Tractable Diversity: Scalable Multiperspective Ontology Management via Standpoint $\mathcal{EL}$

Lucía Gómez Álvarez, Sebastian Rudolph, Hannes Strass
Computational Logic Group, TU Dresden, Germany
{lucia.gomez_alvarez, sebastian.rudolph, hannes.strass}@tu-dresden.de

Abstract

The tractability of the lightweight description logic $\mathcal{EL}$ has allowed for the construction of large and widely used ontologies that support semantic interoperability. However, comprehensive domains with a broad user base are often at odds with strong axiomatisations otherwise useful for inferencing, since these are usually context dependent and subject to diverging perspectives.

In this paper we introduce Standpoint $\mathcal{EL}$, a multi-modal extension of $\mathcal{EL}$ that allows for the integrated representation of domain knowledge relative to diverse, possibly conflicting standpoints (or contexts), which can be hierarchically organised and put in relation to each other. We establish that Standpoint $\mathcal{EL}$ still exhibits $\mathcal{EL}$'s favourable PTIME standard reasoning, whereas introducing additional features like empty standpoints, rigid roles, and nominals makes standard reasoning tasks intractable.

1 Introduction

In many subfields of artificial intelligence, ontologies are used to provide a formal representation of a shared vocabulary, give meaning to its terms, and describe the relations between them. To this end, one of the most prominent and successful class of logic-based knowledge representation formalisms are description logics (DLs) [Baader et al., 2017; Rudolph, 2011], which provide the formal basis for the most recent version of the Web Ontology Language OWL 2 [Bao et al., 2009].

Among the most prominent families of DLs used today is $\mathcal{EL}$ [Baader et al., 2005], which is the formal basis of OWL 2 EL [Motik et al., 2009a], a popular tractable profile of OWL 2. One of the main appeals of $\mathcal{EL}$ is that basic reasoning tasks can be performed in polynomial time with respect to the size of the ontology, enabling reasoning-supported creation and maintenance of very large ontologies. An example of this is the healthcare ontology SNOMED CT [Donnelly, 2006], with worldwide adoption and a broad user base comprising clinicians, patients, and researchers among others.

However, when modelling comprehensive ontologies like SNOMED CT, one is usually facing issues related to context or perspective-dependent knowledge as well as ambiguity of language [Schulz et al., 2017]. For instance, the concept Tumour might denote a process or a piece of tissue; Allergy may denote an allergic reaction or just an allergic disposition.

In a similar vein, the decentralised nature of the Semantic Web has led to the generation of various ontologies of overlapping knowledge that inevitably reflect different points of view. For instance, an initiative has attempted to integrate the FMA1140 (Foundational Model of Anatomy), SNOMED-CT, and the NCIt (National Cancer Institute Thesaurus) into a single combined version called LargeBio and reported ensuing challenges [Osman et al., 2017a]. In this context, frameworks supporting the integrated representation of multiple perspectives seem preferable to recording the distinct views in a detached way, but also to entirely merging them at the risk of causing inconsistencies or unintended consequences.

To this end, Gómez Álvarez and Rudolph [2021] proposed standpoint logic, a formalism inspired by the theory of supervaluationism [Fine, 1975] and rooted in modal logic, which allows for the simultaneous representation of multiple, potentially contradictory, viewpoints in a unified way and the establishment of alignments between them. This is achieved by extending the base language with labelled modal operators, where propositions $\Box_\phi$ and $\Diamond_\phi$ express information relative to the standpoint $S$ and read, respectively: “according to $S$, it is unequivocal/conceivable that $\phi$”. Semantically, standpoints are represented by sets of precifications,\(^1\) such that $\Box_\phi$ and $\Diamond_\phi$ hold if $\phi$ is true in all/some of the precifications associated with $S$. Consider the following example.

