TECHNISCHE UNIVERSITÄT DRESDEN

Technische Universität Dresden
Institute for Theoretical Computer Science
Chair for Automata Theory

LTCS–Report

A Tableau Algorithm for $\mathcal{SROIQ}$ under Infinitely Valued Gödel Semantics

Stefan Borgwardt and Rafael Peñaloza

LTCS-Report 15-18

Postal Address:
Lehrstuhl für Automatentheorie
Institut für Theoretische Informatik
TU Dresden
01062 Dresden

http://lat.inf.tu-dresden.de

Visiting Address:
Nöthnitzer Str. 46
Dresden

A Tableau Algorithm for $\mathcal{SROIQ}$ under Infinitely Valued Gödel Semantics

Stefan Borgwardt and Rafael Peñaloza

Abstract

Fuzzy description logics (FDLs) are knowledge representation formalisms capable of dealing with imprecise knowledge by allowing intermediate membership degrees in the interpretation of concepts and roles. One option for dealing with these intermediate degrees is to use the so-called Gödel semantics. Despite its apparent simplicity, developing reasoning techniques for expressive FDLs under this semantics is a hard task. We present a tableau algorithm for deciding consistency of a $\mathcal{SROIQ}$ ontology under Gödel semantics. This is the first algorithm that can handle the full expressivity of $\mathcal{SROIQ}$ as well as the full Gödel semantics.

1 Introduction

Description logics (DLs) [1] are a family of knowledge representation formalisms designed to represent the terminological knowledge of an application domain in an unambiguous and easy-to-understand manner. They have been successfully applied for the modelling of many real-world domains, including several from the bio-medical sciences. In addition, many efficient reasoners are now available.

As it has been widely argued in the literature, one of the important deficits of classical DLs is their inability to handle imprecise notions for which a clear-cut characterisation cannot be achieved. To cover this gap, it has been proposed to extend the semantics of DLs following the ideas of mathematical fuzzy logic [12]. Briefly, fuzzy description logics allow intermediate truth degrees—usually real numbers between 0 (false) and 1 (true)—to be used in the definition of imprecise knowledge [2]. To interpret these intermediate degrees, the logical connectives need to be extended accordingly. In general, there are many possible extensions that can be used; hence, each (classical) DL gives rise to a family of FDLs. However, for most of these extensions, reasoning becomes undecidable, even if the underlying DL is relatively inexpressive [6]. In fact, the only decidable expressive FDLs are those based on the Gödel semantics, and the variant Zadeh semantics.

Developing a reasoning algorithm for the very expressive DL $\mathcal{SROIQ}$ is far from trivial, as one needs to handle all the constructors, including nominals and number restrictions, adequately. This difficulty is accentuated when the Gödel semantics are considered, since this logic does not have the finitely valued model property [5]. This means that there are ontologies whose models must use infinitely many truth degrees, and hence must have an infinite domain. Indeed, this is one of the reasons why the *crispification approach* as described in [4, 7] is only valid under finitely valued semantics.

To the best of our knowledge, the only existing algorithms for deciding consistency of ontologies in expressive DLs with (infinitely valued) Gödel semantics are the automata-based approach from [5, 8] and the new crispification method from [11]. Rather than trying to find a model directly, the automata-based approach produces an abstract representation of a large class of models. In this representation, the actual degrees of truth used in a model are abstracted to consider only the order among them. This abstraction from the actual degrees is also exploited by the crispification approach, which translates a fuzzy ontology into a classical ontology by using concepts that simulate the order between the relevant truth degrees. As an added benefit, considering only the order between concepts allows for a more flexible representation of the domain knowledge in which, for instance, one can express that an individual is more *tall* than *strong*, without having to specify truth degrees for each of these concepts.

Although they provide good theoretical results such as tight complexity bounds for reasoning, these approaches are restricted to sublogics of $\mathcal{SROIQ}$, and there is no obvious way to extend them to the full expressivity of $\mathcal{SROIQ}$. Moreover, the automata-based approach is not adequate for producing efficient implementations, as it requires the construction of an exponentially large automaton before any reasoning steps are made. In this paper we present a new tableau-based algorithm that combines the ideas of the classical tableau approach for $\mathcal{SROIQ}$ with the order-based abstraction developed to handle the Gödel semantics. The result is the first reasoning algorithm that can handle the full expressivity of fuzzy $\mathcal{SROIQ}$ under Gödel semantics. Interestingly, our algorithm inherits the properties that allow an efficient implementation of the classical tableau algorithm. In particular, the algorithm behaves better when the input ontology only uses less expressive features of the logic, and applies complex constructions only when it is absolutely necessary.

2 Preliminaries

We recall basic definitions about FDLs and weighted automata using Gödel semantics [2, 5, 11].

2.1 Gödel Fuzzy Logic and Order Structures

The two basic operators of Gödel fuzzy logic are conjunction and implication, interpreted by the *Gödel t-norm* and *residuum*, respectively. The Gödel t-norm of two fuzzy degrees $x, y \in [0, 1]$ is defined as minimum function $\min\{x, y\}$. The residuum $\Rightarrow$ is uniquely defined by the equivalence $\min\{x, y\} \leqslant z$ iff $y \leqslant (x \Rightarrow z)$ for all $x, y, z \in [0, 1]$, and can be computed as

$$x \Rightarrow y = \begin{cases} 1 & \text{if } x \leqslant y, \\ y & \text{otherwise.} \end{cases}$$

For a deeper introduction to t-norms and t-norm-based fuzzy logics, see [12, 13, 21].

An *order structure* S is a finite set containing at least the numbers 0, 0.5, and 1, together with an involutive unary operation $\mathsf{inv}\colon S \to S$ such that $\mathsf{inv}(x) = 1 - x$ for all numbers $x \in S \cap [0, 1]$. A *total preorder* (on S) is a transitive and total binary relation $\preceq_* \subseteq S \times S$. For $\alpha, \beta \in S$, we write $\alpha \simeq_* \beta$ if $\alpha \preceq_* \beta$ and $\beta \preceq_* \alpha$, and we write $\alpha \prec_* \beta$ if it is not the case that $\beta \preceq_* \alpha$. Notice that $\simeq_*$ is an equivalence relation on S. For a relation symbol $\bowtie \in \{<, \leqslant, =, \geqslant, >\}$, we denote by $\bowtie_*$ the corresponding relation induced by $\preceq_*$; that is, $\prec_*$, $\preceq_*$, $\simeq_*$, $\succeq_*$, or $\succ_*$, respectively. The set $\mathsf{order}(S)$ contains exactly those total preorders $\preceq_*$ over S which

- have 0 and 1 as least and greatest element, respectively,
- preserve the order of the real numbers on $S \cap [0, 1]$, and
- satisfy $\alpha \preceq_* \beta$ iff $\mathsf{inv}(\beta) \preceq_* \mathsf{inv}(\alpha)$ for all $\alpha, \beta \in S$.

Given $\preceq_* \in \mathsf{order}(S)$, the following functions on S that mimic the operators of Gödel fuzzy logic over $[0, 1]$ are well-defined since $\preceq_*$ is total:

$$\min_*\{\alpha, \beta\} := \begin{cases} \alpha & \text{if } \alpha \preceq_* \beta \\ \beta & \text{otherwise,} \end{cases}$$

$$\alpha \Rightarrow_* \beta := \begin{cases} 1 & \text{if } \alpha \preceq_* \beta \\ \beta & \text{otherwise.} \end{cases}$$

It is easy to see that these operators agree with min and $\Rightarrow$ on the set $S \cap [0, 1]$.

An *order assertion* (over S) is an expression of the form $\alpha \bowtie \beta$, where $\bowtie \in \{<, \leqslant, =, \geqslant, >\}$ and $\alpha, \beta \in S$. An *order formula* is a Boolean combination of order assertions. An element $\preceq_* \in \mathsf{order}(S)$ *satisfies* (or is a *model* of)

- the order assertion $\alpha \bowtie \beta$ iff $\alpha \bowtie_* \beta$ holds;
- an order formula if there is a satisfying Boolean valuation of all its order assertions such that $\preceq_*$ satisfies all order assertions evaluated to true, and does not satisfy any order assertions evaluated to false.
- a set of order assertions if it satisfies all assertions contained in it.

A set of order assertions Φ is *satisfiable* if it has a model, and it *entails* an order assertion ϕ if all models of Φ are also models of ϕ. Deciding satisfiability of a set of order assertions Φ is clearly possible in time polynomial in the size of S: one can saturate Φ w.r.t. the axioms defining $\mathsf{order}(S)$, i.e. transitivity, the order on $S \cap [0, 1]$, and the properties of inv, and then check whether the resulting set contains an assertion of the form $\alpha < \alpha$. Furthermore, Φ entails $\alpha \leqslant \beta$ iff $\Phi \cup \{\beta > \alpha\}$ is unsatisfiable (and similarly for $<$ and $\geqslant$), and Φ entails $\alpha = \beta$ iff it entails both $\alpha \leqslant \beta$ and $\alpha \geqslant \beta$.

For convenience, we also sometimes use expressions like $\alpha \geqslant \min\{\beta, \gamma\}$ or $\alpha = \beta \Rightarrow \gamma$, where min and $\Rightarrow$ are interpreted using the operators $\min_*$ and $\Rightarrow_*$, respectively, introduced above.

2.2 G-$\mathcal{SROIQ}$

We now define the fuzzy description logic G-$\mathcal{SROIQ}$. Let $\mathsf{N_I}$, $\mathsf{N_C}$, and $\mathsf{N_R}$ be three mutually disjoint sets of *individual names*, *concept names*, and *role names*, respectively, where $\mathsf{N_R}$ contains the *universal role* r_u. The set of *roles* is $\mathsf{N_R^-} := \mathsf{N_R} \cup \{r^- \mid r \in \mathsf{N_R}\}$, where the elements of the form r^- are called *inverse roles*. Since we need to make several syntactic restrictions based on which roles appear in which role axioms, we start by defining role hierarchies. A *role hierarchy* $\mathcal{R}_h$ is a finite set of *(complex) role inclusions* of the form $w \sqsubseteq r \geqslant p$, where $r \neq r_u$ is a role name, $w \in (\mathsf{N_R^-})^+$ is a non-empty *role chain* not including the universal role, and $p \in (0, 1]$. Such a role inclusion is called *simple* if $w \in \mathsf{N_R^-}$. We extend the notation $\cdot^-$ to inverse roles and role chains as usual, by setting $(r^-)^- := r$ and $(r_1 \dots r_n)^- := r_n^- \dots r_1^-$.

We recall now the regularity condition from [3,16]. Let $\lessdot$ be a strict partial order on $\mathsf{N_R^-}$ such that $r \lessdot s$ iff $r^- \lessdot s$. A role inclusion $w \sqsubseteq r \geqslant p$ is *$\lessdot$-regular* if

- w is of the form rr or r^-, or
- w is of the form $r_1 \dots r_n$, $rr_1 \dots r_n$, or $r_1 \dots r_n r$, and for all $1 \leqslant i \leqslant n$ it holds that $r_i \lessdot r$.

An role hierarchy $\mathcal{R}_h$ is *regular* if there is a strict partial order $\lessdot$ as above such that each role inclusion in $\mathcal{R}_h$ is $\lessdot$-regular. A role name r is *simple* (w.r.t. $\mathcal{R}_h$) if for each $w \sqsubseteq r \geqslant p \in \mathcal{R}_h$ we have that w is of the form s or s^- for a simple role s. This notion is well-defined since the regularity condition prevents any cyclic dependencies between role names in $\mathcal{R}_h$. An inverse role r^- is *simple* if r is simple. For the rest of this paper, let $\mathcal{R}_h$ be a regular role hierarchy.

Table 1: Syntax and semantics of G-$\mathcal{SROIQ}$

Name	Syntax	Semantics ($C^{\mathcal{I}}(d)$ / $r^{\mathcal{I}}(d,e)$)
concept name	A	$A^{\mathcal{I}}(d) \in [0,1]$
truth constant	$\overline{p}$	p
conjunction	$C \sqcap D$	$\min\{C^{\mathcal{I}}(d), D^{\mathcal{I}}(d)\}$
implication	$C \to D$	$C^{\mathcal{I}}(d) \Rightarrow D^{\mathcal{I}}(d)$
negation	$\neg C$	$1 - C^{\mathcal{I}}(d)$
existential restriction	$\exists r.C$	$\sup_{e \in \Delta^{\mathcal{I}}} \min\{r^{\mathcal{I}}(d,e), C^{\mathcal{I}}(e)\}$
value restriction	$\forall r.C$	$\inf_{e \in \Delta^{\mathcal{I}}} r^{\mathcal{I}}(d,e) \Rightarrow C^{\mathcal{I}}(e)$
nominal	$\{a\}$	$\begin{cases} 1 & \text{if } d = a^{\mathcal{I}} \\ 0 & \text{otherwise} \end{cases}$
at-least restriction	$\geqslant n\, s.C$	$\sup_{\substack{e_1,\dots,e_n \in \Delta^{\mathcal{I}} \\ \text{pairwise different}}} \min_{i=1}^{n} \min\{s^{\mathcal{I}}(d,e_i), C^{\mathcal{I}}(e_i)\}$
local reflexivity	$\exists s.\mathsf{Self}$	$r^{\mathcal{I}}(d,d)$
role name	r	$r^{\mathcal{I}}(d,e) \in [0,1]$
inverse role	r^-	$r^{\mathcal{I}}(e,d)$
universal role	r_u	1

G-$\mathcal{SROIQ}$ *concepts* [11] are built using the constructors listed in the upper part of Table 1, where C, D denote concepts, $p \in [0,1]$, $n \in \mathbb{N}$, $A \in \mathsf{N_C}$, $a \in \mathsf{N_I}$, $r \in \mathsf{N_R^-}$, and $s \in \mathsf{N_R^-}$ is a simple role. The restriction to simple roles in at-least restrictions is necessary to ensure decidability, already in the classical case [18]. We also use the common DL constructors $\top := \overline{1}$ (top concept), $\bot := \overline{0}$ (bottom concept), $C \sqcup D := \neg(\neg C \sqcap \neg D)$ (disjunction), and $\leqslant n\, s.C := \neg(\geqslant (n+1)\, s.C)$ (at-most restriction).

