Towards defeasible $\mathcal{SROIQ}$

Katarina Britz[1] and Ivan Varzinczak[2,1]

[1] CSIR-SU CAIR, Stellenbosch University, South Africa
[2] CRIL, Univ. Artois & CNRS, France
abritz@sun.ac.za, varzinczak@cril.fr

Abstract. We present a decidable extension of the Description Logic $\mathcal{SROIQ}$ that supports defeasible reasoning in the KLM tradition, and extends it through the introduction of defeasible roles. The semantics of the resulting DL $d\mathcal{SROIQ}$ extends the classical semantics with a parameterised preference order on binary relations in a domain of interpretation. This allows for the use of defeasible roles in complex concepts, as well as in defeasible concept and role subsumption, and in defeasible role assertions. Reasoning over $d\mathcal{SROIQ}$ ontologies is made possible by a translation of entailment to concept satisfiability relative to an RBox only. A tableau algorithm then decides on consistency of $d\mathcal{SROIQ}$-concepts in the preferential semantics.

Keywords: $\mathcal{SROIQ}$, non-monotonic reasoning, preferential semantics

1 Introduction

$\mathcal{SROIQ}$ [20] is an expressive, yet decidable Description Logic (DL) that serves as semantic foundation for the OWL 2 profile, on which several ontology languages of various expressivity are based. However, $\mathcal{SROIQ}$ still allows for meaningful, decidable extension, as new knowledge representation requirements are identified. A case in point is the need to allow for exceptions and defeasibility in reasoning over logic-based ontologies [4, 3, 2, 8, 6, 7, 10, 12–14, 17, 18, 27]. Yet, $\mathcal{SROIQ}$ does not allow for the direct expression of and reasoning with different aspects of defeasibility.

Given the special status of subsumption in DLs in particular, and the historical importance of entailment in logic in general, past research efforts in this direction have focused primarily on accounts of defeasible subsumption and the characterisation of defeasible entailment. Semantically, the latter usually take as point of departure orderings on a class of first-order interpretations, whereas the former usually assume a preference order on objects of the domain.

In this paper, we propose a decidable extension of $\mathcal{SROIQ}$ that supports defeasible knowledge representation and reasoning over defeasible ontologies. Our proposal builds on previous work to resolve two important ontological limitations of the preferential approach to defeasible reasoning in DLs — the assumption of a single preference order on all objects in the domain of interpretation, and the assumption that defeasibility is intrinsically linked to argument form [9, 10].

We achieve this by extending $\mathcal{SROIQ}$ with nonmonotonic reasoning features in the concept language, in subsumption statements and in role assertions via an intuitive notion of normality for roles. This parameterises the idea of preference while at the same time introducing the notion of defeasible class membership. Defeasible subsumption allows for the expression of statements of the form "C is usually subsumed by D", for example, "Chenin blanc wines *are usually* unwooded". In our extended language $d\mathcal{SROIQ}$, we can now also refer directly to, for example, "Chenin blanc wines that *usually have* a wood aroma". We can also combine these seamlessly, as in: "Chenin blanc wines that *usually have* a wood aroma *are usually* wooded". Note that this cannot be expressed in terms of defeasible subsumption alone, nor can it be expressed w.l.o.g. using a typicality operator on concepts. This is because the semantics of the expression is inextricably tied to the two distinct uses of the term 'usually'. Another defeasible construct that adds to the expressivity of $d\mathcal{SROIQ}$ is defeasible role inclusion, e.g. "having a given geographic style usually implies having that region as origin". $d\mathcal{SROIQ}$ also includes defeasible role assertions, such as defeasible functionality or defeasible disjointness, and defeasible number- and Self-restrictions.

The remainder of the paper is structured as follows: In Section 2 we introduce the syntax and semantics of the extended language $d\mathcal{SROIQ}$. Section 3 covers a number of rewriting and elimination results required for effective reasoning with $d\mathcal{SROIQ}$ knowledge bases, and which are needed for the tableau algorithm presented in Section 4. The main results of the paper are Theorem 1, which reduces concept satisfiability in $d\mathcal{SROIQ}$ to concept satisfiability relative to only an RBox, and Theorem 2, which establishes the correctness of the tableau procedure. The latter result is established only for the restriction of $d\mathcal{SROIQ}$ which excludes role composition in defeasible RIAs.

Space considerations prevent us from providing a summary of the required logical background. We shall therefore assume the reader's familiarity with DLs in general [1] and with $\mathcal{SROIQ}$ in particular [20], as well as with the preferential approach to non-monotonic reasoning [23, 24, 28]. Whenever necessary, we refer the reader to the definitions and results in the relevant literature.

2 Defeasible $\mathcal{SROIQ}$

2.1 Defeasibility in RBoxes

Let R be a set of *role names*, and let u denote the *universal role*. The set of all roles is given by $\mathbf{R} := \mathsf{R} \cup \{r^- \mid r \in \mathsf{R}\} \cup \{\mathsf{u}\}$. We denote roles with $r, s, \ldots$, possibly with subscripts. Moreover, let $\mathsf{inv} : \mathbf{R} \longrightarrow \mathbf{R}$ be such that $\mathsf{inv} : r \mapsto r^-$, if $r \in \mathsf{R}$, $\mathsf{inv} : r \mapsto s$, if $r = s^-$, and $\mathsf{inv} : \mathsf{u} \mapsto \mathsf{u}$.

Let $r_1, \ldots, r_n, r \in \mathbf{R} \setminus \{\mathsf{u}\}$. A *classical role inclusion axiom* is a statement of the form $r_1 \circ \cdots \circ r_n \sqsubseteq r$. A *defeasible role inclusion axiom* has the form $r_1 \circ \cdots \circ r_n \sqsubseteq\!\!\!\!\sim r$, read "usually, $r_1 \circ \cdots \circ r_n$ is included in r". A finite set of role inclusion axioms (RIAs) is called a *role hierarchy* and is denoted by $\mathcal{R}_h$.

Definition 1 ((Non-)Simple Role). *Let $r \in \mathbf{R}$ and let $\mathcal{R}_h$ be a role hierarchy. Then r is* ***non-simple*** *in $\mathcal{R}_h$ iff:*

1. *There is $r_1 \circ \cdots \circ r_n \sqsubseteq r$ or $r_1 \circ \cdots \circ r_n \sqsubseteq\!\!\!\!\sim r$ in $\mathcal{R}_h$ such that $n > 1$, or*
2. *There is $s \sqsubseteq r$ or $s \sqsubseteq\!\!\!\!\sim r$ in $\mathcal{R}_h$ such that s is non-simple, or*

3. $\mathsf{inv}(r)$ *is non-simple.*

With $\mathbf{R}^n$ *we denote the set of* ***non-simple*** *roles in* $\mathcal{R}_h$. $\mathbf{R}^s := \mathbf{R} \setminus \mathbf{R}^n$ *is the set of* ***simple*** *roles in* $\mathcal{R}_h$.

Intuitively, simple roles are those that are not implied by the composition of roles. They are needed to restrict the type of roles in certain concept constructors (see below), thereby preserving decidability [20].

Definition 2 (Regular Hierarchy). *A role hierarchy* $\mathcal{R}_h$ *is* ***regular*** *if there is a strict partial order* $<$ *on* $\mathbf{R}^n$ *such that:*

1. $s < r$ *iff* $\mathsf{inv}(s) < r$*, for every* r, s *in* $\mathbf{R}^n$*, and*
2. *every role inclusion in* $\mathcal{R}_h$ *is of one of the forms: (1a)* $r \circ r \sqsubseteq r$*, (1b)* $r \circ r \precsim r$*, (2a)* $\mathsf{inv}(r) \sqsubseteq r$*, (2b)* $\mathsf{inv}(r) \precsim r$*, (3a)* $s_1 \circ \cdots \circ s_n \sqsubseteq r$*, (3b)* $s_1 \circ \cdots \circ s_n \precsim r$*, (4a)* $r \circ s_1 \circ \cdots \circ s_n \sqsubseteq r$*, (4b)* $r \circ s_1 \circ \cdots \circ s_n \precsim r$*, (5a)* $s_1 \circ \cdots \circ s_n \circ r \sqsubseteq r$*, (5b)* $s_1 \circ \cdots \circ s_n \circ r \precsim r$*, where* $r \in \mathsf{R}$ *(i.e., a role name), and* $s_i < r$*, for* $i = 1, \ldots, n$.

(Regularity prevents a role hierarchy from inducing cyclic dependencies, which are known to lead to undecidability.)