Example 1 (Tumour Disambiguation). Two derivatives of the SNOMED CT ontology (SN) model tumours differently. According to TP, a Tumour is a process by which abnormal or damaged cells grow and multiply (1), yet according to TT, a Tumour is a lump of tissue (2).

\[
\begin{align*}
\Box_{TP}[\text{Tumour} \sqsubseteq \text{AbnormalGrowthProcess}] & \quad (1) \\
\Box_{TT}[\text{Tumour} \sqsubseteq \text{Tissue}] & \quad (2)
\end{align*}
\]

Both interpretations inherit the axioms of the original SNOMED CT (3) and are such that: if according to SN something is arguably both a Tumour and a Tissue, then it (unequivocally) is a Tumour according to TT (4). The respective

\(^1\)Precifications are analogous to the worlds of modal-logic frameworks with possible-worlds semantics.
assertion is made for TP (5). But Tissue and Process are disjoint categories according to SN (6).

\[(\text{TP} \sqsubseteq \text{SN}) \quad (\text{TT} \sqsubseteq \text{SN}) \quad (7)\]
\[
\diamond_{\text{SN}}[\text{Tissue} \sqcap \text{PhysicalObject}] \sqsubseteq \Box_{\text{TT}}[\text{Tissue}] \quad (8)
\]
\[
\diamond_{\text{SN}}[\text{Tumour} \sqcap \text{Process}] \sqsubseteq \Box_{\text{TT}}[\text{Tissue}] \quad (9)
\]
\[
\Box_{\text{SN}}[\text{Tissue} \sqcap \text{Process}] \sqsubseteq \bot \quad (10)
\]

While clearly incompatible, both perspectives are semantically close and we can establish relations between them. For instance, we might assert that something is unequivocally the product of a Tumour (process) according to TP if and only if it is arguably a Tumour (tissue) according to TT (7). Also, we may specify a subsumption between the classes of unequivocal instances of Tissue according to TT and to TP (8).

\[
\Box_{\text{TP}}[\exists \text{ProductOf. Tumour}] \equiv \diamond_{\text{TT}}[\text{Tissue}] \quad (11)
\]
\[
\Box_{\text{TT}}[\text{Tissue}] \sqsubseteq \Box_{\text{TP}}[\text{Tissue}] \quad (12)
\]

When recording clinical findings, clinicians may use ambiguous language, so an automated knowledge extraction service may obtain the following from text and annotated scans:

\[
\Box_{\text{SN}}[\text{Patient}(p), \exists \text{HasPart}(p, a), \text{Colon}(a)] \quad (9)
\]
\[
\diamond_{\text{SN}}[\exists \text{HasPart}(a, b), \text{Tumour}(b), \text{PhysicalObject}(b)] \quad (10)
\]

The logical statements (1)–(10), which formalise Example 1 by means of a standpoint-enhanced EL description logic, are not inconsistent, so all axioms can be jointly represented. Let us now illustrate the use of standpoint logic for reasoning with and across individual perspectives.

Example 2 (Continued from Example 1). In this case, we can disambiguate the information given by Axiom (10) using Axiom (3) and Axiom (4), which entail that according to TT, b is unequivocally a tumour, \(\Box_{\text{TT}}[\text{Tissue}(b)]\), and with Axiom (2) also a tissue, \(\Box_{\text{TT}}[\text{Tissue}(b)]\). Moreover, we can use the “bridges” to switch to another perspective. From Axiom (8), it is clear that according to TP, b is also a tissue, \(\Box_{\text{TP}}[\text{Tissue}(b)]\), and from Axiom (7) b is the product of a tumour, \(\Box_{\text{TP}}[\exists \text{ProductOf. Tumour}(b)]\). Then Axiom (1) yields

\[
\Box_{\text{TP}}[\exists \text{ProductOf. (Tissue \sqcap \text{AbnormalGrowthProcess})(b)}] \quad (13)
\]

The statement \(\Box_{\text{SN}}[\exists \text{Tumour} \sqcap \text{Process}](d)\), in contrast, will trigger an inconsistency thanks to Axiom (6), which prevents the evaluation of Tumour simultaneously as a Tissue and a Process and Axiom (2), which states that according to some interpretations, a Tumour is a Tissue. Finally, a user (e.g. a specific clinic, CL) may inherit the SNOMED CT (CL \(\sqsubseteq \text{SN}\)) and establish further axioms, e.g.

\[
\Box_{\text{CL}}[\text{Patient} \sqcap \exists \text{HasPart. (Colon} \sqcap \Box_{\text{SN}}[\exists \text{HasPart. Tumour}] 
\sqsubseteq \exists \text{AssociatedWith. ColonCancerRisk}] \quad (14)
\]

To identify patients with cancer risk. Here, Axiom (9) lets us infer that \(\Box_{\text{CL}}[\exists \text{AssociatedWith. ColonCancerRisk}]\).

