled out using atoms $\mathsf{instd}(x, \mathsf{Ctx}, \mathsf{gm}, \mathsf{main})$ and $\mathsf{tripled}(x, \mathsf{mod}, \mathsf{m}, \mathsf{gm}, \mathsf{main})$ that act as guards in rules. The present construction is more readable.

ovr(α, **e**, c), where α is represented in a reified form; the answer set semantics ensures that these literals must be derived from rules whose bodies resemble clashing sets $S_{c,(\alpha,e)}$ for (α, **e**). In turn, the positive literals in $S_{c,(\alpha,e)}$ must be derived via the materialization calculus, and the negative literals via contradiction proofs defined by the test mechanism mentioned at the beginning of Section 5.2 and previously detailed. In all these derivations, the materialization rules for defeasible axiom must respect the ovr-assumptions.

Query answering K ⊨ c:α is then achieved by testing whether the query, translated into its datalog rendering $O(\alpha, c)$, is a consequence of $PK(\mathsf{K})$, i.e., whether $PK(\mathsf{K}) \models O(\alpha, c)$ holds; for global entailment and conjunctive queries, this is analogous.

Example 17. *We consider the translation of* K_{tourD} *from Example 5 into its CKR program* $PK(\mathsf{K}_{tourD})$. *In Step 1, the content of the global context* G *is translated to the global program* $PG(\mathsf{G})$. *In particular, this program contains the structure of the metaknowledge represented as facts, e.g.* insta(cultural_tourist, Ctx, gm, main) *and* triplea(cultural_tourist, mod, ctourist_m, gm, main). *By the rules in* I_D, $PG(\mathsf{G})$ *contains the translation of the defeasible axioms in* G. *E.g. for* $\mathrm{D}(Cheap \sqsubseteq Interesting)$, *it includes the atom*

$$\mathsf{def\_subclass}(Cheap, Interesting).$$

Note that the rules in I_{rl} *also add to* $PG(\mathsf{G})$ *the "non-defeasible" translation of this axiom:* subClass(*Cheap*, *Interesting*, gk). *Furthermore,* $PG(\mathsf{G})$ *also contains the translation of the global assertions Cheap(fbmatch) and Cheap(market):*

$$\mathsf{insta}(fbmatch, Cheap, \mathsf{gk}) \qquad \mathsf{insta}(market, Cheap, \mathsf{gk}).$$

In Step 2 of the translation process, the relevant contexts and their associations to the modules are determined. In particular, from the facts above and the rules prl-instd *and* prl-tripled, *we obtain that* cultural_tourist ∈ $\mathbf{N}_G$ *and that* $\mathrm{K}_{ctourist\_m} \subseteq \mathrm{K}_{cultural\_tourist}$. *Then, the local programs* $PC(\mathsf{c}, \mathsf{K}_{tourD})$ *for all contexts* c *are computed: in the case of context* cultural_tourist, *note that* $PC(\mathsf{cultural\_tourist}, \mathsf{K}_{tourD})$ *contains the fact* ¬ *Interesting(fbmatch), which is represented as* ninsta(*fbmatch*, *Interesting*, cultural_tourist). *In the translation of local programs, we also add the defeasibility deduction rules of* P_D, *defining the rules for overriding and defeasible propagation of the global knowledge: in particular, the following rule* ovr-subc *provides the condition for overriding of atomic inclusion axioms like the one considered in our example:*

$$\begin{aligned}\mathsf{ovr}(\mathsf{subClass}, x, y, z, c) \leftarrow\ & \mathsf{def\_subclass}(y, z), \mathsf{prec}(c, g), \mathsf{instd}(x, y, c, \mathsf{main}),\\ & \mathsf{not\ test\_fails}(\mathsf{nlit}(x, z, c)).\end{aligned}$$

Propagation of defeasible atomic inclusion axioms is defined by the rule prop-subc*:*

$$\mathsf{instd}(x, z, c, t) \leftarrow \mathsf{subClass}(y, z, g), \mathsf{instd}(x, y, c, t), \mathsf{prec}(c, g), \mathsf{not\ ovr}(\mathsf{subClass}, x, y, z, c).$$

In Step 3 of the translation, the final program $PK(\mathsf{K}_{tourD})$ *results as the union of* $PG(\mathsf{G})$ *and all the local programs, including* $PC(\mathsf{cultural\_tourist}, \mathsf{K}_{tourD})$.

Let us now consider what can be inferred from $PK(\mathsf{K}_{tourD})$. *From the contents of* G *and the context* cultural_tourist, *it is easy to verify that by the rules*

$$PK(\mathsf{K}_{tourD}) \models \mathsf{ovr}(\mathsf{subClass}, fbmatch, Cheap, Interesting, \mathsf{cultural\_tourist}).$$

This holds because the test (instantiated by the overriding rule on fbmatch) for negative literal nlit(*fbmatch*, *Interesting*, cultural_tourist) *succeeds:* ¬*Interesting(fbmatch) holds locally and Interesting(fbmatch) is added in the test environment, thus a contradiction is found. Hence the inheritance rule* prop-subc *is not applicable and* $PK(\mathsf{K}_{tourD}) \models \mathsf{instd}(fbmatch, Interesting,$

cultural_tourist, main). *On the other hand, since* $PK(\mathsf{K}_{tourD})$ = ovr(subClass, *market*, *Cheap*, *Interesting*, cultural_tourist), *rule* prop-subc *can be applied and*

$$PK(\mathsf{K}_{tourD}) \models \mathsf{instd}(market, Interesting, \mathsf{cultural\_tourist}, \mathsf{main}).$$

These results coincide with the semantic interpretation of overridings given in Example 12. ◻

5.3. Correctness

The presented rules and translation provide a sound and complete materialization calculus for instance checking (with respect to c-entailment) and conjunctive query answering on SROIQ-RLD CKRs in normal form. This can be shown by establishing a correspondence between minimal justified CKR-models of K and answer sets of $PK(\mathsf{K})$. Having considered UNA and named models in the definition of our translation, we can concentrate on showing the result on Herbrand models: thus, naming v in the definition of least CAS-models $\hat{\mathfrak{I}}_{\mathsf{K}}(\chi, \mathsf{v})$ is irrelevant and we will simplify the denotation of such models as $\hat{\mathfrak{I}}_{\mathsf{K}}(\chi)$.

Let $\mathfrak{I}_{CAS}$ be a justified named CAS-model. We define the set of corresponding overriding assumptions:

$$OVR(\mathfrak{I}_{CAS}) = \{\, \mathsf{ovr}(p(\mathbf{e})) \mid \langle \alpha, \mathbf{e}\rangle \in \chi(\mathsf{c}),\ I_{rl}(\alpha, \mathsf{c}) = p \,\}.$$

Intuitively, given a CAS-interpretation $\mathfrak{I}_{CAS} = \langle \mathfrak{M}, \mathfrak{I}, \chi\rangle$ we can define a corresponding Herbrand interpretation $I(\mathfrak{I}_{CAS})$ of the program $PK(\mathsf{K})$ by including the following atoms in it (see the Appendix for a formal definition):

- all facts of $PK(\mathsf{K})$;
- $\mathsf{instd}(a, A, \mathsf{c}, \mathsf{main})$, if $\mathcal{I}(\mathsf{c}) \models A(a)$;
- $\mathsf{tripled}(a, R, b, \mathsf{c}, \mathsf{main})$, if $\mathcal{I}(\mathsf{c}) \models R(a, b)$;
- each ovr-literal from $OVR(\mathfrak{I}_{CAS})$;
- each literal l with environment $t \neq \mathsf{main}$, if $\mathsf{test}(t) \in I(\mathfrak{I}_{CAS})$ and l is in the head of a rule $r \in grnd(PK(\mathsf{K}))$ with $Body(r) \subseteq I(\mathfrak{I}_{CAS})$;
- $\mathsf{test}(t)$, if $\mathsf{test\_fails}(t)$ appears in the body of an overriding rule r in $grnd(PK(\mathsf{K}))$ and the head of r is an ovr literal in $OVR(\mathfrak{I}_{CAS})$;
- $\mathsf{unsat}(t) \in I(\mathfrak{I}_{CAS})$, if adding the literal corresponding to t to the local interpretation of its context c violates some axiom of the local knowledge $\mathsf{K_c}$;
- $\mathsf{test\_fails}(t)$, if $\mathsf{unsat}(t) \notin I(\mathfrak{I}_{CAS})$.

Note that $\mathsf{unsat}(\mathsf{main})$ is not included in $I(\mathfrak{I}_{CAS})$. We can establish the following property, which informally says that the least models of the global context is represented in the least justified named CAS-models.

Proposition 12. *Let* $\mathsf{K} = \langle \mathsf{G}, \{\mathsf{K_m}\}_{\mathsf{m}\in\mathsf{M}}\rangle$ *be a CKR in* SROIQ*-RLD normal form. Then for every least justified CAS-model* $\hat{\mathfrak{I}}_{\mathsf{K}}(\chi) = \langle \hat{\mathfrak{M}}, \hat{\mathcal{I}}, \chi\rangle$, *it holds that* $\hat{\mathfrak{M}} = \mathfrak{M}_{\mathsf{G}}$, *where* $\mathfrak{M}_{\mathsf{G}}$ *is the least Herbrand model of* G.

This result follows from the intersection property of CAS-models (Proposition 7): it is easy to verify that the CAS-interpretation $\mathfrak{I}'_{CAS} = \langle \hat{\mathfrak{M}} \cap \mathfrak{M}_{\mathsf{G}}, \hat{\mathcal{I}}, \chi\rangle$ is also a CAS-model of the CKR K; as $\hat{\mathfrak{I}}_{\mathsf{K}}(\chi)$ is a least CAS-model, $\hat{\mathfrak{M}} = \mathfrak{M}_{\mathsf{G}}$ must hold.

The next proposition shows that the least Herbrand model $\mathfrak{M}_{\mathsf{G}}$ of G is reflected in the answer set of the global program $PG(\mathsf{G})$. Let $I(\mathfrak{M}_{\mathsf{G}})$ denoted the Herbrand interpretation for $PG(\mathsf{G})$ that is defined analogously to $I(\mathfrak{I}_{CAS})$ above for $PK(K)$.

Proposition 13. *Let* $\mathsf{K} = (\mathsf{G}, \{\mathsf{K_m}\}_{\mathsf{m}\in\mathsf{M}})$ *be a CKR in* SROIQ*-RLD normal form. If* G *is satisfiable, then* $I(\mathsf{M_G})$ *is the unique answer set of* $PG(\mathsf{G})$*; otherwise,* $PG(\mathsf{G})$ *has no answer sets.*

The main result on the correctness of the translation is achieved by showing that the answer sets of the final program $PK(\mathsf{K})$ correspond with the least justified models of K as follows:

Lemma 6. *Let* K *be a CKR in* SROIQ*-RLD normal form. Then*

(i). for every (named) justified clashing assumption χ*, the interpretation* $S = I(\hat{\mathfrak{I}}(\chi))$ *is an answer set of* $PK(\mathsf{K})$*;*

(ii). every answer set S *of* $PK(\mathsf{K})$ *is of the form* $S = I(\hat{\mathfrak{I}}(\chi))$ *where* χ *is a (named) justified clashing assumption for* K.

The correctness result for instance checking of atomic concepts and roles is then an easy consequence of Lemma 6 and Corollary 1 (cf. the discussion at the end of Section 4.1; negative instance checking can be reduced to unsatisfiability).

Theorem 6. *Let* K *be a CKR in* SROIQ*-RLD normal form, and let* α *and* c *such that* $O(\alpha, \mathsf{c})$ *is defined. Then* $\mathsf{K} \models \mathsf{c} : \alpha$ *iff* $PK(\mathsf{K}) \models O(\alpha, \mathsf{c})$.

Similarly, we obtain the correctness for conjunctive query answering from the correspondence of Lemma 6. Given a logic program P and a conjunction $q(\mathbf{t}) = p_1(\mathbf{t}_1) \wedge \cdots \wedge p_m(\mathbf{t}_m)$ of atoms $p_i(\mathbf{t}_i)$ in its language, where all variables in $\mathbf{t}_i$ occur in $\mathbf{t}$, we say that P entails $q(\mathbf{t})$, denoted $P \models q(\mathbf{t})$, if for every answer set I some ground instance $q(\mathbf{c}) = p_1(\mathbf{c}_1) \wedge \cdots \wedge p_m(\mathbf{c}_m)$ of $q(\mathbf{t})$ exists such that $I \models q(\mathbf{c})$.

Now for a Boolean CQ $Q = \exists \mathbf{y}\gamma(\mathbf{y})$ on K, where $\gamma(\mathbf{y}) = \gamma_1 \wedge \cdots \wedge \gamma_m$ and $\gamma_i = \mathsf{c}_i : \alpha_i(\mathbf{t}_i)$, let $O(Q) = O(\alpha_i(\mathbf{t}_1), \mathsf{c}_1) \wedge \cdots \wedge O(\alpha_i(\mathbf{t}_m), \mathsf{c}_m)$ denote its translation into the corresponding conjunction of atoms of a logic program, where variables are treated like special constants. Then we obtain:

Theorem 7. *Let* K *be a CKR in* SROIQ*-RLD normal form, and let* $Q = \exists \mathbf{y}\gamma(\mathbf{y})$ *be a Boolean CQ on* K*. Then* $\mathsf{K} \models Q$ *iff* $PK(\mathsf{K}) \models O(Q)$.

As above, this result is a consequence of Lemma 6 and Corollary 1. Furthermore, it naturally generalizes to the certain answers of general conjunctive queries.

5.4. *Discussion: justification safeness*

Test environments are needed to check the derivation of negative literals in the clashing sets and thus to assure completeness of justifications. Still, this proof-by-contradiction encoding is less natural than a direct encoding of negative reasoning, where strong negation is used to represent negative instance-level literals and rules are used to conclude negative facts by modus ponens.

However, such a direct encoding involves reasoning on disjunctive knowledge which is not easy to represent using ASP interpretation of disjunctive rules [27]. For example, consider the negative version of rule (prl-subcnj) to reason on negative instances of $A_1 \sqcap A_2 \sqsubseteq B$. Using disjunction in the head of rules, one could write the rule as:

$$\neg\mathsf{instd}(x, y_1, c) \vee \neg\mathsf{instd}(x, y_2, c) \leftarrow \mathsf{subConj}(y_1, y_2, z, c), \neg\mathsf{instd}(x, z, c).$$

As noted in the example in Section 5.2, this would lead to a calculus that is incomplete with respect to negative reasoning. For example, from $A \sqcap B \sqsubseteq C$, $\neg C(a)$, $D \sqsubseteq A \sqcap B$ we can classically infer $\neg D(a)$, but neither $\neg A(a)$ nor $\neg B(a)$: however, the interpretation of the rule above would lead to two distinct answer sets, one in which $\neg A(a)$ holds and one where $\neg B(a)$ holds, but in neither of them $\neg D(a)$ is inferred.

One possible solution is to require a notion of *justification safeness* for the input knowledge base. Intuitively, this condition guarantees that whenever an axiom gives rise to reasoning on negative disjunctive cases, one of the disjuncts is provable from the knowledge base. Then, proofs for justifications do not depend on non-deterministic choices. For instance, in the example above, a knowledge base K containing $A \sqcap B \sqsubseteq C$ would be justification safe if, whenever $\neg C(a)$ is derivable, either $\neg A(a)$ or $\neg B(a)$ can be derived from K. If the input CKR is justification safe, the translation can be modified by omitting the test environment mechanism and using direct reasoning on negative instance-level literals (sample deduction rules for this setting are shown in Table A.18).

Furthermore, in such a modified program we could also recognize violations of safeness by reasoning inside the program (namely, on the least justified CAS-models $\hat{I}(\chi)$). In the case of $A \sqcap B \sqsubseteq C$, a violation of safety can be recognized with the following rule:

$$\mathsf{unsafe} \leftarrow \mathsf{subConj}(y_1, y_2, z, c), \neg\mathsf{instd}(x, z, c), \mathsf{not}\ \neg\mathsf{instd}(x, y_1, c), \mathsf{not}\ \neg\mathsf{instd}(x, y_2, c).$$

By adding rules of this kind, if unsafe is derived we recognize that the input CKR was not justification safe and thus *it might happen that* some of the justifications are not established in an answer set. Justification safeness, however, ensures completeness of justification in each answer set. Furthermore, the direct encoding of negative reasoning would also be complete for both positive and negative instance queries.

6. CKR*ew*: CKR datalog Rewriter Prototype

The datalog translation from above has been implemented in a prototype called CKR*ew* *(CKR datalog Rewriter)*. After a brief description of its structure and implementation details, we will report on an experimental evaluation with respect to performance and different degrees of defeasibility.

6.1. Prototype description

CKR*ew* has been implemented as a Java-based command line application. It accepts as input global and local modules of the initial CKR represented as RDF files (either as distinct N3 RDF files or as a single TRIG file) that contain OWL-RL axioms in normal form and produces as output a single .dlv text file that contains the complete datalog rewriting for the input CKR. The newly added contextual primitives have been defined in an RDF vocabulary (imported in the translation); in particular, axiom defeasibility assertions have been encoded as OWL axiom annotations hasAxiomType having the value defeasible. The conceptual system architecture is depicted in Figure 2. The prototype takes advantage of the DL-to-datalog rewriter *DReW* [40], which is used in the translation of global and local OWL axioms into their datalog counterparts. The loading of OWL-RL RDF files is managed using the *OWL API 3.4*.[18] The CKR system structure

[18] http://owlcs.github.io/owlapi/

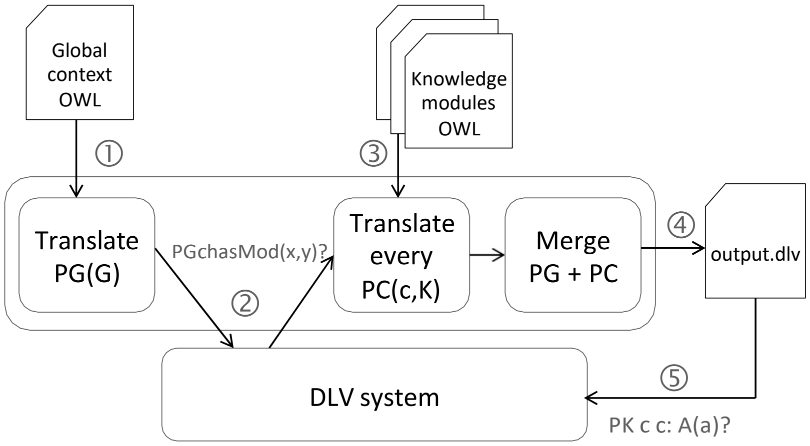


Figure 2: CKR*ew* architecture and dynamic translation process

is managed by the prototype; external calls to the *DLV solver*[19] by means of the *DLVWrapper* Java library [41] are used to determine the set of contexts and their module associations, which are extracted from the computed answer set(s) of the global program $PG(\mathsf{G})$.

The translation process, which is depicted in Figure 2, follows the strategy in Section 5. After checking that the CKR represented in the input files is in the required $\mathcal{SROIQ}$-RLD normal form, the system proceeds to produce the rewriting. First of all, the global module is translated (step ⊠), basically using translation rules from I_{glob} and I_{rl}; if an axiom is recognized as defeasible, the corresponding defeasible declaration in I_D is added to the program. The global program $PG(\mathsf{G})$ is completed by adding the deduction rules from P_{rl}. The set of contexts and their association to local modules are then computed by submitting the global program to DLV and retrieving the instances of Context concept and hasModule role in the resulting answer sets (step ②). Using this information, the prototype computes local knowledge bases for all contexts and applies the rewriting process to compute the local program $PC(\mathsf{c}, \mathsf{K})$ for each of them using rules in I_{loc} and I_{rl} (step ③). The resulting program is completed with deduction rules P_{loc} and P_D and saved in a file (step №.). The final program $PK(\mathsf{K})$ is then evaluated using the DLV solver, resorting to the syntax defined by the output translation O (step ⑤). Note that DLV supports query answering, and also allows the evaluation of non-ground conjunctive queries on the produced CKR program.

A demo of the prototype, together with RDF files implementing the examples in [2, 1], can be found at http://ckrew.fbk.eu/.

6.2. Experimental evaluation

In this section, we describe an experimental evaluation that we have conducted to measure the performance of the prototype, which is similar in spirit to the evaluation of the RDF based implementation of non-defeasible CKRs in [42]. In particular, we want to study the behavior of the translation and the resulting program in presence of different dimensions of the input CKR or degrees of defeasibility. We note, however, that scalability of the approach is inherently

[19] http://www.dlvsystem.com/dlv/

limited by the coNP-completeness of the reasoning problems (reaching Π^p_2-completeness in case of conjunctive queries).

6.2.1. Generation of synthetic testsets

To create our test sets, we developed a simple generator that can output synthetically generated CKRs with certain features. For each generated CKR, the generator takes as input:

- the number n of contexts (i.e. local named graphs) to be generated;
- the dimensions of the signature to be declared (number m of base classes, l of properties and k of individuals);
- the number of axioms for the global and local modules (number of global TBox, ABox and RBox axioms and number of TBox, ABox and RBox axioms per context);
- optionally, the number of additional local *eval* axioms and the number of individuals to be propagated across contexts.
- optionally, the percentage of axioms in the global context to be declared as defeasible and the percentage of local overriding instances of such axioms.

Intuitively, the generation of a CKR proceeds as follows:

1. The contexts (named : c0, . . . , : cn) are declared in the global context named graph and are linked to a different module name (: m0, . . . , : mn), corresponding to the named graph containing their local knowledge.
2. Base classes (named : A0, . . . , : Am), object properties (: R0, . . . , : Rl) and individuals (: a0, . . . , : ak) are added to the global graph: these symbols are used in the generation of global and local axioms.
3. Then generation of global axioms takes place. We chose to generate axioms as follows, in order to create realistic instances of knowledge bases:

   - Classes and properties names are taken from the base signature using random selection criteria in the form of (the positive part of) a Gaussian curve centered in 0: intuitively, classes equal or near to : A0 are more probable in axioms than : Am.
   - Individuals are randomly selected using a uniform distribution.
   - TBox, ABox and RBox axioms in SROIQ-RL are added in the requested number to the global context module following the percentages shown in Table 9 (note that the reported axioms are normal form SROIQ-RL axioms). Such percentages have been manually selected in order to simulate the common distribution in the use of the SROIQ-RL constructs in real knowledge bases.

   The rationale behind such choices for the generation is to produce knowledge bases with a reasonable knowledge structure. Moreover, we want to guarantee that all of the constructs in the language are represented in every generated knowledge base and used in a non-random and rational way; our goal is to avoid unfair behaviors in the experiments due to the lack or over-use of some language constructs.
4. The same generation criteria are then applied in the case of local graphs representing the local knowledge of contexts.

Table 9: Percentages of generated axioms

TBox axiom	%
$A \sqsubseteq B$	50%
$A \sqcap B \sqsubseteq \bot$	20%
$A \sqsubseteq \exists R.\{a\}$	10%
$A \sqcap B \sqsubseteq C$	5%
$\exists R.A \sqsubseteq B$	5%
$A \sqsubseteq \forall R.B$	5%
$A \sqsubseteq \leq 1 R.B$	5%

ABox axiom	%
$A(a)$	50%
$R(a, b)$	40%
$\neg R(a, b)$	10%

RBox axiom	%
$R \sqsubseteq T$	50%
$\mathrm{Inv}(R, S)$	25%
$R \circ S \sqsubseteq T$	10%
$\mathrm{Dis}(R, S)$	10%
$\mathrm{Irr}(R)$	5%

5. If specified, the requested number for *eval* axioms of the form *eval*$(A, \mathrm{C}) \sqsubseteq B$ and a set of individuals in the scope of the *eval* operator (i.e. as local members of A) are added to local contexts graphs.
6. If specified, the requested percentage of global axioms is generated (using the same criteria as above) which are declared defeasible: in particular, in order to control the number of local overridings, the defeasible axioms are generated using "fresh" symbols (i.e. not occurring in the signature of other strict global or local axioms).
7. If defeasible axioms have been generated, a fixed number of instances is added to randomly chosen contexts. For example, for $\mathrm{D}(A \sqsubseteq B)$, in case of a positive instance $A(a)$ is added locally, while in case of a negative (exceptional) instance $A(a), \neg B(a)$ is added. The percentage of such instances that represent overridings (i.e. clashing sets) is specified by the user.

6.2.2. Evaluation setup

Evaluation experiments were carried out on a 4 core Dual Intel Xeon Processor machine with 32GB 1866MHz DDR3 RAM, standard S-ATA (7.200RPM) HDD, running a Linux RedHat 6.5 distribution. We allocated 10Gb of memory to the JVM running the prototype command line application and the utility scripts managing the upload, profiling and reporting of test instances. The datalog engine that we used to test the computation of the answer sets of the translated programs (and in the computation of the global context structure) is the latest DLV solver.[20]

Using the profiling methods, we could measure the time needed (1) to translate the whole CKR program, (2) to interact with DLV in computing the global model and (3) to compute the answer sets for the final program via DLV. We will use these values to evaluate our reasoning method with respect to different dimensions of input CKR and different levels of defeasibility.

6.2.3. TS1: Scalability evaluation

The first experiments served to determine the (average) translation and model computation time depending on the number of contexts and their contents. In other words, we wanted to answer the following question:

> *What is the effect on the amount of time requested for rewriting and model computation with respect to the number and size of contexts of a CKR?*

[20] http://www.dlvsystem.com/dlv/, build 2012-12-17

Using the CKR generator described above, we generated a suite of CKRs whose profiles are shown in Table 10. We will refer to this suite as TS1: this test set (basically corresponding to an extension of TS1 from [42]) contains sets of CKRs with an increasing number of contexts, in which CKRs have an increasing number of axioms; no eval-axioms or defeasible axioms were added during the generation.

We have tested the rewriting and answer set computation over 3 random generations of the TS1 profile and 5 independent test runs: the different generation instances of TS1 are necessary in order to reduce the impact of special cases in the random generation. The results of the experiments on TS1 are reported in Table 11. In the table, for each of the generated CKRs (referred by number of contexts and number of base classes in the first two columns), we show their number of total effective input axioms in column *Statements* (averaged on the 3 versions of TS1). The column *Prog. size* reports the corresponding dimensions of the output program in terms of program statements. The rewriting time is listed in the following two columns: *Gl. time* lists the (average) time in milliseconds for rewriting and computing the answer sets for the global context, and *RW time* lists the (average) time for the rewriting of the complete CKR program. The column *DLV time* lists the (average) time in milliseconds needed for computing the answer sets of the output program. A dash indicates a timeout, which was set to 20 minutes (1.200.000 ms).

In order to analyze the results, the behaviour of the rewriting and answer sets computation has been plotted to graphs, shown in Figure 3. Each of the series represents a set with a fixed number of contexts (1 to 100) and each point a CKR. The *x* axis represents the number of asserted statements, while the *y* axis shows the time in milliseconds. To better visualize the behaviour of the series, we plotted a trend line for each of the series: the lines represent an approximation of the data trend calculated by polynomial regression.[21]

Some conclusions can be drawn from these data and graphs. In particular, we note that the expected behaviour of the rewriting process and answer set computation is reflected by these results. About the rewriting, it is clear that the dimension of the output program (and the corresponding rewriting time) is basically linear in the dimension of the input CKR. In fact, the size of the output program can be determined quite precisely given the applied rules and the translation process: some variability may occur due to the translation of the local signatures, which is determined randomly in the axiom generation. The size of the output program can be estimated by considering how each of its components is built. The global program contains a fixed number of statements to represent CKR primitives and deduction rules; its variable part depends on the size of the signature, the number of global axioms and finally a fixed number of statement for each context (for context declaration, declaration of the associated module and module association). For each of the contexts, every local program needs a fixed number of statements for the declaration of CKR primitives and the local prec statement; then its size depends on the local signature declaration and the number of local axioms.