A *classical role assertion* is a statement of the form $\mathsf{Fun}(r)$ (functionality), $\mathsf{Ref}(r)$ (reflexivity), $\mathsf{Irr}(r)$ (irreflexivity), $\mathsf{Sym}(r)$ (symmetry), $\mathsf{Asy}(r)$ (asymmetry), $\mathsf{Tra}(r)$ (transitivity), and $\mathsf{Dis}(r, s)$ (role disjointness), where $r, s \neq \mathsf{u}$. A *defeasible role assertion* is a statement of the form $\mathsf{dFun}(r)$ (r is usually functional), $\mathsf{dRef}(r)$ (r is usually reflexive), $\mathsf{dIrr}(r)$ (r is usually irreflexive), $\mathsf{dSym}(r)$ (r is usually symmetric), $\mathsf{dAsy}(r)$ (r is usually asymmetric), $\mathsf{dTra}(r)$ (r is usually transitive), and $\mathsf{dDis}(r, s)$ (r and s are usually disjoint), also with $r, s \neq \mathsf{u}$. With $\mathcal{R}_a$ we denote a finite set of role assertions.

Given a role hierarchy $\mathcal{R}_h$, we say that $\mathcal{R}_a$ is *simple* w.r.t. $\mathcal{R}_h$ if all roles r, s appearing in statements of the form $\mathsf{Irr}(r)$, $\mathsf{dIrr}(r)$, $\mathsf{Asy}(r)$, $\mathsf{dAsy}(r)$, $\mathsf{Dis}(r, s)$ or $\mathsf{dDis}(r, s)$ are simple in $\mathcal{R}_h$ (see Definition 1).

A $d\mathcal{SROIQ}$ RBox is a set $\mathcal{R} := \mathcal{R}_h \cup \mathcal{R}_a$, where $\mathcal{R}_h$ is a regular hierarchy and $\mathcal{R}_a$ is a set of role assertions which is simple w.r.t. $\mathcal{R}_h$.

2.2 Defeasibility in Concepts and in TBoxes

Let C be a set of (atomic) *concept names* disjoint from R and of which N, the set of *nominals*, is a subset. We use $A, B, \ldots$, possibly with subscripts, to denote concept names. A nominal will also be denoted by o, possibly with subscripts.

Definition 3 ($d\mathcal{SROIQ}$ Concepts). *The set of $d\mathcal{SROIQ}$* ***complex concepts*** *is the smallest set such that* $\top$*,* $\bot$ *and every* $A \in \mathsf{C}$ *are concepts, and if* C *and* D *are concepts,* $r \in \mathbf{R}$*,* $s \in \mathbf{R}^s$*, and* $n \in \mathbb{N}$*, then* $\neg C$ *(concept complement),* $C \sqcap D$ *(concept conjunction),* $C \sqcup D$ *(concept disjunction),* $\forall r.C$ *(value restriction),* $\exists r.C$ *(existential restriction),* $\mathbin{\rotatebox[origin=c]{180}{$\sim$}}\!\!\forall r.C$ *(****defeasible value restriction****),* $\mathbin{\sim}\!\!\exists r.C$ *(****defeasible existential restriction****),* $\exists r.\mathsf{Self}$ *(self restriction),* $\mathbin{\sim}\!\!\exists r.\mathsf{Self}$ *(****defeasible self restriction****),* $\geq ns.C$ *(at-least restriction),* $\leq ns.C$ *(at-most restriction),* $\gtrsim ns.C$ *(****defeasible at-least restriction****),* $\lesssim ns.C$ *(****defeasible at-most restriction****) are also concepts. With* $\mathbf{C}$ *we denote the set of all complex concepts.*

Note that every $\mathcal{SROIQ}$ concept is a $d\mathcal{SROIQ}$ concept, too. We shall use $C, D \ldots$, possibly with subscripts, to denote complex $d\mathcal{SROIQ}$ concepts.

Given $C, D \in \mathbf{C}$, $C \sqsubseteq D$ is a *classical general concept inclusion*, read "C is subsumed by D". ($C \equiv D$ is an abbreviation for both $C \sqsubseteq D$ and $D \sqsubseteq C$.) $C \precsim D$ is a *defeasible general concept inclusion*, read "C is *usually* subsumed by D". A *$d\mathcal{SROIQ}$ TBox* $\mathcal{T}$ is a finite set of general concept inclusions (GCIs), whether classical or defeasible.

Let I be a set of *individual names* disjoint from both C and R. Given $C \in \mathbf{C}$, $r \in \mathbf{R}$ and $a, b \in \mathsf{I}$, an *individual assertion* is an expression of the form $a : C$, $(a, b) : r$, $(a, b) : \neg r$, $a = b$ or $a \neq b$. A *$d\mathcal{SROIQ}$ ABox* $\mathcal{A}$ is a finite set of individual assertions.

Let $\mathcal{A}$ be an ABox, $\mathcal{T}$ be a TBox and $\mathcal{R}$ an RBox. A *knowledge base* (alias ontology) is a tuple $\mathcal{KB} := \langle \mathcal{A}, \mathcal{R}, \mathcal{T} \rangle$.

2.3 Preferential Semantics

We shall anchor our semantic constructions in the well-known preferential approach to non-monotonic reasoning [23, 24, 28] and its extensions [5, 9, 11], especially those to the DL case [8, 16, 25].

Let X be a set and let $<$ be a strict partial order on X. With $\min_< X := \{x \in X \mid$ there is no $y \in X$ s.t. $y < x\}$ we denote the *minimal elements* of X w.r.t. $<$. With $\#X$ we shall denote the *cardinality* of X.

Definition 4 (Ordered Interpretation). *An **ordered interpretation** is a tuple $\mathcal{O} := \langle \Delta^{\mathcal{O}}, \cdot^{\mathcal{O}}, \prec^{\mathcal{O}}, \ll^{\mathcal{O}} \rangle$ in which $\langle \Delta^{\mathcal{O}}, \cdot^{\mathcal{O}} \rangle$ is a $\mathcal{SROIQ}$ interpretation with $A^{\mathcal{O}} \subseteq \Delta^{\mathcal{O}}$, for every $A \in \mathsf{C}$, $A^{\mathcal{O}}$ a singleton for every $A \in \mathsf{N}$, $r^{\mathcal{O}} \subseteq \Delta^{\mathcal{O}} \times \Delta^{\mathcal{O}}$, for all $r \in \mathbf{R}$, and $a^{\mathcal{O}} \in \Delta^{\mathcal{O}}$, for every $a \in \mathsf{I}$, $\prec^{\mathcal{O}}$ is a strict partial order on $\Delta^{\mathcal{O}}$, and $\ll^{\mathcal{O}} := \langle \ll_1^{\mathcal{O}}, \ldots, \ll_{\#\mathsf{R}}^{\mathcal{O}} \rangle$, where $\ll_i^{\mathcal{O}} \subseteq r_i^{\mathcal{O}} \times r_i^{\mathcal{O}}$, for $i = 1, \ldots, \#\mathsf{R}$, and such that $\prec^{\mathcal{O}}$ and each $\ll_i^{\mathcal{O}}$ satisfy the smoothness condition [23]. Moreover, for any $r, r_1, r_2 \in \mathbf{R} \setminus \{\mathsf{u}\}$, $\mathcal{O}$ interprets **orderings on role inverses** and **on role compositions** as follows:*
$\ll_{r^-}^{\mathcal{O}} := \{((y_1, x_1), (y_2, x_2)) \mid ((x_1, y_1), (x_2, y_2)) \in \ll_r^{\mathcal{O}}\}$, and $\ll_{r_1 \circ r_2}^{\mathcal{O}} := \{((x_1, y_1), (x_2, y_2)) \mid$ for some z_1, z_2 $[((x_1, z_1), (x_2, z_2)) \in \ll_{r_1}^{\mathcal{O}}$ and $((z_1, y_1), (z_2, y_2)) \in \ll_{r_2}^{\mathcal{O}}]$, and for no z_1, z_2 $[((x_2, z_2), (x_1, z_1)) \in \ll_{r_1}^{\mathcal{O}}$ and $((z_2, y_2), (z_1, y_1)) \in \ll_{r_2}^{\mathcal{O}}]\}$.
Let $r_i^{\mathcal{O}|x} := r_i^{\mathcal{O}} \cap (\{x\} \times \Delta^{\mathcal{O}})$ (i.e., the restriction of the domain of $r_i^{\mathcal{O}}$ to $\{x\}$). The interpretation function $\cdot^{\mathcal{O}}$ interprets $d\mathcal{SROIQ}$ concepts in the following way (whenever it is clear which component of $\ll^{\mathcal{O}}$ is used, we shall drop the subscript in $\ll_i^{\mathcal{O}}$):