The need of handling multiple perspectives in the Semantic Web has led to several (non-modal) logic-based proposals. The closest regarding goals are multi-viewpoint ontologies [Hemam and Boufaida, 2011; Hemam, 2018], which model the intuition of viewpoints in a tailored extension of OWL for which no complexity bounds are given. Similar problems are also addressed in the more extensive work on contextuality (e.g. C-OWL and Distributed ontologies [Bouquet et al., 2003; Borgida and Serafini, 2003] and the Contextualised Knowledge Repository (CKR) [Serafini and Homola, 2012]). These frameworks focus on contextual and distributed reasoning and range between different levels of expressivity for modeling the structure of contexts and the bridges between them. In the context of scalable reasoning, one should highlight the implementations that provide support for OWL2-RL based CKR defeasible reasoning [Bozzato et al., 2018].

As for modal logics, their suitability to model perspectives and contexts in a natural way is obvious [Klarman and Gutiérrez-Basulto, 2013; Gómez Álvarez and Rudolph, 2021], they are well-known in the community and their semantics is well-understood. Yet, the interplay between DL constructs and modalities is often not well-behaved and can easily endanger the decidability of reasoning tasks or increase their complexity [Baader and Othlibach, 1995; Mosurović, 1999; Wolter and Zakharyaschev, 1999]. Notable examples are NEXPTIME-completeness of the multi-modal description logic \(K_{\text{ACC}}\) [Lutz et al., 2002] and 2EXPTIME-completeness of \(\mathcal{ALC}_{\text{ACC}}\) [Klarman and Gutiérrez-Basulto, 2013], a modal contextual logic framework in the style proposed by McCarthy [McCarty and Buvac, 1998].

In this work, we focus on the framework of standpoint logics [Gómez Álvarez and Rudolph, 2021], which are modal logics, too, but come with a simplified Kripke semantics. Recently, Gómez Álvarez et al. [2022] introduced First-Order Standpoint Logic (FOSL) and showed favourable complexity results for its sentential fragments,\(^2\) which disallow modal operators being applied to formulas with free variables. In particular, adding sentential standpoints does not increase the complexity for fragments that are NP-hard. Yet, a fine-grained terminological alignment between different perspectives requires concepts preceded by modal operators, as in Axiom (7), leading to non-sentential fragments of FOSL.

Our paper is structured as follows. After introducing the syntax and semantics of Standpoint EL (\(\mathcal{SEL}\)) in a suitable normal form (Section 2), we establish our main result: satisfiability checking in \(\mathcal{SEL}\) is PTIME-complete. We show this by providing a worst-case optimal tableau-based algorithm (Section 3) that takes inspiration from the quasi-modal based methods [Wolter and Zakharyaschev, 1998] as used for \(K_{\text{ACC}}\) [Lutz et al., 2002], but differs in its specifics. Our approach builds a quasi-modal from a graph of (quasi) domain elements, which are annotated with various constraints, to then reconstruct the worlds or, in our case, precifications. We also show that introducing additional features such as empty standpoints, rigid roles, and nominals make standard reasoning tasks intractable (Section 4). In Section 5, we conclude the paper with a discussion of future work, including efficient approaches for reasoner implementations. Altogether, this paper provides a clear pathway for making scalable multiperspective ontology management possible.

An extended version of the paper with proofs of all results is available at https://arxiv.org/abs/2302.13187.

\(^2\)This includes the sentential standpoint variant of the expressive DL \(SROIQb\), a logical basis of OWL 2 DL [Motik et al., 2009b].
2 Syntax, Semantics, and Normalisation

We now introduce syntax and semantics of Standpoint EL (referred to as S_EL) and propose a normal form that is useful for subsequent algorithmic considerations.

2.1 Syntax

A Standpoint DL vocabulary is a traditional DL vocabulary consisting of sets $N_C$ of concept names, $N_R$ of role names, and $N_I$ of individual names, extended by an additional set $N_S$ of standpoint names with $s \in N_S$. A standpoint operator is of the form $\circ_s$, where $\circ_s$ denotes the "diamond" operator.

A concept term is defined via

$$ C := T \ | \ \bot \ | \ A \ | \ C_1 \cap C_2 \ | \ \exists R.C \ | \ \circ_s[C] $$

where $A \in N_C$ and $R \in N_R$. A general concept inclusion (GCI) is of the form $\circ_s[C \subseteq D]$, where $C$ and $D$ are concept terms. A concept assertion is of the form $\circ_s[R(a, b)]$, where $a, b \in N_I$. $C$ is a concept term, and $R \in N_R$. A sharpening statement is of the form $s \preceq s'$ where $s, s' \in N_S$.