On the other hand, DLV answer sets computation for the final program is clearly not linearly dependent from the size of the program, and the computational hardness of the materialization solution is evident in the graphs growth. From the results, it is evident that the feasibility of the reasoning is affected by the number of contexts of the CKR: for example, this can be seen in Figure 3.b by comparing the case of 1 context and 1000 classes (having 7014 statements, with DLV time 33469 ms.) with the cases of 5 contexts and 350 classes (having 7396 statements, with DLV time 14554 ms.) and 50 contexts and 35 classes (having 6655 statements, with DLV

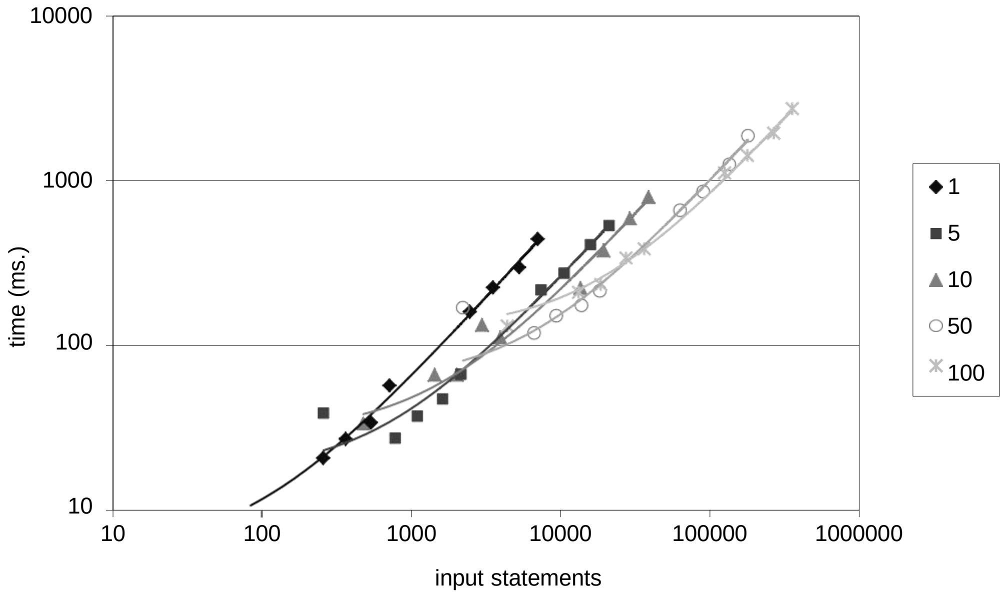
[21]Average R^2 value across all approximations is $\geq 0,996$.

Table 10: Test set TS1.

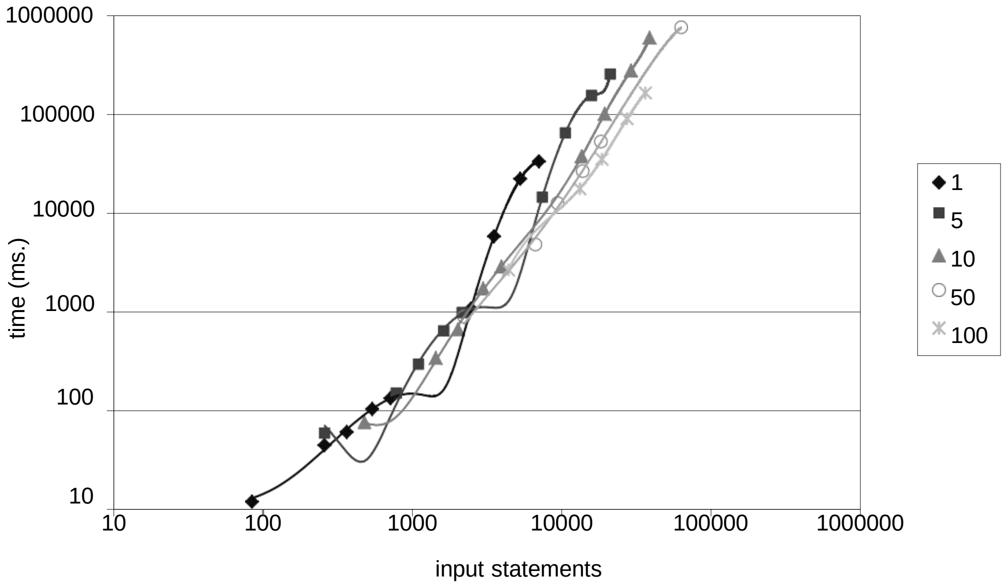
				Global KB			Local KBs			
Contexts	**Classes**	**Roles**	**Indiv.**	**TBox**	**RBox**	**ABox**	**TBox**	**RBox**	**ABox**	**Total ax.**
1	10	10	20	10	5	20	10	5	20	70
1	35	35	70	35	18	70	35	18	70	245
1	50	50	100	50	25	100	50	25	100	350
1	75	75	150	75	38	150	75	38	150	525
1	100	100	200	100	50	200	100	50	200	700
1	350	350	700	350	175	700	350	175	700	2.450
1	500	500	1000	500	250	1000	500	250	1000	3.500
1	750	750	1500	750	375	1500	750	375	1500	5.250
1	1000	1000	2000	1000	500	2000	1000	500	2000	7.000
5	10	10	20	10	5	20	10	5	20	210
5	35	35	70	35	18	70	35	18	70	735
5	50	50	100	50	25	100	50	25	100	1.050
5	75	75	150	75	38	150	75	38	150	1.575
5	100	100	200	100	50	200	100	50	200	2.100
5	350	350	700	350	175	700	350	175	700	7.350
5	500	500	1000	500	250	1000	500	250	1000	10.500
5	750	750	1500	750	375	1500	750	375	1500	15.750
5	1000	1000	2000	1000	500	2000	1000	500	2000	21.000
10	10	10	20	10	5	20	10	5	20	385
10	35	35	70	35	18	70	35	18	70	1.348
10	50	50	100	50	25	100	50	25	100	1.925
10	75	75	150	75	38	150	75	38	150	2.888
10	100	100	200	100	50	200	100	50	200	3.850
10	350	350	700	350	175	700	350	175	700	13.475
10	500	500	1000	500	250	1000	500	250	1000	19.250
10	750	750	1500	750	375	1500	750	375	1500	28.875
10	1000	1000	2000	1000	500	2000	1000	500	2000	38.500
50	10	10	20	10	5	20	10	5	20	1.785
50	35	35	70	35	18	70	35	18	70	6.248
50	50	50	100	50	25	100	50	25	100	8.925
50	75	75	150	75	38	150	75	38	150	13.388
50	100	100	200	100	50	200	100	50	200	17.850
50	350	350	700	350	175	700	350	175	700	62.475
50	500	500	1000	500	250	1000	500	250	1000	89.250
50	750	750	1500	750	375	1500	750	375	1500	133.875
50	1000	1000	2000	1000	500	2000	1000	500	2000	178.500
100	10	10	20	10	5	20	10	5	20	3.535
100	35	35	70	35	18	70	35	18	70	12.373
100	50	50	100	50	25	100	50	25	100	17.675
100	75	75	150	75	38	150	75	38	150	26.513
100	100	100	200	100	50	200	100	50	200	35.350
100	350	350	700	350	175	700	350	175	700	123.725
100	500	500	1000	500	250	1000	500	250	1000	176.750
100	750	750	1500	750	375	1500	750	375	1500	265.125
100	1000	1000	2000	1000	500	2000	1000	500	2000	353.500

Table 11: Scalability results for test set TS1.

Ctx.	Cls.	Statements	Prog. size	Gl. time	RW time	DLV time
1	10	84	278	6	7	12
1	35	258	672	18	21	45
1	50	364	909	23	27	61
1	75	539	1318	27	34	104
1	100	714	1716	45	57	134
1	350	2463	5724	131	161	1082
1	500	3514	8145	190	225	5845
1	750	5264	12157	250	298	22268
1	1000	7014	16169	379	444	33469
5	10	259	669	21	39	59
5	35	779	1874	18	28	151
5	50	1096	2586	22	37	297
5	75	1620	3785	30	48	646
5	100	2147	5000	42	67	987
5	350	7396	17029	136	218	14554
5	500	10548	24167	178	276	65133
5	750	15798	36322	279	410	155483
5	1000	21049	48299	348	535	255716
10	10	477	1189	17	34	77
10	35	1433	3367	36	67	343
10	50	2017	4705	39	67	668
10	75	2971	6891	66	134	1731
10	100	3941	9105	66	112	2890
10	350	13566	31164	116	224	37719
10	500	19343	44445	211	377	101350
10	750	28964	66390	339	591	277843
10	1000	38589	88373	475	798	598679
50	10	2217	5180	58	170	894
50	35	6655	15391	47	119	4803
50	50	9361	21547	55	152	12611
50	75	13797	31717	54	175	26564
50	100	18280	41961	67	215	52935
50	350	62908	144208	176	661	758959
50	500	89679	205503	229	860	—
50	750	134304	307741	380	1257	—
50	1000	178934	409724	493	1871	—
100	10	4392	10217	62	131	2661
100	35	13175	30304	77	211	17677
100	50	18534	42621	70	235	35234
100	75	27321	62633	98	340	90377
100	100	36206	83084	92	386	165006
100	350	124585	285522	177	1114	—
100	500	177609	406964	251	1424	—
100	750	265971	609292	262	1943	—
100	1000	354359	811666	362	2734	—



a) Rewriting time



b) DLV time

Figure 3: Scalability graphs for TS1.

time 4803 ms.). This suggest that the modularization of knowledge provided by contextual sub-programs may allow to limit local reasoning and manage larger numbers of local axioms. On the other hand, it is evident that while the rewriting is applicable to large datasets, as expected, the current materialization based translation does not allow to scale to very large number of (complex) statements and contextual structures.

6.2.4. TS2: Defeasibility evaluation

The second set of experiments over the CKR prototype served to determine the impact of defeasible axioms. Basically they were modelled inspired by the defeasibility evaluation of Casini et. al in [43]. The question is as follows.

> *Which effect does the percentage of global defeasible axioms and their overridings have on the time for rewriting and model computation?*

Using the CKR generator, we generated a second suite TS2 of CKRs. We fixed the number of contexts to 5 and number of global axioms to 175: basically, this amounts in size and form to the case of 5 contexts and 50 classes in TS1; this setting was chosen to fix a reasonable number of contexts and axioms to a case that was proved to be easily treatable from the tests on TS1. We generated 9 groups of CKRs with a percentage of global axioms declared as defeasible varying from 10% to 100%. Each group has 10 CKRs with different percentage of overridings (from 10% to 100%, with an increase of 10% across CKRs). No random local axioms have been generated; instead, 10 local instances of each (strict or defeasible) global axiom scheme have been generated. In case of defeasible axioms, these instances are negative (i.e. clashing sets) and their number yields the specified overriding percentage, while the other instances are positive. In this way, we keep the number of instances fixed. Intuitively, this allows us to verify the behaviour of the prototype in CKRs with equal size but different ratios of defeasibility and overridings.

We tested computing the rewriting and the answer sets over 5 instances of the TS2 profile and 5 independent test runs for each instance. Results of the experiments on TS2 are reported in Table 12, where the first two columns show the percentages of defeasible axioms and overridings, respectively. The following columns (number of statements, output program size, global and total rewriting time, and DLV computation time) are as in Table 11, where DLV time is for computing one answer set. Finally, last column reports the number of test instances (of the nlit kind) in the computed model.

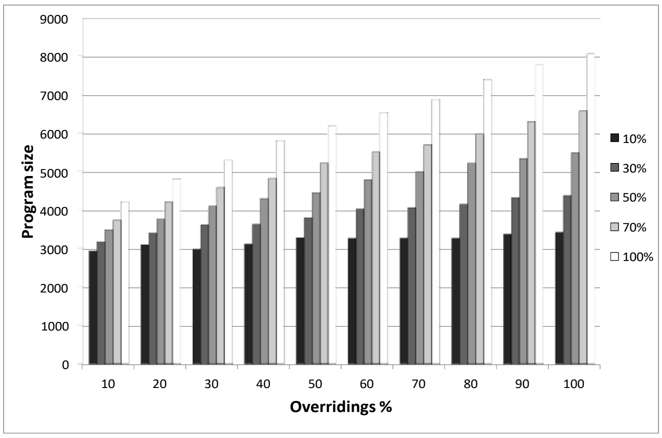
For a representative selection of the data, Figure 4 shows histograms for the output program size, the rewriting time and the DLV time. The *y*-axis represents the number of statements for program size and time in ms. for rewriting and DLV time respectively, while on the *x*-axis the bars are ordered by percentage of overridings (from 10% to 100%). Each series represents a different percentage of defeasibility (from 10% to 100%), i.e. a CKR group in Table 12.

As we can see from the data and the graphs, the program size grows linearly with number of defeasible axioms and overridings. As in the case of scalability, we can precisely count the number of added rules and statements in the output program by considering the translation process. As we specified above, in this experiment the numbers of global and local axioms, contexts and signature size are fixed. For each defeasible global axiom, one corresponding defeasibility atom (e.g. def_subclass) has to be added to the global program. The evident growth in the number of statements w.r.t. the number of overridings is also justified by the larger number of axioms that is needed in general to represent the negative instances of axioms. For example, in the case of an atomic subsumption $A \sqsubseteq B$, its positive instance is a single assertion axiom $A(a)$, while the

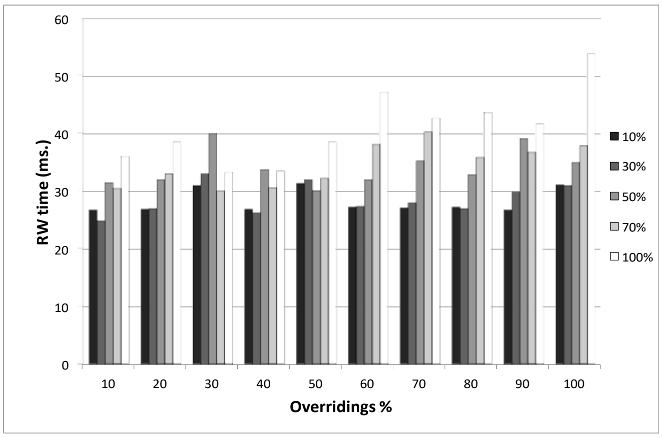
Table 12: Experiments results for test set TS2.

Def.%	Ovr.%	Stat.	Prog.	Gl.t.	RW	DLV	test
10	10	1115	2926	21	27	22113	41
10	20	1203	3092	20	27	21435	40
10	30	1151	2976	24	31	24874	45
10	40	1212	3110	21	27	29706	40
10	50	1288	3274	23	31	27518	47
10	60	1279	3257	20	27	33411	57
10	70	1287	3262	21	27	31699	55
10	80	1293	3259	19	27	32744	65
10	90	1331	3365	20	27	28046	64
10	100	1366	3415	23	31	42126	68
20	10	1193	3154	19	27	35921	78
20	20	1178	3154	21	29	58356	65
20	30	1239	3268	22	31	68159	81
20	40	1325	3489	19	27	55163	88
20	50	1369	3545	21	29	74582	96
20	60	1415	3604	19	28	81698	93
20	70	1429	3665	22	30	120530	103
20	80	1504	3763	21	28	79927	123
20	90	1561	3841	21	30	134042	126
20	100	1573	3905	20	28	110618	132
30	10	1160	3165	19	25	70188	96
30	20	1252	3394	20	27	68560	127
30	30	1351	3613	23	33	107571	130
30	40	1369	3624	19	26	91741	132
30	50	1459	3795	22	32	87452	142
30	60	1559	4020	19	27	131856	144
30	70	1595	4057	19	28	111634	164
30	80	1647	4142	19	27	111544	167
30	90	1755	4317	19	30	149951	204
30	100	1774	4373	22	31	176888	195
40	10	1223	3353	24	34	84649	143
40	20	1354	3715	20	30	108808	155
40	30	1379	3768	23	32	155393	157
40	40	1510	4032	20	31	152683	170
40	50	1650	4285	21	31	187733	202
40	60	1708	4392	22	35	192295	214
40	70	1823	4542	26	40	241755	226
40	80	1931	4777	23	35	255782	214
40	90	1969	4851	19	31	266191	251
40	100	2040	4943	21	32	305133	262
50	10	1224	3481	22	31	102882	158
50	20	1355	3754	21	32	159336	196
50	30	1506	4093	28	40	157123	211
50	40	1610	4294	23	34	184372	225
50	50	1687	4444	20	30	172795	236
50	60	1845	4783	21	32	238078	263
50	70	1977	4992	23	35	257234	271
50	80	2116	5210	20	33	315922	312
50	90	2179	5333	26	39	434312	332
50	100	2269	5482	23	35	349145	325

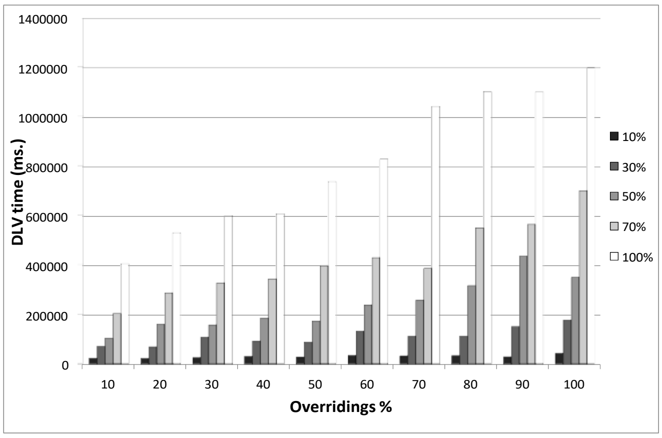
Def.%	Ovr.%	Stat.	Prog.	Gl.t.	RW	DLV	test
60	10	1297	3703	22	31	176656	226
60	20	1419	3951	21	29	180581	257
60	30	1561	4256	21	30	234177	272
60	40	1699	4567	21	32	273768	291
60	50	1886	4927	21	32	328445	312
60	60	1976	5075	23	36	330168	314
60	70	2125	5348	22	35	321582	351
60	80	2208	5487	21	35	410883	344
60	90	2419	5842	21	36	499053	382
60	100	2460	5906	20	32	430448	388
70	10	1274	3730	22	30	202731	232
70	20	1481	4198	24	33	286869	294
70	30	1659	4582	20	30	327092	278
70	40	1790	4815	21	31	343136	314
70	50	1978	5213	21	32	396284	356
70	60	2130	5497	25	38	428827	370
70	70	2247	5688	25	40	385229	396
70	80	2397	5968	23	36	548701	411
70	90	2613	6295	21	37	563161	442
70	100	2764	6570	22	38	699516	486
80	10	1302	3854	21	31	241660	277
80	20	1544	4399	22	32	311755	287
80	30	1706	4759	23	33	310633	333
80	40	1923	5216	19	28	448689	361
80	50	2058	5447	21	32	494499	399
80	60	2269	5871	22	36	635089	407
80	70	2442	6168	23	39	601622	453
80	80	2573	6373	22	40	659166	454
80	90	2861	6848	23	42	694975	519
80	100	2941	7006	21	36	753493	536
90	10	1303	3923	23	30	258290	321
90	20	1576	4560	24	34	368366	334
90	30	1750	4905	20	29	436950	359
90	40	2067	5538	22	36	571478	433
90	50	2279	5947	20	32	631558	435
90	60	2374	6129	22	37	658675	454
90	70	2673	6645	22	37	786983	512
90	80	2827	6923	22	37	763897	535
90	90	2988	7195	23	42	723663	582
90	100	3232	7603	24	43	841494	608
100	10	1396	4206	24	36	404289	375
100	20	1659	4799	25	38	529286	417
100	30	1901	5284	22	33	597376	435
100	40	2130	5796	21	33	605912	474
100	50	2344	6175	23	38	736464	519
100	60	2519	6527	25	47	827167	524
100	70	2724	6866	23	43	1039709	558
100	80	3035	7384	24	44	1099509	598
100	90	3234	7762	23	42	1097829	636
100	100	3414	8053	30	54	1196079	667



a) Program size



b) Rewriting time



c) DLV time

Figure 4: Experiments graphs for TS2.

set expressing its negative assertion (that is, its clashing set), is $\{A(a), A \sqcap B \sqsubseteq \bot, B(a)\}$. Also, negative instances need the declaration of a larger set of auxiliary signature elements.

In the rewriting time histogram, the growth is less evident given the quite limited size of the reference CKRs.[22]

As expected, the number of defeasible axioms and overridings clearly influences the time needed for the model computation by DLV. In particular, if we fix the percentage of overridings, DLV computing time grows polynomially in the percentage of defeasible axioms.[23] Moreover, by fixing the percentage of defeasible axioms (i.e. each sub-table in Table 12), model computation also grows polynomially in the percentage of overridings.[24]

This behaviour can also be justified by the growth in the number of test environments needed to verify the conditions for overriding of such defeasible axioms: in particular, by definition of the rules, a test literal is added to the model for each instance (exceptional or not) of a defeasible axiom. Note that, by definition of test rules, each instantiation of a test environment (corresponding to a different instance of a test literal) leads to a copy of the instance knowledge derived from the main environment. This intuition is reflected by the results in last column of Table 12, representing the number of test literals in the computed model: as expected (cf. introduction of Section 5.2), this value is linearly dependent on the percentage of defeasible axioms in the input CKR.

7. Related Work

In this section, we relate and compare our proposal with other approaches for including notions of defeasibility contextual systems and in description logics. In particular, we compare it to non-monotonic multi-context systems (MCS) [9], multi-context systems under argumentation semantics [29], typicality in DLs [30], and nonmonotonic description logics [31]. We will briefly present these approaches and aim to give an intuition about analogies and differences in our representation of defeasible inheritance (also by means of some representative examples).

7.1. Non-monotonic Multi-Context Systems

The idea of multi-context systems (MCS) is to align knowledge from different contexts in a single system using special bridge rules, dating back to [7]. We consider here the expressive concept of non-monotonic MCS in [9], in which contexts may be based on possibly different monotonic and non-monotonic logic, and bridge rules can be non-monotonic. The semantics of nonmonotonic MCS is defined in terms of *equilibria*: intuitively, an equilibrium is a collection of one belief set (local model) per context that verifies the knowledge content of contexts and the knowledge propagated through bridge rules.

Formally, in this approach a logic is abstractly defined as a triple $L = (\mathbf{KB}_L, \mathbf{BS}_L, \mathbf{ACC}_L)$, where $\mathbf{KB}_L$ is the set of well-formed KBs of L, which are sets of formulas; $\mathbf{BS}_L \subseteq 2^{B_L}$ is the set of possible belief sets of the logic, where B_L is base set of beliefs; and $\mathbf{ACC}_L : \mathbf{KB}_L \to 2^{\mathbf{BS}_L}$ is the set of accepted belief sets, i.e., the set of belief sets associated with a KB *kb* (thus providing

[22] Fluctuations in graphs are due to the random nature of the dataset and may be polished by averaging the results over a larger number of dataset instances.

[23] The trend can be approximated with an average R^2 value ≥ 0.994.

[24] Approximation with an average R^2 value ≥ 0.953.

the "semantics" of L). Propagation of knowledge across knowledge bases in different logics can be defined using bridge rules: given a set of logics $L_1, \ldots, L_n$, a bridge rule for L_i has the form:

$$(i : s) \leftarrow (r_1 : p_1), \ldots, (r_k : p_k), \mathsf{not}(r_{k+1} : p_{k+1}), \ldots, \mathsf{not}(r_m : p_m) \tag{10}$$

where $r_k \in \{1, \ldots, n\}$, $p_k \in B_{L_k}$, and s is a formula of L_i. A nonmonotonic MCS is then a collection $M = (C_1, \ldots, C_n)$ of contexts $C_i = (L_i, kb_i, br_i)$, where $L_i = (\mathbf{KB}_i, \mathbf{BS}_i, \mathbf{ACC}_i)$ is the logic of the context, $kb_i \in \mathbf{KB}_i$ is a knowledge base and br_i is a set of bridge rules for L_i over logics $L_1, \ldots, L_n$. An equilibrium of M is a collection $S = (S_1, \ldots, S_n)$ of belief sets $S_i \in \mathbf{BS}_i$ for the context C_i such that $S_i \in \mathbf{ACC}_i(kb_i \cup \{head(r) \mid r \in app(br_i, S)\})$, where $head(r) = (i : s)$ for any bridge rule r of form (10), and $app(br_i, S)$ consists of all bridge rules $r \in br_i$ such that (i) $p_j \in S_{r_j}$ for $1 \leq j \leq k$ and (ii) $p_j \notin S_{r_j}$, for $k < j \leq n$.

The idea of CKRs with defeasible inheritance based on justifiable assumptions may also be realized within the nonmonotonic MCS framework of [9], where contexts C_i with local semantics (acceptable belief sets over a local knowledge base kb_i) can add via bridge rules formulas to their kb_i depending on the local belief sets of the contexts. Adopting open bridge rules, i.e. bridge rules with variables that are instantiated over a given domain (using standard names in case, similar as in [44]), we may encode the global context G as an MCS context g and associate each element x of the domain with a context name in the MCS. We then may mimic satisfaction relative to assumptions as in CAS-interpretations with bridge rules that access G to determine whether axioms resp. axiom instances must be evaluated at x (if $x \in \mathsf{Ctx}^{\mathsf{M}}$). In particular, defeasible axioms α of the kind $\mathrm{D}(C \sqsubseteq D)$ can be encoded using auxiliary concept names A_α and bridge rules:

$$x : C \sqcap A_\alpha \sqsubseteq D \leftarrow \mathsf{g} : \mathsf{Ctx}(x) \qquad x : A_\alpha(y) \leftarrow \mathsf{g} : \mathsf{Ctx}(x), \mathsf{not}(x : \neg A_\alpha(y))$$

and for defeasible concept assertions $\mathrm{D}(A(c))$ bridge rules

$$x : A(c) \leftarrow \mathsf{g} : \mathsf{Ctx}(x), \mathsf{not}(x : \neg A(c)).$$

Intuitively, A_α serves as guard for the inclusion which by default is true for an individual, and thus the inclusion axiom applies to it; likewise, a concept assertion is true by default. The guard is blocked if a violation of the inclusion (an exception) is provable. The equilibria of the so constructed MCS are then akin to CKR-models. However, while this or a similar MCS approach is elegant, we need to extend the language and basically encode the problem in a framework that is very expressive and propositional in nature. Furthermore, currently only limited computational support is available for MCS. Above we aim at a formalization from first principles (giving a model-based semantics) that is suitable for realization in a well-supported host formalism.

7.2. MCS under Argumentation Semantics

A different non-monotonic semantics for MCS, based on argumentation, was proposed in [29]. The authors aimed at reasoning in presence of "imperfect" knowledge in ambient intelligence where knowledge is distributed across different contexts.