The semantics of G-$\mathcal{SROIQ}$ is based on *(fuzzy) interpretations* $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ over a non-empty *domain* $\Delta^{\mathcal{I}}$, which assign to each individual name $a \in \mathsf{N_I}$ an element $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$, to each concept name $A \in \mathsf{N_C}$ a fuzzy set $A^{\mathcal{I}} \colon \Delta^{\mathcal{I}} \to [0,1]$, and to each role name $r \in \mathsf{N_R}$ a fuzzy binary relation $r^{\mathcal{I}} \colon \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \to [0,1]$. This interpretation is extended to complex concepts and roles as defined in the last column of Table 1, for all $d, e \in \Delta^{\mathcal{I}}$.

As it is common for fuzzy DLs, we restrict reasoning to *witnessed* interpretations [14]. Intuitively, these interpretations require that the suprema and infima in the semantics are in fact maxima and minima, respectively. In other words, the degrees of these constructors are witnessed by an element in the domain. Formally, an interpretation $\mathcal{I}$ is *witnessed* if, for every $d \in \Delta^{\mathcal{I}}$, $n \geqslant 0$, $r \in \mathsf{N_R^-}$, simple $s \in \mathsf{N_R^-}$, and concept C, there are $e, e', e_1, \dots, e_n \in \Delta^{\mathcal{I}}$ such that $e_1, \dots, e_n$ are pairwise different,

$$\begin{aligned}(\exists r.C)^{\mathcal{I}}(d) &= \min\{r^{\mathcal{I}}(d,e), C^{\mathcal{I}}(e)\}, \\ (\forall r.C)^{\mathcal{I}}(d) &= r^{\mathcal{I}}(d,e') \Rightarrow C^{\mathcal{I}}(e'), \qquad \text{and} \\ (\geqslant n\, s.C)^{\mathcal{I}}(d) &= \min_{i=1}^{n} \min\{s^{\mathcal{I}}(d,e_i), C^{\mathcal{I}}(e_i)\}.\end{aligned}$$

As we have seen already with the role inclusions, the axioms of G-$\mathcal{SROIQ}$ extend classical axioms by allowing to state a degree in $(0,1]$ to which the axioms hold. An *ordered ABox* is a finite set of *(fuzzy) concept assertions* of the form $C(a) \bowtie 1$ for $a \in \mathsf{N_I}$, a concept C, and $\bowtie \in \{<, \geqslant\}$. A *TBox* is a finite set of *general concept inclusions (GCIs)* of the form $C \sqsubseteq D \geqslant p$ for concepts C, D and $p \in (0,1]$. An *RBox* $\mathcal{R} = \mathcal{R}_h \cup \mathcal{R}_a$ consists of a role hierarchy $\mathcal{R}_h$ and a finite set $\mathcal{R}_a$ of *disjoint role axioms* $\mathsf{dis}(s_1, s_2) \geqslant p$ and *reflexivity axioms* $\mathsf{ref}(r) \geqslant p$, where r is a role, s_1, s_2 are simple roles, and $p \in (0,1]$. An *ontology* $\mathcal{O} = (\mathcal{A}, \mathcal{T}, \mathcal{R})$ consists of an ABox $\mathcal{A}$, a TBox $\mathcal{T}$, and an RBox $\mathcal{R}$.

For an ontology $\mathcal{O}$, we denote by $\mathsf{rol}(\mathcal{O})$ the set of all roles occurring in $\mathcal{O}$, together with their inverses, and by $\mathsf{ind}(\mathcal{O})$ the set of all individual names occurring in $\mathcal{O}$. We denote by $\mathcal{V}_\mathcal{O}$ the closure under the involutive negation $x \mapsto 1 - x$ of the set of all truth degrees appearing in $\mathcal{O}$ (either in axioms or directly as truth constants), together with 0, 0.5, and 1. The size of this set is linear in the size of $\mathcal{O}$.

An interpretation $\mathcal{I}$ *satisfies* (or is a *model* of)

- the concept assertion $C(a) \bowtie 1$ if $C^\mathcal{I}(a^\mathcal{I}) \bowtie 1$;
- the GCI $C \sqsubseteq D \geqslant p$ iff $C^\mathcal{I}(d) \Rightarrow D^\mathcal{I}(d) \geqslant p$ for all $d \in \Delta^\mathcal{I}$;
- the role inclusion $r_1 \dots r_n \sqsubseteq r \geqslant p$ iff $(r_1 \dots r_n)^\mathcal{I}(d_0, d_n) \Rightarrow r^\mathcal{I}(d_0, d_n) \geqslant p$ for all $d_0, d_n \in \Delta^\mathcal{I}$, where
$$(r_1 \dots r_n)^\mathcal{I}(d_0, d_n) := \sup_{d_1, \dots, d_{n-1} \in \Delta^\mathcal{I}} \min_{i=1}^{n} r_i^\mathcal{I}(d_{i-1}, d_i);$$
- the disjoint role axiom $\mathsf{dis}(s_1, s_2) \geqslant p$ iff $\min\{s_1^\mathcal{I}(d,e), s_2^\mathcal{I}(d,e)\} \leqslant 1 - p$ for all $d, e \in \Delta^\mathcal{I}$;
- the reflexivity axiom $\mathsf{ref}(r) \geqslant p$ iff $r^\mathcal{I}(d,d) \geqslant p$ for all $d \in \Delta^\mathcal{I}$;
- an ontology if it satisfies all its axioms.

An ontology is *consistent* if it has a witnessed model.

Other common reasoning problems for FDLs, such as concept satisfiability and subsumption can be reduced to consistency in linear time [5]. For instance, the subsumption between C and D to degree p w.r.t. a TBox $\mathcal{T}$ and an RBox $\mathcal{R}$ is equivalent to the inconsistency of $(\{\overline{p} \to (C \to D)(a) < 1\}, \mathcal{T}, \mathcal{R})$, and the satisfiability of C to degree p w.r.t. $\mathcal{T}$ and $\mathcal{R}$ is equivalent to the consistency of $(\{\overline{p} \to C(a) \geqslant 1\}, \mathcal{T}, \mathcal{R})$.

Using the axioms previously introduced, it is possible to simulate other axioms that are common for $\mathcal{SROIQ}$ [4, 15] as follows:

- transitivity axioms $\mathsf{tra}(r) \geqslant p$ by $rr \sqsubseteq r \geqslant p$;
- symmetry axioms $\mathsf{sym}(r) \geqslant p$ by $r^- \sqsubseteq r \geqslant p$;
- asymmetry axioms $\mathsf{asy}(s) \geqslant p$ by $\mathsf{dis}(s, s^-) \geqslant p$;
- irreflexivity axioms $\mathsf{irr}(s) \geqslant p$ by $\exists s.\mathsf{Self} \sqsubseteq \neg\overline{p} \geqslant 1$;
- individual equality assertions $a \approx b$ by $\{b\}(a) \geqslant 1$; and
- individual inequality assertions $a \not\approx b$ by $\{b\}(a) < 1$.

Moreover, due to the expressivity of our language, we can express arbitrary order assertions as in [5, 11], even over negated roles. For example, the axiom $r(a,b) > \neg s(c,d)$ can be expressed as
$$\Big(\big(\exists r.\{b\}\big) \to \exists r_u.\big(\{c\} \sqcap \neg\exists s.\{d\}\big)\Big)(a) < 1.$$

2.3 Weighted Automata for Fuzzy Role Inclusions

To deal with complex role inclusions, in [16] an automata construction was developed that allows to break down inferences about long role chains into smaller steps. We recall the generalization of this construction to Gödel semantics from [11].

Definition 2.1 (WFA)**.** A *weighted finite automaton (WFA)* is a tuple $\mathbf{A} = (Q, \Sigma, q_{\mathsf{ini}}, \mathsf{wt}, q_{\mathsf{fin}})$, consisting of a non-empty set Q of *states*, a non-empty *input alphabet* Σ, an *initial state* $q_{\mathsf{ini}} \in Q$, a *transition weight function* $\mathsf{wt}\colon Q \times (\Sigma \cup \{\varepsilon\}) \times Q \to [0,1]$, and a *final state* $q_{\mathsf{fin}} \in Q$. Given an input word $w \in \Sigma^*$, a *run* of $\mathbf{A}$ on w is a non-empty sequence of pairs $\mathsf{r} = \{(w_i, q_i)\}_{0 \leqslant i \leqslant m}$ such that $(w_0, q_0) = (w, q_{\mathsf{ini}})$, $(w_m, q_m) = (\varepsilon, q_{\mathsf{fin}})$, and for each i, $1 \leqslant i \leqslant m$, it holds that $w_{i-1} = x_i w_i$ for some $x_i \in \Sigma \cup \{\varepsilon\}$. The *weight* of this run is $\mathsf{wt}(\mathsf{r}) := \min_{i=1}^{m} \mathsf{wt}(q_{i-1}, x_i, q_i)$. The *behavior* of $\mathbf{A}$ on w is defined as $(\|\mathbf{A}\|, w) := \sup_{\mathsf{r} \in \mathsf{runs}(\mathbf{A}, w)} \mathsf{wt}(\mathsf{r})$, where $\mathsf{runs}(\mathbf{A}, w)$ denotes the set of all runs of $\mathbf{A}$ on w.

We often denote by $q \xrightarrow{x,p} q' \in \mathbf{A}$ the fact that $\mathsf{wt}(q, x, q') = p$. Further, for a state q of $\mathbf{A}$, we denote by $\mathbf{A}^q$ the automaton resulting from $\mathbf{A}$ by making q the initial state. The following connection is a direct consequence of the definition of the behavior of a WFA.

Proposition 2.2. *Let* $\mathbf{A}$ *be a WFA over* Σ, $q \xrightarrow{x,p} q' \in \mathbf{A}$, *and* $w \in \Sigma^*$. *Then we have* $(\|\mathbf{A}^q\|, xw) \geqslant \min\{p, (\|\mathbf{A}^{q'}\|, w)\}$.

Let now $\mathcal{O} = (\mathcal{A}, \mathcal{T}, \mathcal{R}_h \cup \mathcal{R}_a)$ be a G-$\mathcal{SROIQ}$ ontology. In order to characterize the complex role inclusions in $\mathcal{R}_h$, [11,16] construct a family of WFA $(\mathbf{A}_r)_{r \in \mathsf{rol}(\mathcal{O})}$ that "read" role chains, i.e. words over the input alphabet $\Sigma := \mathsf{rol}(\mathcal{O})$. Intuitively, the automaton $\mathbf{A}_r$ recognizes all role chains that imply the role r via the role hierarchy (with associated truth values). The size of these automata may be exponential in the size of $\mathcal{R}_h$.

Lemma 2.3 ([11])**.** *Let* $\mathcal{O} = (\mathcal{A}, \mathcal{T}, \mathcal{R}_h \cup \mathcal{R}_a)$ *be an ontology and* $(\mathbf{A}_r)_{r \in \mathsf{rol}(\mathcal{O})}$ *be the family of WFA constructed in [11], and* $\mathcal{I}$ *be an interpretation. Then* $\mathcal{I}$ *satisfies* $\mathcal{R}_h$ *iff for every* $r \in \mathsf{rol}(\mathcal{O})$, *every* $w \in \mathsf{rol}(\mathcal{O})^+$, *and all* $d, e \in \Delta^{\mathcal{I}}$, *we have*

$$\min\left\{(\|\mathbf{A}_r\|, w), w^{\mathcal{I}}(d,e)\right\} \leqslant r^{\mathcal{I}}(d,e).$$

A *mirrored copy* $\mathbf{A}^-$ is constructed from such a WFA $\mathbf{A}$ over $\mathsf{rol}(\mathcal{O})$ by exchanging initial and final states, and replacing each transition $q \xrightarrow{x,p} q'$ by $q' \xrightarrow{x^-,p} q$, where $\varepsilon^- := \varepsilon$. By the construction in [11], each automaton $\mathbf{A}_{r^-}$ is a mirrored copy of $\mathbf{A}_r$.