$\top^{\mathcal{O}} := \Delta^{\mathcal{O}}$; $\quad \bot^{\mathcal{O}} := \emptyset$; $\quad (\neg C)^{\mathcal{O}} := \Delta^{\mathcal{O}} \setminus C^{\mathcal{O}}$;
$(C \sqcap D)^{\mathcal{O}} := C^{\mathcal{O}} \cap D^{\mathcal{O}}$; $\quad (C \sqcup D)^{\mathcal{O}} := C^{\mathcal{O}} \cup D^{\mathcal{O}}$;
$(\forall r.C)^{\mathcal{O}} := \{x \mid r^{\mathcal{O}}(x) \subseteq C^{\mathcal{O}}\}$; $(\forall\!\!\!\!\sim r.C)^{\mathcal{O}} := \{x \mid \min_{\ll^{\mathcal{O}}}(r^{\mathcal{O}|x})(x) \subseteq C^{\mathcal{O}}\}$;
$(\exists r.C)^{\mathcal{O}} := \{x \mid r^{\mathcal{O}}(x) \cap C^{\mathcal{O}} \neq \emptyset\}$; $(\exists\!\!\!\!\sim r.C)^{\mathcal{O}} := \{x \mid \min_{\ll^{\mathcal{O}}}(r^{\mathcal{O}|x})(x) \cap C^{\mathcal{O}} \neq \emptyset\}$;
$(\exists r.\mathsf{Self})^{\mathcal{O}} := \{x \mid (x, x) \in r^{\mathcal{O}}\}$; $(\exists\!\!\!\!\sim r.\mathsf{Self})^{\mathcal{O}} := \{x \mid (x, x) \in \min_{\ll^{\mathcal{O}}}(r^{\mathcal{O}|x})\}$;
$(\geq nr.C)^{\mathcal{O}} := \{x \mid \# r^{\mathcal{O}}(x) \cap C^{\mathcal{O}} \geq n\}$; $(\leq nr.C)^{\mathcal{O}} := \{x \mid \# r^{\mathcal{O}}(x) \cap C^{\mathcal{O}} \leq n\}$;
$(\gtrsim nr.C)^{\mathcal{O}} := \{x \mid \# \min_{\ll^{\mathcal{O}}}(r^{\mathcal{O}|x})(x) \cap C^{\mathcal{O}}\} \geq n$;
$(\lesssim nr.C)^{\mathcal{O}} := \{x \mid \# \min_{\ll^{\mathcal{O}}}(r^{\mathcal{O}|x})(x) \cap C^{\mathcal{O}} \leq n\}$.

It is not hard to see that, analogously to the classical case, $\forall\!\!\!\forall$ and $\exists\!\!\!\exists$, as well as $\gtrsim$ and $\lesssim$, are duals to each other.

Definition 5 (Satisfaction). *Let $\mathcal{O} = \langle \Delta^{\mathcal{O}}, \cdot^{\mathcal{O}}, \prec^{\mathcal{O}}, \ll^{\mathcal{O}} \rangle$ and let $r_1, \ldots, r_n, r, s \in \mathbf{R}$, $C, D \in \mathbf{C}$, and $a, b \in \mathsf{I}$. The **satisfaction relation** $\Vdash$ is defined as follows:*
$\mathcal{O} \Vdash r \sqsubseteq s$ *if* $r^{\mathcal{O}} \subseteq s^{\mathcal{O}}$; $\mathcal{O} \Vdash r \sqsubsim s$ *if* $\min_{\ll^{\mathcal{O}}} r^{\mathcal{O}} \subseteq s^{\mathcal{O}}$;
$\mathcal{O} \Vdash r_1 \circ \cdots \circ r_n \sqsubseteq r$ *if* $(r_1 \circ \cdots \circ r_n)^{\mathcal{O}} \subseteq r^{\mathcal{O}}$; $\mathcal{O} \Vdash r_1 \circ \cdots \circ r_n \sqsubsim r$ *if* $\min_{\ll^{\mathcal{O}}}(r_1 \circ \cdots \circ r_n)^{\mathcal{O}} \subseteq r^{\mathcal{O}}$;
$\mathcal{O} \Vdash \mathsf{Fun}(r)$ *if* $r^{\mathcal{O}}$ *is a function;* $\mathcal{O} \Vdash \mathsf{dFun}(r)$ *if for all* x, $\#\min_{\ll^{\mathcal{O}}}(r^{\mathcal{O}|x})(x) \leq 1$;
$\mathcal{O} \Vdash \mathsf{Ref}(r)$ *if* $\{(x,x) \mid x \in \Delta^{\mathcal{O}}\} \subseteq r^{\mathcal{O}}$; $\mathcal{O} \Vdash \mathsf{dRef}(r)$ *if for every* $x \in \min_{\prec^{\mathcal{O}}} \Delta^{\mathcal{O}}$, $(x,x) \in r^{\mathcal{O}}$;
$\mathcal{O} \Vdash \mathsf{Irr}(r)$ *if* $r^{\mathcal{O}} \cap \{(x,x) \mid x \in \Delta^{\mathcal{O}}\} = \emptyset$; $\mathcal{O} \Vdash \mathsf{dIrr}(r)$ *if for every* $x \in \min_{\prec^{\mathcal{O}}} \Delta^{\mathcal{O}}$, $(x,x) \notin r^{\mathcal{O}}$;
$\mathcal{O} \Vdash \mathsf{Sym}(r)$ *if* $\mathsf{inv}(r)^{\mathcal{O}} \subseteq r^{\mathcal{O}}$; $\mathcal{O} \Vdash \mathsf{dSym}(r)$ *if* $\min_{\ll^{\mathcal{O}}}(r^{-})^{\mathcal{O}} \subseteq r^{\mathcal{O}}$;
$\mathcal{O} \Vdash \mathsf{Asy}(r)$ *if* $r^{\mathcal{O}} \cap \mathsf{inv}(r)^{\mathcal{O}} = \emptyset$; $\mathcal{O} \Vdash \mathsf{dAsy}(r)$ *if* $\min_{\ll^{\mathcal{O}}} r^{\mathcal{O}} \cap \min_{\ll^{\mathcal{O}}}(r^{-})^{\mathcal{O}} = \emptyset$;
$\mathcal{O} \Vdash \mathsf{Tra}(r)$ *if* $(r \circ r)^{\mathcal{O}} \subseteq r^{\mathcal{O}}$; $\mathcal{O} \Vdash \mathsf{dTra}(r)$ *if* $\min_{\ll^{\mathcal{O}}}(r \circ r)^{\mathcal{O}} \subseteq r^{\mathcal{O}}$;
$\mathcal{O} \Vdash \mathsf{Dis}(r,s)$ *if* $r^{\mathcal{O}} \cap s^{\mathcal{O}} = \emptyset$; $\mathcal{O} \Vdash \mathsf{dDis}(r,s)$ *if* $\min_{\ll^{\mathcal{O}}} r^{\mathcal{O}} \cap \min_{\ll^{\mathcal{O}}} s^{\mathcal{O}} = \emptyset$;
$\mathcal{O} \Vdash C \sqsubseteq D$ *if* $C^{\mathcal{O}} \subseteq D^{\mathcal{O}}$; $\mathcal{O} \Vdash C \sqsubsim D$ *if* $\min_{\prec^{\mathcal{O}}} C^{\mathcal{O}} \subseteq D^{\mathcal{O}}$;
$\mathcal{O} \Vdash a : C$ *if* $a^{\mathcal{O}} \in C^{\mathcal{O}}$; $\mathcal{O} \Vdash (a,b) : r$ *if* $(a^{\mathcal{O}}, b^{\mathcal{O}}) \in r^{\mathcal{O}}$; $\mathcal{O} \Vdash (a,b) : \neg r$ *if* $\mathcal{O} \not\Vdash (a,b) : r$; $\mathcal{O} \Vdash a = b$ *if* $a^{\mathcal{O}} = b^{\mathcal{O}}$; $\mathcal{O} \Vdash a \neq b$ *if* $\mathcal{O} \not\Vdash a = b$.

*If $\mathcal{O} \Vdash \alpha$, then we say $\mathcal{O}$ **satisfies** α. $\mathcal{O}$ satisfies a set of statements or assertions X (denoted $\mathcal{O} \Vdash X$) if $\mathcal{O} \Vdash \alpha$ for every $\alpha \in X$, in which case we say $\mathcal{O}$ is a **model** of X. We say $C \in \mathbf{C}$ is **satisfiable** w.r.t. $\mathcal{KB} = \langle \mathcal{A}, \mathcal{R}, \mathcal{T} \rangle$ if there is a model $\mathcal{O}$ of $\mathcal{KB}$ s.t. $C^{\mathcal{O}} \neq \emptyset$, and **unsatisfiable** otherwise.*

A statement α is (classically) *entailed* by a knowledge base $\mathcal{KB}$, denoted $\mathcal{KB} \models \alpha$, if every model of $\mathcal{KB}$ satisfies α.