In this formulation, a MCS is a collection $C = (C_1, \ldots, C_m)$ of contexts of the form $C_i = (V_i, R_i, T_i)$ where V_i is the vocabulary (i.e. propositional literals) of C_i, R_i is the set of rules of the context, and T_i is a local preference ordering over all contexts. Rules can be either *local rules*, corresponding to local knowledge of C_i, or *mapping rules*, which basically amount to bridge rules across different contexts. Local rules are either strict, denoted $r_i^l : a_i^1, \ldots, q^{n-1} \rightarrow a_i^n$, or defeasible, denoted $r_i^d : b_i^1, \ldots, b_i^{n-1} \Rightarrow b_i^n$, where all a_i^j and b_i^j are from V_i; they represent strict and "soft" knowledge in the local theories, respectively. Mapping rules, denoted $r_i^m : a_{k_1}^1, \ldots, a_{k_{n-1}}^{n-1} \Rightarrow a_i^n$ with every $a_{k_j}^j \in V_{k_j}$ where $k_j \in \{1, \ldots, m\}$ and $q^n \in V_i$, are regarded as defeasible and serve to "import" knowledge from other contexts into C_i. Finally, the *local preference ordering* $T_i = [C_{k_1}, \ldots, C_{k_m}]$, $k_j \in \{1, \ldots, m\}$, on contexts expresses confidence in the knowledge imported from the other contexts by mapping rules.

The argumentation semantics for these MCS is based on the common argumentation semantics of Defeasible Logic (cf. [45]) extended with distribution of knowledge and preferences across contexts. A *support relation* for the MCS C is a set SR_C of triples (C_i, PT_{p_i}, p_i) where $C_i \in C$, $p_i \in V_i$, and PT_{p_i} is a proof tree for p_i using the local and mapping rules in R_i (we omit further details); any such triple is an *argument* for p_i. The set $ARGS_{C_i}$ of arguments for all $p_i \in V_i$ represents all possible logical consequences in C_i that are derivable using local or mapping rules.

If consequences are derived using "external" knowledge by mapping rules, conflicts over a literal p_i are resolved using the local context preference T_i, where clashes across arguments are considered. Intuitively, an argument A *attacks* another argument B if (i) A has a literal p_i in its consequences, (ii) B has the complementary literal $\neg p_i$ in its consequences, and (iii) p_i is a consequence of some defeasible local rule. An argument A *defeats* B at p_i if p_i has lower rank than the complementary $\neg p_i$ in B, where the rank of a literal p_i in C_i is 0 if $p_i \in V_i$ and is the rank of C_j in T_i if $p_i \in V_j \neq V_i$. In case of conflicting literals in mapping rules, an *argumentation line* A_L for a literal p_i can be formed as a sequence of arguments, possibly from different contexts, where attacks are extended to sequences. Based on attacks and defeats across arguments resp. argumentation lines, each literal in a MCS is either found justified, i.e. proved by a non-defeatable argument, or rejected, i.e. it can not overcome attacks from stronger arguments.

Compared to our CKR, we first note the different setting of defeasible MCS as in [29]: every context is seen as an independent agent having its own knowledge and preferences (ordering) on contexts. A CKR instead has a global structure of contexts and it only represents one level of "preference", namely the precedence of G w.r.t. local contexts. Viewing a CKR as a defeasible MCS (with preferences), the local preference ordering of each context c_i may thus be defined as $T_i = [c_i, \mathsf{G}]$. Local and global axioms of a CKR can be translated to local rules and mapping rules, where similar as for nonmonotonic MCS in Section 7.1 schematic (open) rules are used that are instantiated over a concrete domain. In particular, global default axioms can be here introduced as local defeasible rules: e.g., $\mathrm{D}(A \sqsubseteq B)$ can be represented (in every context c_i) as the defeasible rule $A_i(x) \Rightarrow B_i(x)$. Global subsumptions can be propagated to each context as strict local rules: e.g. if $C \sqsubseteq D$ is in G, then every context c_i contains the strict rule $C_i(x) \rightarrow D_i(x)$. We can relate *eval* expressions to mapping rules: e.g. $\mathit{eval}(A, \{c_1\}) \sqsubseteq C$ in context c_2 is expressible by the mapping rule $A_1(x) \Rightarrow C_2(x)$. Note, however, that *eval* expressions are strict inclusions and may contain complex context expressions; thus a proper encoding of *eval*-expression using defeasible and strict rules is more involved.

Our notion of overriding of a defeasible axiom compares to a "conflict" among two arguments for conflicting literals in [29]. In a CKR, conflicts occur only among arguments of the global and the local contexts. Using the above preference ordering, local arguments are preferred over global arguments and thus relate to clashing sets; as in our semantics, they serve to justify the local conclusions. To this extent, the clashing assumptions $\chi(x)$ for context x are akin to the rejected global arguments of x. However, the emerging semantics differs from our CKR-models in an important respect: while defeasible MCS resolve conflicts deterministically using local preferences, CKRs incorporate reasoning by cases; this manifests in tractable inference of literal queries from defeasible MCs, while literal queries on all CKR-models are intractable in general.

As for valuation, [46] showed how to translate defeasible MCSs into single theories of Defeasible Logic, where the idea is to include strict and defeasible (mapping) rules in such a theory and to express the local preference ordering on contexts by rule priorities. Our translation of

CKRs into datalog programs is analogous in this respect, and shows how a fragmented knowledge base can be compiled in a way such that available efficient tools can be utilized for reasoning.

7.3. *Normality in Description Logics*

In the area of Description Logics, a number of different proposals have been made to incorporate non-monotonic features, dating back at least to terminological default logic [47]. We refer for a more extensive bibliography and classification to [48], where preferential approaches (e.g. [49, 50]), circumscription-based approaches (e.g. [51, 52]) and others (e.g., [53]) are distinguished. We concentrate here on some recent proposals that aim at supporting defeasible subsumption respectively entailment, viz. [30, 54, 31, 51, 52], and we omit here works that aim at establishing semantic properties of entailment relations (such as rational closure [50, 55, 56]) or consider a finer grained notion of defeasibility depending on the nature of the relationship between elements of the vocabulary (cf. [48]). This is because our interest stands, at this point, with a basic mechanism for a formalism with explicit hierarchical structure, which is usually not reflected in nonmonotonic entailment relations.

7.3.1. *Typicality in DLs*

Default assumptions about properties of the members in a class C and the properties of prototypical elements of C, as defined in [30, 54], are closely related notions. Giordano et al. [30] formalize in their logic $\mathcal{ALC}+\mathbf{T}_{min}$ the intuition that a prototypical element of a concept C is a "generic element" of C. This definition stands on the possibility to organize objects in a generic-specific hierarchy, formally a partial order $<$, where $y < x$ means that object y is more generic (less specific) than object x. For instance, if x is a red Ferrari car and y is a yellow one, $x < y$ models that a Ferrari is more typically red than yellow.

Formally, the language of the description logic $\mathcal{ALC}$ is extended with the typicality operator $\mathbf{T}$: each (possibly complex) concept C in the language is associated with an extended concept $\mathbf{T}(C)$ representing its "typical" instances. In knowledge bases, extended concepts can appear on the left-hand sides of concept inclusions in the TBox and as concepts of assertions in the ABox.

The semantics of the typicality operator is obtained by extending DL interpretations with a preference relation on the domain. Thus, a $\mathcal{ALC}+\mathbf{T}$ interpretation is a structure $(\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}}, <)$ where $(\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ is a usual $\mathcal{ALC}$ interpretation and $<$ is irreflexive and transitive relation over $\Delta^{\mathcal{I}}$. The relation has to satisfy a *smoothness* condition, which eliminates infinitely decreasing chains: for each subset $S \subseteq \Delta^{\mathcal{I}}$, for every $x \in S$, either $x \in \min_<(S)$ or some $y \in \min_<(S)$ exists such that $y < x$, where $\min_<(S) = \{x \in S \mid \nexists y \in S . y < x\}$ are the minimal elements of S under $<$. The interpretation of the extended concept $\mathbf{T}(C)^{\mathcal{I}}$ corresponds then to $\min_< C^{\mathcal{I}}$. As noted by Giordano et al., this can be seen as a modal expression w.r.t. $<$: by defining $(\Box C)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid \forall y \in \Delta^{\mathcal{I}} . y < x \rightarrow y \notin C^{\mathcal{I}}\}$ the typical elements of a concept C are described by $(C \sqcap \Box\neg C)^{\mathcal{I}}$.

While this models typicality, it does not yet enforce that elements of a concept C must belong to its typical subconcept $\mathbf{T}(C)$, unless known otherwise. Basically, this corresponds to introducing non-monotonicity in the logic $\mathcal{ALC}+\mathbf{T}$. This is achieved by restricting the models to those which minimize the set of exceptional instances of concepts; the resulting logic is $\mathcal{ALC}+\mathbf{T}_{min}$. By exploiting the modal definition of typical concepts, exception minimization is effected by considering the models in which $(\neg\Box\neg C)^{\mathcal{I}}$ is minimized, for each C with typicality, in parallel; respective minimal models are called preferred models. Query entailment in $\mathcal{ALC}+\mathbf{T}_{min}$ is then defined via entailment from all preferred models.

Compared to our approach, the main analogy of the approach by Giordano et al. is that membership of an element in a concept must be blocked. However, the way this is achieved is fundamentally different: Giordano et al. use semantic model minimization, where the blocking results for minimal subsets of non-specific elements of C (i.e., the set $Q(C)$); intuitively, every element of C is regarded as prototypical unless this is infeasible. Our approach instead is syntactic and consequence-based, as exceptions have to be justified in terms of a semantic consequence. Notably, different from preferred models there is no notion of minimality in the definition of our CKR-models, even though it comes as a property (Proposition 6). Furthermore, we deal with explicit modular structure of a knowledge base and cross-references, which Giordano et al. do not consider. We note that, similarly to our approach, the authors recently proposed in [57] a datalog translation for reasoning with a typicality extension of $\mathcal{SROEL}(\sqcap, \times)$.

Our approach is geared towards syntactically guided exception handling in query answering, and not conceived as a logic of typicality of individuals per se. Nonetheless, prototypical concepts may be mimicked in our formalism using an extra concept for each concept C, say C^T, for the typical elements of C, and by the axioms

$$C^T \sqsubseteq C \qquad \mathrm{D}(C \sqsubseteq C^T)$$

which state that prototypical C's are C's and that C's are prototypical unless the contrary is entailed; C^T is then used for the prototypical concept. A deeper formal analysis of the correspondence between the two approaches will require some adaptation of the approach in [30] to $\mathcal{SROIQ}$-RL, and is beyond the present paper.

7.3.2. Normality via Circumscription

Besides [30], another approach to express typicality or "normality" via model-based minimization is to exploit McCarthy's circumscription principle [58]. This has been adopted for DLs e.g. in [51] and [52], where in particular DL-Lite$_R$ and $\mathcal{EL}$ resp. $\mathcal{EL}^\perp$ were considered, which are related to the OWL profiles OWL QL and OWL EL, respectively. From a semantical perspective, similar considerations as for the approach of Giordano et al. apply at a general level. Computationally, instance checking is in circumscribed DL-Lite$_R$ Π^p_2-complete and in $\mathcal{EL}^\perp$ ExpTime-hard; while circumscription of $\mathcal{SROIQ}$-RL has to the best of our knowledge not been studied, results in [59] and Section 3 suggest that instance checking should be coNP-complete, and thus have the same complexity under CKR-model semantics.

7.4. Nonmonotonic description logic $\mathcal{DL}^N$

A recent approach to overriding in description logics was presented by Bonatti et al. [31], which aims at a natural representation of exceptional classes of elements in a DL and retaining its tractability; in this way, applicability to large size knowledge bases should be secured.

A family $\mathcal{DL}^N$ of non monotonic DLs is defined by extending a generic base $\mathcal{DL}$ with an operator NC for *normality concepts*, which are the prototypical "normal" instances of type C, and with *defeasible inclusions (DIs)* $C \sqsubseteq_n D$ between concepts, which can be interpreted as "normally, instances of C are instances of D, unless stated otherwise."

Formally, for each concept C in $\mathcal{DL}$ a normality concept name NC is added to $\mathcal{DL}^N$; defeasible inclusions $C \sqsubseteq_n D$ require that C is from $\mathcal{DL}$ and D from $\mathcal{DL}^N$. A $\mathcal{DL}^N$ knowledge base has the form $\mathcal{K} = \mathcal{S} \cup \mathcal{D}$, where $\mathcal{S}$ and $\mathcal{D}$ are disjoint finite sets of $\mathcal{DL}^N$ axioms and defeasible inclusions, respectively.

In any $\mathcal{DL}^N$ interpretation $\mathcal{I}$, the inclusion N$C \sqsubseteq C$ must hold for each NC. The semantics of a defeasible inclusion $C \sqsubseteq_n D$ w.r.t. normal individuals is defined by resorting to the set

$\mathrm{sat}^{\mathcal{I}}(C \sqsubseteq_n D) = \{ \mathrm{N}E \mid \forall x \in \mathrm{N}E^{\mathcal{I}}, x \notin C^{\mathcal{I}} \vee x \in D^{\mathcal{I}} \}$ which are the normal concepts satisfied by the DI in $\mathcal{I}$: the idea is that normality concepts NE can not satisfy all DIs, and thus some DIs may be in conflict on NE. To decide which DIs then should be overridden, a priority relation $\delta_1 \prec \delta_2$ is used expressing that δ_1 has higher priority and is preferred over δ_2. While $\prec$ could be any strict partial order, Bonatti et al. mainly concentrated on *specificity*, i.e., $(C_1 \sqsubseteq_n D_1) \prec (C_2 \sqsubseteq_n D_2)$ iff $\mathcal{S} \models C_1 \sqsubseteq C_2$ and $\mathcal{S} \not\models C_2 \sqsubseteq C_1$.[25] Using $\mathrm{sat}^{\mathcal{I}}$ and $\prec$, the semantics of overriding is recursively defined by a function ovd: a DI δ is overridden in NC for interpretation $\mathcal{I}$ (denoted N$C \in \mathrm{ovd}^{\mathcal{I}}(\delta)$), if no interpretation $\mathcal{J}$ exists such that: (i) $\mathcal{J} \models \mathcal{S}$; (ii) N$C \notin \mathrm{sat}^{\mathcal{J}}(\delta)$; (iii) N$C^{\mathcal{J}} \neq \emptyset$; and (iv) for any other $\delta' \in \mathcal{D}$ s.t. $\delta' \prec \delta$, it holds that $\mathrm{sat}^{\mathcal{I}}(\delta') \setminus \mathrm{ovd}^{\mathcal{I}}(\delta') \subseteq \mathrm{sat}^{\mathcal{J}}(\delta')$, i.e. $\mathcal{J}$ satisfies all non-overridden higher priority DIs that are satisfied in $\mathcal{I}$. Based on this, an interpretation $\mathcal{I}$ satisfies a $\mathcal{DL}^N$ axiom α, , if $\mathcal{I} \models \alpha$ for $\alpha \in \mathcal{S}$ and if for every normality concept NC, it holds that N$C \in \mathrm{sat}^{\mathcal{I}}(\alpha) \cup \mathrm{ovd}^{\mathcal{I}}(\alpha)$ if $\alpha \in \mathcal{D}$.

To decide satisfaction in the presence of defeasible inclusions, Bonatti et al. provided a translation that compiles defeasible inclusions away. Intuitively, it proceeds as follows. First a set Σ of relevant normality concepts NC has to be fixed, which must include the normality concepts occurring in the initial KB. Given a linearization $\delta_1, \ldots, \delta_n$ of $(\mathcal{D}, \prec)$, the KB $\mathcal{K}^{\Sigma}$ is built from $\mathcal{K} = \mathcal{S} \cup \mathcal{D}$ by augmenting $\mathcal{S}$ along the linearization while omitting overridden DIs, as follows:

1. set $\mathcal{K}_0 = \mathcal{S} \cup \{\mathrm{N}C \sqsubseteq C \mid \mathrm{N}C \in \Sigma\}$;
2. for $i = 1, \ldots, n$: let $\mathcal{K}_i = \mathcal{K}_{(i-1)}$ and consider $\delta_i = D_i \sqsubseteq_n E_i$; add to $\mathcal{K}_i$ every $\delta_i^{\mathrm{N}C} = (\mathrm{N}C \sqcap D_i \sqsubseteq E_i)$, $\mathrm{N}C \in \Sigma$, s.t. $\mathcal{K}_{(i-1)} \setminus \{\delta^{\mathrm{N}C} \mid \delta \nprec \delta_i\} \cup \{\delta_i^{\mathrm{N}C}\} \not\models \mathrm{N}C \sqsubseteq \bot$;
3. finally, let $\mathcal{K}^{\Sigma} = \mathcal{K}_n$.

Intuitively, in each step the KB is enriched with an axiom $\delta_i^{\mathrm{N}C}$ if it does not cause an inconsistency in NC; otherwise, the DI is omitted, corresponding to an overriding w.r.t. NC.

Compared to our approach, the overall idea of individual exceptions and axiom overriding is similar in spirit. A notable difference is the extended notion of precedence between defeasible axioms in $\mathcal{DL}^N$. In our formalism, precedence is via the CKR structure, viz. from the global context to the local contexts. Accordingly, we can simulate a $\mathcal{DL}^N$ knowledge base $\mathcal{K} = \mathcal{S} \cup \mathcal{D}$ with void preference by representing the defeasible inclusions of $\mathcal{D}$ as defeasible axioms in the global context $\mathfrak{G}$ and the strong axioms as $\mathcal{S}$ a single local context.

Another relevant difference is that our formalism has no notion of "normal" concepts: every individual is "normal" w.r.t. all axioms, but can be exceptional w.r.t. given defeasible axioms. Thus while our formalism is not capable of reasoning about classes of "normal" and "exceptional" individuals, it can deal with property inheritance at the instance level; we illustrate this on the situs inversus example.

Example 18 ([31], rephrased)**.** *While normally humans have their heart oriented to the left part of their chest, in the exceptional case of* situs inversus *the heart is positioned towards the right part. We want to represent this fact while ensuring that the other anatomical features of such humans are not overridden. We can represent this scenario as a CKR* $\mathfrak{K}_{si} = (\mathfrak{G}, \{\mathcal{K}_{\mathrm{m1}}\})$ *where*

[25] Note that specificity can be related to similar ordering in [30].

$$\mathfrak{G} = \left\{ \begin{array}{ll} (d1) & \mathrm{D}(Human \sqsubseteq \exists hasHeartPosition.\{chest\text{-}left\}), \\ (d2) & \mathrm{D}(Human \sqsubseteq \exists hasNosePosition.\{face\text{-}center\}), \\ (m1) & \mathsf{mod}(\mathsf{c1}, \mathsf{m1}) \end{array} \right\},$$

$$K_{\mathsf{m1}} = \left\{ \begin{array}{ll} (i1) & SitusInversus \sqsubseteq \exists hasHeartPosition.\{chest\text{-}right\}, \\ (i2) & SitusInversus \sqsubseteq \exists NotHasHeartPosition.\{chest\text{-}left\}, \\ (i3) & \mathrm{Dis}(hasHeartPosition, NotHasHeartPosition), \\ (i4) & SitusInversus \sqsubseteq Human, \\ (a1) & SitusInversus(bob) \end{array} \right\}.$$

In this case, it is easy to verify that $\mathfrak{K}_{si}$ has a CKR model $\mathfrak{I}$ in which $d1$ is overridden, as it has a justified CAS-model with $\chi(\mathsf{c1}) = \{\langle Human \sqsubseteq \exists hasHeartPosition.\{chest\text{-}left\}, bob\rangle\}$ and clashing set $S = \{Human(bob), \neg hasHeartPosition(bob, chest\text{-}left)\}$ for the clashing assumption. On the other hand, $d2$ is naturally inherited even by the exceptional individual: that is, we have that $\mathcal{I}(\mathsf{c1}^{\mathcal{M}}) \models hasNosePosition(bob, face\text{-}center)$. ◇

As for the case of clashing inheritance, let us consider as in [31] a classic example.

Example 19 (Nixon diamond)**.** *We can easily represent the classical* Nixon diamond, *considered in [31, Example 9], also in our formalism as a CKR $\mathfrak{K}_{nd} = \langle \mathfrak{G}, \{K_{\mathsf{m1}}\}\rangle$, where*

$$\mathfrak{G}: \left\{ \begin{array}{ll} (d1) & \mathrm{D}(Quaker \sqsubseteq Pacifist), \\ (d2) & \mathrm{D}(Republican \sqsubseteq \neg Pacifist), \\ (m1) & \mathsf{mod}(\mathsf{c1}, \mathsf{m1}) \end{array} \right\},$$

$$K_{\mathsf{m1}}: \quad \{ Quaker(nixon), Republican(nixon) \}.$$

This CKR has two CKR models corresponding to the two possible overridings of the defeasible axioms (with the same priority). In particular, we have a CKR model $\mathfrak{I}_1$ s.t. $\mathcal{I}_1(\mathsf{c1}) \models Pacifist(nixon)$ with $\chi_1(\mathsf{c1}) = \{\langle Republican \sqsubseteq \neg Pacifist, nixon\rangle\}$ and the clashing set $S = \{Republican(nixon), Pacifist(nixon)\}$, and symmetrically a CKR model $\mathfrak{I}_2$ s.t. $\mathcal{I}_2(\mathsf{c1}) \models \neg Pacifist(nixon)$ with $\chi_2(\mathsf{c1}) = \{\langle Quaker \sqsubseteq Pacifist, nixon\rangle\}$ and clashing set $S = \{Quaker(nixon), \neg Pacifist(nixon)\}$. Thus it holds that $\mathfrak{K}_{nd} \not\models \mathsf{c}: Pacifist(nixon)$ and $\mathfrak{K}_{nd} \not\models \mathsf{c}: \neg Pacifist(nixon)$; similarly, neither $Pacifist(nixon)$ nor $\neg Pacifist(nixon)$ is concluded by the approach of Bonatti et al. However, if we change $d2$ to $\mathrm{D}(Republican \sqsubseteq Hawk)$ and add in $\mathfrak{G}$ the axioms $\mathrm{Dis}(Hawk, Pacifist)$, $Hawk \sqsubseteq Activist$, $Pacifist \sqsubseteq Activist$, then for the modifed CKR $\mathfrak{K}'_{nd}$ we obtain $\mathfrak{K}'_{nd} \models \mathsf{c}: Activist(nixon)$ in our approach, while one can not infer $Activist(nixon)$ from the corresponding knowledge base using Bonatti et al.'s approach; this demonstrates that the latter is not geared towards reasoning by cases if conflicts surface. ◇

A distinctive feature of our approach over [31] and others is of course the possibility to define a complex contextual structure of the knowledge base, allowing contextual reasoning inside each module. Moreover, our definition of defeasibility allows for an extension of the materialization calculus that was developed for the monotonic case and its implementation in the datalog rewriter, where the rewriting is not data dependent and can be done without solving reasoning problems on the knowledge base.

Regarding implementation and test results, we note that the defeasibility tests are similar: both study the effect of varying the degree of defeasible axioms and their overridings (which in turn is similar to the tests carried out by Casini et al. in [43]). However, the results are hard to compare, as they have been carried out on ontologies in $\mathcal{EL}$, which is a DL language of different characteristics than $\mathcal{SROIQ}$-RL.

8. Conclusion

We have considered the description logic-based Contextualized Knowledge Repository (CKR) framework [12, 16, 17, 2], which serves to represent and reason about information in contexts that model individual views within a global environment. Notably, the description of the global part of a CKR comprises both general information and knowledge about the structure of the contexts, which can be interrelated through extensional access among each other. To address inconsistency due to inheritance of global information to contexts, we have presented an extension of CKRs with defeasible axioms whose instances can be overridden, viewing them as exceptions that are justified by provable evidence. We have discussed some semantic properties of this approach for CKRs based on $\mathcal{SROIQ}$-RL (a description logic underlying the OWL RL profile) and we have introduced and studied the computational complexity of major reasoning tasks for CKRs. As for realization, we have developed reasoning on CKRs as a translation into datalog under stable models semantics: such encoding, that matches the intrinsic complexity, follows a materialization calculus approach as in [20, 60, 1]. We then described a prototype implementation for such translation (called CKR*ew*), as well as experimental results. Finally, we have compared the approach with related proposals for contextual reasoning and defeasibility in description logics. Notably, reasoning by cases as it emerges in the well-known Nixon Diamond scenario, for instance, can be properly handled by it.

Our work contributes to a general program of providing extensions for formalisms based on description logics, where the use of database technology such as SQL and datalog play a prominent role, based on the fact that a rich body of work in this area is available, with ongoing improvements of which the reasoning systems on top can take advantage. Nonetheless, however, future work is suggestive to address issues both on the computational and the modeling side.

As regards computation, in order to increase the practical applicability of the defeasible CKRs to larger sets of data, the translation described in this paper and its implementation need optimization. One possible direction for this regards the study of alternative datalog translations that limit the need for materialization; furthermore, engines other than DLV supporting non-ground query answering, such as the recent *s(ASP)* solver[26], could be explored. As discussed in Section 5.4, one possibility is to limit the use of tests environments only to CKRs that are not “safe” with respect to a direct reasoning on negative facts. Orthogonal to this is to use, instead of a uniform (factual) CKR encoding, one where datalog rules are generated ad-hoc: this could offer the possibility to take advantage of internal optimizations of the used datalog engine. Another possibility is to study different approaches, like e.g. abstraction refinement methods [61]. Moreover, such alternative translations might open the possibility to treat different or more expressive description logics (e.g. non-Horn fragments of $\mathcal{SROIQ}$). This also includes the challenge to identify and study fragments of the CKR formalism in which reasoning is highly efficiently realizable; this includes, for instance, syntactic criteria which ensure justification safeness. On the other hand, approximation of query results may be considered: the well-founded semantics can be readily applied to our datalog translation to be used as a tractable approximation.

On the modeling side, a natural continuation of our work is to allow defeasible axioms across local contexts, possibly along an explicit hierarchical relation between contexts (as the *coverage* relation [12]), or across knowledge modules, so as to allow for overriding in specific instances of context classes that are associated with such modules. In this respect, a notion of priority across

[26] https://sourceforge.net/projects/sasp-system/

defeasible axioms in local contexts should be defined to resolve the clashes among instances of such axioms at different contexts (cf. [31]); naturally, a respective priority order could be defined exactly as (or compatible to) the hierarchical order defined by the contexts coverage relation.

Another way to allow defeasible axioms in local contexts is to interpret them only inside the local context interpretation (i.e. CKRs become structures of locally defeasible knowledge bases): in this case, different interpretations of defeasibility can be adopted and compared (as, e.g., the semantics described in Section 7) and we may study the interaction of such "local defeasibility" with the interpretation of the current "global defeasible" axioms and its inheritance across local contexts. Another direction would be to extend the current CKR definition and allow multiple global contexts: in particular, this may require a preference order among these global contexts, in order to decide clashes in the inheritance of defeasible axioms in local contexts.

Allowing defeasible axioms in local contexts also opens the discussion on how the *eval* operator should be interpreted when used as a local defeasible axiom, thus allowing a notion of "defeasible propagation" of knowledge along local contexts (cf. mapping rules in [29]). Furthermore, for interactions across contexts, currently the knowledge of the CKR is consistent: another direction would be to allow local inconsistencies in contexts, similarly to defeasible MCS [29].