Proposition 2.4. *Let* $\mathbf{A}$ *be a WFA over* Σ, $\mathbf{A}'$ *be a mirrored copy of* $\mathbf{A}$, *and* $w \in \Sigma^*$. *Then we have* $(\|\mathbf{A}\|, w) = (\|\mathbf{A}'\|, w^-)$.

We define now the relation $\sqsubseteq_p$, which can be understood as the "transitive closure" of the simple role inclusions in $\mathcal{R}$: we set $r \sqsubseteq_p s$ iff p is the supremum of the values $\min\{p_1, \ldots, p_n\}$ over all sequences of simple role inclusions $r \sqsubseteq r_1 \geqslant p_1, \ldots, r_{n-1} \sqsubseteq s \geqslant p_n$ in $\mathcal{R}$. Note in particular that $r \sqsubseteq_1 r$ holds for every simple role r. The following is a special case of Lemma 2.3.

Proposition 2.5 ([11])**.** *For a simple role* r *and* $w \in \mathsf{rol}(\mathcal{O})^*$, *we have*

$$(\|\mathbf{A}_r\|, w) = \begin{cases} p & \text{if } w = s \in \mathsf{rol}(\mathcal{O}) \text{ and } s \sqsubseteq_p r, \\ 0 & \text{otherwise.} \end{cases}$$

3 From Models to Tableaux

In this section, we extend the classical tableau construction from [15, 17] with the ideas developed in [5, 11] to produce a reasoning algorithm for G-$\mathcal{SROIQ}$. For the rest of this paper, $\mathcal{O} = (\mathcal{A}, \mathcal{T}, \mathcal{R})$ is an arbitrary, but fixed, G-$\mathcal{SROIQ}$ ontology. To simplify the presentation, we will assume in the following that $\mathcal{O}$ contains no existential restrictions. These concepts can be handled in the same way as value restrictions, by dualizing the constructions, i.e. replacing $\leqslant$ with $\geqslant$ and $\Rightarrow$ with min.

The set $\mathsf{sub}(C)$ of (extended) sub-concepts of a concept C contains at least C and $\neg C$, as well as the concepts of the following set, which is defined recursively:

$$\begin{cases} \mathsf{sub}(D) & \text{if } C = \neg D, \\ \mathsf{sub}(D) \cup \mathsf{sub}(E) & \text{if } C \in \{D \sqcap E, D \to E\}, \\ \mathsf{sub}(D) \cup \{\forall \mathbf{A}_r^q.D, \neg\forall \mathbf{A}_r^q.D \mid q \text{ is a state of } \mathbf{A}_r\} & \text{if } C = \forall r.D, \\ \mathsf{sub}(D) \cup \{\geqslant m\, r.D, \neg\geqslant m\, r.D \mid 1 \leqslant m \leqslant n-1\} & \text{if } C = \geqslant n\, r.D, \\ \emptyset & \text{otherwise.} \end{cases}$$

In the following, we consider $\neg\neg C$ to be equal to C. The set $\mathsf{sub}(\mathcal{O})$ of sub-concepts of $\mathcal{O}$ is defined as the union of all sets $\mathsf{sub}(C), \mathsf{sub}(D)$ for all axioms $C \sqsubseteq D \geqslant p \in \mathcal{T}$ and $C(a) \bowtie 1 \in \mathcal{A}$. The size of $\mathsf{sub}(\mathcal{O})$ is exponential in the size of the role hierarchy (due to the automata $\mathbf{A}_r$) and exponential in the largest number appearing in number restrictions in $\mathcal{O}$ (if such numbers are given in binary encoding).

The semantics of the newly introduced expressions of form $\forall \mathbf{A}.C$ is defined by

$$(\forall \mathbf{A}.C)^{\mathcal{I}}(d) := \inf_{w \in \mathsf{rol}(\mathcal{O})^*} \inf_{e \in \Delta^{\mathcal{I}}} \min\{(\|\mathbf{A}\|, w), w^{\mathcal{I}}(d,e)\} \Rightarrow C^{\mathcal{I}}(e),$$

where $\varepsilon^{\mathcal{I}}(d,e) := 1$ if $d = e$, and $\varepsilon^{\mathcal{I}}(d,e) := 0$ otherwise (cf. Lemma 2.3).

Our construction uses order assertions to deal with relations between the values of concepts. The domain of these assertions is the order structure

$$\mathcal{U}(\Delta) := \mathcal{V}_{\mathcal{O}} \cup \big\{C(x) \mid C \in \mathsf{sub}(\mathcal{O}),\ x \in \Delta\big\} \cup \big\{r(x,y), \neg r(x,y) \mid r \in \mathsf{rol}(\mathcal{O}),\ x, y \in \Delta\big\},$$

where Δ is a set of *nodes*, $\mathsf{inv}(C(x)) := \neg C(x)$, and $\mathsf{inv}(r(x,y)) := \neg r(x,y)$. To simplify dealing with inverse roles, we will treat the expressions $r(x,y)$ and $r^-(y,x)$ as if they were the same; this is clearly in line with the semantics of inverse roles.

As in [15, 17], we define the notion of a *tableau*, which is an abstract version of a model of $\mathcal{O}$ that may still be infinite, but allows us to simplify the semantics: for example, all complex role inclusions are handled by three simple rules for the behavior of the concepts $\forall \mathbf{A}.C$. Recall that we use the term "satisfiability" only w.r.t. total preorders over $\mathcal{U}(\Delta)$, and not w.r.t. full DL interpretations.

Definition 3.1 (Tableau)**.** A *tableau* for $\mathcal{O}$ is a pair $(\Delta, \mathcal{A}^*)$, where Δ is a non-empty set of *nodes* and $\mathcal{A}^*$ is a satisfiable set of order assertions over $\mathcal{U}(\Delta)$ such that the following conditions are satisfied, for all $x, y \in \Delta$, $C, D \in \mathsf{sub}(\mathcal{O})$, $r, s \in \mathsf{rol}(\mathcal{O})$, and $a \in \mathsf{ind}(\mathcal{O})$:

(T1) If $\overline{p}(x)$ occurs in $\mathcal{A}^*$, then $\mathcal{A}^*$ entails $\overline{p}(x) = p$.
(T2) If $(\exists r.\mathsf{Self})(x)$ occurs in $\mathcal{A}^*$, then $\mathcal{A}^*$ entails $(\exists r.\mathsf{Self})(x) = r(x,x)$.
(T3) If $(\neg C)(x)$ occurs in $\mathcal{A}^*$, then $C(x)$ also occurs in $\mathcal{A}^*$.
(T4) If $(C \sqcap D)(x)$ occurs in $\mathcal{A}^*$, then $\mathcal{A}^*$ entails $(C \sqcap D)(x) = \min\{C(x), D(x)\}$.
(T5) If $(C \to D)(x)$ occurs in $\mathcal{A}^*$, then $\mathcal{A}^*$ entails $(C \to D)(x) = C(x) \Rightarrow D(x)$.
(T6) If $(\forall r.C)(x)$ occurs in $\mathcal{A}^*$, then there is a $y \in \Delta$ such that $\mathcal{A}^*$ entails

$$(\forall r.C)(x) \geqslant r(x,y) \Rightarrow C(y).$$

(T7) If $(\forall r.C)(x)$ occurs in $\mathcal{A}^*$, then $\mathcal{A}^*$ entails $(\forall r.C)(x) \leqslant (\forall \mathbf{A}_r.C)(x)$.
(T8) If $(\forall \mathbf{A}^q.C)(x)$ and $r(x,y)$ occur in $\mathcal{A}^*$ and $q \xrightarrow{r,p} q' \in \mathbf{A}$, then $\mathcal{A}^*$ entails

$$(\forall \mathbf{A}^q.C)(x) \leqslant \min\{p, r(x,y)\} \Rightarrow (\forall \mathbf{A}^{q'}.C)(y).$$

(T9) If $(\forall \mathbf{A}^q.C)(x)$ occurs in $\mathcal{A}^*$ and $q \xrightarrow{\varepsilon,p} q' \in \mathbf{A}$, then $\mathcal{A}^*$ entails $(\forall \mathbf{A}^q.C)(x) \leqslant p \Rightarrow (\forall \mathbf{A}^{q'}.C)(x)$.

(T10) If $(\forall \mathbf{A}^q.C)(x)$ occurs in $\mathcal{A}^*$ and q is final, then $\mathcal{A}^*$ entails $(\forall \mathbf{A}^q.C)(x) \leqslant C(x)$.

(T11) If $(\geqslant n\, r.C)(x)$ occurs in $\mathcal{A}^*$, then there are at least n elements $y \in \Delta$ for which $\mathcal{A}^*$ entails $(\geqslant n\, r.C)(x) \leqslant \min\{r(x,y), C(y)\}$.

(T12) If $(\geqslant n\, r.C)(x)$ occurs in $\mathcal{A}^*$, then there are at most $n-1$ elements $y \in \Delta$ for which $\mathcal{A}^*$ entails $(\geqslant n\, r.C)(x) < \min\{r(x,y), C(y)\}$.

(T13) If $(\geqslant n\, r.C)(x)$ and $r(x,y)$ occur in $\mathcal{A}^*$, then $\mathcal{A}^*$ entails either

$$(\geqslant n\, r.C)(x) \geqslant \min\{r(x,y), C(y)\} \text{ or } (\geqslant n\, r.C)(x) < \min\{r(x,y), C(y)\}.$$

(T14) If $\{a\}(x)$ occurs in $\mathcal{A}^*$, then $\mathcal{A}^*$ entails either $\{a\}(x) \geqslant 1$ or $\{a\}(x) \leqslant 0$.

(T15) There is exactly one $x_a \in \Delta$ such that $\mathcal{A}^*$ entails $\{a\}(x_a) \geqslant 1$.

(T16) If $r(x,y)$ occurs in $\mathcal{A}^*$, then $\mathcal{A}^*$ entails $r(x,y) = r^-(y,x)$.

(T17) If $C(a) \bowtie 1 \in \mathcal{A}$, then $\mathcal{A}^*$ entails $C(x_a) \bowtie 1$.

(T18) If $C \sqsubseteq D \geqslant p \in \mathcal{T}$, then $\mathcal{A}^*$ entails $C(x) \Rightarrow D(x) \geqslant p$.

(T19) If $r \sqsubseteq s \geqslant p \in \mathcal{R}$, then $\mathcal{A}^*$ entails $r(x,y) \Rightarrow s(x,y) \geqslant p$.

(T20) If $\mathsf{ref}(r) \geqslant p \in \mathcal{R}$, then $\mathcal{A}^*$ entails $r(x,x) \geqslant p$.

(T21) If $\mathsf{dis}(r,s) \geqslant p \in \mathcal{R}$, then $\mathcal{A}^*$ entails $\min\{r(x,y), s(x,y)\} \leqslant 1-p$.

(T22) $\mathcal{A}^*$ entails $r_u(x,y) \geqslant 1$.

The main differences to the classical tableau conditions of [15,17], in addition to the use of order assertions, are the following:

- We do not need a dedicated condition for negation other than to add the relevant subconcept to the tableau. The semantics of the involutive negation is handled by the conditions that define the order structure $\mathcal{U}(\Delta)$.
- We do not internalize the ABox, TBox, or the universal role, which is why we need to include dedicated conditions for them.
- Although we do not consider existential restrictions or at-most number restrictions here, the corresponding conditions from [15, 17] can also be found in Definition 3.1. The reason for this is that fuzzy value restrictions and at-least number restrictions additionally exhibit a behavior similar to that of (classical) existential restrictions and at-most number restrictions. For example, due to the supremum used in its semantics, the value of an at-least restriction $\geqslant n\, r.C$ at a node x also imposes an *upper* bound (namely $n-1$) on the number of nodes y for which $\min\{r(x,y), C(y)\}$ can exceed this value. Similarly, a fuzzy value restriction also enforces the existence of a successor node (classically the behavior of an existential restriction) due to our witnessing conditions.

The following lemma shows that it suffices to construct a countable tableau to show that $\mathcal{O}$ is consistent. The requirement on the cardinality of the tableau is the main difference to the corresponding result in [15]. It is due to the necessity of finding enough values to instantiate all relevant concepts and roles. We could also use a tableau having the same cardinality as the real numbers, but a countable tableau suffices for our purposes (see Lemma 4.3).