3 Reasoning with $d\mathcal{SROIQ}$ Knowledge Bases

As for classical $\mathcal{SROIQ}$ [20, Lemma 7], it is possible to eliminate an ABox $\mathcal{A}$ by compiling all individual assertions in $\mathcal{A}$ as follows:

1. Let $\mathsf{N}' := \mathsf{N} \cup \{o_a \mid a \text{ appears in } \mathcal{A}\}$ (i.e., extend the signature with new nominals);
2. Let $\mathcal{A}' := \{a : C \in \mathcal{A}\} \cup \{a : \exists r.o_b \mid (a,b) : r \in \mathcal{A}\} \cup \{a : \forall r.\neg o_b \mid (a,b) : \neg r \in \mathcal{A}\} \cup \{a : \neg o_b \mid a \neq b \in \mathcal{A}\}$;
3. For every $C \in \mathbf{C}$, let $C' := C \sqcap \bigsqcap_{a:D \in \mathcal{A}'} \exists \mathsf{u}.(o_a \sqcap D)$.

It is then easy to see that C is satisfiable w.r.t. $\langle \mathcal{A}, \mathcal{R}, \mathcal{T} \rangle$ if and only if C' is satisfiable w.r.t. $\langle \emptyset, \mathcal{R}, \mathcal{T} \rangle$, which allows us to assume from now on and w.l.o.g. that ABoxes have been eliminated.

Next, in the same way that most of the classical role assertions can equivalently be replaced by GCIs or RIAs, under our preferential semantics, all of our defeasible role assertions, with the exception of $\mathsf{dAsy}(\cdot)$ and $\mathsf{dDis}(\cdot)$, can be reduced to defeasible RIAs in the following way. $\mathsf{dFun}(r)$ can be replaced by $\top \sqsubsim\, \lesssim 1r.\top$ — to be 'usually functional' means only non-normal arrows can break functionality. (Note that, since the number restriction is unqualified, r need not be simple.) $\mathsf{dRef}(r)$ and $\mathsf{dIrr}(r)$ can, respectively, be replaced with $\top \sqsubseteq\!\!\!\sim \exists r.\mathsf{Self}$ and $\top \sqsubseteq\!\!\!\sim \neg\exists r.\mathsf{Self}$. $\mathsf{dSym}(r)$ can be reduced to $r^- \sqsubseteq\!\!\!\sim r$ and $\mathsf{dTra}(r)$ to $r \circ r \sqsubseteq\!\!\!\sim r$. Furthermore, note that $\mathsf{dAsy}(r)$ can be reduced to $\mathsf{dDis}(r, r^-)$ (cf. Definition 5). Hence, from now on we can assume, w.l.o.g., that the set of role assertions $\mathcal{R}_a$ contains only statements of the form $\mathsf{Dis}(r, s)$ and $\mathsf{dDis}(r, s)$.

Next, we observe that defeasible concept inclusions can be made classical by introducing a new role name $r_\prec$ to encode $\prec$ at the object level. This is similar to the $\mathcal{SROIQ}$ encoding of the typicality operator of Giordano et al. [16, 15].

Finally, we can apply the same procedure for eliminating both the TBox and the universal role u defined for classical $\mathcal{SROIQ}$ [20, Lemma 8][26], extended to the case of $d\mathcal{SROIQ}$ concepts. Hence, from now on we can assume TBoxes (as well as occurrences of u therein) have been eliminated.

The next theorem summarises the reduction outlined in this section:

Theorem 1. *Satisfiability of $d\mathcal{SROIQ}$-concepts w.r.t. TBoxes, ABoxes and RBoxes can be polynomially reduced to satisfiability of $d\mathcal{SROIQ}$-concepts w.r.t. RBoxes in which all role assertions are of the form* $\mathsf{Dis}(r, s)$ *and* $\mathsf{dDis}(r, s)$.

It is known that classical RIAs with role composition on the LHS can be eliminated via automata-based procedures [21] or regular expressions [29]. Hence, we can assume w.l.o.g. that all classical RIAs are of the form $r \sqsubseteq s$, with $r, s \in \mathbf{R} \setminus \{\mathsf{u}\}$. Whether analogous procedures for getting rid of role composition on the LHS of *defeasible* RIAs are devisable and, if so, feasible in practice, is an open question that we leave for future investigation. (Roughly, the automaton used to 'memorise' role-paths $r_1, \ldots, r_n$ in the classical case must be carefully adapted in order to also recognise preferred role-paths so that a normal $r_1, \ldots, r_n$-path warrants the existence of an s-path, whenever $r_1 \circ \ldots \circ r_n \sqsubseteq\!\!\!\sim s$ follows from $\mathcal{R}$). Hence, in the remainder of the paper, we shall make the assumption that all defeasible RIAs are of the form $r \sqsubseteq\!\!\!\sim s$, for $r, s \in \mathbf{R} \setminus \{\mathsf{u}\}$ (and therefore $\mathcal{R}$ contains no assertions of the form $\mathsf{dTra}(\cdot)$ — see above).

Furthermore, note that the special role name $r_\prec$ used in the internalisation of defeasible concept inclusions does not appear in $\forall\!\!\!\!\forall$-, $\exists\!\!\!\exists$-, $\gtrsim$- or $\lesssim$-concepts or in defeasible RIAs, for $r_\prec \notin \mathbf{R}$.

4 A Tableau Proof Procedure for $d\mathcal{SROIQ}$

We shall now present a tableau-based algorithm for deciding consistency of $d\mathcal{SROIQ}$-concepts w.r.t. an RBox. Thanks to the results in Section 3, it also allows for checking concept satisfiability w.r.t. knowledge bases $\mathcal{KB} = \langle \mathcal{A}, \mathcal{R}, \mathcal{T} \rangle$.

The algorithm extends that for $\mathcal{SROIQ}$ [20] to deal with defeasible constructs and also works by generating a completion graph, which, if complete and clash-free (see below), can be used to construct a (possibly infinite) model for the input concept and the RBox.

With $\mathsf{nnf}(C)$ we denote the *negation normal form* (NNF) of $C \in \mathbf{C}$, i.e., the result of transforming C into an equivalent concept by pushing negation inwards and applying De Morgan's laws as well as the duality between $\forall$ and $\exists$, $\geq$ and $\leq$, $\forall\!\!\!\!\forall$ and $\exists\!\!\!\exists$, and $\gtrsim$

and $\lesssim$. Note that in NNF negation occurs only in front of concept names or in front of $\exists r.\mathsf{Self}$ or $\boxminus r.\mathsf{Self}$.

If $C \in \mathbf{C}$, $\mathsf{sub}(C)$ denotes the set of all (syntactic) *sub-concepts* of C (including C itself), and $\mathsf{clos}(C)$ is the smallest set containing C that is closed under sub-concepts and negation: $\mathsf{clos}(C) := \{D \mid D \in \mathsf{sub}(C)\} \cup \{\mathsf{nnf}(\neg D) \mid D \in \mathsf{sub}(C)\}$.

Definition 6 (Completion Graph). *Let $C \in \mathbf{C}$ be in NNF and such that the universal role* u *does not occur in C, and let $\mathcal{R}$ be an RBox. A* ***completion graph*** *for C w.r.t. $\mathcal{R}$ is a directed graph $\mathcal{G} := \langle V, E, M, \mathcal{L}, \mathcal{N}, \not\approx \rangle$ where V is a set of* ***nodes****, $E \subseteq V \times V$ is a set of* ***edges****, $M \subseteq E \times E$ is a* ***relation on edges****, $\mathcal{L}(\cdot)$ is a* ***labelling function*** *defined by:*

1. *for every $v \in V$, $\mathcal{L}(v) \subseteq \mathsf{clos}(C) \cup \mathsf{N} \cup \{\leq mr.D \mid \leq nr.D \in \mathsf{clos}(C)$ and $m \leq n\} \cup \{\lesssim mr.D \mid \lesssim nr.D \in \mathsf{clos}(C)$ and $m \leq n\}$;*
2. *for every $e = (v, v') \in E$, $\mathcal{L}(e) \subseteq \mathbf{R} \setminus \{\mathsf{u}\}$;*
3. *for every $m = (e, e') \in M$, $\mathcal{L}(m) \subseteq \mathcal{L}(e) \cap \mathcal{L}(e')$,*

$\mathcal{N} \subseteq E \times \mathbf{R}$, with $(e, r) \in \mathcal{N}$ only if $r \in \mathcal{L}(e)$, and $\not\approx\, \subseteq V \times V$ is a symmetric relation.