Acknowledgments

We would like to thank the reviewers for their careful reading of the manuscript and for their constructive comments to improve this article. This work has been supported in part by the Austrian Science Fund (FWF) via the projects P24090 and P27730 and by the FBK Mobility program.

Table A.13: Syntax and Semantics of $\mathcal{SROIQ}$, where A is any atomic concept, C and D are any concepts, P and R are any atomic roles (and for $*$ *simple* in the context of a knowledge base $\mathcal{K}$, cf. Section 2.1), S and Q are any (possibly complex) roles, a and b are any individual constants, and n stands for any positive integer.

Concept constructors	Syntax	Semantics
atomic concept	A	$A^{\mathcal{I}}$
top concept	$\top$	$\Delta^{\mathcal{I}}$
bottom concept	$\bot$	$\emptyset$
complement	$\neg C$	$\Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$
intersection	$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
union	$C \sqcup D$	$C^{\mathcal{I}} \cup D^{\mathcal{I}}$
existential restriction	$\exists R.C$	$\{x \in \Delta^{\mathcal{I}} \mid \exists y.\, (x,y) \in R^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\}$
self restriction$*$	$\exists R.\mathrm{Self}$	$\{x \in \Delta^{\mathcal{I}} \mid (x,x) \in R^{\mathcal{I}}\}$
universal restriction	$\forall R.C$	$\{x \in \Delta^{\mathcal{I}} \mid \forall y.\, (x,y) \in R^{\mathcal{I}} \rightarrow y \in C^{\mathcal{I}}\}$
min. card. restriction$*$	$\geq nR.C$	$\{x \in \Delta^{\mathcal{I}} \mid \#\{y \mid (x,y) \in R^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\} \geq n\}$
max. card. restriction	$\leq nR.C$	$\{x \in \Delta^{\mathcal{I}} \mid \#\{y \mid (x,y) \in R^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\} \leq n\}$
cardinality restriction$*$	$= nR.C$	$\{x \in \Delta^{\mathcal{I}} \mid \#\{y \mid (x,y) \in R^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\} = n\}$
nominal	$\{a\}$	$\{a^{\mathcal{I}}\}$

Role constructors	Syntax	Semantics
atomic role	R	$R^{\mathcal{I}}$
inverse role	R^{-}	$\{(y,x) \mid (x,y) \in R^{\mathcal{I}}\}$
role composition	$S \circ Q$	$\{(x,z) \mid (x,y) \in S^{\mathcal{I}}, (y,z) \in Q^{\mathcal{I}}\}$
Axioms	Syntax	Semantics
concept inclusion	$C \sqsubseteq D$	$C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$
concept definition	$C \equiv D$	$C^{\mathcal{I}} = D^{\mathcal{I}}$
role inclusion	$S \sqsubseteq R$	$S^{\mathcal{I}} \subseteq R^{\mathcal{I}}$
role disjointness$*$	$\mathrm{Dis}(P,R)$	$P^{\mathcal{I}} \cap R^{\mathcal{I}} = \emptyset$
reflexivity assertion$*$	$\mathrm{Ref}(R)$	$\{(x,x) \mid x \in \Delta^{\mathcal{I}}\} \subseteq R^{\mathcal{I}}$
irreflexivity assertion$*$	$\mathrm{Irr}(R)$	$R^{\mathcal{I}} \cap \{(x,x) \mid x \in \Delta^{\mathcal{I}}\} = \emptyset$
symmetry assertion	$\mathrm{Sym}(R)$	$(x,y) \in R^{\mathcal{I}} \Rightarrow (y,x) \in R^{\mathcal{I}}$
asymmetry assertion$*$	$\mathrm{Asym}(R)$	$(x,y) \in R^{\mathcal{I}} \Rightarrow (y,x) \notin R^{\mathcal{I}}$
transitivity assertion	$\mathrm{Tra}(R)$	$\{(x,y),(y,z)\} \subseteq R^{\mathcal{I}} \Rightarrow (x,z) \in R^{\mathcal{I}}$
concept assertion	$C(a)$	$a^{\mathcal{I}} \in C^{\mathcal{I}}$
role assertion	$R(a,b)$	$\langle a^{\mathcal{I}}, b^{\mathcal{I}}\rangle \in R^{\mathcal{I}}$
negated role assertion	$\neg R(a,b)$	$\langle a^{\mathcal{I}}, b^{\mathcal{I}}\rangle \notin R^{\mathcal{I}}$
equality assertion	$a = b$	$a^{\mathcal{I}} = b^{\mathcal{I}}$
inequality assertion	$a \neq b$	$a^{\mathcal{I}} \neq b^{\mathcal{I}}$

Appendix A. Further Details

Appendix A.1. $\mathcal{SROIQ}$ *syntax and semantics*

Table A.13 shows the syntax and semantics of $\mathcal{SROIQ}$.

Appendix A.2. FO-translation for $\mathcal{SROIQ}$*-RL*

In Table A.14, we present a FO-translation for contextualized $\mathcal{SROIQ}$-RL axioms: given a $\mathcal{SROIQ}$-RL axiom α in $\mathcal{L}^{e}_{\Sigma}$, the formula $\forall\, \mathbf{x}.\varphi_{\alpha}(\mathbf{x}, x_c)$, where $\mathbf{x} = x_1, x_2, \ldots x_n$ is a list of variables and x_c is expresses α in the context x_c.

The translation rules are recursively defined using additional set of rules for left-side and right-side expressions: translations $\beta_E(\mathbf{x}, x_c)$ and $\gamma_E(\mathbf{x}, x_c)$ for left-side and right-side expressions E are defined in Table A.15 and A.16, respectively. Note that the translation of *eval*-expressions, i.e., $\beta_{eval(A,C)}(\mathbf{x}, x_c)$ and $\beta_{eval(R,C)}(\mathbf{x}, x_c)$, omits the context argument x_c in $\beta_C(y_c)$; this context-free form serves to represent the global knowledge base $\mathfrak{G}$. Thus, a concept A is represented for contexts by the binary predicate $A(x, x_c)$, while in the global knowledge base by the unary predicate $A(x)$; similar for roles etc.

More generally, the context-free form $\varphi_{\alpha}(\mathbf{x})$ (resp., $\beta_E(\mathbf{x})$, $\gamma_E(\mathbf{x})$) of $\varphi_{\alpha}(\mathbf{x}, x_c)$ (resp., $\beta_E(\mathbf{x}, x_c)$, $\gamma_E(\mathbf{x}, x_c)$) is obtained by omitting the context argument x_c; it is applicable to axioms $\alpha \in \mathcal{L}_{\Gamma} \cup \mathcal{L}_{\Sigma}$, i.e., of the global knowledge base $\mathfrak{G}$ (and expressions E occurring in them).

Then, more formally Lemma 1 can be established.

Proof of Lemma 1. We show the claim for the contextualized FO-translation φ_{K,x_c} in (5); the result for the ordinary translation φ_K is then obvious.

Table A.14: Translation $\varphi_\alpha(\mathbf{x}, x_c)$, $\mathbf{x} = x_1, \dots, x_n$, of SROIQ-RL axioms α in L^e_Σ in context x_c to first-order logic.

$D(a) \mapsto$	$\gamma_D(a, x_c)$	$\mathrm{Sym}(R) \mapsto$	$R(x_1, x_2, x_c) \rightarrow R(x_2, x_1, x_c)$
$R(a, b) \mapsto$	$R(a, b, x_c)$	$\mathrm{Trans}(R) \mapsto$	$R(x_1, x_2, x_c) \wedge R(x_2, x_3, x_c) \rightarrow R(x_1, x_3, x_c)$
$\neg R(a, b) \mapsto$	$\neg R(a, b, x_c)$	$\mathrm{Asym}(R) \mapsto$	$R(x_1, x_2, x_c) \rightarrow \neg R(x_2, x_1, x_c)$
$a = b \mapsto$	$= (a, b, x_c)$	$\mathrm{Dis}(R, S) \mapsto$	$(R(x_1, x_2, x_c) \rightarrow \neg S(x_1, x_2, x_c)) \wedge$ $(S(x_1, x_2, x_c) \rightarrow \neg R(x_1, x_2, x_c))$
$a \neq b \mapsto$	$\neq (a, b, x_c)$		
$C \pm D \mapsto$	$\beta_C(x_1, x_c) \rightarrow \gamma_D(x_1, x_c)$	$\mathrm{Inv}(R, S) \mapsto$	$(R(x_1, x_2, x_c) \rightarrow S(x_2, x_1, x_c)) \wedge$ $(S(x_1, x_2, x_c) \rightarrow R(x_2, x_1, x_c))$
$R \pm T \mapsto$	$R(x_1, x_2, x_c) \rightarrow T(x_1, x_2, x_c)$	$\mathrm{Irr}(R) \mapsto$	$\neg R(x_1, x_1, x_c)$
$R \circ S \pm T \mapsto$	$(R(x_1, x_2, x_c) \wedge S(x_2, x_3, x_c)) \rightarrow T(x_1, x_3, x_c)$		

Table A.15: Translation $\beta_E(\mathbf{x}, x_c)$, $\mathbf{x} = x_1, \dots, x_n$, of SROIQ-RL (left-side) expressions E in L^e_Σ to first-order logic

$A \mapsto$	$A(x_1, x_c)$	$\exists R.C_1 \mapsto$	$R(x_1, x_2, x_c) \wedge \beta_{C_2}(x_1, x_c)$
		$\exists R.\{a\} \mapsto$	$R(x_1, a, x_c)$
$\{a\} \mapsto$	$= (x_1, a, x_c)$	$\exists R.\top \mapsto$	$R(x_1, x_2, x_c)$
$C_1 \sqcap C_2 \mapsto$	$\beta_{C_1}(x_1, x_c) \wedge \beta_{C_2}(x_1, x_c)$	$eval(A, C) \mapsto$	$\exists y_c.\beta_C(y_c) \wedge \beta_A(x_1, y_c)$
$C_1 \sqcap C_2 \mapsto$	$\beta_{C_1}(x_1, x_c) \vee \beta_{C_2}(x_1, x_c)$	$eval(R, C) \mapsto$	$\exists y_c.\beta_C(y_c) \wedge R(x_1, x_2, y_c)$

Table A.16: Translation $\gamma_E(\mathbf{x}, x_c)$, $\mathbf{x} = x_1, \dots, x_n$, of SROIQ-RL (right-side) expressions E in L^e_Σ to first-order logic

$A \mapsto$	$A(x_1, x_c)$	$\exists R.\{a\} \mapsto$	$R(x_1, a, x_c)$
$\neg C_1 \mapsto$	$\neg \beta_{C_1}(x_1, x_c)$	$\forall R.D_1 \mapsto$	$R(x_1, x_2, x_c) \rightarrow \gamma_{D_1}(x_2, x_c)$
		$\leq 0R.\top \mapsto$	$\neg R(x_1, x_2, x_c)$
$D_1 \sqcap D_2 \mapsto$	$\gamma_{D_1}(x_1, x_c) \wedge \gamma_{D_2}(x_1, x_c)$	$\leq 1R.\top \mapsto$	$(R(x_1, x_2, x_c) \wedge R(x_1, x_3, x_c)) \rightarrow = (x_2, x_3, x_c)$

To prove the result, we need to show that each axiom α in K can be written as a universal Horn sentence $\forall\mathbf{x}\varphi_\alpha(\mathbf{x}, x_c)$, where $\mathbf{x} = x_1, \dots, x_n$ is a list of free variables in $\varphi_\alpha(\mathbf{x}, x_c)$ depending on the type of the axiom α as follows:

- if α is an assertion, then $n = 0$ (and $\mathbf{x}$ is omitted);
- if α is a concept inclusion axiom, then $n = 1$; and
- if α is a role inclusion axiom or a role constraint, then $n = 2$.

More in detail we argue that $\varphi_\alpha(\mathbf{x}, x_c)$ can be written as $\varphi_\alpha(\mathbf{x}, x_c) = \bigvee_{i=1}^{4} \forall \mathbf{x}_i \gamma_i(\mathbf{x}, \mathbf{x}_i, x_c)$, where each γ_i is a Horn clause of the form

$$\gamma_i(\mathbf{x}, \mathbf{x_i}, x_c) = p_1(\mathbf{x}, \mathbf{x}_{i,1}, y_1) \wedge \cdots \wedge p_k(\mathbf{x}, \mathbf{x}_{i,k}, y_k) \rightarrow p_0(\mathbf{x}, \mathbf{x}_{i,0}, x_c) \qquad \text{(A.1)}$$

where

1. each p_i is a concept name, a role name, or equality $\approx$;
2. $p_0 = \perp$ (falsum) is possible;
3. each variable in $\mathbf{x}$, $\mathbf{x}_{i,j}$ occurs in the antecedent (safety), and $\mathbf{x}_i = \mathbf{x}_{i,0}, \dots, \mathbf{x}_{i,k}$;
4. each y_i is either x_c or a variable from $\mathbf{x}_i$, and then some $p_j(\mathbf{x}, \mathbf{x}_{i,j}, y_j)$ is of the form $\mathrm{Ctx}(y_j)$.

For the non-contextualized form, we analogously have $\varphi_\alpha(\mathbf{x}) = \bigvee_{i=1}^{4} \forall \mathbf{x}_i \gamma_i(\mathbf{x}, \mathbf{x}_i)$, where

$$\gamma_i(\mathbf{x}, \mathbf{x_i}) = p_1(\mathbf{x}, \mathbf{x}_{i,1}) \wedge \cdots \wedge p_k(\mathbf{x}, \mathbf{x}_{i,k}) \rightarrow p_0(\mathbf{x}, \mathbf{x}_{i,0}) \tag{A.2}$$

This form can be obtained by applying the contextualized FO-translation described in Table A.14, and is immediate for all axioms except concept inclusions $C \sqsubseteq D$ (writing $\neg R(a, b, x_c)$ as $R(a, b, x_c) \rightarrow \bot$, and moving $\neg R(\cdot\,\cdot)$, $\neg S(\cdot\,\cdot)$, and $\neg(a \approx b, x_c)$ from the consequent to the antecedent). Regarding the latter negation $\neg\beta_{C_1}(X_1, x_c)$ is moved from the consequent to the antecedent of the respective implication, $\exists y_c$ in the antecedent is turned into $\forall y_c$ (after standardizing apart) and pulled to the front; nested implications $\alpha \rightarrow (\beta \rightarrow \ldots)$ are rewritten as conjunctions $\alpha \wedge \beta \rightarrow \ldots$; eventually, disjunction in the antecedent resp. conjunction on the consequent is split into two clauses. □

Furthermore, the *context-constraint translation* of $\alpha \in \mathcal{L}_\Sigma$ is $\varphi_\alpha^{\mathrm{Ctx}}(\mathbf{x}, x_c) := Ctx(x_c) \rightarrow \varphi_\alpha(\mathbf{x}, x_c)$, and the *clashing-constraint translation* is $\varphi_\alpha^{CAS}(\mathbf{x}, x_c) := Ctx(x_c) \wedge app_\alpha(\mathbf{x}, x_c) \rightarrow \varphi_\alpha(\mathbf{x}, x_c)$, where app_α is a predicate that informally represents that in context x_c the axiom α is applicable for $\mathbf{x}$. That is, app_α represents the complement of the clashing assumptions in a *CAS*-interpretation $\mathcal{I} = \langle \mathfrak{M}, \chi \rangle$ and contains all tuples $\mathbf{e}, c$ over NI_Σ such that $\langle \alpha, \mathbf{e} \rangle$ is not in $\chi(c)$. If the interpretation $\mathcal{I}$ is named relative to N, we can restrict to tuples over N.

In abuse of notation, we also denote with $\forall \mathbf{x}.\varphi_\alpha(\mathbf{x}, x_c)$ such a Horn rewriting. Note that in the presence of disjunction $\sqcup$, the natural rewriting (which treats $(C_1 \sqcup C_2) \sqcap C_3 \sqsubseteq D$ as $C_1 \sqcap C_3 \sqsubseteq D$, $C_2 \sqcap C_3 \sqsubseteq D$) may have exponential size; otherwise, it is of polynomial size. The blowup can be avoided by normalizing axioms using auxiliary concepts, or at the level of translation, auxiliary predicates $P_{C_1 \sqcap C_2}(x_1, x_c)$ for $C_1 \sqcap C_2$ that are defined by Horn clauses $\forall x_1.\beta_{C_i}(x_1, x_c) \rightarrow P_{C_1 \sqcap C_2}(x_1, x_c)$, for $i = 1, 2$.

Furthermore, also the context-free translation $\varphi_\alpha(\mathbf{x})$, the *context-constraint translation* $\varphi_\alpha^{\mathrm{Ctx}}(\mathbf{x}, x_c)$, and the *context-constraint translation* $\varphi_\alpha^{CAS}(\mathbf{x}, x_c)$ can be easily rewritten to conjunctions of Horn clauses of the form (A.1), where now in the antecedent Ctx and app_α may occur; again we shall assume in abuse of notation that they are of this form.

Proof of Proposition 2. The FO-translation $\forall \mathbf{x}.\varphi_\alpha(\mathbf{x})$ of any $\mathcal{SROIQ}$-RL axiom α can be rewritten as a conjunction $\bigvee_{i=1}^{4} \forall \mathbf{x}_i \gamma_i(\mathbf{x}, \mathbf{x}_i)$ of Horn clauses with γ_i is in (A.2). If $\varphi_\alpha(\mathbf{e})$ is not valid, then for some clause γ_i a substitution $\theta : \mathbf{x}_i \rightarrow \mathrm{NI}$ must exist such that the assertion set

$$S_\theta = \{p_1(\mathbf{e}, \mathbf{x}_{i,1}\theta), \ldots p_k(\mathbf{e}, \mathbf{x}_{i,k}\theta), \neg p_0(\mathbf{e}, \mathbf{x}_{0,k}\theta)\}$$

is satisfiable; as $S_\theta \cup \{\varphi_\alpha(\mathbf{e})\}$ is unsatisfiable, S_θ is a clashing set. On the other hand, each clashing set S for $\langle \alpha, \mathbf{e} \rangle$ must violate some ground instance of some clause γ_i, and thus modulo equality among constants (as determined by all CAS-models $\mathcal{I}'_{CAS}$ that are NI-congruent with some justified CAS-model $\mathcal{I}_{CAS}$, we must have $S_\theta \subseteq S$ for some θ; the size of γ_i is linear in the size of $\gamma_i(\mathbf{e}, \mathbf{x}_i\theta)$ and thus of α, and likewise is the size of S_θ. □

Appendix A.3. CKR models: semantic properties

Proof of Proposition 3. If $\alpha \equiv \beta$ holds, then $\varphi_\alpha(\mathbf{x})$ and $\varphi_\beta(\mathbf{x})$ are logically equivalent. Thus items (ii) and (v) in the definition of CAS-model hold for α iff they hold for β replacing α (in particular also in $\chi(x)$). Furthermore, $S \cup \{\varphi_\alpha(\mathbf{e})\}$ and $S \cup \{\varphi_\beta(\mathbf{e})\}$ are equi-satisfiable for all sets of formulas S; hence, the clashing assumptions $\langle \alpha, \mathbf{e} \rangle$ and $\langle \beta, \mathbf{e} \rangle$ have the same clashing sets. It follows that a CAS-model $\mathcal{I}_{CAS} = \langle \mathfrak{M}, \chi \rangle$ for $\mathfrak{K}$ is justified iff the CAS-model $\mathcal{I}'_{CAS} = \langle \mathfrak{M}, \chi' \rangle$ in which β replaces α in χ is justified for $\mathfrak{K}'$. Hence the CKR-models of $\mathfrak{K}$ and $\mathfrak{K}'$ coincide. □

Proof of Proposition 5. We proceed by contraposition. Suppose that $\mathbf{I}'_{CAS} \not\models \mathfrak{K}$. Hence, item (v) of a CAS-model must be violated for some $D(\alpha) \in \mathfrak{G}$ and $x \in \mathsf{Ctx}^{\mathfrak{M}}$; the same pair also violates (v) for $\mathbf{I}_{CAS}$. For the second part, assume that $\mathbf{I}'_{CAS}$ is not justified. Hence, some $\langle\alpha, \mathbf{e}\rangle \in \chi'(x)$ for some $x \in \mathsf{Ctx}^{\mathfrak{M}}$ is not justified. That is, no clashing set S for $\langle\alpha, \mathbf{e}\rangle \in \chi(x)$ exists such that $\mathbf{I}''_{CAS} \models S$ for every CAS-model $\mathbf{I}''_{CAS} = (\mathfrak{M}'', \mathcal{I}'', \chi')$ that is NI-congruent with $\mathbf{I}'_{CAS}$. As every such $\mathbf{I}''_{CAS}$ is also NI-congruent with $\mathbf{I}_{CAS}$ and $\chi(x) = \chi'(x)$, no clashing set S for $\langle\alpha, \mathbf{e}\rangle \in \chi(x)$ either can exist to witness that $\langle\alpha, \mathbf{e}\rangle \in \chi(x)$ is justified wrt. $\mathbf{I}_{CAS}$; hence $\mathbf{I}_{CAS}$ is not justified. □

Proof of Proposition 6. Towards a contradiction, suppose that some $x \in \mathsf{Ctx}^{\mathfrak{M}}$ exists such that $\chi'(x) \subsetneq \chi(x)$, i.e., some $\langle\alpha, \mathbf{e}\rangle \in \chi(x) \setminus \chi'(x)$ exists. As $\mathbf{I}'_{CAS} \models \mathfrak{K}$ it follows that (a) $\mathcal{I}'(x) \models \phi_\alpha(\mathbf{e})$. Furthermore, as $\mathbf{I}_{CAS} = (\mathfrak{M}, \mathcal{I}, \chi)$ is justified for $\mathfrak{K}$, there exists a clashing set $S = S_{\langle\alpha,\mathbf{e}\rangle,x}$ such that $\mathbf{I}''_{CAS} \models S$ for every CAS-model $\mathbf{I}''_{CAS} = (\mathfrak{M}'', \mathcal{I}'', \chi)$ of $\mathfrak{K}$ that is NI-congruent with $\mathbf{I}_{CAS}$. Now let $\mathbf{I}'''_{CAS} = (\mathfrak{M}', \mathcal{I}', \chi)$. As $\mathbf{I}'_{CAS} \models \mathfrak{K}$ and $\chi'(x) \subseteq \chi(x)$, it follows that $\mathbf{I}'''_{CAS} \models \mathfrak{K}$ as well (item (v) is monotonic in enlarging clashing assumptions). Furthermore, by construction $\mathbf{I}'''_{CAS}$ is NI-congruent with $\mathbf{I}'_{CAS}$ and thus also with $\mathbf{I}_{CAS}$. Thus it follows that (b) $\mathcal{I}'(x) \models S$. From (a) and (b) follows $\mathcal{I}'(x) \models S \cup \{\alpha(\mathbf{e})\}$. This however contradicts that S is a clashing set for $\langle\alpha, \mathbf{e}\rangle$, as $S \cup \{\alpha(\mathbf{e})\}$ would be satisfiable. This proves the result. □

Proof of Proposition 7. The sentence $\phi_{\mathfrak{K}}$ defined in (8) characterizes the CAS-models of a $\mathcal{SROIQ}$-RL CKR $\mathfrak{K}$; as it amounts to a Horn sentence, its models relative to an interpretation of the constants symbols NI are closed under intersections. As $\mathbf{I}^i_{CAS} = (\mathfrak{M}_i, \mathcal{I}_i, \chi)$, $i \in \{1, 2\}$, are CAS-models of $\mathfrak{K}$, it follows that $\mathbf{I}_{CAS} = (\mathfrak{M}_1 \cap \mathfrak{M}_2, \mathcal{I}_1 \cap \mathcal{I}_2, \chi)$ is a CAS-model of $\mathfrak{K}$ as well. (Alternatively, this can be assessed straight from the definition of CAS-model (Definition 11), as α resp. $\forall \mathbf{x}.\phi_\alpha(\mathbf{x})$ and K_m resp. ϕ_{K_m} amount to Horn sentences, and their models wrt. an interpretation of NI are closed under intersection.)

For the second part, assume without loss of generality that $\mathbf{I}^1_{CAS}$ is justified. That is, $\langle\alpha, \mathbf{e}\rangle \in \chi(x)$ for $x \in \mathsf{Ctx}^{\mathfrak{M}_1}$ implies that some clashing set $S = S_{\langle\alpha,\mathbf{e}\rangle,x}$ for $\langle\alpha, \mathbf{e}\rangle$ exists such that $\mathbf{I}'_{CAS} \models S$ for every CAS-model $\mathbf{I}'_{CAS}$ of $\mathfrak{K}$ that is NI-congruent with $\mathbf{I}^1_{CAS}$. As $\mathsf{Ctx}^{\mathfrak{M}} \subseteq \mathsf{Ctx}^{\mathfrak{M}_1}$ and $\mathbf{I}^1_{CAS}$ is NI-congruent with $\mathbf{I}_{CAS}$, it follows that if $x \in \mathsf{Ctx}^{\mathfrak{M}}$ then $\langle\alpha, \mathbf{e}\rangle \in \chi(x)$ has the witnessing clashing set S relative to $\mathbf{I}_{CAS}$. Consequently, $\mathbf{I}_{CAS}$ is justified for $\mathfrak{K}$ as well. □

Proof of Lemma 2. By Lemma 1 the sentence $\phi_{\mathfrak{K}}$ captures $\mathfrak{K}$; these sentences amount to a conjunction $\bigwedge_{i=1}^{4} \forall \mathbf{x}_i \gamma_i(\mathbf{x}, \mathbf{x}_i)$ where $\gamma_i(\mathbf{x}, \mathbf{x}_i)$ is the Horn clause (A.2). By construction, the interpretation $\mathcal{I}^N$ is named relative to N. Furthermore for each assignment $\theta : \mathbf{x}_i \to \mathrm{NI}$, the antecedents of $\gamma_i\theta$ evaluates to false in $\mathcal{I}^N$, if for some p_i, $i \geq 1$, we have $\theta(x) \notin N$ for some variable x in $\mathbf{x}_i$: any concept or role is false outside N^I, and equality atoms in the antecendent are of the form $x_j \approx c$ where $c \in N$. Thus it follows that $\mathcal{I}, \theta \models \gamma_i(\mathbf{e}, \mathbf{x}_i)$ implies that $\mathcal{I}^N, \theta \models \gamma_i(\mathbf{e}, \mathbf{x}_i)$ holds. Hence, it follows that $\mathcal{I} \models \phi_{\mathfrak{K}}$ implies $\mathcal{I}^N \models \phi_{\mathfrak{K}}$ which proves the claim. □

Proof of Theorem 1. Suppose that $\mathbf{I}_{CAS} \models \mathfrak{K}$, where $\mathbf{I}_{CAS} = (\mathfrak{M}, \mathcal{I}, \chi)$. Then it follows from Lemma 2 that for any $\mathbf{I}^N_{CAS} = (\mathfrak{M}^N, \mathcal{I}^N, \chi^N)$ the items (i), (ii) and (iv) of Definition 11 hold. Item (iii) holds by an extension of the argument in Lemma 2 to the contextualized form of Horn clauses in (A.1), which respects *eval* expressions: if $\langle x, y\rangle \in \mathsf{mod}^{\mathfrak{M}^N}$ such that $y = \mathsf{m}^{\mathfrak{M}^N}$, then $\langle x, y\rangle \in \mathsf{mod}^{\mathfrak{M}}$ and $y = \mathsf{m}^{\mathfrak{M}}$, and thus it follows from Lemma 1 we have that $\mathcal{I}(x) \models \phi_{K_m, x_c}(x)$. Furthermore, we have $x \in N^{\mathfrak{M}}$; if for some axiom $\alpha \in K_m$ an assignment θ to $\mathbf{x}, \mathbf{x}_i, y_1, \ldots y_k$ in (A.1) maps some variable outside N^I, then some $p_j(\mathbf{x}, \mathbf{x}_i, y_j)$ evaluates to false under $\theta' = \theta \cup \{x_c \mapsto x\}$, and hence $\mathcal{I}^N(x), \theta' \models \gamma_i(\mathbf{x}, \mathbf{x}_i, x_c)$; otherwise, as $\mathcal{I}^N(x)$ and $\mathcal{I}(x)$ coincide on θ' and

$\mathcal{I}(x), \theta' \models \gamma_i(\mathbf{x}, \mathbf{x}_i, x_c)$, we again have $\mathcal{I}^N(x), \theta' \models \gamma_i(\mathbf{x}, \mathbf{x}_i, x_c)$. Hence, it follows $\mathcal{I}^N(x) \models \varphi_\alpha(x)$, and consequently $\mathcal{I}^N(x) \models \varphi_{K_{\mathsf{m}},x_c}(x)$.