Lemma 3.2. *If $\mathcal{O}$ is consistent, then there is a tableau, and if there is a countable tableau, then $\mathcal{O}$ is consistent.*

Proof. Let first $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$ be a model of $\mathcal{O}$. We construct the tableau $(\Delta^\mathcal{I}, \mathcal{A}^*)$, where $\mathcal{A}^*$ is the set of all order assertions $u \bowtie v$ such that $u, v \in \mathcal{U}(\Delta^\mathcal{I})$ with $u^\mathcal{I} \bowtie v^\mathcal{I}$, where $p^\mathcal{I} := p$, $C(d)^\mathcal{I} := C^\mathcal{I}(d)$, $r(d,e)^\mathcal{I} := r^\mathcal{I}(d,e)$, and $(\neg r(d,e))^\mathcal{I} := 1 - r^\mathcal{I}(d,e)$. By construction, $\mathcal{A}^*$ is satisfiable, and hence we only need to prove that it satisfies the conditions (T1)–(T22) of Definition 3.1. This can be verified by a trivial, but lengthy, case analysis, very similar to the proof of the corresponding result in [11]. For example, if $(\forall \mathbf{A}^q.C)(d)$ occurs in $\mathcal{A}^*$ and q is final, then we know that $(\|\mathbf{A}^q\|, w) = 1$ for every word $w \in \mathsf{rol}(\mathcal{O})^*$, and hence

$$(\forall \mathbf{A}^q.C)^\mathcal{I}(d) = \inf_{w \in \mathsf{rol}(\mathcal{O})^*} \inf_{e \in \Delta^\mathcal{I}} w^\mathcal{I}(d,e) \Rightarrow C^\mathcal{I}(e) \leqslant \varepsilon^\mathcal{I}(d,d) \Rightarrow C^\mathcal{I}(d) = C^\mathcal{I}(d).$$

By our construction, this means that $(\forall \mathbf{A}^q.C)(d) \leqslant C(d)$ is contained in $\mathcal{A}^*$, and thus (T10) is satisfied.

For the second part of the lemma, let $(\Delta, \mathcal{A}^*)$ be a tableau where Δ is countable, and hence $\mathcal{U}(\Delta)$ and $\mathcal{A}^*$ are also countable. Since $\mathcal{A}^*$ is satisfiable, it has a model $\preceq_*$. Since this model is an element of $\mathsf{order}(\mathcal{U}(\Delta))$, there must exist a mapping $v \colon \mathcal{U}(\Delta) \to [0,1]$ with the following properties:

(P1) for all $p \in \mathcal{V}_\mathcal{O}$, we have $v(p) = p$;
(P2) for all $\alpha, \beta \in \mathcal{U}(\Delta)$, we have $v(\alpha) \leqslant v(\beta)$ iff $\alpha \preceq_* \beta$; and
(P3) for all $\alpha \in \mathcal{U}(\Delta)$, we have $v(\mathsf{inv}(\alpha)) = 1 - v(\alpha)$.

We now define the interpretation $\mathcal{I}$ as follows, for all $a \in \mathsf{ind}(\mathcal{O})$, $A \in \mathsf{sub}(\mathcal{O}) \cap \mathsf{N_C}$, and $x \in \Delta$:

- $\Delta^\mathcal{I} := \Delta$;
- $a^\mathcal{I} := a$; and
- $A^\mathcal{I}(x) := v(A(x))$ if $A(x)$ occurs in $\mathcal{A}^*$, and $A^\mathcal{I}(x) := 0$ otherwise.

The interpretation of all other individual names and concept names can be fixed arbitrarily. For the role names $r \in \mathsf{rol}(\mathcal{O}) \cap \mathsf{N_R}$, we first define a "simple" interpretation $\mathcal{I}_0$ as follows: $r^{\mathcal{I}_0}(x,y) := v(r(x,y))$ if $r(x,y)$ occurs in $\mathcal{A}^*$, and $r^{\mathcal{I}_0}(x,y) := 0$ otherwise. By (T16) and (P2), for every inverse role $r^- \in \mathsf{rol}(\mathcal{O})$ for which $r^-(x,y)$ occurs in $\mathcal{A}^*$, we know that $r(y,x)$ also occurs in $\mathcal{A}^*$ and we have $(r^-)^{\mathcal{I}_0}(x,y) = r^{\mathcal{I}_0}(y,x) = v(r(y,x)) = v(r^-(x,y))$, similar to the definition of $r^{\mathcal{I}_0}$. We now use the automaton $\mathbf{A}_r$ to "complete" $\mathcal{I}_0$ with additional links as follows: we set

$$r^\mathcal{I}(x,y) := \sup_{w \in \mathsf{rol}(\mathcal{O})^+} \min\{(\|\mathbf{A}_r\|, w), w^{\mathcal{I}_0}(x,y)\} \tag{1}$$

for all $x, y \in \Delta$. Note that this expression is equal to $r^{\mathcal{I}_0}(x,y)$ if r is simple: by Proposition 2.5, we have $(\|\mathbf{A}_r\|, s) = p$ whenever $s \sqsubseteq_p r$, $(\|\mathbf{A}_r\|, r) = 1$, and $(\|\mathbf{A}_r\|, w) = 0$ for all other words w, and moreover (T19) yields

$$\min\{(\|\mathbf{A}_r\|, r), r^{\mathcal{I}_0}(x,y)\} = r^{\mathcal{I}_0}(x,y) \geqslant \min\{p, s^{\mathcal{I}_0}(x,y)\} = \min\{(\|\mathbf{A}_r\|, s), s^{\mathcal{I}_0}(x,y)\}.$$

The expression (1) can also be used to evaluate inverse roles due to the semantics of role chains, the fact that $\mathbf{A}_{r^-}$ is a mirrored copy of $\mathbf{A}_r$ [11], and Proposition 2.4. Finally, for the universal role r_u, we have $r_u^{\mathcal{I}_0}(x,y) = 1$ due to (T22). By the construction of $\mathbf{A}_{r_u}$ [11], we have $(\|\mathbf{A}_{r_u}\|, r_u) = 1$, and hence the expression (1) also holds for the universal role.

To show that $\mathcal{I}$ is a model of $\mathcal{O}$, we first prove the following claim by induction on the structure of C:

$$\text{For all } x \in \Delta \text{ and } C \in \mathsf{sub}(\mathcal{O}) \text{ for which } C(x) \text{ occurs in } \mathcal{A}^*, \text{ we have } C^\mathcal{I}(x) = v(C(x)). \tag{2}$$

For most concept constructors, this easily follows from the conditions in Definition 3.1 and the fact that $\preceq_*$, and hence v, satisfies all entailments of $\mathcal{A}^*$.

For negation, assume that $(\neg C)(x)$ occurs in $\mathcal{A}^*$. We get

$$(\neg C)^\mathcal{I}(x) = 1 - C^\mathcal{I}(x) = 1 - v(C(x)) = v((\neg C)(x))$$

by (T3), (P3), and the induction hypothesis.

Assume now that $(\forall r.C)(x)$ occurs in $\mathcal{A}^*$. By (T11), there must be a $y_0 \in \Delta$ such that $v((\forall r.C)(x)) \geqslant v(r(x, y_0)) \Rightarrow v(C(y_0)) = r^{\mathcal{I}_0}(x, y_0) \Rightarrow C^{\mathcal{I}}(y_0) \geqslant r^{\mathcal{I}}(x, y_0) \Rightarrow C^{\mathcal{I}}(y_0)$. Hence, y_0 can act as a witness for $(\forall r.C)^{\mathcal{I}}(x)$ if we can show that the latter implication is $\geqslant v((\forall r.C)(x))$ for all elements $y \in \Delta$. For this purpose, we consider the remaining tableau conditions for value restrictions. By (T7), we get

$$
\begin{aligned}
r^{\mathcal{I}}(x, y) \Rightarrow C^{\mathcal{I}}(y) &= \Big(\sup_{w \in \mathsf{rol}(\mathcal{O})^+} \min\{(\|\mathbf{A}_r\|, w), w^{\mathcal{I}_0}(x, y)\} \Big) \Rightarrow C^{\mathcal{I}}(y) \\
&= \inf_{w \in \mathsf{rol}(\mathcal{O})^+} \min\{(\|\mathbf{A}_r\|, w), w^{\mathcal{I}_0}(x, y)\} \Rightarrow C^{\mathcal{I}}(y) \\
&\overset{(*)}{\geqslant} v((\forall \mathbf{A}_r.C)(x)) \\
&\geqslant v((\forall r.C)(x)).
\end{aligned}
$$

as required, if we can show $(*)$, i.e. it remains to show that

$$\min\{(\|\mathbf{A}_r\|, w), w^{\mathcal{I}_0}(x, y)\} \Rightarrow C^{\mathcal{I}}(y) \geqslant v((\forall \mathbf{A}_r.C)(x))$$

holds for all $w = r_1 \dots r_n \in \mathsf{rol}(\mathcal{O})^+$. If y is not connected to x, i.e. we have $w^{\mathcal{I}_0}(x, y) = 0$ for all such w, then this is trivial. The claim for all other y can be shown exactly as in [10, Section 4.1].

Consider now a number restriction for which $({\geqslant} n\, r.C)(x)$ occurs in $\mathcal{A}^*$. Recall that r must be simple, and hence we have $r^{\mathcal{I}} = r^{\mathcal{I}_0}$. If, for some $y \in \Delta$, $r(x, y)$ does not occur in $\mathcal{A}^*$, then we know that $v(({\geqslant} n\, r.C)(x)) \geqslant 0 = \min\{r^{\mathcal{I}}(x, y), C^{\mathcal{I}}(y)\}$. By (T12), (T13), and the induction hypothesis, we know that there are at most $n - 1$ elements $y \in \Delta$ for which $\min\{r^{\mathcal{I}}(x, y), C^{\mathcal{I}}(y)\}$ is strictly greater than $v(({\geqslant} n\, r.C)(x))$. This means that for any n different elements $y_1, \dots, y_n \in \Delta$, we have $v(({\geqslant} n\, r.C)(x)) \geqslant \min_{i=1}^{n} \min\{r^{\mathcal{I}}(x, y_i), C^{\mathcal{I}}(y_i)\}$. Hence, to prove $v(({\geqslant} n\, r.C)(x)) = ({\geqslant} n\, r.C)^{\mathcal{I}}(x)$, it suffices to find n witnessing elements where the latter inequation holds with $=$ instead of only $\geqslant$. This follows directly from (T11). □

4 A Tableau Algorithm

The construction of a possibly infinite tableau for $\mathcal{O}$ is hardly a decision procedure for consistency. For that purpose, we need to appropriately lift the notion of blocking [15,17] to our sets of order assertions, in order to arrive at a *finite* structure. We also need to take a more fine-grained view at the structure of a tableau. First, we designate a subset $\Delta_o \subseteq \Delta$ as *nominal nodes*. Furthermore, in the tableau algorithm we need to introduce new individual names that do not occur in $\mathsf{ind}(\mathcal{O})$, and we assume that the corresponding nominals are contained in $\mathsf{sub}(\mathcal{O})$. Instead of allowing to connect arbitrary pairs of individuals with roles, we will maintain a binary *neighbor relation* $\mathcal{N}$ on Δ. For a node x, we consider its *neighborhood*

$$\mathcal{N}(x) := \{x\} \cup \big\{y \mid (x, y) \in \mathcal{N} \text{ or } (y, x) \in \mathcal{N}\big\}.$$

Further, if $(x, y) \in \mathcal{N}$, then y is a *successor* of x, and x is a *predecessor* of y. *Ancestor* is the transitive closure of predecessor, and *descendant* the transitive closure of successor. Finally, instead of a global set $\mathcal{A}^*$, we will maintain for each node $x \in \Delta$ a local set of order assertions $\mathcal{L}(x)$ over the localized order structure

$$
\begin{aligned}
\mathcal{U}(x) := \mathcal{V}_{\mathcal{O}} &\cup \big\{C(a) \mid C \in \mathsf{sub}(\mathcal{O}),\ a \in \Delta_o\big\} \cup \big\{r(a, b), \neg r(a, b) \mid r \in \mathsf{rol}(\mathcal{O}),\ a, b \in \Delta_o\big\} \\
&\cup \big\{C(y) \mid C \in \mathsf{sub}(\mathcal{O}),\ y \in \mathcal{N}(x)\big\} \cup \big\{r(x, y), \neg r(x, y) \mid r \in \mathsf{rol}(\mathcal{O}),\ y \in \mathcal{N}(x)\big\},
\end{aligned}
$$

where inv is defined as for $\mathcal{U}(\Delta)$. It follows that $\mathcal{U}(x)$ is a subset of $\mathcal{U}(\Delta)$ if all nominal nodes are neighbors of each other. However, order assertions over $\mathcal{U}(x)$ can only contain information about the concepts at neighbors of x and role connections between x and its neighbors; additionally, information about concepts and roles at nominal nodes is shared by all nodes.

Definition 4.1. A *completion graph* for $\mathcal{O}$ is a tuple $\mathcal{G} = (\Delta, \mathcal{N}, \mathcal{L}, \neq)$, where Δ is a finite set of *nodes*, $\mathcal{N}$ is a binary *neighbor* relation on Δ, $\mathcal{L}$ is a *labeling function* that assigns each node $x \in \Delta$ a set $\mathcal{L}(x)$ of oder assertions over $\mathcal{U}(x)$, and $\neq$ is a binary relation on Δ.

The relation $\neq$ indicates which nodes must be kept different. If $x \neq y$ does not hold, then we can merge them, which may be necessary in order to satisfy some number restrictions. We denote by $\doteq$ the complement of $\neq$; if $x \doteq y$, then this does not mean that x and y will necessarily be merged, but only that it is possible to do so.