A S_EL knowledge base is a tuple $\mathcal{K} = (S, T, \mathcal{A})$, where $T$ is a set of GCIs, called TBox; $\mathcal{A}$ is a set of (concept or role) assertions, called ABox; and $S$ is a set of sharpening statements, called SBox. We refer to arbitrary statements from $K$ as axioms. Since the axiom types in $S$, $T$, and $A$ are syntactically well-distinguished, we sometimes identify $K$ as $S \cup T \cup A$. Note that all axioms except sharpening statements are preceeded by modal operators ("modali-sed" for short). In case the preceding operator happens to be $\circ_s$, we may omit it.

2.2 Semantics

The semantics of $S_EL$ is defined via standpoint structures. For a Standpoint DL vocabulary $\langle N_C, N_R, N_I, N_S \rangle$, a description logic standpoint structure is a tuple $\mathcal{D} = \langle \Delta, \Pi, \sigma, \gamma \rangle$ where:

- $\Delta$ is a non-empty set, the domain of $\mathcal{D}$;
- $\Pi$ is a set, called the precisions of $\mathcal{D}$;
- $\sigma$ is a function mapping each standpoint name to a non-empty subset of $\Pi$;
- $\gamma$ is a function mapping each precisation from $\Pi$ to an "ordinary" DL interpretation $\mathcal{I} = \langle \Delta, \mathcal{I} \rangle$ over the domain $\Delta$, where the interpretation function $\mathcal{I}$ maps:
  - each concept name $A \in N_C$ to a set $A^I \subseteq \Delta$,
  - each role name $R \in N_R$ to a binary relation $R^I \subseteq \Delta \times \Delta$,
  - each individual name $a \in N_I$ to an element $a^I \in \Delta$,
  - and we require $a^\gamma(\pi) = a^\gamma(\pi')$ for all $\pi, \pi' \in \Pi$ and $a \in N_I$.

Note that by this definition, individual names (also referred to as constants) are interpreted rigidly, i.e., each individual name $a$ is assigned the same $a^\mathcal{I}$ across all precisciations $\pi \in \Pi$. We will refer to this uniform $a^\mathcal{I}$ by $a^\mathcal{D}$. For each $\pi \in \Pi$, the interpretation function $\mathcal{I}$ is extended to concept terms via structural induction as follows:

$$ (\circ_s C)^\mathcal{I} := \{ a^I \mid a \in (s^\mathcal{I}) \} $$

$$ (\pi \circ_s \mathcal{D})^\mathcal{I} := \emptyset $$

$$ (C_1 \cap C_2)^\mathcal{I} := C_1^\mathcal{I} \cap C_2^\mathcal{I} $$

$$ (\exists R.C)^\mathcal{I} := \{ a \in \Delta \mid \exists \delta \in R^I.a \} $$

We observe that modali-sed concepts $\circ_s C$ are interpreted uniformly across all precisciations $\pi \in \Pi$, which allows us to denote their extensions with $\circ_s C^\mathcal{D}$.

A DL standpoint structure $\mathcal{D}$ satisfies a sharpening statement $s \preceq s'$, written as $\mathcal{D} \models s \preceq s'$ iff $\sigma(s) \subseteq \sigma(s')$. For the other axiom types, satisfaction by $\mathcal{D}$ is defined as follows:

- $\mathcal{D} \models \circ_s[C \subseteq D]$ for all $\pi \in \sigma(s)$
- $\mathcal{D} \models \circ_s[R(a, b)]$ for all $\pi \in \sigma(s)$
- $\mathcal{D} \models \circ_s[R(a, b)]$ for all $\pi \in \sigma(s)$
- $\mathcal{D} \models \circ_s[R(a, b)]$ for all $\pi \in \sigma(s)$

As usual, $\mathcal{D}$ is a model of $S$ iff it satisfies every sharpening statement in $S$; it is a model of $T$ iff it satisfies every GCI $\tau \in T$; it is a model of $A$ iff it satisfies every assertion $\alpha \in A$; it is a model of $K = (S, T, A)$ (written $\mathcal{D} \models K$) iff it is a model of $S$ and a model of $T$ and a model of $A$. Figure 1 illustrates an example of Model 1.