Intuitively, $\mathcal{N}$ tells us whether (v, v') is a *normal* r-edge among those leaving v. It is used along with M in the model-unravelling phase to construct a preference relation for each role name. (For the sake of readability, we shall henceforth write $\mathcal{L}(v, v')$ and $\mathcal{L}((v, v'), (u, u'))$ instead of $\mathcal{L}((v, v'))$ and $\mathcal{L}(((v, v'), (u, u')))$.) M is the explicit construction of the skeleton of the preference relation on the edges, and is used to construct the model resulting from the unravelling of the completion graph.

If $(v, v') \in E$, then v' is a *successor* of v, and v is a *predecessor* of v'. *Ancestor* is the transitive closure of predecessor, and *descendant* is the transitive closure of successor. We say v' is an *r-successor* of v if $r \in \mathcal{L}(v, v')$. v is an *r-predecessor* of v' if v' is an r-successor of v. *Neighbour* (resp. *r-neighbour*) is the union of successor (resp. r-successor) and predecessor (r-predecessor). If $r \in \mathbf{R} \setminus \{\mathsf{u}\}$, $C \in \mathbf{C}$ and $v \in V$ in $\mathcal{G}$, then

$$r^{\mathcal{G}}(v, C) := \{v' \mid (v, v') \in E, r \in \mathcal{L}(v, v'), \text{ and } C \in \mathcal{L}(v')\}$$

denotes all r-successors of v with C in their label, and

$$r^{\mathcal{G}}_{\mathcal{N}}(v, C) := r^{\mathcal{G}}(v, C) \cap \{v' \mid ((v, v'), r) \in \mathcal{N}\}$$

denotes (intuitively) the r-successors of v with C in their label that are accessible via an r-edge which is minimal among r-edges leaving v.

Definition 7 (Clash). *Let $\mathcal{G} = \langle V, E, M, \mathcal{L}, \mathcal{N}, \not\approx \rangle$ be a completion graph. We say $\mathcal{G}$ contains a* ***clash*** *if there are nodes $v, v', v'', v_1, \ldots, v_k, v'_1, \ldots, v'_k \in V$ such that:*

1. *$\bot \in \mathcal{L}(v)$, or for some $A \in \mathsf{C}$, $\{A, \neg A\} \subseteq \mathcal{L}(v)$;*
2. *$r \in \mathcal{L}(v, v)$ and $\neg\exists r.\mathsf{Self} \in \mathcal{L}(v)$;*
3. *$r \in \mathcal{L}(v, v)$, $((v, v), r) \in \mathcal{N}$ and $\neg\boxminus r.\mathsf{Self} \in \mathcal{L}(v)$;*
4. *$\mathsf{Dis}(r, s) \in \mathcal{R}_a$, $(v, v') \in E$ and $\{r, s\} \subseteq \mathcal{L}(v, v')$;*
5. *$\mathsf{dDis}(r, s) \in \mathcal{R}_a$, $(v, v') \in E$, $\{r, s\} \subseteq \mathcal{L}(v, v')$ and $((v, v'), r), ((v, v'), s) \in \mathcal{N}$;*
6. *$\leq nr.C \in \mathcal{L}(v)$ and $\{v_0, \ldots, v_n\} \subseteq r^{\mathcal{G}}(v, C)$, where $v_i \not\approx v_j$ for $0 \leq i < j \leq n$;*

7. *$\lesssim nr.C \in \mathcal{L}(v)$ and $\{v_0, \ldots, v_n\} \subseteq r^{\mathcal{G}}_{\mathcal{N}}(v, C)$, where $v_i \neq v_j$ for $0 \leq i < j \leq n$;*
8. *$((v, v'), r) \in \mathcal{N}$ and $r \in \mathcal{L}((v, v''), (v, v'))$;*
9. *$r \in \mathcal{L}((v_i, v'_i), (v_{i+1}, v'_{i+1}))$, for $i = 1, \ldots, k-1$, and $v_k = v_1$, $v'_k = v'_1$, or*
10. *for some $o \in \mathsf{N}$, $v \neq v'$ and $o \in \mathcal{L}(v) \cap \mathcal{L}(v')$.*

In order to ensure termination of the algorithm in the presence of transitive roles, we extend the standard (classical) *blocking* technique [19, 22] to the case of our richer structures as follows:

Definition 8 (Blocking). *Let $\mathcal{G} = \langle V, E, M, \mathcal{L}, \mathcal{N}, \neq \rangle$ be a completion graph and $v \in V$. If $\mathcal{L}(v) \cap \mathsf{N} \neq \emptyset$, then v is a **nominal** node; otherwise v is a **blockable** node. We say v is **label blocked** if v has ancestors v', u and u' such that:*

1. *$(v', v), (u', u) \in E$ and there is a path $u, \ldots, v', v$ with $u, \ldots v', v$ blockable;*
2. *$\mathcal{L}(v) = \mathcal{L}(u)$, $\mathcal{L}(v') = \mathcal{L}(u')$ and $\mathcal{L}(v', v) = \mathcal{L}(u', u)$;*
3. *for all $r \in \mathcal{L}(v', v)$, $((v', v), r) \in \mathcal{N}$ iff $((u', u), r) \in \mathcal{N}$;*
4. *for every (x, y) such that $((x, y), (v', v)) \in M$, there is (x', y') such that $\mathcal{L}(x) = \mathcal{L}(x')$, $\mathcal{L}(y) = \mathcal{L}(y')$, $\mathcal{L}(x, y) = \mathcal{L}(x', y')$, $((x', y'), (u', u)) \in M$ and $\mathcal{L}((x', y'), (u', u)) = \mathcal{L}((x, y), (v', v))$.*
5. *for every (x, y) such that $((v', v), (x, y)) \in M$, there is (x', y') such that $\mathcal{L}(x) = \mathcal{L}(x')$, $\mathcal{L}(y) = \mathcal{L}(y')$, $\mathcal{L}(x, y) = \mathcal{L}(x', y')$, $((u', u), (x', y')) \in M$ and $\mathcal{L}((u', u)(x', y')) = \mathcal{L}((v', v), (x, y))$.*

*If (1)–(5) hold, we say u **blocks** v. We say $v \in V$ is **blocked** if either (a) v is label blocked, or (b) v is blockable and there is $(v', v) \in E$ such that v' is blocked. If v is blocked but is not label blocked, then we say v is **indirectly blocked**.*

Let C be the concept of which the satisfiability w.r.t. an RBox $\mathcal{R}$ one wants to check, and let $o_1, \ldots, o_k$ be the nominals occurring in C. The tableau algorithm is initialised with a completion graph $\mathcal{G} = \langle \{v_0, v_1, \ldots, v_k\}, \emptyset, \emptyset, \mathcal{L}, \emptyset, \emptyset \rangle$, where $\mathcal{L}(v_0) := \{C\}$, $\mathcal{L}(v_i) := \{o_i\}$, for $1 \leq i \leq k$. We then expand $\mathcal{G}$ by decomposing concepts in its nodes through the application of the expansion rules in Figures 1–3. These rules are repeatedly applied until either no more rules are applicable or a clash (Definition 7) is found. In either case, we say the completion graph is *complete*. The algorithm returns "C is satisfiable w.r.t. $\mathcal{R}$", if the result of the application of the expansion rules to C and $\mathcal{R}$ is a complete and clash-free graph, and "C is unsatisfiable w.r.t. $\mathcal{R}$", otherwise.

Note that the rules in Figure 1 are the same as the corresponding ones for $\mathcal{SROIQ}$ modulo the new definitions of blocking (see Definition 8), and of merging and pruning (see below). The rules in Figure 3 deal specifically with our new non-monotonic constructs. The rules in Figure 2 correspond to those classical rules that had to be modified in the light of our richer semantics. We here detail the case of the $\exists$-rule, from which the respective explanations for the Self-, $\geq$- and NN-rules can be constructed. Unlike in the $\exists$-rule, we cannot assume the newly added r-edge is minimal among r-successors of v. We therefore need to consider the additional possibility that the new r-edge is not normal. (This has to be dealt with explicitly in order to ensure soundness of the algorithm.) Therefore, when creating a new r-successor, there are two possibilities: either (*i*) the new edge is normal among the r-edges leaving v, in which case the result is the