Nominal nodes and blockable nodes. The set Δ_o is not fixed, but rather defined as the set of all $x \in \Delta$ such that $\mathcal{L}(x)$ entails $\{a\}(x) \geqslant 1$ for some $a \in \mathsf{N_I}$. Recall that we may need to introduce more such individual names in the construction of the completion graph. All nodes in $\Delta \setminus \Delta_o$ are called *blockable nodes*. The idea is that nominal nodes may be arbitrarily interconnected in $\mathcal{N}$, but blockable nodes always form a tree structure among themselves. Each nominal node may be the root of such a tree, and additionally all blockable nodes may have $\mathcal{N}$-successors that are nominal nodes.

Rule applications. The *initial completion graph* for $\mathcal{O}$ is $\mathcal{G}_0 := (\Delta_0, \mathcal{N}_0, \mathcal{L}_0, \neq_0)$, where $\Delta_0 := \mathsf{ind}(\mathcal{O})$, $\mathcal{N}_0 := \Delta_0 \times \Delta_0$, $\mathcal{L}_0(a)$ consists of all assertions containing a, and $\neq_0$ is empty. Starting from $\mathcal{G}_0$, the tableau algorithm nondeterministically applies the rules in Tables 2–6, which modify the completion graph according to the semantics of concepts and axioms. To guarantee termination, it is important that we only add order assertions to a label if they are not already entailed by this label. Since labels of nodes may refer to neighboring nodes, we need the dedicated rule ($\rightsquigarrow$) to ensure that labels of neighbors are synchronized.

When we extend the neighborhood of x by adding a new node y with $(x, y) \in \mathcal{N}$, $\mathcal{L}(x)$ still contains order assertions over the (extended) order structure $\mathcal{U}(x)$. The same holds when we introduce a new nominal node using the rule (NN).

We now explain all relevant notions used in the tableau rules, most of which are suitably lifted variants of the definitions in [15, 17]. For the following exposition, let $\mathcal{G} = (\Delta, \mathcal{N}, \mathcal{L}, \neq)$ be an arbitrary completion graph produced during the tableau algorithm.

Complete and clash-free. Our completion graph contains a *clash* if one of the following conditions holds:

- For some node $x \in \Delta$, the set $\mathcal{L}(x)$ is unsatisfiable.
- For some $(\geqslant n\, r.C) \in \mathsf{sub}(\mathcal{O})$, there are nodes $x, y_1, \ldots, y_n \in \Delta$, such that $\mathcal{L}(x)$ entails $(\geqslant n\, r.C)(x) < \min\{r(x, y_i), C(y_i)\}$, $1 \leqslant i \leqslant n$, and $y_i \neq y_j$, $1 \leqslant i < j \leqslant n$.
- For some $a \in \mathsf{N_I}$, there are nodes $x, y \in \Delta$ such that $x \neq y$, $\mathcal{L}(x)$ entails $\{a\}(x) \geqslant 1$, and $\mathcal{L}(y)$ entails $\{a\}(y) \geqslant 1$.

A completion graph is *complete* if it contains a clash or none of the tableau rules are applicable.

If the tableau rules can be applied to $\mathcal{G}_0$ such that a complete and clash-free completion graph is obtained, then the algorithm has successfully proven the consistency of $\mathcal{O}$. If we obtain a clash, then either we have made the wrong choices in the rule applications, or $\mathcal{O}$ is inconsistent.

Blocking and safe neighbors. We adapt the notion of blocking from [17] to sets of order assertions. A node x is *directly blocked* if it has ancestors x', y, and y' such that

- $(x', x), (y', y) \in \mathcal{N}$;

Table 2: The tableau rules for the propositional constructors

$(\overline{p})$	If $\overline{p}(x)$ occurs in $\mathcal{L}(x)$, then add $\overline{p}(x) = p$ to $\mathcal{L}(x)$ unless it is already entailed.
(Self)	If $(\exists r.\mathsf{Self})(x)$ occurs in $\mathcal{L}(x)$, then then add $(\exists r.\mathsf{Self})(x) = r(x,x)$ to $\mathcal{L}(x)$ unless it is already entailed.
$(\neg)$	If $(\neg C)(x)$ occurs in $\mathcal{L}(x)$, then add $C(x) \leqslant C(x)$ to $\mathcal{L}(x)$.
$(\sqcap)$	If $(C \sqcap D)(x)$ occurs in $\mathcal{L}(x)$ and $\mathcal{L}(x)$ does not entail $(C \sqcap D)(x) = \min\{C(x), D(x)\}$, then add either • $(C \sqcap D)(x) \leqslant C(x)$ and $(C \sqcap D)(x) = D(x)$, or • $(C \sqcap D)(x) = C(x)$ and $(C \sqcap D)(x) \leqslant D(x)$ to $\mathcal{L}(x)$.
$(\rightarrow)$	If $(C \rightarrow D)(x)$ occurs in $\mathcal{L}(x)$ and $\mathcal{L}(x)$ does not entail $(C \rightarrow D)(x) = C(x) \Rightarrow D(x)$, then add either • $C(x) \leqslant D(x)$ and $(C \rightarrow D)(x) \geqslant 1$, or • $C(x) > D(x)$ and $(C \rightarrow D)(x) = D(x)$ to $\mathcal{L}(x)$.
(o)	If $\{a\}(x)$ occurs in $\mathcal{L}(x)$, then add either $\{a\}(x) \geqslant 1$ or $\{a\}(x) \leqslant 0$ to $\mathcal{L}(x)$ unless one of them is already entailed.
$(o_{\leqslant})$	If for some $a \in \mathsf{N_I}$ there are two nodes x, y such that $\mathcal{L}(x)$ entails $\{a\}(x) \geqslant 1$, $\mathcal{L}(y)$ entails $\{a\}(y) \geqslant 1$, and $x \doteq y$, then merge x into y.

- x, y, and all nodes on the path from y to x are blockable;
- for all order assertions ϕ over $\mathcal{U}(\Delta)$ involving only the nodes x and x', we have that $\mathcal{L}(x)$ entails ϕ iff $\mathcal{L}(y)$ entails $\sigma(\phi)$, where σ replaces x by y and x' by y'.

In this case, we say that y *blocks* x. A node is *blocked* if it is directly blocked or it is blockable and its predecessor is blocked, i.e. it is *indirectly blocked.*

The rules $(\forall)$, $(\geqslant)$, and (NN) are called *generating*, and the rules $(\geqslant_{\leqslant})$, $(o_{\leqslant})$ and $(\mathsf{NN}_{\leqslant})$ are called *shrinking.* Note that generating rules are not applicable to any blocked nodes, but the other rules may be applied to all nodes. The reason for this is that, due to inverse roles, by applying these rules to blocked nodes, the order assertions at unblocked nodes may change, possibly leading to a clash or the breaking of a blocking relation.

A neighbor y of a node x is *safe* if (i) x is blockable, or (ii) x is a nominal node and y is not blocked. The reason for this definition is that only safe neighbors really count for the satisfaction of the witnessing conditions for value and number restrictions, since a blocked predecessor of a nominal node does not correspond to an individual in the tableau that will be constructed in Lemma 4.3.

Merging and pruning. We sometimes need to merge nodes in order to satisfy nominals or number restrictions. When we merge y into x, we replace all occurrences of y in $\mathcal{L}(y)$ by x and add these assertions to $\mathcal{L}(x)$. Additionally, we modify the neighborhood of x such that it inherits all neighbors of y, and then remove y (and all blockable subtrees) from the completion graph. For this reason, we also call x a *direct heir* of y. Formally, to *merge y into x*, we do the

Table 3: The tableau rules for inverse roles, ontology axioms, and transfer of order assertions between neighbors

(r^-)	If $r(x,y)$ occurs in $\mathcal{L}(x)$, then add $r(x,y) = r^-(y,x)$ to $\mathcal{L}(x)$ unless it is already entailed.
$(\sqsubseteq_\mathcal{T})$	If $C \sqsubseteq D \geqslant p \in \mathcal{T}$, then add either $C(x) \leqslant D(x)$ or $p \leqslant D(x)$ to $\mathcal{L}(x)$ unless one of them is already entailed.
$(\sqsubseteq_\mathcal{R})$	If $r \sqsubseteq s \geqslant p \in \mathcal{R}$ and $y \in \mathcal{N}(x)$, then add either $r(x,y) \leqslant s(x,y)$ or $p \leqslant s(x,y)$ to $\mathcal{L}(x)$ unless one of them is already entailed.
(ref)	If $\mathsf{ref}(r) \geqslant p \in \mathcal{R}$, then add $r(x,x) \geqslant p$ to $\mathcal{L}(x)$ unless it is already entailed.
(dis)	If $\mathsf{dis}(r,s) \geqslant p \in \mathcal{R}$ and $y \in \mathcal{N}(x)$, then add either $r(x,y) \leqslant 1-p$ or $s(x,y) \leqslant 1-p$ to $\mathcal{L}(x)$ unless one of them is already entailed.
(r_u)	If $y \in \mathcal{N}(x)$, then add $r_u(x,y) \geqslant 1$ to $\mathcal{L}(x)$ unless it is already entailed.
$(\rightsquigarrow)$	If 1. $\mathcal{L}(x)$ entails $\alpha \lhd \beta$ and 2. there is a $y \in \Delta$ such that $\{\alpha, \beta\} \subseteq \mathcal{U}(y)$, then add $\alpha \lhd \beta$ to $\mathcal{L}(y)$ unless it is already entailed.

following:

1. For all nodes $z \in \Delta$ with $(z,y) \in \mathcal{N}$, we replace (z,y) by (z,x) in $\mathcal{N}$.
2. For all nominal nodes $z \in \Delta$ with $(y,z) \in \mathcal{N}$, we replace (y,z) by (x,z) in $\mathcal{N}$.
3. We collect all blockable nodes $z \in \Delta$ with $(y,z) \in \mathcal{N}$ into the set Z; these nodes will be removed from the completion graph together with y.
4. For all order assertions $\phi \in \mathcal{L}(y)$ that do not involve nodes from Z, we add $\sigma(\phi)$ to $\mathcal{L}(x)$, where σ is a substitution that replaces y by x.
5. For all nodes $z \in \Delta$ with $y \not\doteq z$, we add $x \not\doteq z$.
6. We prune y from the completion graph,

where the operation of *pruning* y is defined recursively as follows:

1. For all nodes $z \in \Delta$ with $(y,z) \in \mathcal{N}$, we remove (y,z) from $\mathcal{N}$ and, if z is blockable, prune z from the completion graph.
2. We remove y from Δ.

The tree-like structure of the blockable parts of a completion graph ensures that pruning removes only subtrees, but no ancestors of y.

Strategy of rule applications. The *level* of a nominal node is defined as follows: Every nominal node x where $\mathcal{L}(x)$ entails $\{a\}(x) \geqslant 1$ for some $a \in \mathsf{ind}(\mathcal{O})$ (e.g. one of the initial nominal nodes) is of level 0; and a nominal node is of level $i > 0$ if it is not of some lower level $j < i$ and has a neighbor that is of level $i-1$. Note that merging can only decrease the levels of nodes, but not increase them.

We use a similar strategy of rule applications as in [17], i.e. the priority order between rules is as follows:

Table 4: The tableau rules for value restrictions

Rule	
$(\forall)$	If 1. $(\forall r.C)(x)$ occurs in $\mathcal{L}(x)$, x is not blocked, and 2. there is no safe neighbor $y \in \mathcal{N}(x)$ such that $\mathcal{L}(x)$ entails $(\forall r.C)(x) \geqslant r(x,y) \Rightarrow C(y)$, then 1. introduce a new node y, add (x,y) to $\mathcal{N}$, and 2. add either • $r(x,y) \leqslant C(y)$ and $(\forall r.C)(x) \geqslant 1$, or • $r(x,y) > C(y)$ and $(\forall r.C)(x) \geqslant C(y)$ to $\mathcal{L}(x)$.
$(\forall_{\mathbf{A}})$	If $(\forall r.C)(x)$ occurs in $\mathcal{L}(x)$, then add $(\forall r.C)(x) \leqslant (\forall \mathbf{A}_r.C)(x)$ to $\mathcal{L}(x)$ unless it is already entailed.
$(\mathbf{A})$	If $(\forall \mathbf{A}^q.C)(x)$ and $r(x,y)$ occur in $\mathcal{L}(x)$ and $q \xrightarrow{r,p} q' \in \mathbf{A}$, then add either $p \leqslant (\forall \mathbf{A}^{q'}.C)(y)$, $r(x,y) \leqslant (\forall \mathbf{A}^{q'}.C)(y)$, or $(\forall \mathbf{A}^q.C)(x) \leqslant (\forall \mathbf{A}^{q'}.C)(y)$ to $\mathcal{L}(x)$ unless one of them is already entailed.
$(\mathbf{A}_\varepsilon)$	If $(\forall \mathbf{A}^q.C)(x)$ occurs in $\mathcal{L}(x)$ and $q \xrightarrow{\varepsilon,p} q' \in \mathbf{A}$, then add either $p \leqslant (\forall \mathbf{A}^{q'}.C)(x)$ or $(\forall \mathbf{A}^q.C)(x) \leqslant (\forall \mathbf{A}^{q'}.C)(x)$ to $\mathcal{L}(x)$ unless one of them is already entailed.
$(\mathbf{A}_{\mathsf{fin}})$	If $(\forall \mathbf{A}^q.C)(x)$ occurs in $\mathcal{L}(x)$ and q is final, then add $(\forall \mathbf{A}^q.C)(x) \leqslant C(x)$ to $\mathcal{L}(x)$ unless it is already entailed.