Finally, by construction of χ^N also item (v) holds: let $\mathbf{d} \notin \{\mathbf{e} \mid (\alpha, \mathbf{e}) \in \chi^N(x)\}$ for $D(\alpha) \in \mathcal{G}$ and $x \in \mathsf{Ctx}^{\mathcal{M}^N}$. If $\mathbf{d}$ is over $N \cup N^I$, then $\mathbf{d} \notin \{\mathbf{e} \mid (\alpha, \mathbf{e}) \in \chi(x)\}$, and hence $\mathcal{I}(x) \models \varphi_\alpha(\mathbf{d})$ holds; similar as in Lemma 2, we then conclude that $\mathcal{I}^N(x) \models \varphi_\alpha(\mathbf{d})$ holds. Otherwise, if $\mathbf{d}$ is not over $N \cup N^{\mathcal{M}}$, then in the Horn rewriting of $\varphi_\alpha(\mathbf{d})$, in each clause $\gamma_i(\mathbf{d}, \mathbf{x}_i)$ in (A.2) some constant $d \notin N \cup N^I$ occurs in the antecedent; hence, $\gamma_i(\mathbf{d}, \mathbf{x}_i)$ evaluates to false under any assignment to $\mathbf{x}_i$.

We next show that $\mathfrak{I}^N_{CAS}$ is justified if $\mathfrak{I}_{CAS}$ is justified. Towards a contradiction, suppose $\mathfrak{I}^N_{CAS}$ is not justified. Then some $(\alpha, \mathbf{e}) \in \chi^N(x)$ is not justified, i.e., for every clashing set $S = S_{(\alpha,\mathbf{e}),x}$ for $(\alpha, \mathbf{e})$, there exists some CAS-model $\mathfrak{I}'_{CAS^N} = (\mathcal{M}^{N\prime}, \mathcal{I}^{N\prime}, \chi^N)$ of $\mathfrak{K}$ that is NI-congruent to $\mathfrak{I}_{CAS}$ and $\mathcal{I}^{N\prime}(x) \not\models S$. In particular, this must hold for the S witnessing that $(\alpha, \mathbf{e})$ is justified wrt. $\mathfrak{I}_{CAS}$; without loss of generality, in S no constants from $\mathrm{NI} \setminus (\mathrm{NI}_S \cup N)$ occur. Consider then the CAS-interpretation $\mathfrak{I}'_{CAS} = (\mathcal{M}', \mathcal{I}', \chi)$ that results from $\mathfrak{I}_{CAS}$ by changing the interpretations of concept and role names in $\mathcal{M}$ resp. $\mathcal{I}$ to those in $\mathcal{M}^{N\prime}$ resp. $\mathcal{I}^{N\prime}$. It holds that $\mathfrak{I}'_{CAS} \models \mathfrak{K}$ and $\mathfrak{I}'_{CAS}$ is NI-congruent with $\mathfrak{I}_{CAS}$ (as no $c \in \mathrm{NI} \setminus N$ occurs in $\mathfrak{K}$ and $\chi^N \subseteq \chi$), but $\mathcal{I}'(x) \not\models S$; this contradicts that $(\alpha, \mathbf{e})$ is justified at x w.r.t. $\mathfrak{I}_{CAS}$. Hence, $\mathfrak{I}^N_{CAS}$ is justified.

It remains to show that each $(\alpha, \mathbf{e}) \in \chi(x)$ is justified by some clashing set $S = S_{(\alpha,\mathbf{e}),x}$ with constants from N. By Proposition 2, without loss of generality S amounts to a ground instance of some clause $\gamma_i(\mathbf{e}, \mathbf{x}_i)$ in (A.2). As $\mathfrak{I}^N_{CAS} \models S$ holds, by construction we must have that $c^{\mathcal{M}^N} \in N^{\mathcal{M}^N}$ holds for each constant c that occurs in $p_j(\mathbf{e}, \mathbf{x}_{i,j}\theta)$, $j > 0$; thus, we can replace each c in S with some $d \in N$ such that $c^{\mathcal{M}^N} = d^{\mathcal{M}^N}$. By safety of (A.2), this turns S into an equivalent clashing set w.r.t. the name assignment given by $\mathfrak{I}_{CAS}$ resp. $\mathfrak{I}^N_{CAS}$. □

Appendix A.4. Proofs of Section on Reasoning and Complexity

Proof of Theorem 3 (continued). The hardness part is shown by a reduction from 3SAT, and in fact for a fixed set of inclusion axioms, that is, under data complexity. Let $E = \bigwedge_{i=1}^{m} \gamma_i$ be an instance of 3SAT over propositional atoms $X = \{x_1, \ldots, x_n\}$. Without loss of generality, each clause γ_i in E is either positive or negative. Then we construct $\mathfrak{K}$ as follows, where V, F, T, A are concepts, $P_1, P_2, P_3, N_1, N_2, N_3$ are roles, and $x_1, \ldots, x_n, c_1, \ldots, c_m$ are individual constants.

- the global knowledge $\mathcal{G}$ contains defeasible axioms $D(V \sqsubseteq T)$ and $D(V \sqsubseteq F)$ and a module association $\mathsf{mod}(\mathsf{c}, \mathsf{m})$;
- a single module K_{m} that contains the inclusion axioms:

$$T \sqcap F \sqsubseteq \bot, \quad T \sqsubseteq A, \quad F \sqsubseteq A, \quad \textstyle\bigsqcap_{j=1}^{3} \exists N_j.(T \sqcap A) \sqsubseteq \bot, \quad \text{and} \quad \bigsqcap_{j=1}^{3} \exists P_j.(F \sqcap A) \sqsubseteq \bot,$$

  where $\bot$ stands for $A \sqcap \neg A$. Furthermore, K_{m} contains assertions
- $V(x_h)$, $h = 1, \ldots, n$, and
- $P_j(c_i, x_{i_j})$ for $i = 1, \ldots, m$ and $j = 1, 2, 3$ such that the clause γ_i is of form $x_{i_1} \vee x_{i_2} \vee x_{i_3}$,
- $N_j(c_i, x_{i_j})$ for $i = 1, \ldots, m$ and $j = 1, 2, 3$ such that the clause γ_i is of form $\neg x_{i_1} \vee \neg x_{i_2} \vee \neg x_{i_3}$.

Intuitively, we must at context c make for x_h an exception to either $V \sqsubseteq F$ or $V \sqsubseteq T$; the respective single minimal clashing set is $\{V(x_h), \neg F(x_h)\}$ resp. $\{V(x_h), \neg T(x_h)\}$. Keeping $V \sqsubseteq T$ (resp. $V \sqsubseteq F$) justifies $\neg F(x_h)$ (resp. $\neg T(x_h)$) via the axiom $T \sqcap F \sqsubseteq \bot$. Every truth assignment σ to X thus gives rise to a natural clashing assumption CAS_σ that at c includes $(V \sqsubseteq F, x_h)$ if $\sigma(x_h) = \mathit{true}$ and

$(V \sqsubseteq T, x_h)$ if $\sigma(x_h) = false$; if σ satisfies E, then in a Herbrand interpretation all axioms in K_m are satisfied, if all $A(x_h)$ are true. Formally, we can construct a justified named CAS-model of K.

Conversely, if K has some justified CAS-model $I_{CAS}(M, I, \chi)$, without loss of generally we assume it is named, and moreover that it is a pseudo Herbrand model (as no equality occurs in G and K_m). Due to $T \sqcap F \sqsubseteq \perp$, we have that $\chi(c)$ must contain at least one of $(V \sqsubseteq F, x_h)$ and $(V \sqsubseteq T, x_h)$; on the other hand, it can not contain both: indeed, in this case $I(c) \models \neg T(x_h)$ and $I(c) \models \neg F(x_h)$; we then can modify $I(c)$ to $I'(c)$ by switching $T(x_h)$ true and $A(x_h)$ to false. The resulting CAS-interpretation $I'_{CAS}(M, I', \chi)$ satisfies K; this however implies that $(V \sqsubseteq T, x_h)$ is not justified, and thus $I_{CAS}(M, I, \chi)$ is not a justified, which is a contradiction.

It is then easy to check that the natural truth assignment σ to X represented by $\chi(c)$, viz. $\sigma(x_h) = true$ if $(V \sqsubseteq F, x_h) \in \chi(c)$ and $\sigma(x_h) = false$ otherwise, satisfies the formula E, as for each c_i at least one of $T(x_{i_1})$, $T(x_{i_2})$, $T(x_{i_3})$ resp. of $F(x_{i_1})$, $F(x_{i_2})$, $F(x_{i_3})$ must be false, and thus the clause γ_i evaluates under σ to true. As K is easily constructed from E, this proves the NP-hardness. □

Proof of Theorem 5 (continued). To show Π^p_2 hardness under the stated restriction, consider evaluating a QBF Φ of the form $\forall X \exists Y E$, where $E = \bigvee_{i=1}^{k} \gamma_i$ is a conjunction of clauses $\gamma_i = l_{i,1} \vee l_{i,2} \vee l_{i,3}$ over atoms $X = \{x_1, \dots, x_n\}$ and $Y = \{y_1, \dots, y_m\}$.

We construct K as follows, where V, F, T, X are concepts, val, opp, l_1, l_2, l_3 are roles, and $0, 1, \dots, 7, c_1, \dots, c_n$, are individual constants.

- the global knowledge G contains defeasible axioms $D(X \sqsubseteq T)$ and $D(X \sqsubseteq F)$ and a module association mod.{m}(c);
- a single module K_m that contains the inclusion axioms:
  - $T \sqcap F \sqsubseteq \perp$, $F \sqsubseteq \exists val.\{0\}$, $T \sqsubseteq \exists val.\{1\}$,

  (where $\perp$ stands for falsity) and the assertions
  - $X(c_1), \dots, X(c_n)$,
  - $opp(0, 1)$, $opp(1, 0)$,
  - $as_i(v, w)$ where $(v, w) \in \{0, 1\} \times \{1, \dots, 7\}$ such that v is the i-th bit of the integer w in binary, $i = 1, 2, 3$. Informally, v is the projection of an assignment w to the variables z_1, z_2, z_3 that satisfies the clause $z_1 \vee z_2 \vee z_3$ to the i-th component.

Intuitively, we must at context c make for c_i an exception to either $V \sqsubseteq F$ or $V \sqsubseteq T$, for each i; the respective single minimal clashing set is $\{V(c_i), \neg F(c_i)\}$ resp. $\{V(c_i), \neg T(c_i)\}$. Keeping $V \sqsubseteq T$ (resp. $V \sqsubseteq F$) justifies $\neg F(c_i)$ (resp. $\neg T(c_i)$) via the axiom $T \sqcap F \sqsubseteq \perp$. Every truth assignment σ to $x_1, \dots, x_n$ thus gives rise to a natural justified clashing assumption χ_σ that for c includes $(V \sqsubseteq F, c_i)$ if $\sigma(x_i) = true$ and $(V \sqsubseteq T, c_i)$ if $\sigma(x_i) = false$; furthermore, we obtain from the axiom $T \sqsubseteq \exists val.\{1\}$ (resp. $T \sqsubseteq \exists val.\{0\}$) the conclusion $val(c_i, 1)$ (resp., $val(c_i, 0)$).

The Boolean query that we construct is

$$Q = \exists \mathbf{y} \bigwedge_{i=1}^{n} \mathrm{c} : val(c_i, x_i) \wedge \bigwedge_{j=1}^{k} (\tau(l_{j,1}, j) \wedge \tau(l_{j,2}, 2) \wedge \tau(l_{j,3}, 3)),$$

where for every literal 4 over $X \cup Y$ and $j \in \{1, 2, 3\}$,

$$\tau(4, j) = \begin{cases} \mathsf{c} : as_j(v, w_j), & \text{if } 4 = v \in X \cup Y, \\ \mathsf{c} : as_j(\bar{v}, w_j) \wedge \mathsf{c} : opp(\bar{v}, v), & \text{if } 4 = \neg v,\ v \in X \cup Y. \end{cases}$$

Informally, the first part of the query must match each variable x_i to the value that has been chosen for it by the defeasible axioms; the second part must match all y_i variables then to either 0 or 1, such that each clause γ_j is satisfied: the assignment to the variables of $4_{j,1}$, $4_{j,2}$ and $4_{j,3}$ must be such that w_j represents a satisfying assignment to γ_j that is via "*opp*" renamed to a positive clause.

It is then not difficult to verify that for the least CAS-model for χ_σ w.r.t. identity v, denoted $\mathbf{I}_\sigma = \hat{\mathbf{I}}(\chi_\sigma, \mathsf{v})$, we have that $\mathbf{I}_\sigma \models Q$, iff the formula $\exists Y E\sigma$ evaluates to true, i.e., after applying σ on E, the resulting CNF $E\sigma$ is satisfiable. Hence, $\mathsf{K} \models Q$ iff the QBF Φ evaluates to true. Furthermore, the module structure in K is fixed and only the assertions in the modules $\mathsf{K_m}$ vary in order to encode Φ. This proves Π^p_2-hardness under the stated restriction.

Remark. To establish the result for constant-free queries, we can easily remove in Q above the constants c_i by introducing in K_m assertions $R(c_i, c_{i+1})$, $1 \leq i < n$, where R is a fresh role, replacing each c_i in Q with a fresh variable z_i and adding $\bigvee_{i=1}^{n-1} \mathsf{c} : R(z_i, z_{i+1})$ (assuming $n \geq 2$).

We can alternatively establish Π^p_2-hardness of CQ answering under the restriction that the set of assertions (the data) is fixed by a reduction from CERT3COL [62], which is a generalization of graph 3-colorability: given a graph $G = (V, E)$ where each edge $e \in E$ is labeled with a disjunction $\delta_e = 4_{e,1} \vee 4_{e,2}$ of literals $4_{e,j}$ over propositional variables $x_1, \ldots, x_m$, decide whether each graph $G_\sigma = (V, E_\sigma)$ where E_σ contains all edges $e \in E$ such that $\vartheta(\delta_e)$ evaluates to true, is 3-colorable. Note that the graph 3-colorability problem results if $4_{e,1}$ and $4_{e,2}$ are always opposite literals, i.e., δ_e is a tautology and then e is always selected.

Our reduction is inspired by a well-known reduction from deciding graph 3-colorability to CQ answering over a relational database, cf. [63], where the database holds tuples that state admissible color combinations of adjacent nodes, and the query describes the graph; the query answers correspond then to the legal 3-colorings of the graph.

We construct K as follows, where V, F_{x_i}, T_{x_i}, C_e are concepts and R, E_e are roles, for all $e \in E$ and x_i, and where a, r, g, b are individual constants:

- the global knowledge G contains defeasible axioms $D(V \sqsubseteq T_{x_i})$ and $D(V \sqsubseteq F_{x_i})$, $1 \leq i \leq m$ and a module association $\mathsf{mod}.\{\mathsf{m}\}(\mathsf{c})$;
- a single module K_m that contains the inclusion axioms:
  - $T_{x_i} \sqcap F_{x_i} \sqsubseteq \bot$,

  (where $\bot$ stands for falsity) and for all $e \in E$ and $c \in \{r, g, b\}$:

  - $\Lambda(4_{e,2}) \sqcap \Lambda(4_{e,2}) \sqsubseteq \exists R.(\{c\} \sqcap C_e)$, where $\Lambda(x_i) = F_{x_i}$ and $\Lambda(\neg x_i) = T_{x_i}$ for each atom x_i, and
  - $C_e \sqsubseteq \exists E_e.\{c\}$,

  and assertions

  - $V(a)$, and
  - $E_e(r, g)$, $E_e(r, b)$, $E_e(g, r)$, $E_e(b, r)$, $E_e(b, g)$, $E_e(g, b)$.

Table A.17: (Minimal) clashing sets for normal form SROIQ-RL clashing assumptions.

$(A(a), a)$:	$\{\neg A(a)\}$	$(a = b, \_)$:	$\{a \neq b\}$
$(\neg A(a), a)$:	$\{A(a)\}$	$(a \neq b, \_)$:	$\{a = b\}$
$(R(a, b), (a, b))$:	$\{\neg R(a, b)\}$	$(A \sqsubseteq \forall R.B, e)$:	$\{A(e), R(e, f), \neg B(f)\}$
$(\neg R(a, b), (a, b))$:	$\{R(a, b)\}$	$(A \sqsubseteq \leq 1R.\top, e)$:	$\{A(e), R(e, f_1), R(e, f_2), f_1 \neq f_2\}$
$(\{a\} \sqsubseteq B, a)$:	$\{\neg B(a)\}$	$(R \sqsubseteq T, (e_1, e_2))$:	$\{R(e_1, e_2), \neg T(e_1.e_2)\}$
$(A \sqsubseteq B, e)$:	$\{A(e), \neg B(e)\}$	$(R \circ S \sqsubseteq T, (e_1, e_2))$:	$\{R(e_1, f), S(f, e_2), \neg T(e_1, e_2)\}$
$(A_1 \sqcap A_2 \sqsubseteq B, e)$:	$\{A_1(e), A_2(e), \neg B(e)\}$	$(\mathrm{Dis}(R, S), (e_1, e_2))$:	$\{R(e_1, e_2), S(e_1, e_2)\}$
$(\exists R.A \sqsubseteq B, e)$:	$\{R(e, f), A(f), \neg B(e)\}$	$(\mathrm{Inv}(R, S), (e_1, e_2))$:	$\{R(e_1, e_2), \neg S(e_1, e_2)\}$,
$(A \sqsubseteq \exists R.\{a\}, e)$:	$\{A(e), \neg R(e, a)\}$		$\{\neg R(e_1, e_2), S(e_2, e_1)\}$
		$(\mathrm{Irr}(R), e)$:	$\{R(e, e)\}$

Intuitively, we must at context c make for a an exception to either $V \sqsubseteq F_{x_i}$ or $V \sqsubseteq T_{x_i}$, for each i; the respective single minimal clashing set is $\{V(a), \neg F_{x_i}(a)\}$ resp. $\{V(a), \neg T_{x_i}(a)\}$. Keeping $V \sqsubseteq T_{x_i}$ (resp. $V \sqsubseteq F_{x_i}$) justifies $\neg F_{x_i}(a)$ (resp. $\neg T_{x_i}(a)$) via the axiom $T_{x_i} \sqcap F_{x_i} \sqsubseteq \perp$. Every truth assignment σ to $x_1, \ldots, x_m$ thus gives rise to a natural justified clashing assumption χ_σ that for c includes $\langle V \sqsubseteq F_{x_i}, a\rangle$ if $\sigma(x_i) = true$ and $\langle V \sqsubseteq T_{x_i}, a\rangle$ if $\sigma(x_i) = false$. Furthermore, if σ does not satisfy the disjunct δ_e, then $\exists R.(\{c\} \sqcap C_e)$ for a is concluded; in further steps then, $E_e(x, y)$ is concluded for each pair $(c, c') \in \{r, g, b\}^2$. If σ satisfies the disjunct δ_e, then no such conclusion is made, and we have $E_e(c, c')$ only if $c \neq c'$. On the other hand, no justified clashing assumption χ for c can include both $\langle V \sqsubseteq F_{x_i}, c\rangle$ or $\langle V \sqsubseteq T_{x_i}, c\rangle$ where c is a or its standard name, as without an inherited axiom $V \sqsubseteq F$ resp. $V \sqsubseteq T$ it is not possible to derive $\neg F_{x_i}(a)$ and $\neg T_{x_i}(a)$.

The Boolean query that we construct is

$$Q = \exists \mathbf{y} \bigwedge_{e=(i,j)\in E} \mathrm{c} : E_e(y_i, y_j).$$

Informally, the graph G is encoded in Q, where the variables y_i, y_j amount to nodes i and j. If in the assignment σ, the edge e is selected (i.e., δ_e is satisfied), then we must match y_u and y_v to different colors, as all $E_e(c, c)$, where $c \in \{r, g, b\}$ are not possible; if e is not selected, y_i and y_j can be matched to any colors. For the least CAS-model for χ_σ w.r.t. identity ν, $\mathbf{I}_\sigma = \hat{\mathbf{I}}(\chi_\sigma, \nu)$, we have that $\mathbf{I}_\sigma \models Q$ iff the graph G_σ is 3-colorable.

It is then is not difficult to verify that $\mathfrak{K} \models Q$ holds iff for every assignment σ, we have $\mathbf{I}_\sigma \models Q$, i.e., G_σ is 3-colorable. Hence, $\mathfrak{K} \models Q$ iff G is a yes-instance of CERT3COL. Furthermore, in $\mathfrak{K}$ the module structure and all assertions are fixed, and only the inclusion axioms vary; hence Π^p_2-hardness under the stated restriction holds.

For the data-complexity (i.e., fixed module structure, only the assertions in the modules $\mathrm{K_m}$ vary, and the query Q fixed), the coNP-hardness follows from the reduction of 3SAT to CKR-model existence in the proof of Theorem 3: the 3SAT instance E is unsatisfiable iff $\mathfrak{K} \models \mathrm{c} : V(c_1)$ resp. $\mathfrak{K} \models V(c_1)$, say, as this is equivalent to $\mathfrak{K}$ not having a CKR-model. □

Appendix A.5. Proofs of Section on Datalog Translation

Appendix A.5.1. Normal form

Table A.17 shows minimal clashing sets for instances of defeasible axioms in normal form (under unique name assumption, some particular instances can be further be simplified).

Proof of Lemma 5. Intuitively, to prove item (a) we need to show that the rules for normal form transformation are complete with respect to the possible input axioms; to prove item (b) one has to show that the translations produce at most a linear increase in the size of the output CKR. Finally, to prove item (c), we have to show that any interpretation satisfying the original axiom can be extended to an interpretation satisfying the transformed axiom set, and, conversely, that any interpretation satisfying the transformed axioms set satisfies the original axiom: in verifying this, we have also to prove that the interpretations agree on the justification of the overridings.

(a). Let $\Sigma = NC_\Sigma \uplus NR_\Sigma \uplus NI_\Sigma$. and $\Gamma = NC_\Gamma \uplus NR_\Gamma \uplus NI_\Gamma$. We extend such vocabulary to $\overline{\Sigma}$ and $\overline{\Gamma}$ in all of its components by adding a distinct set of new symbols: that is $\overline{\Sigma} = \overline{NC}_\Sigma \uplus \overline{NR}_\Sigma \uplus \overline{NI}_\Sigma$ with $\overline{NC}_\Sigma = NC_\Sigma \cup X_\Sigma$, $\overline{NR}_\Sigma = NR_\Sigma \cup W_\Sigma$, $\overline{NI}_\Sigma = NI_\Sigma \cup Z_\Sigma$. similarly, $\overline{\Gamma} = \overline{NC}_\Gamma \uplus \overline{NR}_\Gamma \uplus \overline{NI}_\Gamma$ with $\overline{NC}_\Gamma = NC_\Gamma \cup X_\Gamma$, $\overline{NR}_\Gamma = NR_\Gamma \cup W_\Gamma$, $\overline{NI}_\Gamma = NI_\Gamma \cup Z_\Gamma$. The extended vocabularies are only used to consider the new symbols added in the translation.

We prove the assertion showing the following claim: *the CKR* $\mathfrak{K}'$ *over extended vocabulary* $\langle \overline{\Sigma}, \overline{\Gamma} \rangle$, *that is obtained by exhaustively applying the rules in Table 2 to axioms in* $\mathfrak{K}$, *is in* $\mathcal{SROIQ}$*-RL and in normal form*. We can prove the claim by cases on the possible form of input axioms.

Let $\alpha \in \mathfrak{K}$ be a $\mathcal{SROIQ}$-RL axiom. We assume $\alpha \in \mathcal{L}_\Sigma$: the case for $\alpha \in \mathcal{L}_\Gamma$ can be shown similarly. We consider all of the possible cases in which α is not already in normal form and show how the rules can be applied to yield a normal form equivalent. In the following we use the same conventions on symbols used in Table 2 (e.g., C, D represent complex concepts while A, B are concept names).

- If $\alpha = D(a)$, by applying the corresponding rule in Table 2 we obtain $S = \{X(a), X \sqsubseteq D\}$. We note that since X is a new concept name, $X(a)$ is in normal form, while $X \sqsubseteq D$ need further expansion (that will be shown as one of the cases below): we remark however that the latter axiom is in $\mathcal{SROIQ}$-RL, since by definition concept assertions can be only defined over right concept expressions.
- If $\alpha = C \sqsubseteq D$, by applying the corresponding rule in Table 2 we obtain $S = \{C \sqsubseteq X, X \sqsubseteq D\}$. As in the case above, both $C \sqsubseteq X$ and $X \sqsubseteq D$ need further expansion (treated in the cases below), but the axioms are indeed in $\mathcal{SROIQ}$-RL.
- If $\alpha = C \sqsubseteq A$, then we can recognize the following cases:
  - If $C = A$, $\{a\}$ or $\exists R.\top$, then α is already in normal form.
  - If $C = C_1 \sqcap C_2$, then by applying the rule of Table 2 we obtain the set $\{C_1 \sqsubseteq Y_1, C_2 \sqsubseteq Y_2, Y_1 \sqcap Y_2 \sqsubseteq X\}$. All of the axioms are in $\mathcal{SROIQ}$-RL and the first two axioms can be further expanded following the case for $\alpha = C \sqsubseteq A$.
  - If $C = C_1 \sqcap C_2$, then by applying the rule of Table 2 we obtain the set $\{C_1 \sqsubseteq A, C_2 \sqsubseteq A\}$. All of the axioms are in $\mathcal{SROIQ}$-RL and the two axioms can be then expanded following the case for $\alpha = C \sqsubseteq A$.
  - If $C = \exists R.C_1$ (or, similarly, if $C = \exists R.\{a\}$), then by applying the rule of Table 2 we obtain the set $\{C_1 \sqsubseteq X, \exists R.X \sqsubseteq A\}$. The axioms are in $\mathcal{SROIQ}$-RL and the second axiom is in normal form while the first can be expanded following the case for $\alpha = C \sqsubseteq A$.
  - If $C = \mathit{eval}(C_1, \mathrm{C})$, then by applying the rule of Table 2 we obtain the set $\{\mathit{eval}(X, Y) \sqsubseteq B \in \mathrm{K_m}, C_1 \sqsubseteq X \in \mathrm{K_{mx}}, \mathrm{C} \sqsubseteq Y \in \mathfrak{G}, Y \sqsubseteq \exists \mathsf{mod}.\{\mathsf{mx}\} \in \mathfrak{G}\}$. Note that the axioms are in

SROIQ-RL and the first and last axioms are in normal form. The axioms $C_1 \sqsubseteq X$ and $C \sqsubseteq Y$ can be expanded following again the case for $\alpha = C \sqsubseteq A$.