$\sqcap$-rule:		
if		$C_1 \sqcap C_2 \in \mathcal{L}(v)$, v is not indirectly blocked, and $\{C_1, C_2\} \not\subseteq \mathcal{L}(v)$
then		$\mathcal{L}(v) := \mathcal{L}(v) \cup \{C_1, C_2\}$
$\sqcup$-rule:		
if		$C_1 \sqcup C_2 \in \mathcal{L}(v)$, v is not indirectly blocked, $\{C_1, C_2\} \cap \mathcal{L}(v) = \emptyset$
then		$\mathcal{L}(v) := \mathcal{L}(v) \cup \{C'\}$, for some $C' \in \{C_1, C_2\}$
$\forall$-rule:		
if		$\forall r.C \in \mathcal{L}(v)$, v is not indirectly blocked, $r \in \mathcal{L}(v, v')$, $C \notin \mathcal{L}(v')$
then		$\mathcal{L}(v') := \mathcal{L}(v') \cup \{C\}$
ch-**rule**:		
if		$\leq nr.C \in \mathcal{L}(v)$, v is not indirectly blocked, $r \in \mathcal{L}(v, v')$, and $\{C, \mathsf{nnf}(\neg C)\} \cap \mathcal{L}(v') = \emptyset$
then		$\mathcal{L}(v') := \mathcal{L}(v') \cup \{C'\}$, for some $C' \in \{C, \mathsf{nnf}(\neg C)\}$
$\leq$-rule:		
if		$\leq nr.C \in \mathcal{L}(v)$, v is not indirectly blocked, $\# r^{\mathcal{G}}(v, C) > n$, and there are v_1, v_2 s.t. $r \in \mathcal{L}(v, v_1) \cap \mathcal{L}(v, v_2)$, $C \in \mathcal{L}(v_1) \cap \mathcal{L}(v_2)$, but not $v_1 \dot{\neq} v_2$
then	a.	if v_1 is a nominal node, then $\mathsf{merge}(v_2, v_1)$,
	b.	else if v_2 is a nominal node or an ancestor of v_1, then $\mathsf{merge}(v_1, v_2)$
	c.	else $\mathsf{merge}(v_2, v_1)$
o-**rule**:		
if		for some $o \in \mathsf{N}$ there are v, v' s.t. $o \in \mathcal{L}(v) \cap \mathcal{L}(v')$ and not $v \dot{\neq} v'$
then		$\mathsf{merge}(v, v')$

Fig. 1. Classical expansion rules for $d\mathcal{SROIQ}$.

same as that of applying the $\exists$-rule, or (*ii*) it is not normal, in which case there must be a most preferred r-edge, which is also more preferred than the newly created one. (This splitting is of the same nature as that in the $\sqcup$-rule, fitting the purpose of a proof by cases.) The additional index k in the $\geq$- and NN-rules serve a similar purpose.

The result of $\mathsf{prune}(v)$ in $\mathcal{G} = \langle V, E, M, \mathcal{L}, \mathcal{N}, \dot{\neq} \rangle$ is a new completion graph constructed from $\mathcal{G}$ as follows: (1) For every successor v' of v, $E := E \setminus \{(v, v')\}$ and if v' is blockable, then $\mathsf{prune}(v')$; (2) $V := V \setminus \{v\}$. (We assume these changes are propagated to $\mathcal{L}$, M, $\mathcal{N}$ and $\dot{\neq}$ in the expected way.)

The result of $\mathsf{merge}(v', v)$ in $\mathcal{G} = \langle V, E, M, \mathcal{L}, \mathcal{N}, \dot{\neq} \rangle$ is a new completion graph constructed from $\mathcal{G}$ in the following way (conditions (d)–(f) in both clauses (1) and (2) below are used to preserve the relative normality of the edges):

1. For every u s.t. $(u, v') \in E$:
   (a) if $\{(v, u), (u, v)\} \cap E = \emptyset$, then $E := E \cup \{(u, v)\}$ and $\mathcal{L}(u, v) := \mathcal{L}(u, v')$;
   (b) if $(u, v) \in E$, then $\mathcal{L}(u, v) := \mathcal{L}(u, v) \cup \mathcal{L}(u, v')$;
   (c) if $(v, u) \in E$, then $\mathcal{L}(v, u) := \mathcal{L}(v, u) \cup \{\mathsf{inv}(r) \mid r \in \mathcal{L}(u, v')\}$;
   (d) if $(x, y) \in E$ and $((x, y), (u, v')) \in M$, then $M := M \setminus \{((x, y), (u, v'))\} \cup \{((x, y), (u, v))\}$ and $\mathcal{L}((x, y), (u, v)) := \mathcal{L}((x, y), (u, v)) \cup \mathcal{L}((x, y), (u, v'))$;
   (e) if $(x, y) \in E$ and $((u, v'), (x, y)) \in M$, then $M := M \setminus \{((u, v'), (x, y))\} \cup \{((u, v), (x, y))\}$ and $\mathcal{L}((u, v), (x, y)) := \mathcal{L}((u, v), (x, y)) \cup \mathcal{L}((u, v'), (x, y))$;
   (f) if $((u, v'), r) \in \mathcal{N}$, then $\mathcal{N} := \mathcal{N} \cup \{((u, v), r)\}$;
   (g) $E := E \setminus \{(u, v')\}$;
2. For every nominal node u s.t. $(v', u) \in E$:
   (a) if $\{(v, u), (u, v)\} \cap E = \emptyset$, then $E := E \cup \{(v, u)\}$ and $\mathcal{L}(v, u) := \mathcal{L}(v', u)$;
   (b) if $(v, u) \in E$, then $\mathcal{L}(v, u) := \mathcal{L}(v, u) \cup \mathcal{L}(v', u)$;
   (c) if $(u, v) \in E$, then $\mathcal{L}(u, v) := \mathcal{L}(u, v) \cup \{\mathsf{inv}(r) \mid r \in \mathcal{L}(v', u)\}$;

$\exists$-**rule**:		
if		$\exists r.C \in \mathcal{L}(v)$, v is not blocked, and there is no v' s.t. $r \in \mathcal{L}(v, v')$ and $C \in \mathcal{L}(v')$
then	1.	create a new node v' and edge (v, v') with $\mathcal{L}(v') := \{C\}$, $\mathcal{L}(v, v') := \{r\}$ and $\mathcal{N} := \mathcal{N} \cup \{((v, v'), r)\}$
or	2.	create two new nodes v', v'' and new edges (v, v'), (v, v'') with $\mathcal{L}(v') := \{C\}$, $\mathcal{L}(v, v') := \{r\}$, $M := M \cup \{((v, v''), (v, v'))\}$, $\mathcal{L}((v, v''), (v, v')) := \{r\}$ and $\mathcal{N} := \mathcal{N} \cup \{((v, v''), r)\}$
Self-**rule**:		
if		$\exists r.\mathsf{Self} \in \mathcal{L}(v)$, v is not blocked, and $r \notin \mathcal{L}(v, v)$
then	1.	add an edge (v, v), if it does not exist, $\mathcal{L}(v, v) := \mathcal{L}(v, v) \cup \{r\}$, and $\mathcal{N} := \mathcal{N} \cup \{((v, v), r)\}$
or	2.	create a node v' and edges (v, v), (v, v'), $\mathcal{L}(v, v) := \mathcal{L}(v, v) \cup \{r\}$, $\mathcal{L}(v, v') := \{r\}$, $M := M \cup \{((v, v'), (v, v))\}$, $\mathcal{L}((v, v'), (v, v)) := \{r\}$, and $\mathcal{N} := \mathcal{N} \cup \{((v, v'), r)\}$
$\geq$-**rule**:		
if		$\geq nr.C \in \mathcal{L}(v)$, v is not blocked, and there are no $v_1, \ldots, v_n$ s.t. $r \in \mathcal{L}(v, v_i)$, $C \in \mathcal{L}(v_i)$, $i = 1, \ldots, n$, and $v_i \dot{\neq} v_j$, for $1 \leq i < j \leq n$, and each v_i is not blocked if v is not blockable
then	a.	guess $k \in \{0, \ldots, n\}$,
	b.	create k new nodes $v_1, \ldots, v_k$ and edges (v, v_i), for $i = 1, \ldots, k$, with $\mathcal{L}(v, v_i) := \{r\}$, $\mathcal{L}(v_i) := \{C\}$ and $\mathcal{N} := \mathcal{N} \cup \{((v, v_i), r)\}$,
	c.	create $2(n - k)$ new nodes $v_{k+1}, \ldots, v_n$ and $v'_{k+1}, \ldots, v'_n$ and edges (v, v_i) and (v, v'_i), for $i = k + 1, \ldots, n$, with $\mathcal{L}(v_i) := \{C\}$, $\mathcal{L}(v, v_i) := \{r\}$, $\mathcal{L}(v, v'_i) := \{r\}$, $M := M \cup \{((v, v'_i), (v, v_i))\}$, $\mathcal{L}((v, v'_i), (v, v_i)) := \{r\}$ and $\mathcal{N} := \mathcal{N} \cup \{((v, v'_i), r)\}$, and
	d.	set $v_i \dot{\neq} v_j$, for $1 \leq i < j \leq n$
NN-rule:		
if	1.	$\leq nr.C \in \mathcal{L}(v)$, v is not blockable, $r \in \mathcal{L}(v', v)$, v' is blockable, and $C \in \mathcal{L}(v')$
	2.	there is no $m \in \{1, \ldots, n\}$ s.t. $\leq mr.C \in \mathcal{L}(v)$ and s.t. there are m nominal r-successors $v_1, \ldots, v_m$ of v with $C \in \mathcal{L}(v_i)$ and $v_i \dot{\neq} v_j$ for all $1 \leq i < j \leq m$
then	a.	guess $m \in \{1, \ldots, n\}$, set $\mathcal{L}(v) := \mathcal{L}(v) \cup \{\lesssim mr.C\}$ and guess $k \in \{0, \ldots, m\}$,
	b.	create k new nodes $v_1, \ldots, v_k$ and edges (v, v_i), for $i = 1, \ldots, k$, with $\mathcal{L}(v, v_i) := \{r\}$, $\mathcal{L}(v_i) := \{C, o_i\}$ with each $o_i \in \mathsf{N}$ new in $\mathcal{G}$ and $\mathcal{N} := \mathcal{N} \cup \{((v, v_i), r)\}$,
	c.	create $2(m - k)$ new nodes $v_{k+1}, \ldots, v_m$ and $v'_{k+1}, \ldots, v'_m$ and edges (v, v_i) and (v, v'_i), for $i = k + 1, \ldots, m$, with $\mathcal{L}(v, v_i) := \{r\}$, $\mathcal{L}(v_i) := \{C, o_i\}$, with each $o_i \in \mathsf{N}$ new in $\mathcal{G}$, $\mathcal{L}(v, v'_i) := \{r\}$, $M := M \cup \{((v, v'_i), (v, v_i))\}$, $\mathcal{L}((v, v'_i), (v, v_i)) := \{r\}$ and $\mathcal{N} := \mathcal{N} \cup \{((v, v'_i), r)\}$, and
	d.	set $v_i \dot{\neq} v_j$, for $1 \leq i < j \leq m$