1. $(\rightsquigarrow)$
2. (o) and $(o_\leqslant)$
3. (NN) and $(\mathsf{NN}_\leqslant)$ (first applied to nominal nodes with lower levels)
4. all other rules.

Now that we have introduced all relevant definitions, we can prove termination and correctness of our algorithm. The proof of the following lemma closely follows the one from [17], but has to deal with order assertions instead of concepts in the node labels.

Lemma 4.2. *Every sequence of applications of the tableau rules to $\mathcal{G}_0$ terminates.*

Proof. Let $m := |\mathsf{sub}(\mathcal{O})|$, $k := |\mathsf{rol}(\mathcal{O})|$, n be the maximal number occurring in number restrictions in $\mathsf{sub}(\mathcal{O})$, $\ell := |\mathsf{ind}(\mathcal{O})|$, and $o := |\mathcal{V}_\mathcal{O}|$. Recall that in the worst case m is exponential in n and in the size of the role hierarchy. However, the exponential blowup in n is irrelevant since each set $\mathcal{L}(x)$ can contain at most one additional at-least restriction ${\geqslant}m\,r.C$ for each ${\geqslant}n\,r.C$ that occurs in $\mathcal{O}$.

Observe first that the relation $\mathcal{N}$ restricted to the blockable nodes is always tree-shaped. More precisely, such trees are rooted in nominal nodes and leaves may have outgoing $\mathcal{N}$-edges to nominal nodes. To see this, assume that the application of one of the tableau rules destroys this property, i.e. creates a completion graph with a blockable node x that has two different predecessors, i.e. $(y_1,x),(y_2,x) \in \mathcal{N}$. Then the rule that was applied must be a shrinking rule, and it must further be the case that y_1 and y_2 each had a blockable successor, x and x', respectively, and x' was merged into x by the rule $(\geqslant_\leqslant)$ (the other two shrinking rules do merge two blockable nodes). But then there must be a common neighbor z of x and x' such that $\mathcal{L}(z)$ entails $({\geqslant}n\,r.C)(z) < \min\{r(z,x),C(x)\}$ and $({\geqslant}n\,r.C)(z) < \min\{r(z,x'),C(x')\}$. This means that z is a nominal node since otherwise the structure among blockable nodes would already have been non-tree-shaped before. Furthermore, we must have either $(x,z) \in \mathcal{N}$ or $(x',z) \in \mathcal{N}$ since otherwise either $z = y_1 = y_2$ or either x or x' already had two different predecessors. This

Table 5: The tableau rules for number restrictions

$(\geqslant)$	If 1. $(\geqslant n\, r.C)(x)$ occurs in $\mathcal{L}(x)$, x is not blocked, and 2. there do not exist n safe neighbors $y_1, \dots, y_n \in \mathcal{N}(x)$ with $y_i \not\doteq y_j$, $1 \leqslant i < j \leqslant n$, such that $\mathcal{L}(x)$ entails $(\geqslant n\, r.C)(x) \leqslant \min\{r(x, y_i), C(y_i)\}$, $1 \leqslant i \leqslant n$, then 1. introduce n new nodes $y_1, \dots, y_n$ with $(x, y_1), \dots, (x, y_n) \in \mathcal{N}$ and $y_i \not\doteq y_j$, $1 \leqslant i < j \leqslant n$, and 2. for each i, $1 \leqslant i \leqslant n$, add $(\geqslant n\, r.C)(x) \leqslant r(x, y_i)$ and $(\geqslant n\, r.C)(x) \leqslant C(y_i)$ to $\mathcal{L}(x)$.
$(\geqslant_\leqslant)$	If 1. $(\geqslant n\, r.C)(x)$ occurs in $\mathcal{L}(x)$ and 2. there exist at least n neighbors $y \in \mathcal{N}(x)$ such that $\mathcal{L}(x)$ entails $(\geqslant n\, r.C)(x) < \min\{r(x, y), C(y)\}$, then 1. choose two such neighbors y, z such that $y \doteq z$ and 2. do the following: • if y is a nominal node, then merge z into y; • else if z is a nominal node or an ancestor of y, then merge y into z; • else merge z into y.
(ch)	If 1. $(\geqslant n\, r.C)(x)$ and $r(x, y)$ occur in $\mathcal{L}(x)$ and 2. $\mathcal{L}(x)$ entails neither $(\geqslant n\, r.C)(x) < \min\{r(x, y), C(y)\}$ nor its negation, then add either • $(\geqslant n\, r.C)(x) < r(x, y)$ and $(\geqslant n\, r.C)(x) < C(y)$, • $(\geqslant n\, r.C)(x) \geqslant r(x, y)$, or • $(\geqslant n\, r.C)(x) \geqslant C(y)$ to $\mathcal{L}(x)$.

means that there must be $m \leqslant n-1$ nominal neighbors $z_1, \dots, z_m$ with $z_i \not\doteq z_j$, $1 \leqslant i < j \leqslant m$, and $\mathcal{L}(z)$ entails $(\geqslant (m+1)\, r.C)(z) < \min\{r(z, z_i), C(z_i)\}$, $1 \leqslant i \leqslant m$; otherwise the rule (NN) would be applicable and would have been applied before $(\geqslant_\leqslant)$. But then immediately afterwards the rule $(\mathsf{NN}_\leqslant)$ would have to be applied, and would merge x and x' into one of the nominal nodes $z_1, \dots, z_m$, thus invalidating our assumption.

Further observe that nodes and elements from node labels can only be removed by the shrinking rules, and new nodes can only be added by the generating rules. Moreover, each generating rule can be triggered at most once for each concept in $\mathsf{sub}(\mathcal{O})$ occurring in the label of a node x. For the rules involving role connections to neighboring nodes, this observation is due to the fact that, if a neighbor y of x is merged into another node z, then z inherits all relevant order assertions from y, and either z is then a neighbor of x (if x is a nominal node or y is a successor of x) or x is removed by pruning (if x is a blockable node and x is a successor of y). This means that each node can have at most mn blockable successors.

The next crucial observation is that blocking, which can occur only within a path consisting only of blockable nodes, occurs after at most $\lambda := 2^{2(o+2m+4k+2)^2} + 1$ steps. For this, it suffices to determine the total number of possible order assertions (not involving min or $\Rightarrow$, and using only $\leqslant$ or $>$) that can be formulated about two neighboring nodes. The underlying order structure contains $o + 2m + 4k + 2$ elements, and hence there are $2(o + 2m + 4k + 2)^2$ such order assertions. This means that each $\mathcal{N}$-chain of blockable nodes must contain a directly blocked node after at most λ steps. This implies that all blockable subtrees of the completion graph have branching degree at most mn and depth at most λ.

The last step is to show that the number of nominal nodes is bounded by $O(\ell(mn)^\lambda)$. The proof of this proceeds as in [17]: The rule (NN) can initially only be triggered due to a newly created

Table 6: The special tableau rules for nominals

(NN)	If 1. $({\geqslant}n\, r.C)(x)$ occurs in $\mathcal{L}(x)$, x is a nominal node, 2. there is a blockable node y with $(y, x) \in \mathcal{N}$ such that $\mathcal{L}(x)$ entails $({\geqslant}n\, r.C)(x) < \min\{r(x, y), C(y)\}$, and 3. there do not exist $\ell \leqslant n - 1$ nominal nodes $z_1, \dots, z_\ell$ such that $\mathcal{L}(x)$ entails $({\geqslant}(\ell+1)\, r.C)(x) = ({\geqslant}n\, r.C)(x)$ and $({\geqslant}(\ell+1)\, r.C)(x) < \min\{r(x, z_i), C(z_i)\}$, $1 \leqslant i \leqslant \ell$, and $z_i \not\doteq z_j$, $1 \leqslant i < j \leqslant \ell$, then 1. guess a number m between 1 and $n - 1$, and add $({\geqslant}(m+1)\, r.C)(x) = ({\geqslant}n\, r.C)(x)$ to $\mathcal{L}(x)$, 2. introduce m new nodes $y_1, \dots, y_m$ with $(x, z_1), \dots, (x, z_m) \in \mathcal{N}$ and $z_i \not\doteq z_j$, $1 \leqslant i < j \leqslant m$, 3. introduce m new individual names $a_1, \dots, a_m$, and for each i, $1 \leqslant i \leqslant m$, add $({\geqslant}(m+1)\, r.C)(x) < r(x, z_i)$, $({\geqslant}(m+1)\, r.C)(x) < C(z_i)$, and $\{a_i\}(z_i) \geqslant 1$ to $\mathcal{L}(x)$.
$(\mathsf{NN}_{\leqslant})$	If 1. $({\geqslant}n\, r.C)(x)$ occurs in $\mathcal{L}(x)$, x is a nominal node, 2. there is a blockable node $y \in \mathcal{N}(x)$ such that $\mathcal{L}(x)$ entails $({\geqslant}n\, r.C)(x) < \min\{r(x, y), C(y)\}$, and 3. there are nominal nodes $z_1, \dots, z_{n-1} \in \mathcal{N}(x)$ with $z_i \not\doteq z_j$, $1 \leqslant i < j \leqslant n - 1$, such that $\mathcal{L}(x)$ entails $({\geqslant}n\, r.C)(x) < \min\{r(x, z_i), C(z_i)\}$, $1 \leqslant i \leqslant n - 1$, then 1. choose a z_i, $1 \leqslant i \leqslant n - 1$, such that $y \doteq z_i$ and 2. merge y into z_i.

nominal node x, which must be of level 0 since the only individual names occurring in $\mathsf{sub}(\mathcal{O})$ are those in $\mathsf{ind}(\mathcal{O})$. Afterwards, it may be applied to predecessors of x that were originally blockable but were then merged into nominal neighbors of x. Since the length of such a chain of blockable nodes is at most λ, the rule (NN) can be applied only to nominal nodes of level below λ. Furthermore, this rule can be applied at most m times to each node of level i (or its heirs), each time generating at most n new nominals, and hence at most $\ell(mn)^{i+1}$ nominal nodes of level $i+1$. Since it can only be applied up to level λ, this gives an upper bound of $O(\ell(mn)^\lambda)$ new nominal nodes. Additionally, each nominal node may be the root of a blockable tree of size $O((mn)^\lambda)$. Hence, the total number of nodes in a completion graph is finite, and thus each completion graph must become complete after finitely many steps. ☐

We now prove that the algorithm correctly decides consistency of $\mathcal{O}$ (cf. Lemma 3.2).

Lemma 4.3. *If the tableau rules can be applied to $\mathcal{G}_0$ in such a way that a complete and clash-free completion graph is obtained, then there exists a countable tableau for $\mathcal{O}$.*

Proof. Assume that the tableau rules have been applied to $\mathcal{G}_0$, resulting in a complete and clash-free completion graph $\mathcal{G} = (\Delta, \mathcal{N}, \mathcal{L}, \not\doteq)$. We first modify the labeling function for all nominal nodes x, by removing all assertions from $\mathcal{L}(x)$ that refer to blockable nodes, and adding all entailments about nominal nodes that may have been lost in this process. More formally, we define $\mathcal{L}'(x)$ as the restriction of $\mathcal{L}(x)$ to the order structure

$$\mathcal{U}_o := \mathcal{V}_{\mathcal{O}} \cup \big\{C(a) \mid C \in \mathsf{sub}(\mathcal{O}),\ a \in \Delta_o\big\} \cup \big\{r(a, b), \neg r(a, b) \mid r \in \mathsf{rol}(\mathcal{O}),\ a, b \in \Delta_o\big\},$$

and then add all order assertions over $\mathcal{U}_o$ that are entailed by the original $\mathcal{L}(x)$ to $\mathcal{L}'(x)$. This allows a better separation of the behavior of the nominal nodes from that of the blockable nodes. Observe that all relevant order assertions that refer to the connection between a nominal node x and a blockable neighbor y have already been transferred to $\mathcal{L}(y)$ by the rule $(\rightsquigarrow)$. Furthermore, $\mathcal{G}$ is still clash-free after this modification; however, it may not be complete anymore for the nominal nodes. The blocking relationships between nodes are not affected since they do not involve nominal nodes.

We now construct a countable tableau $(\Delta', \mathcal{A}^*)$ of $\mathcal{O}$ by following the structure of $\mathcal{N}$ and, at each directly blocked node x, unraveling the structure by replacing x with a copy of an ancestor that blocks it. At the same time, we will construct a function $f\colon \Delta' \to \Delta$ that specifies which node was used to construct each element of Δ'.