  - If $C = \exists eval(R, C).A$, then by applying the rule of Table 2 we obtain the set $\{\exists W.A \sqsubseteq B, eval(R, C) \sqsubseteq W\}$. The axioms are in SROIQ-RL, the first is in normal form, while the second can be further expanded following the case for $\alpha = eval(R, C) \sqsubseteq T$.

- If $\alpha = A \sqsubseteq D$, then we can recognize the following cases:
  - If $D = A$, $\exists R.\{a\}$ or $\leq 1R.\top$, then α is already in normal form.
  - If $D = \neg C_1$, then by applying the corresponding rule of Table 2 we obtain the axiom $\{A \sqcap C \sqsubseteq \bot\}$. The axiom is in SROIQ-RL and can be further expanded following the case for $\alpha = C \sqsubseteq A$.
  - If $D = D_1 \sqcap D_2$, then by applying the corresponding rule of Table 2 we obtain the set $\{A \sqsubseteq D_1, A \sqsubseteq D_2\}$. Both axioms are in SROIQ-RL and can be further expanded following the case for $\alpha = A \sqsubseteq D$.
  - If $D = \forall R.D_1$, then by applying the corresponding rule of Table 2 we obtain the set $\{A \sqsubseteq \forall R.X, X \sqsubseteq D_1\}$. Both axioms are in SROIQ-RL and the first is in normal form: the second axiom can be further expanded following the case for $\alpha = A \sqsubseteq D$.
  - If $D = \leq 0R.\top$, then by consecutively applying the rule of Table 2 relative for axioms $A \sqsubseteq \leq 0R.\top$ and then the rule for $A \sqsubseteq \forall R.D$, we obtain the set $\{X \sqsubseteq \forall R.Y_1, Y_1 \sqsubseteq \neg\top\}$. It is easy to check that all of the axioms are in SROIQ-RL and in normal form.

- If $\alpha = \mathrm{Sym}(P)$, $\mathrm{Trans}(P)$ or $\mathrm{Asym}(P)$, then by applying the corresponding rules of Table 2 we directly obtain a set of SROIQ-RL axioms in normal form.
- If $\alpha = eval(R, C) \sqsubseteq T$, then by applying the rule of Table 2 we obtain the set $\{eval(R, Y) \sqsubseteq T \in K_m, C \sqsubseteq Y \in G\}$. Both axioms are in SROIQ-RL and the first is in normal form. The second one can be expanded following the case for $\alpha = C \sqsubseteq A$. This can be shown analogously for $\alpha = eval(R, C) \circ S \sqsubseteq T$ and $\alpha = \mathrm{Dis}(eval(R, C), S)$.

In the cases where $\alpha = \mathrm{D}(\beta)$, transformations are similar to the non-defeasible cases, and thus the assertion can be shown using an analogous reasoning as follows.

- If $\alpha = \mathrm{D}(D(a))$, by applying the corresponding rule in Table 2 we obtain the set $S = \{X(a), \mathrm{D}(X \sqsubseteq D)\}$. We note that $X(a)$ is in normal form, while $X \sqsubseteq D$ need further expansion (shown in the cases below); both axioms are in SROIQ-RLD.
- If $\alpha = \mathrm{D}(C \sqsubseteq D)$, by applying the corresponding rule in Table 2 we obtain $S = \{C \sqsubseteq X, \mathrm{D}(X \sqsubseteq D)\}$. As in the strict case, both $C \sqsubseteq X$ and $X \sqsubseteq D$ need further expansion, but the axioms are indeed in SROIQ-RLD. Note that this case covers also the case $\mathrm{D}(C \sqsubseteq A)$ (i.e. where C is complex and A is atomic): the further expansion of the left side can be completed using the transformation rules for the strict axioms.
- If $\alpha = \mathrm{D}(A \sqsubseteq D)$, then we can recognize the following cases:
  - If $D = A$, $\exists R.\{a\}$ or $\leq 1R.\top$, then α is already in normal form.
  - If $D = \neg C_1$, then by applying the corresponding rule of Table 2 we obtain $\{\mathrm{D}(A \sqcap C \sqsubseteq \bot)\}$. The axiom is in SROIQ-RLD and can be further expanded following the strict case for $\alpha = C \sqsubseteq A$.
  - If $D = \leq 0R.\top$, then by the rules of Table 2 relative for axioms $A \sqsubseteq \leq 0R.\top$ and then the rule for $A \sqsubseteq \forall R.D$ (as in the strict case), we obtain the set $\{X \sqsubseteq \forall R.Y_1, \mathrm{D}(Y_1 \sqsubseteq \neg\top)\}$. It is easy to check that all of the axioms are in SROIQ-RLD and in normal form.

- As in the strict case, if $\alpha = \mathrm{D}(\mathrm{Sym}(P))$, $\mathrm{D}(\mathrm{Trans}(P))$ or $\mathrm{D}(\mathrm{Asym}(P))$, then by applying the rules of Table 2 we obtain a set of axioms in SROIQ-RLD and in normal form.

(b). The assertion can be proved by introducing a measure on input axioms from K. Given a concept C over Σ or Γ, we define its size $||C||$ as:

- $||A|| = 0$ for $A \in \mathrm{NC}_\Sigma$, $||R|| = 0$ for $R \in \mathrm{NR}_\Sigma$, $||\{a\}|| = 0$ for $a \in \mathrm{NI}_\Sigma$;
- $||\neg C|| = ||C|| + 1$;
- $||C_1 \, \mathsf{S} \, C_2|| = ||C_1|| + ||C_2|| + 1$, for $\mathsf{S} \in \{\sqcap, \sqcap\}$;
- $||\mathsf{Q}R.C_1|| = ||R|| + ||C_1|| + 1$, for $\mathsf{Q} \in \{\exists, \forall, \leq n\}$;
- $||\mathit{eval}(C_1, \mathrm{C})|| = ||C_1|| + ||\mathrm{C}|| + 1$ and $||\mathit{eval}(R, \mathrm{C})|| = ||\mathrm{C}|| + 1$;

We extend the definition to axioms α in L^e_Σ as:

- $||C(a)|| = ||C||$;
- $||R(a, b)|| = 0$;
- $||C \pm D|| = ||C|| + ||D|| + 1$;
- $||\mathit{char}_1(R)|| = ||\mathit{char}_2(R, S)|| = 0$, for $\mathit{char}_1 \in \{\mathrm{Sym}, \mathrm{Trans}, \mathrm{Asym}, \mathrm{Irr}\}$ and $\mathit{char}_2 \in \{\mathrm{Inv}, \mathrm{Dis}\}$.

For defeasible axioms, $||\mathrm{D}(\alpha)|| = ||\alpha||$. The size of sets of axioms is the sum of sizes of all their components. We can prove that every rule in Table 2 adds in the size $||S||$ of the resulting set at most a linear increase w.r.t. the size $||\alpha||$ of the input axiom. This can be easily proved by cases on the rules of Table 2; for example:

- Let $\alpha = C(a)$, then $||\alpha|| = ||C||$ and $||S|| = ||C|| + 1$. Thus $||S|| = ||\alpha|| + 1$.
- Let $\alpha = C \pm D$, then $||\alpha|| = ||C|| + ||D|| + 1$ and $||S|| = ||C|| + 1 + ||D|| + 1$. Thus $||S|| = ||\alpha|| + 1$.
- Let $\alpha = A \pm \neg C$, then $||\alpha|| = ||\neg C|| + 1 = ||C|| + 2$ and $||S|| = ||A|| + ||C|| + 1 + ||\bot|| + 1 = ||C|| + 2$. Thus $||S|| = ||\alpha||$.
- Let $\alpha = A \pm C \sqcap D$, then $||\alpha|| = ||C \sqcap D|| + 1 = ||C|| + ||D|| + 2$ and $||S|| = ||C|| + 1 + ||D|| + 1 = ||C|| + ||D|| + 2$. Thus $||S|| = ||\alpha||$.
- Let $\alpha = \mathit{eval}(C_1, \mathrm{C}) \pm B$ then $||\alpha|| = ||\mathit{eval}(C_1, \mathrm{C})|| + 1 = ||C_1|| + ||\mathrm{C}|| + 2$ and $||S|| = 1 + (||C_1|| + 1) + (||\mathrm{C}|| + 1) + 2 = ||C_1|| + ||\mathrm{C}|| + 2$. Thus $||S|| = ||\alpha|| + 3$.

Note that the cases for defeasible axioms are similar to the corresponding strict cases.

(c). For every axiom α in L^e_Σ (as above, the case for $\alpha \in \mathrm{L}_\Gamma$ can be proved similarly), let S be the set of axioms obtained from the application of the corresponding rule in Table 2 to α. Let $\mathrm{I}^{N\mathrm{K}}_{CAS}$ be a justified named CAS-model for K on (Σ, Γ) such that $\mathrm{I}^{N\mathrm{K}}_{CAS} \models \mathsf{c} : \alpha$ for $\mathsf{c} \in \mathbf{N}$. We can extend this interpretation to the interpretation $\bar{\mathrm{I}}^{N\mathrm{K}}_{CAS}$ on $\langle \bar{\Sigma}, \bar{\Gamma} \rangle$ such that:

- Let $A \in \overline{\mathrm{NC}_\Sigma}$. If $A \in \mathrm{NC}_\Sigma$ then $A^{\bar{\mathrm{I}}(\mathsf{c})} = A^{\mathrm{I}(\mathsf{c})}$. Otherwise, if $A \in \overline{X_\Sigma}$ then it has been introduced in the translation in an axiom set S: then $A^{\bar{\mathrm{I}}(\mathsf{c})}$ is the least set of $d \in \Delta^{\mathrm{I}(\mathsf{c})}$ such that $\bar{\mathrm{I}}(\mathsf{c}) \models S$. Similarly for $A \in \mathrm{NC}_\Gamma$ and the global interpretation M.

- Let $R \in \overline{\mathrm{NR}_\Sigma}$. If $R \in \mathrm{NR}_\Sigma$ then $R^{\overline{\mathcal{I}(c)}} = R^{\mathcal{I}(c)}$. Otherwise, if $R \in \overline{W_\Sigma}$ then it has been introduced in the translation in an axiom set S: then $R^{\overline{\mathcal{I}(c)}}$ is the least set of $(d, d) \in \Delta^{\mathcal{I}(c)} \times \Delta^{\mathcal{I}(c)}$ such that $\overline{\mathcal{I}(c)} \models S$. Similarly for $R \in \mathrm{NR}_\Gamma$ and the global interpretation $\mathcal{M}$.
- Let $a \in \overline{\mathrm{NI}_\Sigma}$. If $a \in \mathrm{NI}_\Sigma$ then $a^{\overline{\mathcal{I}(c)}} = a^{\mathcal{I}(c)}$. Otherwise, if $a \in \overline{Z_\Sigma}$ then it has been introduced in the translation in an axiom set S: then $a^{\overline{\mathcal{I}(c)}}$ is a new domain element $d \in \Delta^{\mathcal{I}(c)}$ such that $\overline{\mathcal{I}(c)} \models S$. Similarly for $a \in \mathrm{NI}_\Gamma$ and the global interpretation $\mathcal{M}$.
- If $(\alpha, \mathbf{e}) \in \chi(x)$: if α is in normal form, then $(\alpha, \mathbf{e}) \in \chi'(x)$; otherwise, let α' be the (single) defeasible axiom in the set obtained by exhaustively applying the rules in Table 2 to α, then $(\alpha', \mathbf{e}) \in \chi'(x)$.

For the first direction (i), we show that for all axioms α in $\mathcal{L}^e_\Sigma$, with $\mathfrak{I}^{N_K}_{CAS} \models c : \alpha$, we have $\overline{\mathfrak{I}}^{N_K}_{CAS}$ built as above is a model of K' and $\overline{\mathfrak{I}}^{N_K}_{CAS} \models c : \alpha$.

For strict axiom cases, we can show the claim by induction on the form of α and transformation rules, for example:

- If $\alpha = C \sqsubseteq D$, then $S = \{C \sqsubseteq X, X \sqsubseteq D\}$. By hypothesis $\mathcal{I}(c) \models C \sqsubseteq D$, that is $C^{\mathcal{I}(c)} \subseteq D^{\mathcal{I}(c)}$: by construction we have that $C^{\overline{\mathcal{I}(c)}} \subseteq X^{\overline{\mathcal{I}(c)}}$ and $X^{\overline{\mathcal{I}(c)}} \subseteq D^{\overline{\mathcal{I}(c)}}$. This implies that $C^{\overline{\mathcal{I}(c)}} \subseteq D^{\overline{\mathcal{I}(c)}}$ and thus $\overline{\mathcal{I}(c)} \models \alpha$.
- If $\alpha = C(a)$, then $S = \{X(a), X \sqsubseteq C\}$. By hypothesis $\mathcal{I}(c) \models C(a)$ that is $a^{\mathcal{I}(c)} \in C^{\mathcal{I}(c)}$: by construction, $a^{\overline{\mathcal{I}(c)}} \in X^{\overline{\mathcal{I}(c)}}$ and $X^{\overline{\mathcal{I}(c)}} \subseteq C^{\overline{\mathcal{I}(c)}}$, which directly implies $a^{\overline{\mathcal{I}(c)}} \in X^{\overline{\mathcal{I}(c)}}$ and thus $\overline{\mathcal{I}(c)} \models \alpha$.
- If $\alpha = A \sqsubseteq C \sqcap D$, then $S = \{A \sqsubseteq C, A \sqsubseteq D\}$. By hypothesis $\mathcal{I}(c) \models \alpha$, thus $A^{\mathcal{I}(c)} \subseteq C^{\mathcal{I}(c)} \cap D^{\mathcal{I}(c)}$. This implies that $A^{\mathcal{I}(c)} \subseteq C^{\mathcal{I}(c)}$ and $A^{\mathcal{I}(c)} \subseteq D^{\mathcal{I}(c)}$: then, by definition of $\overline{\mathcal{I}(c)}$, $A^{\overline{\mathcal{I}(c)}} \subseteq C^{\overline{\mathcal{I}(c)}}$ and $A^{\overline{\mathcal{I}(c)}} \subseteq D^{\overline{\mathcal{I}(c)}}$ which directly implies that $\overline{\mathcal{I}(c)} \models \alpha$.
- If $\alpha = \mathit{eval}(C_1, \mathsf{C}) \sqsubseteq B$, then $S = \{\mathit{eval}(X, Y) \sqsubseteq B \in \mathsf{K}_m, C_1 \sqsubseteq X \in \mathsf{K}_{mx}, \mathsf{C} \sqsubseteq Y \in \mathsf{G}, Y \sqsubseteq \exists\mathrm{mod}.\{\mathrm{mx}\} \in \mathsf{G}\}$. By hypothesis $\mathcal{I}(c) \models \mathit{eval}(C_1, \mathsf{C}) \sqsubseteq B$: hence for every $x^{\mathcal{M}} \in \mathsf{C}^{\mathcal{M}}$, $a^{\mathcal{I}(x)} \in C_1^{\mathcal{I}(x)}$ implies $a^{\mathcal{I}(c)} \in B^{\mathcal{I}(c)}$. Considering the translated set S and the definition of $\overline{\mathfrak{I}}^{N_K}_{CAS}$ we have that $x^{\overline{\mathcal{M}}} \in Y^{\overline{\mathcal{M}}}$ and, by the content of its newly associated module mx, $C_1^{\overline{\mathcal{I}(x)}} \subseteq X^{\overline{\mathcal{I}(x)}}$. By the interpretation of the eval expression in S, we have that for every $a^{\overline{\mathcal{I}(x)}} \in C_1^{\overline{\mathcal{I}(x)}}$ then $a^{\overline{\mathcal{I}(c)}} \in B^{\overline{\mathcal{I}(c)}}$, and thus $\overline{\mathcal{I}(c)} \models \alpha$.

In the case of defeasible axioms, we can moreover show that the translation preserves the justification of the axioms. For all defeasible axioms, if we consider the case of $\mathcal{M}$ and the non-exceptional case for every $\mathcal{I}(c)$ (i.e. for instances $\mathbf{e}$ s.t. $(\alpha, \mathbf{e}) \notin \chi(c)$), the result can be shown as in the strict case. Otherwise, consider the case in which there exists some $(\alpha, \mathbf{e}) \in \chi(c)$. Then, for example:

- If $\alpha = \mathrm{D}(C \sqsubseteq D)$, then $S = \{C \sqsubseteq X, \mathrm{D}(X \sqsubseteq D)\}$. Then, by construction, if $(\alpha, e) \in \chi(c)$, we have $(X \sqsubseteq D, e) \in \chi'(c)$. Since $\mathfrak{I}^{N_K}_{CAS}$ is justified for K, then there exists a clashing set $Q = \{\neg D(e), C(e)\}$ that is satisfied by $\mathcal{I}(c)$. By definition, we also have that $\overline{\mathcal{I}(c)} \models Q$. Then, in $\overline{\mathfrak{I}}^{N_K}_{CAS}$ we also have a justification for $(X \sqsubseteq D, e)$, since it holds that $Q' = \{\neg D(e), X(e)\}$ is a clashing set that is satisfied in $\overline{\mathcal{I}(c)}$.

– If $\alpha = \mathrm{D}(D(a))$, then $S = \{X(a), \mathrm{D}(X \pm D)\}$. Then, if $(\alpha, a) \in \chi(c)$, we have $(X \pm D, a) \in \chi'(c)$. Since $\mathrm{I}^{N_K}_{CAS}$ is justified for K, then there exists a clashing set $Q = \{\neg D(a)\}$ that is satisfied by $\blacksquare(c)$. We also have that, by definition, $\overline{\blacksquare(c)} \blacktriangleright Q$. Then, $\overline{\mathrm{I}}^{N_K}_{CAS}$ contains a justification for $(X \pm D, a)$, since $\blacksquare(c)$ satisfies the clashing set $Q' = \{\neg D(a), X(a)\}$.

Finally, the converse direction (ii) is proved by showing that: considering the definition of $\overline{\mathrm{I}}^{N_K}_{CAS}$ given above, for all axioms α in L_Σ, for every justified $\overline{\mathrm{I}}^{N_K}_{CAS}$ s.t. $\overline{\mathrm{I}}^{N_K}_{CAS} \blacktriangleright \mathrm{K}'$ and $\overline{\mathrm{I}}^{N_K}_{CAS} \blacktriangleright c : \alpha$, then $\mathrm{I}^{N_K}_{CAS}$ (i.e. the original model on (Σ, Γ)) is a justified model of K and $\mathrm{I}^{N_K}_{CAS} \blacktriangleright c : \alpha$. We can then again show the claim proceeding by induction on the form of α and transformation rules, for example:

– If $\alpha = C \pm D$, then $S = \{C \pm X, X \pm D\}$. By hypothesis $\overline{\blacksquare(c)} \blacktriangleright \alpha$, thus $C^{\overline{\blacksquare(c)}} \subseteq X^{\overline{\blacksquare(c)}} \subseteq D^{\blacksquare(c)}$. This implies that $C^{\blacksquare(c)} \subseteq D^{\blacksquare(c)}$ and thus $C^{\blacksquare(c)} \subseteq D^{\blacksquare(c)}$ (since $C, D \in \Sigma$), that is $\blacksquare(c) \blacktriangleright \alpha$.

– If $\alpha = C(a)$, then $S = \{X(a), X \pm C\}$. By hypothesis $\overline{\blacksquare(c)} \blacktriangleright \alpha$, thus $a^{\overline{\blacksquare(c)}} \in X^{\overline{\blacksquare(c)}}$ and $X^{\blacksquare(c)} \subseteq C^{\blacksquare(c)}$. This implies that $a^{\blacksquare(c)} \in C^{\blacksquare(c)}$ and thus $a^{\blacksquare(c)} \in C^{\blacksquare(c)}$ (since $a, C \in \Sigma$), that is $\blacksquare(c) \blacktriangleright \alpha$.

– If $\alpha = C \sqcap A \pm B$, then $S = \{C \pm X, X \sqcap A \pm B\}$. By hypothesis $\overline{\blacksquare(c)} \blacktriangleright \alpha$, thus $C^{\blacksquare(c)} \subseteq X^{\blacksquare(c)}$ and $X^{\blacksquare(c)} \cap A^{\blacksquare(c)} \subseteq B^{\blacksquare(c)}$. This implies that $C^{\blacksquare(c)} \cap A^{\blacksquare(c)} \subseteq B^{\blacksquare(c)}$. Hence $C^{\blacksquare(c)} \cap A^{\blacksquare(c)} \subseteq B^{\blacksquare(c)}$ (since $C, A, B \in \Sigma$), that is $\blacksquare(c) \blacktriangleright \alpha$.

– If $\alpha = eval(C_1, C) \pm B$ then $S = \{eval(X, Y) \pm B \in \mathrm{K_m}, C_1 \pm X \in \mathrm{K_{mx}}, C \pm Y \in \mathrm{G}, Y \pm \exists\mathrm{mod}.\{mx\} \in \mathrm{G}\}$. It holds that $\overline{\mathrm{I}}^{N_K}_{CAS} \blacktriangleright c : \alpha$, then we have that $\bigcup_{e \in Y^M} X^{\overline{\blacksquare(e)}} \subseteq B^{\overline{\blacksquare(c)}}$. This implies that $\bigcup_{e \in C} C_1^{\blacksquare(e)} \subseteq B^{\blacksquare(c)}$, that is $\mathrm{I}^{N_K}_{CAS} \blacktriangleright c : \alpha$.

In the case of defeasible axioms, we can again show that the justification of the original axiom is preserved. For all defeasible axioms, if we consider the case of M and the non-exceptional case for every $\blacksquare(c)$ (i.e. for instances **e** s.t. $(\alpha', \mathbf{e}) \notin \chi'(c)$), the result can be shown as in the strict cases. Thus, let us consider the case in which there exists some $(\alpha', \mathbf{e}) \in \chi'(c)$. Then, for example:

– If $\alpha = \mathrm{D}(C \pm D)$, then $S = \{C \pm X, \mathrm{D}(X \pm D)\}$. Then, if $(X \pm D, e) \in \chi'(c)$, by construction we have $(\alpha, e) \in \chi(c)$. By hypothesis $\overline{\mathrm{I}}^{N_K}_{CAS}$ is justified for K': then there exists a clashing set $Q' = \{\neg D(e), X(e)\}$ that is satisfied by $\overline{\blacksquare(c)}$. By definition of the translation, we also have that $\blacksquare(c) \blacktriangleright C(e)$. Then, in $\mathrm{I}^{N_K}_{CAS}$ we also have a justification for (α, e), since the corresponding clashing set $Q = \{\neg D(e), C(e)\}$ is satisfied in $\blacksquare(c)$.

– If $\alpha = \mathrm{D}(D(a))$, then $S = \{X(a), \mathrm{D}(X \pm D)\}$. Supposing $(X \pm D, a) \in \chi'(c)$, we have $(\alpha, a) \in \chi(c)$. Since $\overline{\mathrm{I}}^{N_K}_{CAS}$ is justified for K', then there exists a clashing set $Q' = \{\neg D(a), X(a)\}$ that is satisfied by $\blacksquare(c)$. This directly implies that $\blacksquare(c) \blacktriangleright \neg D(a)$. Since $Q = \{\neg D(a)\}$ is a clashing set for (α, a), then we also have a justification in $\mathrm{I}^{N_K}_{CAS}$.

□

Appendix A.5.2. Translation correctness

Given a CAS-interpretation $\mathrm{I}_{CAS} = \langle \mathcal{M}, \mathrm{I} \rangle$ we construct a corresponding Herbrand interpretation $S = I(\mathrm{I}_{CAS})$ of the program $PK(\mathrm{K})$ as the smallest set of literals so defined:

- $l \in S$, if $l \in PK(\mathrm{K})$;
- $\mathrm{instd}(a, A, \mathrm{c}, \mathrm{main}) \in S$, if $\mathrm{I}(\mathrm{c}) \models A(a)$;
- $\mathrm{tripled}(a, R, b, \mathrm{c}, \mathrm{main}) \in S$, if $\mathrm{I}(\mathrm{c}) \models R(a, b)$;
- $\mathrm{ovr}(p(\mathbf{e})) \in S$, if $\mathrm{ovr}(p(\mathbf{e})) \in OVR(\mathrm{I}_{CAS})$;
- $l \in S$ with $l = \mathrm{instd}(a, A, \mathrm{c}, t), \mathrm{tripled}(a, R, b, \mathrm{c}, t)$ and $t \neq \mathrm{main}$, if $\mathrm{test}(t) \in S$ and $l \leftarrow b_1, \ldots, b_n \in grnd(PK(\mathrm{K}))^{OVR(\mathrm{I}_{CAS})}$ where $\{b_1, \ldots, b_n\} \subseteq S$.
- $\mathrm{test}(t) \in S$, if $\mathrm{ovr}(p(\mathbf{e})) \in OVR(\mathrm{I}_{CAS})$ and $r \in grnd(PK(\mathrm{K}))$, with $\mathrm{ovr}(p(\mathbf{e})) \in Head(r)$ and $\mathrm{test\_fails}(t) \in Body(r)$;
- $\mathrm{unsat}(\mathrm{nlit}(a, A, \mathrm{c})) \in S$, if $\mathrm{I}(\mathrm{c}) \not\models \mathrm{K_c} \cup \{A(a)\}$;
- $\mathrm{unsat}(\mathrm{nrel}(a, R, b, \mathrm{c})) \in S$, if $\mathrm{I}(\mathrm{c}) \not\models \mathrm{K_c} \cup \{R(a, b)\}$;
- $\mathrm{test\_fails}(t) \in S$, if $\mathrm{unsat}(t) \notin S$.

Note that $\mathrm{unsat}(\mathrm{main}) \notin S$.