Fig. 2. New classical expansion rules for $d\mathcal{SROIQ}$.

(d) if $(x, y) \in E$ and $((x, y), (v', u)) \in M$, then $M := M \setminus \{((x, y), (v', u))\} \cup \{((x, y), (v, u))\}$ and $\mathcal{L}((x, y), (v, u)) := \mathcal{L}((x, y), (v, u)) \cup \mathcal{L}((x, y), (v', u))$;
(e) if $(x, y) \in E$ and $((v', u), (x, y)) \in M$, then $M := M \setminus \{((v', u), (x, y))\} \cup \{((v, u), (x, y))\}$ and $\mathcal{L}((v, u), (x, y)) := \mathcal{L}((v, u), (x, y)) \cup \mathcal{L}((v', u), (x, y))$;
(f) if $((v', u), r) \in \mathcal{N}$, then $\mathcal{N} := \mathcal{N} \cup \{((v, u), r)\}$;
(g) $E := E \setminus \{(v', u)\}$;

3. $\mathcal{L}(v) := \mathcal{L}(v) \cup \mathcal{L}(v')$;
4. $\dot{\neq} := \dot{\neq} \cup \{(v, w) \mid v' \dot{\neq} w\}$; and
5. $\mathsf{prune}(v')$.

As in the classical case, in order to ensure termination of the tableau algorithm, one has to assign higher priorities to certain rules. Here we assume the following strategy is adopted: The o-rule is applied with the highest priority; the NN- and dNN-rules are applied before the $\leq$- and $\lesssim$-rules; the other rules are applied with a lower priority.

Theorem 2. *Let $C \in \mathbf{C}$ and let $\mathcal{R}$ be an RBox.*

1. *The algorithm terminates if started with $\mathsf{nnf}(C)$ and $\mathcal{R}$;*
2. *When exhaustively applied to $\mathsf{nnf}(C)$ and $\mathcal{R}$, the expansion rules yield a complete and clash-free completion graph iff C is satisfiable w.r.t. $\mathcal{R}$.*

$\mathcal{R}_h$-**rule**:		
if		$r \in \mathcal{L}(v, v')$, v is not indirectly blocked and either $r \sqsubseteq s \in \mathcal{R}$ or both $r \underset{\sim}{\sqsubset} s \in \mathcal{R}$ and $((v, v'), r) \in \mathcal{N}$
then		$\mathcal{L}(v, v') := \mathcal{L}(v, v') \cup \{s\}$
$\tilde{\exists}$-**rule**:		
if		$\tilde{\exists} r.C \in \mathcal{L}(v)$, v is not blocked and there is no v' s.t. $r \in \mathcal{L}(v, v')$, $C \in \mathcal{L}(v')$ and $((v, v'), r) \in \mathcal{N}$
then		create a new node v' and edge (v, v') with $\mathcal{L}(v') := \{C\}$, $\mathcal{L}(v, v') := \{r\}$ and $\mathcal{N} := \mathcal{N} \cup \{((v, v'), r)\}$
dSelf-**rule**:		
if		$\tilde{\exists} r.\mathsf{Self} \in \mathcal{L}(v)$, v is not blocked and either $r \notin \mathcal{L}(v, v)$ or $((v, v), r) \notin \mathcal{N}$
then		add a new edge (v, v), if required, $\mathcal{L}(v, v) := \mathcal{L}(v, v) \cup \{r\}$, and $\mathcal{N} := \mathcal{N} \cup \{((v, v), r)\}$
$\tilde{\forall}$-**rule**:		
if		$\tilde{\forall} r.C \in \mathcal{L}(v)$, v is not indirectly blocked, $r \in \mathcal{L}(v, v')$, $((v, v'), r) \in \mathcal{N}$ and $C \notin \mathcal{L}(v')$
then		$\mathcal{L}(v') := \mathcal{L}(v') \cup \{C\}$
dch-**rule**:		
if		$\lesssim nr.C \in \mathcal{L}(v)$, v is not indirectly blocked, $r \in \mathcal{L}(v, v')$, $((v, v'), r) \in \mathcal{N}$ and $\{C, \mathsf{nnf}(\neg C)\} \cap \mathcal{L}(v') = \emptyset$
then		$\mathcal{L}(v') := \mathcal{L}(v') \cup \{C'\}$, for some $C' \in \{C, \mathsf{nnf}(\neg C)\}$
$\gtrsim$-**rule**:		
if		$\gtrsim nr.C \in \mathcal{L}(v)$, v is not blocked, and there are no $v_1, \ldots, v_n$ s.t. $r \in \mathcal{L}(v, v_i)$, $((v, v_i), r) \in \mathcal{N}$, $C \in \mathcal{L}(v_i)$, for $i = 1, \ldots, n$, and s.t. $v_i \not\doteq v_j$, for $1 \leq i < j \leq n$, and each v_i is not blocked if v is not blockable
then		create n new nodes $v_1, \ldots, v_n$ with $\mathcal{L}(v, v_i) = \{r\}$, $\mathcal{N} := \mathcal{N} \cup \{((v, v_i), r)\}$, $\mathcal{L}(v_i) = \{C\}$, for $i = 1, \ldots, n$, and set $v_i \not\doteq v_j$, $1 \leq i < j \leq n$
$\lesssim$-**rule**:		
if		$\lesssim nr.C \in \mathcal{L}(v)$, v is not indirectly blocked, $\#r^{\mathcal{G}}_{\mathcal{N}}(v, C) > n$, and there are v_1, v_2 s.t. $r \in \mathcal{L}(v, v_1) \cap \mathcal{L}(v, v_2)$, $((v, v_1), r), ((v, v_2), r) \in \mathcal{N}$, $C \in \mathcal{L}(v_1) \cap \mathcal{L}(v_2)$ but not $v_1 \not\doteq v_2$
then	a.	if v_1 is a nominal node, then $\mathsf{merge}(v_2, v_1)$, else
	b.	if v_2 is a nominal node or an ancestor of v_1, then $\mathsf{merge}(v_1, v_2)$
	c.	else $\mathsf{merge}(v_2, v_1)$
dNN-**rule**:		
if	1.	$\lesssim nr.C \in \mathcal{L}(v)$, v is not blockable, $r \in \mathcal{L}(v', v)$, v' is blockable and $C \in \mathcal{L}(v')$
	2.	there is no $m \in \{1, \ldots, n\}$ s.t. $\lesssim mr.C \in \mathcal{L}(v)$ and s.t. there are m nominal nodes $v_1, \ldots, v_m$ with $(v, v_i) \in E$, $r \in \mathcal{L}(v, v_i)$, $((v, v_i), r) \in \mathcal{N}$, $C \in \mathcal{L}(v_i)$, for $i = 1, \ldots, m$, and with $v_i \not\doteq v_j$, for all $1 \leq i < j \leq m$
then	a.	guess $m \in \{1, \ldots, n\}$ and set $\mathcal{L}(v) := \mathcal{L}(v) \cup \{\lesssim mr.C\}$
	b.	create m new nodes $v'_1, \ldots, v'_m$ with $\mathcal{L}(v, v'_i) := \{r\}$, $\mathcal{N} := \mathcal{N} \cup \{((v, v'_i), r) \mid 1 \leq i \leq n\}$, $\mathcal{L}(v'_i) := \{C, o_i\}$, with $o_i \in \mathsf{N}$ new in $\mathcal{G}$, $i = 1, \ldots, m$, and set $v'_i \not\doteq v'_j$, $1 \leq i < j \leq m$

Fig. 3. Defeasible expansion rules for $d\mathcal{SROIQ}$.