We initially set

- $\Delta' := \{x \in \Delta \mid x \text{ is not blocked}\}$,
- $\mathcal{A}^* := \bigcup_{x\in\Delta'} \mathcal{L}'(x)$, and
- $f(x) := x$ for all $x \in \Delta'$,

and assume first that $\mathcal{A}^*$ is unsatisfiable. This can be the case only because of a sequence of elements $\alpha_1 \trianglelefteq_1 \cdots \trianglelefteq_{n-1} \alpha_n$, where $\trianglelefteq_i \in \{<, \leqslant, =\}$, each order assertion $\alpha_i \trianglelefteq_i \alpha_{i+1}$, $1 \leqslant i \leqslant n-1$, is entailed by some $\mathcal{L}'(x_i)$ with $x_i \in \Delta'$, we have $\alpha_1 = \alpha_n$, and at least one of the $\trianglelefteq_i$ is a strict inequality ($<$). Let this be a sequence that has minimal length among all sequences with this property. We show the following properties:

(a) We have $n > 2$. Otherwise, $\alpha_1 <_1 \alpha_1$ would be entailed by $\mathcal{L}'(x_1)$, which contradicts the clash-freeness of $\mathcal{G}$.

(b) We have $x_i \neq x_{i+1}$ for all i, $1 \leqslant i \leqslant n-2$, and $x_{n-1} \neq x_1$. Otherwise, by (a) we would have a situation like $\alpha_1 \trianglelefteq_1 \alpha_2 \trianglelefteq_2 \alpha_3$ (modulo cyclic index shifts), where both order assertions are entailed by the same $\mathcal{L}'(x)$. But then also $\alpha_1 \trianglelefteq \alpha_3$ would be entailed by $\mathcal{L}'(x)$, where $\trianglelefteq$ is $<$ if $< \in \{\trianglelefteq_1, \trianglelefteq_2\}$, and otherwise it is $\leqslant$. But this shows the existence of a shorter sequence with the same properties as before, in contradiction to our minimality assumption.

(c) There is no α_i that refers only to nominal nodes. Otherwise, by (a) we would have a situation like $\alpha_1 \trianglelefteq_1 \alpha_2 \trianglelefteq_2 \alpha_3$, where α_1 is of the form $p \in \mathcal{V}_{\mathcal{O}}$, $C(x_a)$, or $r(x_a, x_b)$, where x_a and x_b are nominal nodes. But then we would have $\alpha_1, \alpha_2 \in \mathcal{U}(x_2)$; in particular, if x_2 is a nominal node, then both α_1 and α_2 would refer only to nominal nodes due to our modification. But then the rule ($\rightsquigarrow$) implies that $\alpha_1 \trianglelefteq_1 \alpha_2$ would also entailed by $\mathcal{L}'(x_2)$, which contradicts (b).

Since the (modified) labels of nominal nodes may only refer to nominal nodes, (c) implies that the x_i are all blockable nodes. Furthermore, since we have $\alpha_i \in \mathcal{U}(x_{i-1}) \cap \mathcal{U}(x_i)$, $2 \leqslant i \leqslant n-1$, and $\alpha_1 = \alpha_n \in \mathcal{U}(x_1) \cap \mathcal{U}(x_{n-1})$, each pair (x_{i-1}, x_i), $2 \leqslant i \leqslant n-1$, and (x_1, x_{n-1}) must be neighbors. The tree structure of $\mathcal{N}$ on the blockable nodes and (b) imply that there is a situation like $\alpha_1 \trianglelefteq_1 \alpha_2 \trianglelefteq_2 \alpha_3 \trianglelefteq_3 \alpha_4$ such that $x_1 = x_3$. Hence, we have $\alpha_2, \alpha_3 \in \mathcal{U}(x_1)$, which shows by ($\rightsquigarrow$) that $\alpha_1 \trianglelefteq \alpha_4$ is already entailed by $\mathcal{L}'(x_1)$, where $\trianglelefteq$ is obtained as in the proof of (b). But this again contradicts the minimality of n.

This shows that the initial tableau constructed above is satisfiable. We now iteratively expand it by unraveling $\mathcal{N}$ at the blocked nodes. Let $x' \in \Delta'$ be such that $f(x')$ has a directly blocked successor x in $\mathcal{G}$ that is not yet represented in our tableau, i.e. there is no $x'' \in \Delta'$ such that $f(x'') = x$ and x'' is connected to x' by some role assertions. Let further $x^@$ be the name that is used in $\mathcal{A}^*$ to refer to this still missing successor of x', $y \in \Delta$ be a node that blocks x in $\mathcal{G}$, and y' be the predecessor of y in $\mathcal{N}$. Recall that y', y, $f(x')$, x, and all nodes in between are blockable. Consider now the subtree $\Delta_y \subseteq \Delta$ consisting of all blockable descendants of y that are not blocked. To distinguish these nodes from those already present in Δ', for each blockable node z occurring in $\mathcal{L}'(v)$ for some $v \in \Delta_y$, let $z^@$ be a unique new node name that does not yet occur in Δ'. We now do the following:

- add $\{z^@ \mid z \in \Delta_y\}$ to Δ';
- replace all occurrences $x^@$ in $\mathcal{A}^*$ by $y^@$ and add $\bigcup_{z\in\Delta_y} \mathcal{L}'(z)^@$ to $\mathcal{A}^*$, where $\mathcal{L}'(z)^@$ is

obtained from $\mathcal{L}'(z)$ by replacing all occurrences of $f(x')$ by x', and of any other blockable nodes z by $z^{@}$; and

- set $f(z^{@}) := z$ for all $z \in \Delta_y$.

The resulting set $\mathcal{A}^*$ is such that it looks as if x has never been blocked, but rather that the tableau rules have been applied to it and its successors without restrictions. Assume now that $\mathcal{A}^*$ has become unsatisfiable by this construction, and hence there is a sequence $\alpha_1 \trianglelefteq_1 \cdots \trianglelefteq_{n-1} \alpha_n$ as above, where n is minimal. Since the original $\mathcal{A}^*$, and hence also $\bigcup_{z\in\Delta_y} \mathcal{L}'(z)^{@}$, are satisfiable, it must be the case that this sequence involves nodes from the previous Δ' as well some of the form $z^{@}$ for $z \in \Delta_y$. We can show the properties (a)–(c) as before. Moreover, the tree-shape of the connections between blockable nodes is maintained by our construction. To derive a contradiction in the same way as above, it suffices to note that due to the blocking condition all order assertions shared by $f(x')$ and x' are also shared by x' and $y^{@}$ (after renaming $x^{@}$ to $y^{@}$ and $f(x')$ to x'), and hence $\mathcal{L}'(y)^{@}$ and the set corresponding to $\mathcal{L}'(f(x'))$ in $\mathcal{A}^*$ behave as if the rule $(\rightsquigarrow)$ has been applied exhaustively.

If we continue this process infinitely, taking care that every directly blocked node is unraveled eventually, we obtain the final tableau $(\Delta', \mathcal{A}^*)$. The set $\mathcal{A}^*$ is satisfiable due to the compactness theorem of first-order logic. It remains to verify the tableau conditions. It is easy to verify that the local conditions (T1)–(T5), (T7), (T9), (T10), (T14), (T18), and (T20), are satisfied due to the corresponding tableau rules. We consider the remaining ones:

(T6) If $x \in \Delta'$ is such that $(\forall r.C)(x)$ occurs in $\mathcal{A}^*$, then we know that $f(x)$ is not blocked in $\mathcal{G}$. Hence, by the rule $(\forall)$ there must be a safe neighbor $y \in \mathcal{N}(x)$ such that $\mathcal{L}(x)$ entails $(\forall r.C)(f(x)) \geqslant r(f(x), y) \Rightarrow C(y)$.
Consider first the case that x is blockable. If y is not blocked, then we have directly introduced (a copy y' of) y into Δ', together with (the copy x of) $f(x)$, and the (renamed) entailment still holds in $\mathcal{A}^*$. If y is a successor of x, then it may be the case that y is directly blocked in $\mathcal{G}$. But then we have introduced a node y' into Δ' that can serve as a replacement for this missing successor, i.e. $\mathcal{A}^*$ entails $(\forall r.C)(x) \geqslant r(x, y') \Rightarrow C(y')$ due to the blocking condition.
If x is a nominal node, then we know that y is not blocked since it is a safe neighbor of x. Nevertheless, it may be that we have removed from $\mathcal{L}(x)$ some assertions that were necessary to derive the above entailment; this can only be the case if y is blockable. However, by the rule $(\rightsquigarrow)$, this entailment has been transferred to $\mathcal{L}(y)$, and is still present in $\mathcal{L}'(y)$, which is why it is still entailed by $\mathcal{A}^*$.

(T8) If $(\forall \mathbf{A}.C)(x)$ and $r(x, y)$ occur in $\mathcal{A}^*$, then the required entailment is provided by the rule $(\mathbf{A})$. Again, the modification of $\mathcal{L}$ to $\mathcal{L}'$ for the nominal nodes is rendered irrelevant by the rule $(\rightsquigarrow)$.

(T11) This case can be handled by similar arguments as for (T6). Additionally, the n safe neighbors created by the rule $(\geqslant)$ are still distinct in Δ' since they can never be merged.

(T12) Assume that $({\geqslant}n\, r.C)(x)$ occurs in $\mathcal{A}^*$ and there are different $y_1, \dots, y_n \in \Delta'$ such that $\mathcal{A}^*$ entails $({\geqslant}n\, r.C)(x) < \min\{r(x, y_i), C(y_i)\}$. By our construction, we know that $({\geqslant}n\, r.C)(f(x))$ occurs in $\mathcal{L}(f(x))$ and there exist n neighbors $y'_1, \dots, y'_n$ of $f(x)$ (which are possibly blocked) for which similar assertions are entailed by $\mathcal{L}(f(x))$. Since $\mathcal{G}$ is clash-free, there must be two of these neighbors that are not in the relation $\not\approx$, and hence the rule $(\geqslant_{\leqslant})$ is applicable to $\mathcal{G}$. This contradicts our assumption that $\mathcal{G}$ is complete.

(T15) For each $a \in \mathsf{ind}(\mathcal{O})$, the existence of exactly one nominal node for a is due to the definition of the initial completion graph $\mathcal{G}_0$, clash-freeness of $\mathcal{G}$, the rule $(o_{\leqslant})$, and our construction of the tableau.

(T17) We consider the example of an assertion $r(a, b) \geqslant C(c)$ in $\mathcal{A}$. In $\mathcal{G}_0$, there exist nodes a, b, c that are all neighbors, and each label entails $r(a, b) \geqslant C(c)$. Due to merging, in $\mathcal{G}$ there exist heirs x_a, x_b, x_c of these original nodes, which inherit the neighbor relationships as well as the required entailment. Hence, $\mathcal{A}^*$ also entails $r(x_a, x_b) \geqslant C(x_c)$. The proofs for the other kinds of assertions are similar.

Finally, (T13), (T16), (T19), (T21), and (T22) can be shown using similar arguments. □

The other direction is easier to show.

Lemma 4.4. *If there is a tableau for $\mathcal{O}$, then the tableau rules can be applied to $\mathcal{G}_0$ in such a way that a complete and clash-free completion graph is obtained.*

Proof. If $\mathcal{O}$ is consistent, then by Lemma 3.2 there exists a tableau $(\Delta', \mathcal{A}^*)$ for $\mathcal{O}$. We use this tableau to guide the application of the completion rules to $\mathcal{G}_0$. We will maintain a function $f\colon \Delta \to \Delta'$ that matches the nodes of our completion graph to the nodes of the tableau, such that the following conditions are satisfied:

- (i) If $\alpha \bowtie \beta$ occurs in $\mathcal{L}(x)$ and α, β do not involve number restrictions or nominals that do not occur in $\mathcal{O}$, then $\mathcal{A}^*$ entails $f(\alpha) \bowtie f(\beta)$, where $f(\alpha)$ is obtained from α by replacing all nodes according to f.
- (ii) If $x \neq y$, then $f(x) \neq f(y)$.
- (iii) If $\geqslant m\, r.C \in \mathsf{sub}(\mathcal{O})$ does not occur in $\mathcal{O}$ and $(\geqslant m\, r.C)(x)$ occurs in $\mathcal{L}(x)$, then there are exactly $m-1$ elements $y \in \Delta'$ such that $\mathcal{A}^*$ entails
$$(\geqslant n\, r.C)(f(x)) < \min\{r(f(x), y), C(y)\}.$$

For each $(\geqslant m\, r.C)(x)$ occurring in $\mathcal{L}(x)$ for which $\geqslant m\, r.C$ does not occur in $\mathcal{O}$, we know that $\mathcal{L}(x)$ entails $(\geqslant m\, r.C)(x) = (\geqslant n\, r.C)(x)$ for some $\geqslant n\, r.C$ that does occur in $\mathcal{O}$. Hence, (i) and the satisfiability of $\mathcal{A}^*$ imply that all node labels of our completion graph will be satisfiable. Furthermore, clashes due to number restrictions are ruled out by (i)–(iii) and (T12). Finally, nominals behave correctly due to (i), (ii), and (T15). Hence, our final completion graph will be clash-free.