Proof of Proposition 13. We will prove that $I(\mathcal{M}_\mathrm{G})$ is an answer set for $PG(\mathrm{G})$ if G is satisfiable. Note that, by restricting the definition of $I(\mathrm{I}_{CAS})$ to the global context, $I(\mathcal{M}_\mathrm{G})$ is as follows:

- Facts of $PG(\mathrm{G})$ are included in $I(\mathcal{M}_\mathrm{G})$;
- $\mathrm{instd}(a, A, \mathrm{g}, \mathrm{main}) \in I(\mathcal{M}_\mathrm{G})$ iff $\mathcal{M}_\mathrm{G} \models A(a)$ for $A \in \mathrm{NC}$ and $a \in \mathrm{NI}$;
- $\mathrm{tripled}(a, R, b, \mathrm{g}, \mathrm{main}) \in I(\mathcal{M}_\mathrm{G})$ iff $\mathcal{M}_\mathrm{G} \models R(a, b)$ for $R \in \mathrm{NR}$ and $a, b \in \mathrm{NI}$;
- $\mathrm{unsat}(\mathrm{main}) \notin I(\mathcal{M}_\mathrm{G})$;

We can show that $I(\mathcal{M}_\mathrm{G}) \models grnd(PG(\mathrm{G}))$, that is for every rule instance $r \in grnd(PG(\mathrm{G}))$ it holds that $I(\mathcal{M}_\mathrm{G}) \models r$. This can be readily established by examining the possible rules that appear in $grnd(PG(\mathrm{G}))$; we list here some representative cases. Suppose that $I(\mathcal{M}_\mathrm{G}) \models Body(r)$:

- **(prl-instd):** then $\mathrm{insta}(a, A, \mathrm{g}, \mathrm{main}) \in I(\mathcal{M}_\mathrm{G})$[27] and, by definition of the translation, $A(a) \in \mathrm{G}$. This implies that $\mathcal{M}_\mathrm{G} \models A(a)$ and thus $\mathrm{instd}(a, A, \mathrm{g}, \mathrm{main})$ is added to $I(\mathcal{M}_\mathrm{G})$.
- **(prl-ninstd):** then we would have $\{\mathrm{ninsta}(a, A, \mathrm{g}), \mathrm{instd}(a, A, \mathrm{g}, \mathrm{main})\} \subseteq I(\mathcal{M}_\mathrm{G})$. This would mean that $\neg A(a) \in \mathrm{G}$, but $\mathcal{M}_\mathrm{G} \models A(a)$. Assuming that G is satisfiable this is not possible and $\mathrm{unsat}(\mathrm{main})$ is not added to $I(\mathcal{M}_\mathrm{G})$. On the other hand, if G is not satisfiable, no answer set would exist due to the constraint (prl-sat).
- **(prl-eq):** then we would have $\mathrm{eq}(a, b, \mathrm{g}, \mathrm{main}) \in I(\mathcal{M}_\mathrm{G})$. By definition of the translation, $a = b \in \mathrm{G}$: since we are assuming UNA, this contradicts the assumption that G is satisfiable. Thus $\mathrm{unsat}(\mathrm{main})$ is not added to $I(\mathcal{M}_\mathrm{G})$. However, if G is not satisfiable, no answer set would exist due to the constraint (prl-sat).
- **(prl-subc):** then $\{\mathrm{subClass}(A, B, \mathrm{g}), \mathrm{instd}(a, A, \mathrm{g}, \mathrm{main})\} \subseteq I(\mathcal{M}_\mathrm{G})$. By definition of the translation we have $A \sqsubseteq B \in \mathrm{G}$ and, for the construction of $I(\mathcal{M}_\mathrm{G})$, $\mathcal{M}_\mathrm{G} \models A(a)$ which implies that $\mathcal{M}_\mathrm{G} \models B(a)$. Thus $\mathrm{instd}(a, B, \mathrm{g}, \mathrm{main})$ is added to $I(\mathcal{M}_\mathrm{G})$.

[27] In this proof, for simplicity of notation, we consider $\mathrm{g} = \mathrm{gm}$ or $\mathrm{g} = \mathrm{gk}$.

– **(prl-subex):** then $\{\mathsf{subEx}(A, R, B, \mathsf{g}, \mathsf{main}), \mathsf{instd}(b, A, \mathsf{g}, \mathsf{main}), \mathsf{tripled}(a, R, b, \mathsf{g}, \mathsf{main})\} \subseteq I(\mathfrak{M}_{\mathfrak{G}})$. By definition of the translation we have $\exists R.A \sqsubseteq B \in \mathfrak{G}$ and, for the construction of $I(\mathfrak{M}_{\mathfrak{G}})$, $\mathfrak{M}_{\mathfrak{G}} \models A(b), R(a, b)$. By definition of the semantics, this proves $\mathfrak{M}_{\mathfrak{G}} \models (\exists R.A)(a)$ which implies that $\mathfrak{M}_{\mathfrak{G}} \models B(a)$ and $\mathsf{instd}(a, B, \mathsf{g}, \mathsf{main})$ is added to $I(\mathfrak{M}_{\mathfrak{G}})$.

– **(prl-supex):** then $\{\mathsf{supEx}(A, R, a, \mathsf{g}, \mathsf{main}), \mathsf{instd}(b, A, \mathsf{g}, \mathsf{main})\} \subseteq I(\mathfrak{M}_{\mathfrak{G}})$. By definition of the translation we have $A \sqsubseteq \exists R.\{a\} \in \mathfrak{G}$ and, for the construction of $I(\mathfrak{M}_{\mathfrak{G}})$, $\mathfrak{M}_{\mathfrak{G}} \models A(b)$: this implies that $b^{\mathfrak{M}_{\mathfrak{G}}} \in (\exists R.\{a\})^{\mathfrak{M}}$, that is $(b^{\mathfrak{M}_{\mathfrak{G}}}, a^{\mathfrak{M}_{\mathfrak{G}}}) \in R^{\mathfrak{M}_{\mathfrak{G}}}$. Thus $\mathsf{tripled}(b, R, a, \mathsf{g}, \mathsf{main})$ is added to $I(\mathfrak{M}_{\mathfrak{G}})$.

On the other hand, we can show that if M is the (unique) answer set of $PG(\mathfrak{G})$, then we can build an interpretation $\mathfrak{M}$ (using the conditions above) such that $\mathfrak{M} \models \mathfrak{G}$. Specifically, $\mathfrak{M} = \langle \Delta^{\mathfrak{M}}, \cdot^{\mathfrak{M}} \rangle$ is as follows:

– $\Delta^{\mathfrak{M}} = \{c \mid c \in \mathrm{NI}_{\Gamma} \cup \mathrm{NI}_{\Sigma}\}$;

– $c^{\mathfrak{M}} = c$, for every $c \in \mathrm{NI}_{\Gamma} \cup \mathrm{NI}_{\Sigma}$;

– $A^{\mathfrak{M}} = \{d \in \Delta^{\mathfrak{M}} \mid M \models \mathsf{instd}(d, A, \mathsf{g}, \mathsf{main})\}$, for every $A \in \mathrm{NC}_{\Gamma} \cup \mathrm{NC}_{\Sigma}$;

– $R^{\mathfrak{M}} = \{(d, d') \in \Delta^{\mathfrak{M}} \times \Delta^{\mathfrak{M}} \mid M \models \mathsf{tripled}(d, R, d', \mathsf{g}, \mathsf{main})\}$ for $R \in \mathrm{NR}_{\Gamma} \cup \mathrm{NR}_{\Sigma}$.

We then can show that $\mathfrak{M} \models \mathfrak{G}$; hence $\mathfrak{M} \in \mathcal{M}$ holds. Furthermore, by construction of $I(\mathfrak{M}_{\mathfrak{G}})$ from $\mathfrak{G}$ and $\mathfrak{M}$ from M, as $M \subseteq I(\mathfrak{M}_{\mathfrak{G}})$ we obtain $\mathfrak{M} \subseteq \mathfrak{M}_{\mathfrak{G}}$. It follows that $\mathfrak{M}_{\mathfrak{G}} = \mathfrak{M}$ which proves that $M = I(\mathfrak{M}_{\mathfrak{G}})$ is the unique answer set of $PG(\mathfrak{G})$.

To show that $\mathfrak{M}$ is a model for $\mathfrak{G}$, we must verify that $\mathfrak{M}$ satisfies (a) the condition (i) of a CKR interpretation and (b) the conditions (i) and (ii) of a CAS-model. As for (a), we easily prove that $\mathbf{N}^{\mathfrak{M}} \subseteq \mathsf{Ctx}^{\mathfrak{M}}$: by rule (igl-subctx2), for every $\mathsf{c} \in \mathbf{N}$ we have $M \models \mathsf{instd}(\mathsf{c}, \mathsf{Ctx}, \mathsf{gm}, \mathsf{main})$, which implies $\mathsf{c}^{\mathfrak{M}} \in \mathsf{Ctx}^{\mathfrak{M}}$. The condition $\mathsf{C}^{\mathfrak{M}} \subseteq \mathsf{Ctx}^{\mathfrak{M}}$ for every $\mathsf{C} \in \mathbf{C}$ can be shown similarly by the rule (igl-subctx1). As for (b), we consider the form of all axioms β in $\mathcal{L}_{\Gamma}$ resp. $\mathcal{L}_{\Sigma}$ that can occur in $\mathfrak{G}$. We show here only some of the cases (the others are similar):

– Let $\beta = A(a) \in \mathfrak{G}$, then, by rule (prl-instd), $M \models \mathsf{instd}(a, A, \mathsf{g}, \mathsf{main})$. This directly implies that $a^{\mathfrak{M}} \in A^{\mathfrak{M}}$.

– Let $\beta = \neg A(a) \in \mathfrak{G}$, then $M \models \mathsf{ninsta}(a, A, \mathsf{g}, \mathsf{main})$. Assuming that $\mathfrak{M} \models A(a)$, we would have that $M \models \mathsf{instd}(a, A, \mathsf{g}, \mathsf{main})$. But, by the rule (prl-ninstd), we would have that $\mathsf{unsat}(\mathsf{main}) \in M$ and M would violate the constraint (prl-sat). Thus $\mathfrak{M} \not\models A(a)$.

– Let $\beta = A \sqsubseteq B \in \mathfrak{G}$, then $M \models \mathsf{subClass}(A, B, \mathsf{g})$. If $d \in A^{\mathfrak{M}}$, then by definition $M \models \mathsf{instd}(d, A, \mathsf{g}, \mathsf{main})$: by rule (prl-subc) we obtain that $M \models \mathsf{instd}(d, B, \mathsf{g}, \mathsf{main})$ and thus $d \in B^{\mathfrak{M}}$.

– Let $\beta = A_1 \sqcap A_2 \sqsubseteq B \in \mathfrak{G}$. Then $M \models \mathsf{subConj}(A_1, A_2, B, \mathsf{g})$. Supposing $d \in A_1^{\mathfrak{M}} \cap A_2^{\mathfrak{M}}$, then by definition $M \models \mathsf{instd}(d, A_1, \mathsf{g}, \mathsf{main})$ and $\mathsf{instd}(d, A_2, \mathsf{g}, \mathsf{main})$: by rule (prl-subcnj) we obtain that $M \models \mathsf{instd}(d, B, \mathsf{g}, \mathsf{main})$ and thus $d \in B^{\mathfrak{M}}$.

– Let $\beta = \exists R.A \sqsubseteq B \in \mathfrak{G}$, then $M \models \mathsf{subEx}(R, A, B, \mathsf{g})$. Let $d \in (\exists R.A)^{\mathfrak{M}}$: by definition of the semantics this means that some $d' \in A^{\mathfrak{M}}$ exists such that $(d, d') \in R^{\mathfrak{M}}$. Thus, $M \models \mathsf{instd}(d', A, \mathsf{g}, \mathsf{main})$ and $M \models \mathsf{tripled}(d, R, d', \mathsf{g}, \mathsf{main})$. By rule (prl-subex), we obtain that $M \models \mathsf{instd}(d, B, \mathsf{g}, \mathsf{main})$: thus $d \in B^{\mathfrak{M}}$ as required.

– Let $\beta = A \sqsubseteq \forall R.B \in \mathfrak{G}$. Then $M \models \mathsf{supForall}(R, A, B, \mathsf{g})$. Let $d \in A^{\mathfrak{M}}$ and there is $(d, d') \in R^{\mathfrak{M}}$: then $M \models \mathsf{instd}(d, A, \mathsf{g}, \mathsf{main})$ and $M \models \mathsf{tripled}(d, R, d', \mathsf{g}, \mathsf{main})$. By rule (prl-forall), we have that $M \models \mathsf{instd}(d', B, \mathsf{g}, \mathsf{main})$ which implies $\mathfrak{M} \models B(d')$.

– Let $\beta = R \circ S \sqsubseteq T \in \mathsf{G}$. Then $M \models \mathsf{subRChain}(R, S, T, \mathsf{g})$. Supposing $(a, c) \in R^{\mathsf{M}}$ and $(c, b) \in R^{\mathsf{M}}$, we have $M \models \mathsf{tripled}(a, R, c, \mathsf{g}, \mathsf{main})$ and $M \models \mathsf{tripled}(c, S, b, \mathsf{g}, \mathsf{main})$. By rule (prl-subrc), we have that $M \models \mathsf{tripled}(a, T, b, \mathsf{g}, \mathsf{main})$ which implies $\mathsf{M} \models T(a, b)$.

– Let $\beta = \mathrm{Dis}(R, S) \in \mathsf{G}$, then $M \models \mathsf{dis}(R, S, \mathsf{g})$. Suppose that $(d, d') \in R^{\mathsf{M}}$ and $(d, d') \in S^{\mathsf{M}}$: then we have $M \models \mathsf{tripled}(d, R, d', \mathsf{g}, \mathsf{main})$ and $M \models \mathsf{tripled}(d, S, d', \mathsf{g}, \mathsf{main})$. By rule (prl-dis) we would obtain $\mathsf{unsat}(\mathsf{main})$; but then M would violate (prl-sat), a contradiction.

By definition of the translation, for every $\mathrm{D}(\beta) \in \mathsf{G}$ with $\beta \in \mathsf{L}$, it also holds that $\mathsf{M} \models \beta$: for example, let $\beta = A(b)$; then the fact $\mathsf{insta}(b, A, \mathsf{gk}, \mathsf{main})$. is added to $PG(\mathsf{G})$, which implies that $M \models \mathsf{instd}(b, A, \mathsf{gk}, \mathsf{main})$ and, by construction of the model, $\mathsf{M} \models A(b)$. □

Proof of Lemma 6. Let us consider $S = I(\hat{\mathrm{I}}(\chi))$ defined above and the reduct $G_S(PK(\mathsf{K}))$ of $PK(\mathsf{K})$ with respect S. That is, $G_S(PK(\mathsf{K}))$ is the set of rules obtained from all ground instances of rules in $PK(\mathsf{K})$ by removing: (i) every rule r such that $S \models l$ for some NAF literal $\mathsf{not}\ l \in Body(r)$; and (ii) the NAF part from the bodies of the remaining rules. Note that the NAF literals in $PK(\mathsf{K})$ involve instances of ovr, $\mathsf{test\ fails}$ and unsat.

We can then proceed to prove items (i) and (ii) of the lemma, showing that the answer sets of $PK(\mathsf{K})$ coincide with the sets $S = I(\hat{\mathrm{I}}(\chi))$ where χ is a justified clashing assumption of K.

(i). Assuming that χ is a justified clashing assumption, we show that $S = I(\hat{\mathrm{I}}(\chi))$ is an answer set of $PK(\mathsf{K})$.

We first show that $S \models G_S(PK(\mathsf{K}))$, that is for every rule instance $r \in G_S(PK(\mathsf{K}))$ it holds that $S \models r$. We can prove this by examining the possible rule forms that occur in $G_S(PK(\mathsf{K}))$. The cases for the rules in P_{rl} are analogous to the proof of Proposition 13. Here we show some of the representative cases (other cases can be shown by similar reasoning). Assuming that $S \models Body(r)$ and r stems from a rule r' in $grnd(PK(\mathsf{K}))$ of the following form, we show that $Head(r) \in S$ and thus r is satisfied:

– **(prl-instd):** then $\mathsf{insta}(a, A, \mathsf{c}, t) \in I(\hat{\mathrm{I}}(\chi))$ and, by definition of the translation, $A(a) \in \mathsf{K_c}$ (as t can only be main). This implies that $\mathsf{I}(\mathsf{c}) \models A(a)$ and thus $\mathsf{instd}(a, A, \mathsf{c}, \mathsf{main})$ is added to $I(\hat{\mathrm{I}}(\chi))$.

– **(prl-subc):** then $\{\mathsf{subClass}(A, B, \mathsf{c}), \mathsf{instd}(a, A, \mathsf{c}, t)\} \subseteq I(\hat{\mathrm{I}}(\chi))$. By definition of the translation we have $A \sqsubseteq B \in \mathsf{K_c}$. For the construction of $I(\hat{\mathrm{I}}(\chi))$, if $t = \mathsf{main}$ then $\mathsf{I}(\mathsf{c}) \models A(a)$. This implies that $\mathsf{I}(\mathsf{c}) \models B(a)$ and $\mathsf{instd}(a, B, \mathsf{c}, t)$ is added to $I(\hat{\mathrm{I}}(\chi))$. Otherwise, if $t \neq \mathsf{main}$ then $\mathsf{instd}(a, B, \mathsf{c}, t)$ is directly added to $I(\hat{\mathrm{I}}(\chi))$ by its construction.

– **(plc-evalat):** then $\{\mathsf{subEval}(A, \mathsf{C}, B, \mathsf{c}), \mathsf{instd}(\mathsf{c}_1, \mathsf{C}, \mathsf{g}, t), \mathsf{instd}(a, A, \mathsf{c}_1, t)\} \subseteq I(\hat{\mathrm{I}}(\chi))$. Thus we have that $eval(A, \mathsf{C}) \sqsubseteq B \in \mathsf{G}$ and $\mathsf{G} \models \mathsf{C}(\mathsf{c}_1)$. For the construction of $I(\hat{\mathrm{I}}(\chi))$, if $t = \mathsf{main}$ then $\mathsf{I}(\mathsf{c}_1) \models A(a)$; This implies that $\mathsf{I}(\mathsf{c}) \models B(a)$ and $\mathsf{instd}(a, B, \mathsf{c}, t)$ is added to $I(\hat{\mathrm{I}}(\chi))$. Otherwise, if $t \neq \mathsf{main}$ then $\mathsf{instd}(a, B, \mathsf{c}, t)$ is directly added to $I(\hat{\mathrm{I}}(\chi))$ by its construction.

– **(prop-inst):** then $\mathsf{insta}(a, A, \mathsf{g}) \in I(\hat{\mathrm{I}}(\chi))$. Since $r \in G_S(PK(\mathsf{K}))$, we have that $\mathsf{ovr}(\mathsf{insta}, a, A, \mathsf{c}) \notin OVR(\hat{\mathrm{I}}(\chi))$, thus $\langle A(a) \rangle \notin \chi(\mathsf{c})$. By definition of the translation, $A(a) \in \mathsf{G}$ and thus $\mathsf{I}(\mathsf{c}) \models A(a)$; hence $\mathsf{instd}(a, A, \mathsf{c}, \mathsf{main})$ is added to $I(\hat{\mathrm{I}}(\chi))$.

– **(prop-subc):** then $\{\mathsf{subClass}(A, B, \mathsf{g}), \mathsf{instd}(a, A, \mathsf{c}, t)\} \subseteq I(\hat{\mathrm{I}}(\chi))$. As $r \in G_S(PK(\mathsf{K}))$, we have $\mathsf{ovr}(\mathsf{subClass}, a, A, B, \mathsf{c}) \notin OVR(\hat{\mathrm{I}}(\chi))$ and hence $\langle A \sqsubseteq B, a \rangle \notin \chi(\mathsf{c})$. By definition, $A \sqsubseteq B \in \mathsf{G}$ and, if $t = \mathsf{main}$, $\mathsf{I}(\mathsf{c}) \models A(a)$ Thus, for the definition of *CAS*-model and the semantics, $\mathsf{instd}(a, B, \mathsf{c}, t)$ is added to $I(\hat{\mathfrak{I}}(\chi))$. If $t \neq \mathsf{main}$, then $\mathsf{instd}(a, B, \mathsf{c}, t)$ is added to $I(\hat{\mathfrak{I}}(\chi))$ by construction.

- **(ovr-subc):** then $\{\mathsf{def\_subclass}(A, B), \mathsf{prec}(\mathsf{c}, g), \mathsf{instd}(a, A, \mathsf{c}, \mathsf{main})\} \subseteq I(\hat{\mathfrak{I}}(\chi))$. As $r \in G_S(PK(\mathfrak{K}))$, we have $\mathsf{test\_fails}(\mathsf{nlit}(a, B, \mathsf{c})) \notin I(\hat{\mathfrak{I}}(\chi))$. By construction of $I(\hat{\mathfrak{I}}(\chi))$ this implies that $\mathsf{unsat}(\mathsf{nlit}(a, B, \mathsf{c})) \in I(\hat{\mathfrak{I}}(\chi))$, meaning that $\mathcal{I}(\mathsf{c}) \models \neg B(a)$. Thus, $\mathcal{I}(\mathsf{c})$ satisfies the clashing set $\{A(a), \neg B(a)\}$ for the clashing assumption $\langle A \sqsubseteq B, a\rangle$. Consequently, $\langle A \sqsubseteq B, a\rangle \in \chi$ and by construction $\mathsf{ovr}(\mathsf{subClass}, a, A, B, \mathsf{c})$ is added to $I(\hat{\mathfrak{I}}(\chi))$.
- **(test-subc):** then $\{\mathsf{def\_subclass}(A, B), \mathsf{prec}(\mathsf{c}, g), \mathsf{instd}(a, A, \mathsf{c}, \mathsf{main})\} \subseteq I(\hat{\mathfrak{I}}(\chi))$. Thus $\mathrm{D}(A \sqsubseteq B) \in \mathfrak{G}$ and $\mathcal{I}(\mathsf{c}) \models A(a)$ (an instance of such defeasible axiom). By the construction of $I(\hat{\mathfrak{I}}(\chi))$ we have that $\mathsf{test}(\mathsf{nlit}(a, B, \mathsf{c})) \in I(\hat{\mathfrak{I}}(\chi))$.
- **(test-fails1):** then $\mathsf{instd}(a, A, \mathsf{c}, \mathsf{nlit}(a, A, \mathsf{c})) \in I(\hat{\mathfrak{I}}(\chi))$. As $r \in G_S(PK(\mathfrak{K}))$, we have that $\mathsf{unsat}(\mathsf{nlit}(a, A, \mathsf{c})) \notin I(\hat{\mathfrak{I}}(\chi))$. By construction of S, $\mathsf{test\_fails}(\mathsf{nlit}(a, A, \mathsf{c})) \in I(\hat{\mathfrak{I}}(\chi))$.
- **(test-add1):** then $\mathsf{test}(\mathsf{nlit}(a, A, \mathsf{c})) \in I(\hat{\mathfrak{I}}(\chi))$. By definition of S, this directly implies that $\mathsf{instd}(a, A, \mathsf{c}, \mathsf{nlit}(a, A, \mathsf{c})) \in I(\hat{\mathfrak{I}}(\chi))$.
- **(test-copy1):** then $\{\mathsf{test}(\mathsf{nlit}(a, A, \mathsf{c})), \mathsf{instd}(b, B, \mathsf{c}, \mathsf{main})\} \subseteq I(\hat{\mathfrak{I}}(\chi))$. By definition of S, this directly implies that $\mathsf{instd}(b, B, \mathsf{c}, \mathsf{nlit}(a, A, \mathsf{c})) \in I(\hat{\mathfrak{I}}(\chi))$.

Minimality of $S = I(\hat{\mathfrak{I}}(\chi))$ w.r.t. the (positive) deduction rules of $G_S(PK(\mathfrak{K}))$ can be motivated as in case of the global least model $\hat{\mathfrak{M}}(\mathfrak{G})$ for $PG(\mathfrak{G})$. Indeed, no model $S' \subseteq S$ of $G_S(PK(\mathfrak{K}))$ such that $S' \neq S$ can exist: as $\hat{\mathfrak{I}}(\chi)$ is the least model of $\mathfrak{K}$ w.r.t. χ, S' can not be a proper subset of S on any of the facts from the input translations, nor on insta, $\mathsf{triplea}$, instd, $\mathsf{tripled}$ for the environment (i.e., last argument) main. Consequently, S' will also contain all atoms on ovr from S, as for every corresponding clashing assumption $\langle \alpha, \mathbf{e}\rangle \in \chi(\mathsf{c})$, the body of the (reduct of) some overriding rule in $PK(\mathfrak{K})$ that encodes a clashing set for $\langle \alpha, \mathbf{e}\rangle$ will be satisfied. Furthermore, S' will also contain all literals over test from S: consequently S' has to contain all literals instd, $\mathsf{tripled}$ for environments different from main and all literals on unsat and $\mathsf{test\_fails}$ from S. That is, $S' = S$ must hold.

(ii). Let S be an answer set of $PK(\mathfrak{K})$. We show that there is some justified clashing assumption χ for $\mathfrak{K}$ such that $S = I(\hat{\mathfrak{I}}(\chi))$ holds.

First of all, we note that as S is an answer set for the CKR program, all literals on ovr and $\mathsf{test\_fails}$ in S are derivable from the reduct $G_S(PK(\mathfrak{K}))$.

By the definition of $I(\hat{\mathfrak{I}}(\chi))$ we can easily build a model $\mathfrak{I}_S = \langle \mathfrak{M}_S, \mathcal{I}_S, CAS_S\rangle$ from the answer set S as follows: the global interpretation $\mathfrak{M}_S$ is analogous to the structure $\mathfrak{M}$ that was defined for the answer set M of $PG(\mathfrak{G})$ in Proposition 13. We note that, by well-known modularity properties of answer set semantics (splitting sets [64]), the restriction of S to the vocabulary of the global part $\mathfrak{G}$, denoted $S|_{\mathfrak{G}}$, is an answer set of $PG(\mathfrak{G})$: thus by Proposition 13, it follows that $\mathfrak{M}_S = \mathfrak{M}_{\mathfrak{G}}$ and that $\mathfrak{M}_S$ is the least Herbrand model of $\mathfrak{G}$. Thus, if $\mathsf{c} \in \mathbf{N}_{\mathfrak{G}}$ (that is, if $grnd(PG(\mathfrak{G})) \models \mathsf{instd}(\mathsf{c}, \mathsf{Ctx}, \mathsf{gm}, \mathsf{main})$) then $\mathsf{c}^{\mathfrak{M}} \in \mathsf{Ctx}^{\mathfrak{M}}$; if $K_m \in K_{\mathsf{c}}$ (that is, if $grnd(PG(\mathfrak{G})) \models \mathsf{tripled}(\mathsf{c}, \mathsf{mod}, \mathsf{m}, \mathsf{gm}, \mathsf{main})$) then $\langle \mathsf{c}^{\mathfrak{M}}, \mathsf{m}^{\mathfrak{M}}\rangle \in \mathsf{mod}^{\mathfrak{M}}$. For every $c \in \mathbf{N}_{\mathfrak{G}}$ (and thus, for every $\mathsf{c}^{\mathfrak{M}} \in \mathsf{Ctx}^{\mathfrak{M}}$), we build the local interpretation $\mathcal{I}_S(c) = \langle \Delta_c, \cdot^{\mathcal{I}(c)}\rangle$ as follows:

- $\Delta_c = \{d \mid d \in \mathrm{NI}_\Sigma\}$;
- $a^{\mathcal{I}(c)} = a$, for every $a \in \mathrm{NI}_\Sigma$;

– $A^{I(c)} = \{d \in \Delta_c \mid S \models \mathsf{instd}(d, A, c, \mathsf{main})\}$, for every $A \in \mathrm{NC}_\Sigma$;

– $R^{I(c)} = \{(d, d') \in \Delta_c \times \Delta_c \mid S \models \mathsf{tripled}(d, R, d', c, \mathsf{main})\}$ for $R \in \mathrm{NR}_\Sigma$;

Finally, $\chi_S(c) = \{(\alpha, \mathbf{e}) \mid I_{rl}(\alpha, c) = p, \mathsf{ovr}(p(\mathbf{e})) \in S\}$. To prove the claim, we have to show that I_S meets the definition of a least justifed CAS-model for K, that is:

(i). $c^{\mathrm{M}} \in \mathrm{Ctx}^{\mathrm{M}}$, for every $c \in \mathbf{N}$, and $C^{\mathrm{M}} \subseteq \mathrm{Ctx}^{\mathrm{M}}$, for every $C \in \mathbf{C}$;

(ii). for every $x \in \mathrm{Ctx}^{\mathrm{M}}$, $\Delta^{I(x)} = \Delta^{\mathrm{M}}$ and $a^{I(x)} = a^{\mathrm{M}}$, for $a \in \mathrm{NI}_\Sigma$;

(iii). for every $\alpha \in \mathrm{L}_\Sigma \cup \mathrm{L}_r$ in G, $\mathrm{M} \models \alpha$;

(iv). for every $\mathrm{D}(\alpha) \in \mathrm{G}$ (where $\alpha \in \mathrm{L}_\Sigma$), $\mathrm{M} \models \alpha$;

(v). for every $(x, y) \in \mathrm{mod}^{\mathrm{M}}$ s.t. $y = \mathrm{m}^{\mathrm{M}}$, then $I(x) \models \mathrm{K_m}$;

(vi). for every $\alpha \in \mathrm{G} \cap \mathbf{L}_\Sigma$ and $x \in \mathrm{Ctx}^{\mathrm{M}}$, $I(x) \models \alpha$, and

(vii). for every $\mathrm{D}(\alpha) \in \mathrm{G}$ (with $\alpha \in \mathrm{L}_\Sigma$), $x \in \mathrm{Ctx}^{\mathrm{M}}$, and $|\mathbf{x}|$-tuple $\mathbf{d}$ of elements in NI_Σ such that $\mathbf{d} \notin \{\mathbf{e} \mid (\alpha, \mathbf{e}) \in \chi(x)\}$, $I(x) \models \alpha(\mathbf{d})$.