5 Summary and Future Work

The main contributions of the present paper are: (*i*) a meaningful extension of $\mathcal{SROIQ}$ with defeasible reasoning constructs in the concept language, in both concept and role inclusions, and in role assertions, together with an intuitive KLM-style preferential semantics; (*ii*) a translation of the entailment problem w.r.t. $d\mathcal{SROIQ}$ knowledge bases to concept satisfiability relative to an RBox only, and (*iii*) a terminating, sound and complete tableau-based algorithm for checking concept satisfiability w.r.t. $d\mathcal{SROIQ}$ RBoxes.

As for the next steps, we have (*i*) extending the tableau procedure to allow role composition in defeasible RIAs, (*ii*) an analysis of the computational complexity of concept satisfiability for $d\mathcal{SROIQ}$, (*iii*) an investigation of the correspondence between $d\mathcal{SROIQ}$ and an extension of the OWL 2 RDF semantics[3], and (*iv*) the definition of an appropriate notion of *non-monotonic* entailment for $d\mathcal{SROIQ}$ ontologies.

[3] https://www.w3.org/TR/2012/REC-owl2-rdf-based-semantics-20121211

References

1. Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P. (eds.): The Description Logic Handbook: Theory, Implementation and Applications. Cambridge University Press, 2 edn. (2007)
2. Bonatti, P., Faella, M., Petrova, I., Sauro, L.: A new semantics for overriding in description logics. Artificial Intelligence 222, 1–48 (2015)
3. Bonatti, P., Faella, M., Sauro, L.: Defeasible inclusions in low-complexity DLs. Journal of Artificial Intelligence Research 42, 719–764 (2011)
4. Bonatti, P., Lutz, C., Wolter, F.: The complexity of circumscription in description logic. Journal of Artificial Intelligence Research 35, 717–773 (2009)
5. Boutilier, C.: Conditional logics of normality: A modal approach. Artificial Intelligence 68(1), 87–154 (1994)
6. Britz, K., Casini, G., Meyer, T., Moodley, K., Varzinczak, I.: Ordered interpretations and entailment for defeasible description logics. Tech. rep., CAIR, CSIR Meraka and UKZN, South Africa (2013), `http://tinyurl.com/cydd6yy`
7. Britz, K., Casini, G., Meyer, T., Varzinczak, I.: Preferential role restrictions. In: Proceedings of the 26th International Workshop on Description Logics. pp. 93–106 (2013)
8. Britz, K., Meyer, T., Varzinczak, I.: Semantic foundation for preferential description logics. In: Wang, D., Reynolds, M. (eds.) Proceedings of the 24th Australasian Joint Conference on Artificial Intelligence. pp. 491–500. No. 7106 in LNAI, Springer (2011)
9. Britz, K., Varzinczak, I.: Defeasible modalities. In: Proceedings of the 14th Conference on Theoretical Aspects of Rationality and Knowledge (TARK). pp. 49–60 (2013)
10. Britz, K., Varzinczak, I.: Introducing role defeasibility in description logics. In: Michael, L., Kakas, A. (eds.) Proceedings of the 15th European Conference on Logics in Artificial Intelligence (JELIA). pp. 174–189. No. 10021 in LNCS, Springer (2016)
11. Britz, K., Varzinczak, I.: Preferential modalities revisited. In: Proceedings of the 16th International Workshop on Nonmonotonic Reasoning (NMR) (2016)
12. Casini, G., Meyer, T., Moodley, K., Sattler, U., Varzinczak, I.: Introducing defeasibility into OWL ontologies. In: Arenas, M., Corcho, O., Simperl, E., Strohmaier, M., d'Aquin, M., Srinivas, K., Groth, P., Dumontier, M., Heflin, J., Thirunarayan, K., Staab, S. (eds.) Proceedings of the 14th International Semantic Web Conference (ISWC). pp. 409–426. No. 9367 in LNCS, Springer (2015)
13. Casini, G., Straccia, U.: Rational closure for defeasible description logics. In: Janhunen, T., Niemelä, I. (eds.) Proceedings of the 12th European Conference on Logics in Artificial Intelligence (JELIA). pp. 77–90. No. 6341 in LNCS, Springer-Verlag (2010)
14. Casini, G., Straccia, U.: Defeasible inheritance-based description logics. Journal of Artificial Intelligence Research (JAIR) 48, 415–473 (2013)
15. Giordano, L., Gliozzi, V.: Encoding a preferential extension of the description logic $\mathcal{SROIQ}$ into $\mathcal{SROIQ}$. In: Foundations of Intelligent Systems. pp. 248–258. No. 9384 in LNCS, Springer (2015)
16. Giordano, L., Gliozzi, V., Olivetti, N., Pozzato, G.: $\mathcal{ALC} + T$: a preferential extension of description logics. Fundamenta Informaticae 96(3), 341–372 (2009)
17. Giordano, L., Gliozzi, V., Olivetti, N., Pozzato, G.: A non-monotonic description logic for reasoning about typicality. Artificial Intelligence 195, 165–202 (2013)
18. Giordano, L., Gliozzi, V., Olivetti, N., Pozzato, G.: Semantic characterization of rational closure: From propositional logic to description logics. Artificial Intelligence 226, 1–33 (2015)
19. Heuerding, A., Seyfried, M., Zimmermann, H.: Efficient loop-check for backward proof search in some non-classical propositional logics. In: Miglioli, P., Moscato, U., Mundici,

D., Ornaghi, M. (eds.) Proceedings of the 5th International Workshop on Theorem Proving with Analytic Tableaux and Related Methods (TABLEAUX). pp. 210–225. No. 1071 in LNAI, Springer (1996)
20. Horrocks, I., Kutz, O., Sattler, U.: The even more irresistible $\mathcal{SROIQ}$. In: Doherty, P., Mylopoulos, J., Welty, C. (eds.) Proceedings of the 10th International Conference on Principles of Knowledge Representation and Reasoning (KR). pp. 57–67. Morgan Kaufmann (2006)
21. Horrocks, I., Sattler, U.: Decidability of $\mathcal{SHIQ}$ with complex role inclusion axioms. Artificial Intelligence 160, 79–104 (2004)
22. Horrocks, I., Sattler, U., Tobies, S.: Reasoning with individuals for the description logic $\mathcal{SHIQ}$. In: MacAllester, D. (ed.) Proceedings of the 17th International Conference on Automated Deduction (CADE). pp. 482–496. No. 1831 in LNCS, Springer (2000)
23. Kraus, S., Lehmann, D., Magidor, M.: Nonmonotonic reasoning, preferential models and cumulative logics. Artificial Intelligence 44, 167–207 (1990)
24. Lehmann, D., Magidor, M.: What does a conditional knowledge base entail? Artificial Intelligence 55, 1–60 (1992)
25. Quantz, J., Royer, V.: A preference semantics for defaults in terminological logics. In: Proceedings of the 3rd International Conference on Principles of Knowledge Representation and Reasoning (KR). pp. 294–305 (1992)
26. Schild, K.: A correspondence theory for terminological logics: Preliminary report. In: Proceedings of the 12th International Joint Conference on Artificial Intelligence (IJCAI). pp. 466–471 (1991)
27. Sengupta, K., Alfa Krisnadhi, A., Hitzler, P.: Local closed world semantics: Grounded circumscription for OWL. In: Aroyo, L., Welty, C., Alani, H., Taylor, J., Bernstein, A., Kagal, L., Noy, N., Blomqvist, E. (eds.) Proceedings of the 10th International Semantic Web Conference (ISWC). pp. 617–632. No. 7031 in LNCS, Springer (2011)
28. Shoham, Y.: Reasoning about Change: Time and Causation from the Standpoint of Artificial Intelligence. MIT Press (1988)
29. Simančík, F.: Elimination of complex RIAs without automata. In: Proceedings of the 25th International Workshop on Description Logics. CEUR, vol. 846 (2012)