For the initial completion graph, we set $f(a) := x_a$ for all $\mathsf{ind}(\mathcal{O})$, where x_a is the nominal node that exists by (T15). Due to (T15) and (T17), this mapping satisfies all our conditions. We now show by induction on the sequence of rule applications how the tableau rules can be applied in order to maintain the conditions (i)–(iii). For most of the tableau rules, it is trivial to show that they can be applied in such a way that the conditions remain satisfied. In particular, for the simple rules that only have to make nondeterministic choices because of the semantics of $\Rightarrow$ and min (i.e. the rules ($\sqcap$), ($\to$), ($\sqsubseteq_{\mathcal{T}}$), ($\sqsubseteq_{\mathcal{R}}$), ($\mathsf{dis}$), ($\mathbf{A}$), ($\mathbf{A}_\varepsilon$), ($\mathsf{ch}$), and ($o$)), we know by the corresponding conditions of Definition 3.1 and our semantics that we can always choose one of the alternatives such that (i) is not violated. It is also clear that the rule ($\rightsquigarrow$) does not affect this condition.

Consider now the generating rule ($\geqslant$); the arguments needed for ($\forall$) are similar. Assume that we have to apply this rule because $(\geqslant n\, r.C)(x)$ occurs in $\mathcal{L}(x)$, and hence by (i) the element $(\geqslant n\, r.C)(f(x))$ occurs in $\mathcal{A}^*$. Due to (T11), there are at least n elements $y_1, \dots, y_n \in \Delta'$ such that $\mathcal{A}^*$ entails $(\geqslant n\, r.C)(x) \leqslant \min\{r(x, y), C(y)\}$, and hence we can introduce n new neighbors $y'_1, \dots, y'_n$ according to ($\geqslant$) and set $f(y'_i) := y_i$, $1 \leqslant i \leqslant n$, in order to keep the conditions (i)–(iii) satisfied.

For the shrinking rule ($\mathsf{NN}_\leqslant$), consider any $(\geqslant n\, r.C)(x)$, y, and $z_1, \dots, z_{n-1}$ as in the preconditions of this rule. Then by (iii) or (i) and (T12), we know that Δ' contains at most $n-1$ nodes z for which $\mathcal{A}^*$ entails $(\geqslant n\, r.C)(f(x)) < \min\{r(f(x), z), C(z)\}$. Furthermore, by (i), the nodes $f(y), f(z_1), \dots, f(z_{n-1})$ all satisfy this condition. By (ii), this implies that there is an index i, $1 \leqslant i \leqslant n-1$, such that $f(y) = f(z_i)$, and hence $y \doteq z_i$. This shows that the rule ($\mathsf{NN}_\leqslant$) can be applied in such a way that all conditions remain satisfied. The same can be shown for ($\geqslant_\leqslant$) using similar arguments.

For $(o_\leqslant)$, assume that there exist an $a \in \mathsf{N_I}$ and two nodes x, y whose labels entail $\{a\}(x) \geqslant 1$ and $\{a\}(y) \geqslant 1$, respectively. This can only be the case for $a \in \mathsf{ind}(\mathcal{O})$ since the rule (NN) always introduces new individual names. Hence, (i) and (T15) imply that $\pi(x) = \pi(y)$, and thus we can again merge these two nodes.

Finally, consider the rule (NN). If all its preconditions are satisfied by $(\geqslant n\, r.C)(x)$ and y, then we know that it has not been applied to this number restriction at x (or any node that was merged into x) before. Hence, $\geqslant n\, r.C$ must occur in $\mathcal{O}$, and thus (i) and (T12) imply that there are exactly $m \leqslant n-1$ elements $z'_1, \dots, z'_m \in \Delta'$ for which $\mathcal{L}(x)$ entails $(\geqslant n\, r.C)(x) < \min\{r(f(x), z'_i), C(z'_i)\}$, $1 \leqslant i \leqslant m$. This shows that we can apply the rule and create m new nominal nodes $z_1, \dots, z_m \in \Delta$, for which we set $f(z_i) := z'_i$, $1 \leqslant i \leqslant m$, without violating the conditions.

Using Lemma 4.2, this shows that after finitely many steps we will have produced a complete and clash-free completion graph. $\square$

Note that the bound on the number of nodes derived in Lemma 4.2 is triply exponential in the size of $\mathcal{O}$, and hence Lemmas 3.2, 4.3, and 4.4 prove a 3-NExpTime upper bound on the complexity of consistency in G-$\mathcal{SROIQ}$, which is the same bound that is obtained from the classical tableau algorithm for $\mathcal{SROIQ}$ [15]. This is in contrast to 2-NExpTime-completeness of classical $\mathcal{SROIQ}$ [20], where the upper bound is obtained by a reduction to the two-variable fragment of first-order logic with counting quantifiers. The 2-NExpTime-hardness can be transferred to our setting via a linear reduction from consistency in a sublogic of G-$\mathcal{SROIQ}$ to consistency in classical $\mathcal{SROIQ}$ [6].

5 Conclusions

This paper continues the study of fuzzy extensions of expressive DLs with the Gödel semantics of [5, 11]. We extend the previous constructions to develop a goal-oriented tableau algorithm that can be the basis of a practical implementation to decide consistency in G-$\mathcal{SROIQ}$. In contrast, the reduction to a classical ontology proposed in [11] exhibits an exponential blowup if both nominals and number restrictions are used, and furthermore is restricted to sublogics of G-$\mathcal{SROIQ}$ with the (quasi-)forest model property only.

Tableau algorithms developed for FDLs using the closely related Zadeh semantics [24, 25] or finitely valued semantics [9, 19, 26] are conceptually much closer to the classical algorithms, and explicitly represent the truth degrees of all concepts in the node labels. In contrast, due to the lack of the finitely valued model property, our tableau algorithm needs to use a novel data structure to reason about order relations between truth degrees for different concepts. The main technical challenge was to find a way to represent these relations that still allowed us to keep the good characteristics of the classical algorithm.

Nevertheless, in order to obtain a practically feasible implementation, more optimizations are necessary. For example, we can reduce the amount of nondeterminism in some of the tableau rules by exploiting the order assertions that are already entailed by $\mathcal{L}(x)$; e.g., in the rule $(\to)$, if $\mathcal{L}(x)$ already entails $C(x) \leqslant D(x)$, then it suffices to add $(C \to D)(x) \geqslant 1$ to $\mathcal{L}(x)$, and the other case can be discarded immediately. Moreover, the naive algorithm suggested in Section **??** to decide satisfiability and entailment for sets of order assertions is hardly practical. This should be done by a more streamlined algorithm, e.g. using reachability analysis in the graph formed by an order structure S and a set of order assertions over S. Moreover, this algorithm should employ caching techniques in order to avoid having to repeat the whole reachability analysis every time a single assertion (i.e. edge) is added. Another obvious optimization would be to put all order assertions that refer only to the nominal nodes into a global set that is shared by all nodes, instead of replicating these assertions in all node labels.

On the theoretical side, future work includes the analysis of sublogics with better complexities, e.g. Horn variants of G-$\mathcal{SROIQ}$ [22], and other reasoning problems such as answering (fuzzy) conjunctive queries [7, 23].

Acknowledgments

We want to thank Daniel Borchmann for fruitful discussions on the topic of this paper.

References

[1] Franz Baader, Diego Calvanese, Deborah L. McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, 2nd edition, 2007.

[2] Fernando Bobillo, Marco Cerami, Francesc Esteva, Àngel García-Cerdaña, Rafael Peñaloza, and Umberto Straccia. Fuzzy description logics. In Petr Cintula, Christian G. Fermüller, and Carles Noguera, editors, *Handbook of Mathematical Fuzzy Logic*, volume 3 of *Studies in Logic*. College Publications, 2015. In press.

[3] Fernando Bobillo, Miguel Delgado, and Juan Gómez-Romero. Optimizing the crisp representation of the fuzzy description logic $\mathcal{SROIQ}$. In *Uncertainty Reasoning for the Semantic Web I*, volume 5327 of *Lecture Notes in Artificial Intelligence*, pages 189–206. Springer-Verlag, 2008.

[4] Fernando Bobillo, Miguel Delgado, Juan Gómez-Romero, and Umberto Straccia. Joining Gödel and Zadeh fuzzy logics in fuzzy description logics. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 20(4):475–508, 2012.

[5] Stefan Borgwardt, Felix Distel, and Rafael Peñaloza. Decidable Gödel description logics without the finitely-valued model property. In Chitta Baral, Giuseppe De Giacomo, and Thomas Eiter, editors, *Proc. of the 14th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR'14)*, pages 228–237. AAAI Press, 2014.

[6] Stefan Borgwardt, Felix Distel, and Rafael Peñaloza. The limits of decidability in fuzzy description logics with general concept inclusions. *Artificial Intelligence*, 218:23–55, 2015.

[7] Stefan Borgwardt, Theofilos P. Mailis, Rafael Peñaloza, and Anni-Yasmin Turhan. Answering fuzzy conjunctive queries over finitely valued fuzzy ontologies. *CoRR*, abs/1508.02626, 2015.

[8] Stefan Borgwardt and Rafael Peñaloza. Reasoning in fuzzy description logics using automata. *Fuzzy Sets and Systems*, 2015. In press.

[9] Stefan Borgwardt and Rafael Peñaloza. Consistency reasoning in lattice-based fuzzy description logics. *International Journal of Approximate Reasoning*, 55(9):1917–1938, 2014.

[10] Stefan Borgwardt and Rafael Peñaloza. Infinitely valued Gödel semantics for expressive description logics. LTCS-Report 15-11, Chair for Automata Theory, TU Dresden, Germany, 2015. See http://lat.inf.tu-dresden.de/research/reports.html.

[11] Stefan Borgwardt and Rafael Peñaloza. Reasoning in expressive description logics under infinitely valued Gödel semantics. In Carsten Lutz and Silvio Ranise, editors, *Proc. of the 10th Int. Symp. on Frontiers of Combining Systems (FroCoS'15)*, volume 9322 of *Lecture Notes in Artificial Intelligence*, pages 49–65. Springer-Verlag, 2015.

[12] Petr Cintula, Petr Hájek, and Carles Noguera, editors. *Handbook of Mathematical Fuzzy Logic*, volume 37–38 of *Studies in Logic*. College Publications, 2011.

[13] Petr Hájek. *Metamathematics of Fuzzy Logic (Trends in Logic)*. Springer-Verlag, 2001.

[14] Petr Hájek. Making fuzzy description logic more general. *Fuzzy Sets and Systems*, 154(1):1–15, 2005.

[15] Ian Horrocks, Oliver Kutz, and Ulrike Sattler. The even more irresistible $\mathcal{SROIQ}$. In Patrick Doherty, John Mylopoulos, and Christopher Welty, editors, *Proc. of the 10th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR'06)*, pages 57–67. AAAI Press, 2006.

[16] Ian Horrocks and Ulrike Sattler. Decidability of $\mathcal{SHIQ}$ with complex role inclusion axioms. *Artificial Intelligence*, 160(1–2):79–104, 2004.

[17] Ian Horrocks and Ulrike Sattler. A tableau decision procedure for $\mathcal{SHOIQ}$. *Journal of Automated Reasoning*, 39(3):249–276, 2007.

[18] Ian Horrocks, Ulrike Sattler, and Stephan Tobies. Practical reasoning for very expressive description logics. *Logic Journal of the Interest Group in Pure and Applied Logic*, 8(3):239–263, 2000.

[19] Yuncheng Jiang, Yong Tang, Ju Wang, Peimin Deng, and Suqin Tang. Expressive fuzzy description logics over lattices. *Knowledge-Based Systems*, 23(2):150–161, 2010.

[20] Yevgeny Kazakov. $\mathcal{RIQ}$ and $\mathcal{SROIQ}$ are harder than $\mathcal{SHOIQ}$. In Gerhard Brewka and Jérôme Lang, editors, *Proc. of the 11th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR'08)*, pages 274–284. AAAI Press, 2008.

[21] Erich Peter Klement, Radko Mesiar, and Endre Pap. *Triangular Norms*. Trends in Logic, Studia Logica Library. Springer-Verlag, 2000.

[22] Markus Krötzsch, Sebastian Rudolph, and Pascal Hitzler. Complexities of horn description logics. *ACM Transactions on Computational Logic*, 14(1):1–36, 2013.

[23] Jeff Z. Pan, Giorgos B. Stamou, Giorgos Stoilos, Stuart Taylor, and Edward Thomas. Scalable querying services over fuzzy ontologies. In Jinpeng Huai, Robin Chen, Hsiao-Wuen Hon, Yunhao Liu, Wei-Ying Ma, Andrew Tomkins, and Xiaodong Zhang, editors, *Proc. of the 17th Int. World Wide Web Conference (WWW'08)*, pages 575–584. ACM, 2008.

[24] Giorgos Stoilos and Giorgos B. Stamou. Reasoning with fuzzy extensions of OWL and OWL 2. *Knowledge and Information Systems*, 40(1):205–242, 2014.

[25] Umberto Straccia. Reasoning within fuzzy description logics. *Journal of Artificial Intelligence Research*, 14:137–166, 2001.

[26] Umberto Straccia. Uncertainty in description logics: A lattice-based approach. In *Proc. of the 10th Int. Conf. on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU'04)*, pages 251–258, 2004.