Conditions (i), (iii) and (iv) directly follow from Proposition 13. Condition (ii) holds since, given $x \in \mathrm{Ctx}^{\mathrm{M}}$, for every $a \in \mathrm{NI}_\Sigma$ it holds that $a^{I(x)} = a^{\mathrm{M}} = a$. Condition (v) is verified by showing that for every $\mathrm{K_m}$ s.t. $\langle c, \mathrm{m}^{\mathrm{M}} \rangle \in \mathrm{mod}^{\mathrm{M}}$ (that is, every $\mathrm{K_m} \in \mathrm{K}_c$) we have $I(c) \models \mathrm{K_m}$. We proceed by cases and consider the form of all of the axioms $\beta \in \mathrm{L}_\Sigma$ that can occur in K_c. The case for axioms in the general normal form of Table 1 can be proved analogously as in the cases of Proposition 13: thus we have to prove the case of local reference axioms.

– Let $\beta = \mathit{eval}(A, \mathrm{C}) \sqsubseteq B \in \mathrm{K}_c$, then $S \models \mathsf{subEval}(A, \mathrm{C}, B, c)$. If $c' \in \mathrm{C}^{\mathrm{M}}$ and $d \in A^{I(c')}$, then by definition $S \models \mathsf{instd}(d, A, c', \mathsf{main})$ and $S \models \mathsf{instd}(c', \mathrm{C}, \mathsf{gm}, \mathsf{main})$. By rule (plc-evalat) we obtain that $S \models \mathsf{instd}(d, B, c, \mathsf{main})$: hence, by definition $d \in B^{I(c)}$.

– The case for $\beta = \mathit{eval}(R, \mathrm{C}) \sqsubseteq T$ can be shown analogously.

To prove, condition (vii), let us assume that $\mathrm{D}(\beta) \in \mathrm{G}$ with $\beta \in \mathrm{L}_\Sigma$. We proceed by cases on the possible forms of β. In the following we only show some of the relevant cases.

– Let $\beta = A(a)$. Then, by definition of the translation, we have that $S \models \mathsf{insta}(a, A, \mathsf{g}, \mathsf{main})$. Suppose that $\langle A(x), a \rangle \notin \chi_{\mathrm{N}}(c)$, and hence $\langle A(x), a \rangle \notin \chi_S(c^{\mathrm{M}})$. Then by definition, $\mathsf{ovr}(\mathsf{insta}, a, A, c) \notin OVR(\hat{\mathrm{I}}(\chi))$. By the definition of the reduction, the corresponding instantiation of rule (prop-inst) has not been removed from $G_S(PK(\mathrm{K}))$: this implies that $S \models \mathsf{instd}(a, A, c, \mathsf{main})$. By definition, this means that $a^{I(c)} \in A^{I(c)}$.

– Let $\beta = A \sqsubseteq B$. Then, by definition of the translation, we have that $S \models \mathsf{subClass}(A, B, \mathsf{g})$. Let us suppose that $b^{I(c)} \in A^{I(c)}$: then $S \models \mathsf{instd}(b, A, c, \mathsf{main})$. Suppose that $\langle A \sqsubseteq B, b \rangle \notin \chi_S(c)$: by definition, $\mathsf{ovr}(\mathsf{subClass}, b, A, B, c) \notin OVR(\hat{\mathrm{I}}(\chi))$. By the definition of the reduction, the corresponding instantiation of rule (prop-subc) has not been removed from $G_S(PK(\mathrm{K}))$: this implies that $S \models \mathsf{instd}(b, B, c, \mathsf{main})$. Thus, by definition, this means that $b^{I(c)} \in B^{I(c)}$.

– Let $\beta = \exists R.A \sqsubseteq B$. Then $S \models \mathsf{subEx}(R, A, B, \mathsf{g})$. Suppose that $d \in (\exists R.A)^{I(c)}$: by definition of the semantics this means that some $d' \in A^{I(c)}$ exists such that $\langle d, d' \rangle \in R^{I(c)}$. Thus, $S \models \mathsf{instd}(d', A, c, \mathsf{main})$ and $S \models \mathsf{tripled}(d, R, d', c, \mathsf{main})$. Suppose that $\langle \exists R.A \sqsubseteq B, d \rangle \notin \chi_S(c)$: by definition, $\mathsf{ovr}(\mathsf{subEx}, d, R, A, B, c) \notin OVR(\hat{\mathrm{I}}(\chi))$. By the definition of the reduction, the corresponding instantiation of rule (prop-subex) has not been removed from $G_S(PK(\mathrm{K}))$. Thus it holds that $S \models \mathsf{instd}(d, B, c, \mathsf{main})$, and hence $d^{I(c)} \in B^{I(c)}$.

Table A.18: Negative deduction rules P_{nd}

(pnd-instd)	$\neg$instd(x, z, c) $\leftarrow$ $\neg$insta(x, z, c).
(pnd-tripled)	$\neg$tripled(x, r, y, c) $\leftarrow$ $\neg$triplea(x, r, y, c).
(pnd-subc)	$\neg$instd(x, y, c) $\leftarrow$ subClass(y, z, c), $\neg$instd(x, z, c).
(pnd-cnj1)	$\neg$instd(x, y_1, c) $\leftarrow$ subConj(y_1, y_2, z, c), $\neg$instd(x, z, c), instd(x, y_2, c).
(pnd-cnj2)	$\neg$instd(x, y_2, c) $\leftarrow$ subConj(y_1, y_2, z, c), $\neg$instd(x, z, c), instd(x, y_1, c).
(pnd-subex1)	$\neg$instd(x', y, c) $\leftarrow$ subEx(v, y, z, c), $\neg$instd(x, z, c), tripled(x, v, x', c).
(pnd-subex2)	$\neg$tripled(x, v, x', c) $\leftarrow$ subEx(v, y, z, c), $\neg$instd(x, z, c), instd(x', y, c).
(pnd-supex)	$\neg$instd(x, y, c) $\leftarrow$ supEx(y, r, x', c), $\neg$tripled(x, r, x', c).
(pnd-supforall)	$\neg$instd(x, z, c) $\leftarrow$ supForall(z, r, z', c), $\neg$instd(y, z', c), tripled(x, r, y, c).
(pnd-leqone1)	$\neg$tripled(x, r, x_2, c) $\leftarrow$ supLeqOne(z, r, c), tripled(x, r, x_1, c), instd(x, z, c).
(pnd-leqone2)	$\neg$instd(x, z, c) $\leftarrow$ supLeqOne(z, r, c), tripled(x, r, x_1, c), tripled(x, r, x_2, c).
(pnd-subr)	$\neg$tripled(x, v, x', c) $\leftarrow$ subRole(v, w, c), $\neg$tripled(x, w, x', c).
(pnd-subrc1)	$\neg$tripled(x, u, y, c) $\leftarrow$ subRChain(u, v, w, c), $\neg$tripled(x, w, z, c), tripled(y, v, z, c).
(pnd-subrc2)	$\neg$tripled(y, v, z, c) $\leftarrow$ subRChain(u, v, w, c), $\neg$tripled(x, w, z, c), tripled(x, u, y, c).
(pnd-dis1)	$\neg$tripled(x, v, y, c) $\leftarrow$ dis(u, v, c), tripled(x, u, y, c).
(pnd-dis2)	$\neg$tripled(x, u, y, c) $\leftarrow$ dis(u, v, c), tripled(x, v, y, c).
(pnd-inv1)	$\neg$tripled(y, v, x, c) $\leftarrow$ inv(u, v, c), $\neg$tripled(x, u, y, c).
(pnd-inv2)	$\neg$tripled(y, u, x, c) $\leftarrow$ inv(u, v, c), $\neg$tripled(x, v, y, c).

Condition (vi) can be verified similarly as (vii): non-defeasible global axioms $\beta \in \mathcal{L}_\Sigma$ can not appear in clashing assumptions in χ_S, thus they can not give rise to the corresponding overriding assumptions in $OVR(\chi_S)$; this implies that the corresponding instantiations of propagation rules in P_D are never removed from $G_S(PK(\mathrm{K}))$ and thus the cases can be proved like in the proof for the previous condition.

Thus, $\mathbf{I}_S$ is a CAS-model of K. We next argue that in fact $\mathbf{I}_S = \hat{\mathbf{I}}(\chi_S)$, i.e., $\mathbf{I}_S$ is the least CAS-model of K for the clashing assumption χ_S as in Corollary 1. We already noted that w.r.t. the global context, M_S coincides with $\hat{\mathsf{M}}$ from Proposition 13. Assuming that $\mathbf{I} \subset \mathbf{I}_S$ is a CAS-model of K with clashing assumption χ_S, we can construct an interpretation $S' \subset S$ such that $S' \models G_S(PK(\mathrm{K}))$, by removing (at least) one fact insta$(d, A, c,$ main$)$ or triple$(d, R, d', c,$ main$)$ from S; however, this would contradict that S is an answer set of $PK(\mathrm{K})$. Hence, $\mathbf{I}_S = \hat{\mathbf{I}}(\chi_S)$ holds.

Finally, it remains to show that χ_S is justified. As $\langle\alpha, \mathbf{e}\rangle \in \chi_S(c)$ is due to ovr$(p(\mathbf{e})) \in S$ and ovr$(p(\mathbf{e}))$ is derived from the reduct $G_S(PK(\mathrm{K}))$, it follows that S must satisfy the body $Body(r)$ of some overriding rule r for $p(\mathbf{e})$. Consequently, $\mathcal{I}_S(c)$ must satisfy the clashing set $S_{c,\langle\alpha,\mathbf{e}\rangle}$ for $\langle\alpha, \mathbf{e}\rangle$ that is encoded in $Body(r)$; note that satisfaction of not test_fails(nlit$(x, y,$ c$)$) means that test_fails(nlit$(x, y,$ c$)$) is not satisfied, which due to the rule (test-fails1) means that unsat(nlit$(x, y,$ c$)$) is derived. From the latter, however, we conclude that for the negative literal $\neg\beta$ that is encoded by nlit$(x, y,$ c$)$, item (ii) of Theorem 2 holds. As item (i) of this theorem holds for every positive literal $\beta \in S_{c,\langle\alpha,\mathbf{e}\rangle}$, it follows that the clashing assumption $\langle\alpha, \mathbf{e}\rangle$ at c is justified. In conclusion, χ_S is justified. This proves the result. □

Appendix A.6. Justification safeness

Table A.18 shows a set of negative deduction rules (corresponding to the positive rules in P_{rl}) that can be added to the translation under the assumption of justification safeness (see Section 5.4).

References

[1] L. Bozzato, L. Serafini, Materialization Calculus for Contexts in the Semantic Web, in: [66], 2013, pp. 552 – 572.

[2] L. Bozzato, T. Eiter, L. Serafini, Defeasibility in contextual reasoning with CKR, in: 29th Italian Conference on Computational Logic, Torino (CILC 2014), Italy, June 16-18, 2014., volume 1195 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2014, pp. 132–146.

[3] L. Bozzato, T. Eiter, L. Serafini, Contextualized knowledge repositories with justifiable exceptions, in: M. Bienvenu, M. Ortiz, R. Rosati, M. Simkus (Eds.), 27th International Workshop on Description Logics (DL2014), Vienna, Austria, July 17-20, 2014, volume 1193 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2014, pp. 112–123.

[4] J. McCarthy, Notes on formalizing context, in: R. Bajcsy (Ed.), 13th International Joint Conference on Artificial Intelligence (IJCAI'93), Morgan Kaufmann, 1993, pp. 555–562.

[5] D. Lenat, The Dimensions of Context Space, Technical Report, CYCorp, 1998. Published online http://www.cyc.com/doc/context-space.pdf.

[6] F. Giunchiglia, C. Ghidini, Local Models Semantics, or Contextual Reasoning = Locality + Compatibility, Artificial Intelligence 127 (2001) 2001.

[7] F. Giunchiglia, L. Serafini, Multilanguage hierarchical logics, or: how we can do without modal logics, Artificial Intelligence 65 (1994) 29–70.

[8] G. Brewka, F. Roelofsen, L. Serafini, Contextual default reasoning, in: Proceedings of the 20th international joint conference on Artificial intelligence (IJCAI'07), Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2007, pp. 268–273.

[9] G. Brewka, T. Eiter, Equilibria in heterogeneous nonmonotonic multi-context systems, in: Proceedings of the Twenty-Second Conference on Artificial Intelligence (AAAI-07), Vancouver, Canada, 2007, pp. 385–390.

[10] O. Khriyenko, V. Terziyan, A framework for context sensitive metadata description, Int. J. Metadata, Semantics and Ontologies 1 (2006) 154–164.

[11] S. Klarman, Reasoning with Contexts in Description Logics, Ph.D. thesis, Free University of Amsterdam, 2013.

[12] L. Serafini, M. Homola, Contextualized knowledge repositories for the semantic web, J. of Web Semantics 12 (2012) 64–87.

[13] U. Straccia, N. Lopes, G. Lukácsy, A. Polleres, A general framework for representing and reasoning with annotated semantic web data, in: Proceedings of the 24th AAAI Conference on Artificial Intelligence (AAAI 2010), Special Track on Artificial Intelligence and the Web, 2010.

[14] L. Tanca, Context-based data tailoring for mobile users, in: Datenbanksysteme in Business, Technologie und Web (BTW 2007), Workshop Proceedings, Verlagshaus Mainz, 2007, pp. 282–295.

[15] O. Udrea, D. Recupero, V. S. Subrahmanian, Annotated RDF, ACM Transactions on Computational Logic 11 (2010) 1–41.

[16] L. Bozzato, M. Homola, L. Serafini, Towards more effective tableaux reasoning for CKR, in: Y. Kazakov, D. Lembo, F. Wolter (Eds.), Proceedings of the 2012 International Workshop on Description Logics, DL-2012, Rome, Italy, June 7-10, 2012, volume 846 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2012, pp. 114–124.

[17] L. Bozzato, C. Ghidini, L. Serafini, Comparing contextual and flat representations of knowledge: a concrete case about football data, in: V. R. Benjamins, M. d'Aquin, A. Gordon (Eds.), Proceedings of the 7th International Conference on Knowledge Capture (K-CAP 2013), 2013, ACM, 2013, pp. 9–16.

[18] B. Motik, U. Sattler, R. Studer, Query answering for OWL-DL with rules, J. Web Semantics 3 (2005) 41–60.

[19] D. Calvanese, G. D. Giacomo, D. Lembo, M. Lenzerini, R. Rosati, Tractable reasoning and efficient query answering in description logics: The *DL-Lite* family, J. Automated Reasoning 39 (2007) 385–429.

[20] M. Krötzsch, Efficient inferencing for OWL EL, in: [65], 2010, pp. 234–246.

[21] M. Krötzsch, A. Mehdi, S. Rudolph, Orel: Database-driven reasoning for OWL 2 profiles, in: V. Haarslev, D. Toman, G. E. Weddell (Eds.), 23rd International Workshop on Description Logics (DL 2010), volume 573 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2010.

[22] R. Kontchakov, C. Lutz, D. Toman, F. Wolter, M. Zakharyaschev, The Combined Approach to Query Answering in DL-Lite, in: F. Lin, U. Sattler, M. Truszczynski (Eds.), 12th International Conference on Principles of Knowledge Representation and Reasoning (KR 2010), AAAI Press, 2010.

[23] T. Eiter, M. Ortiz, M. Šimkus, K. Trung-Tran, G. Xiao, Query rewriting for Horn-SHIQ plus rules, in: 26th Conference on Artificial Intelligence (AAAI '12), AAAI Press, 2012.

[24] S. Ahmetaj, M. Ortiz, M. Simkus, Polynomial datalog rewritings for expressive description logics with closed predicates, in: S. Kambhampati (Ed.), 25th International Joint Conference on Artificial Intelligence (IJCAI 2016), IJCAI/AAAI Press, 2016, pp. 878–885.

[25] F. Buccafurri, W. Faber, N. Leone, Disjunctive logic programs with inheritance, in: D. D. Schreye (Ed.), Logic Programming: The 1999 International Conference (ICLP 1999), MIT Press, 1999, pp. 79–93.

[26] B. Motik, A. Fokoue, I. Horrocks, Z. Wu, C. Lutz, B. C. Grau, OWL 2 Web Ontology Language Profiles, W3C Recommendation, W3C, 2009. Http://www.w3.org/TR/2009/REC-owl2-profiles-20091027/.

[27] M. Gelfond, V. Lifschitz, Classical Negation in Logic Programs and Disjunctive Databases, New Generation Computing 9 (1991) 365–385.
[28] M. Gelfond, V. Lifschitz, The Stable Model Semantics for Logic Programming, in: Logic Programming: Proceedings Fifth Intl. Conference and Symposium, MIT Press, Cambridge, Mass., 1988, pp. 1070–1080.
[29] A. Bikakis, G. Antoniou, Defeasible contextual reasoning with arguments in ambient intelligence, IEEE Transactions on Knowledge and Data Engineering 22 (2010) 1492–1506.
[30] L. Giordano, V. Gliozzi, N. Olivetti, G. L. Pozzato, A non-monotonic description logic for reasoning about typicality, Artificial Intelligence 195 (2013) 165–202.
[31] P. A. Bonatti, M. Faella, I. Petrova, L. Sauro, A new semantics for overriding in description logics, Artificial Intelligence 222 (2015) 1–48.
[32] I. Horrocks, O. Kutz, U. Sattler, The even more irresistible SROIQ, in: Procs. of the 10th International Conference on Principles of Knowledge Representation and Reasoning (KR 2006), AAAI Press, 2006, pp. 57–67.
[33] F. Baader, D. Calvanese, D. McGuinness, D. Nardi, P. Patel-Schneider (Eds.), The Description Logic Handbook, Cambridge University Press, 2003.
[34] W3C, OWL 2 Web Ontology Language Document Overview, W3C Recommendation, 2009.
[35] B. Motik, R. Rosati, Reconciling description logics and rules, J. ACM 57 (2010).
[36] T. Eiter, G. Ianni, T. Lukasiewicz, R. Schindlauer, H. Tompits, Combining answer set programming with description logics for the semantic web, Artif. Intell. 172 (2008) 1495–1539.
[37] S. Abiteboul, R. Hull, V. Vianu, Foundations of Databases, Addison-Wesley, 1995. URL: http://webdam.inria.fr/Alice/.
[38] A. K. Chandra, P. M. Merlin, Optimal Implementation of Conjunctive Queries in Relational Data Bases, in: Conference Record of the 9th Annual ACM Symposium on Theory of Computing (STOC 1977), 1977, pp. 77–90.
[39] F. Buccafurri, W. Faber, N. Leone, Disjunctive logic programs with inheritance, Theory and Practice of Logic Programming (TPLP) 2 (2002) 293–321.
[40] G. Xiao, S. Heymans, T. Eiter, DReW: a reasoner for datalog-rewritable description logics and dl-programs, in: 1st International Workshop on Business Models, Business Rules and Ontologies (BuRO 2010), 2010.
[41] F. Ricca, The DLV java wrapper, in: F. Buccafurri (Ed.), 2003 Joint Conference on Declarative Programming, (AGP-2003), 2003, pp. 263–274.
[42] L. Bozzato, L. Serafini, Knowledge propagation in contextualized knowledge repositories: An experimental evaluation, in: Knowledge Engineering and Knowledge Management (EKAW 2014) Satellite Events. Revised Selected Papers, volume 8982 of *Lecture Notes in Computer Science*, Springer, 2015, pp. 35–51.
[43] G. Casini, T. Meyer, K. Moodley, I. J. Varzinczak, Towards Practical Defeasible Reasoning for Description Logics, in: [66], 2013, pp. 587–599.
[44] M. Fink, L. Ghionna, A. Weinzierl, Relational information exchange and aggregation in multi-context systems, in: J. P. Delgrande, W. Faber (Eds.), Logic Programming and Nonmonotonic Reasoning - 11th International Conference (LPNMR 2011), volume 6645 of *Lecture Notes in Computer Science*, Springer, 2011, pp. 120–133.
[45] D. Nute, Defeasible Logic, in: D. Gabbay, C. Hogger, J. Robinson (Eds.), Handbook of Logic in Artificial Intelligence and Logic Programming, volume III, Clarendon Press, Oxford, 1994, pp. 353–395.
[46] A. Bikakis, G. Antoniou, Local and distributed defeasible reasoning in multi-context systems, in: Rule Representation, Interchange and Reasoning on the Web, International Symposium (RuleML'08), 2008, pp. 135–149.
[47] F. Baader, B. Hollunder, Embedding defaults into terminological knowledge representation formalisms, Journal of Automated Reasoning 14 (1995) 149–180.
[48] K. Britz, I. J. Varzinczak, Introducing role defeasibility in description logics, in: L. Michael, A. C. Kakas (Eds.), Logics in Artificial Intelligence - 15th European Conference (JELIA 2016), volume 10021 of *Lecture Notes in Computer Science*, 2016, pp. 174–189.
[49] K. Britz, T. Meyer, I. J. Varzinczak, Semantic foundation for preferential description logics, in: D. Wang, M. Reynolds (Eds.), 24th Australasian Joint Conference on Artificial Intelligence (AI 2011), volume 7106 of *Lecture Notes in Computer Science*, Springer, 2011, pp. 491–500.
[50] L. Giordano, V. Gliozzi, N. Olivetti, G. L. Pozzato, Semantic characterization of rational closure: From propositional logic to description logics, Artificial Intelligence 226 (2015) 1–33.
[51] P. A. Bonatti, C. Lutz, F. Wolter, Description logics with circumscription, in: 10th International Conference on Principles of Knowledge Representation and Reasoning (KR 2006), AAAI Press, 2006, pp. 400–410.
[52] P. A. Bonatti, M. Faella, L. Sauro, Defeasible Inclusions in Low-Complexity DLs, J. Artificial Intelligence Research 42 (2011) 719–764.
[53] G. Governatori, Defeasible description logics, in: G. Antoniou, H. Boley (Eds.), 3rd International Workshop on Rules and Rule Markup Languages for the Semantic Web (RuleML 2004), volume 3323 of *Lecture Notes in Computer Science*, Springer, 2004, pp. 98–112.
[54] L. Giordano, V. Gliozzi, N. Olivetti, G. L. Pozzato, Reasoning about Typicality in Low Complexity DLs: The Logics $EL^{\perp}T_{min}$ and DL-Litec T_{min}, in: T. Walsh (Ed.), 22nd International Joint Conference on Artificial Intelligence

(IJCAI 2011), IJCAI/AAAI, 2011, pp. 894–899.
[55] G. Casini, U. Straccia, Rational closure for defeasible description logics, in: [65], 2010, pp. 77–90.
[56] G. Casini, T. Meyer, I. J. Varzinczak, K. Moodley, Nonmonotonic Reasoning in Description Logics: Rational Closure for the ABox, in: [66], 2013, pp. 600–615.
[57] L. Giordano, D. T. Dupré, Reasoning in a rational extension of SROEL, in: M. Lenzerini, R. Peñaloza (Eds.), 29th International Workshop on Description Logics (DL2016), volume 1577 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2016.
[58] J. McCarthy, Circumscription – a form of non-monotonic reasoning, Artificial Intelligence 13 (1980) 27–39.
[59] M. Cadoli, L. Palopoli, Circumscribing DATALOG: expressive power and complexity, Theoretical Computer Science 193 (1998) 215–244.
[60] S. Heymans, T. Eiter, G. Xiao, Tractable Reasoning with DL-Programs over Datalog-rewritable Description Logics, in: 9th European Conference on Artificial Intelligence (ECAI 2010), 2010, pp. 35–40.
[61] B. Glimm, Y. Kazakov, T. Tran, Scalable Reasoning by Abstraction Beyond DL-Lite, in: M. Ortiz, S. Schlobach (Eds.), 10th International Conference on Web Reasoning and Rule Systems (RR 2016), volume 9898 of *Lecture Notes in Computer Science*, Springer, 2016, pp. 77–93.
[62] I. Stewart, Complete Problems Involving Boolean Labelled Structures and Projection Transactions, Journal of Logic and Computation 1 (1991) 861–882.
[63] G. Gottlob, N. Leone, F. Scarcello, Hypertree decompositions: A survey, in: J. Sgall, A. Pultr, P. Kolman (Eds.), Mathematical Foundations of Computer Science 2001, 26th International Symposium, MFCS 2001 Marianske Lazne, Czech Republic, August 27-31, 2001, Proceedings, volume 2136 of *Lecture Notes in Computer Science*, Springer, 2001, pp. 37–57. URL: https://doi.org/10.1007/3-540-44683-4_5. doi:10.1007/3-540-44683-4_5.
[64] V. Lifschitz, H. Turner, Splitting a logic program, in: P. V. Hentenryck (Ed.), 11th International Conference on Logic Programming (ICLP 1994), MIT Press, 1994, pp. 23–37.
[65] T. Janhunen, I. Niemela¨ (Eds.), Logics in Artificial Intelligence - 12th European Conference (JELIA 2010), Helsinki, Finland, September 13-15, 2010. Proceedings, volume 6341 of *Lecture Notes in Computer Science*, Springer, 2010.
[66] T. Eiter, B. Glimm, Y. Kazakov, M. Krötzsch (Eds.), Informal Proceedings of the 26th International Workshop on Description Logics, Ulm, Germany, July 23 - 26, 2013, volume 1014 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2013.