![Figure 1: Model of Example 1.](image_url)

Our investigations regarding reasoning in $S_EL$ will focus on standpoint versions of the well-known standard reasoning tasks, and we will make use of variations of established techniques to (directly or indirectly) reduce all of them to the first.

**Knowledge base satisfiability:** Given a knowledge base $K$, is there a DL standpoint structure $\mathcal{D}$ such that $\mathcal{D} \models K$?

**Axiom entailment:** Given $K$ and some $SBox$, $TBox$, or $ABox$ axiom $\phi$, does $K \models \phi$ hold, that is, is it the case that for every model $\mathcal{D}$ of $K$ we have $\mathcal{D} \models \phi$?

To show that axiom entailment can be polynomially reduced to knowledge base unsatisfiability, we exhibit for every axiom type $\phi$ a knowledge base $K_{\phi}$ such that $K \models \phi$ coincides with unsatisfiability of $K \cup K_{\phi}$.

$$ K_{s \preceq s'} := \{ \circ_s T \sqsubseteq A, \circ_s [A \sqsubseteq T] \} $$

$$ K_{\circ_s[C \subseteq D]} := \{ \circ_s T \sqsubseteq C, A \sqsubseteq D, \circ_s [T \sqsubseteq \exists R.A] \} $$

$$ K_{\circ_s[R(a, b)]} := \{ \circ_s T \sqsubseteq R(a, b), A \sqsubseteq \exists R.B, \circ_s [R(a, b)] \} $$

3The square brackets $[\ldots]$ indicate the scope of the modality, as the same modalities may be used inside concept terms.

4As shown in Section 4, allowing for "empty standpoints" immediately incurs intractability, even for an otherwise empty vocabulary.
Therein, $\Box^d := \Diamond_\alpha$ and $\Diamond^d := \Box_\alpha$ are modality duals, and $\bar{A}, \bar{B}$ denote fresh concept names and $\bar{R}$ a fresh role name.

**Concept satisfiability (w.r.t. $\mathcal{K}$):** Given $\mathcal{K}$ and a modalised concept term $C$, is there a model $\mathcal{D}$ of $\mathcal{K}$ with $C^\mathcal{D} \neq \emptyset$?

This task can be solved by checking the axiom entailment $\mathcal{K} \models \Box_\alpha [C(a)]$. If the entailment holds, then $C$ is unsatisfiable w.r.t. $\mathcal{K}$, otherwise it is satisfiable.

**Instance retrieval:** Given $\mathcal{K}$ and a modalised concept term $C$, obtain all $a \in N_1$ with $a^\mathcal{D} \in C^\mathcal{D}$ for every model $\mathcal{D}$ of $\mathcal{K}$.

This task can be solved by checking, for all individuals $a$, if the entailment $\mathcal{K} \models \Box_\alpha [C(a)]$ holds and returning all such $a$.

### 2.3 Normalisation

Before we can describe a PTIME algorithm for checking satisfiability of $\mathcal{S}_{\mathcal{FL}}$ KBs, we define an appropriate normal form.

**Definition 1 (Normal Form of $\mathcal{S}_{\mathcal{FL}}$ Knowledge Bases).** A $TBox$ $\mathcal{T}$ is in normal form iff, for all its GChs $\Box_\alpha [C \sqsubseteq D]$,

1. $C$ is of the form $A$, $\exists R.A$ or $A \sqcap A'$ with $A, A' \in N_C \cup \{\top\}$,
2. $D$ is of the form $B$, $\exists R.B$, $\Box_\alpha [C \sqsubseteq D]$, or $\Box_\alpha [R(a, b)]$ with $B \in N_B \cup \{\bot\}$, and
3. at least one of $C, D$ is from $\mathcal{N}_C \cup \{\top, \bot\}$; where $R \in N_R$, and $s, s' \in N_S$.

An ABox $\mathcal{A}$ is in normal form iff all assertions have the form $\Box_\alpha [A(a)]$ or $\Box_\alpha [R(a, b)]$ for $a, b \in N_1$, $A \in N_C$, and $R \in N_R$.

$\mathcal{K} = (\mathcal{S}, \mathcal{T}, \mathcal{A})$ is in normal form whenever $\mathcal{T}$ and $\mathcal{A}$ are.

For a given $\mathcal{S}_{\mathcal{FL}}$ knowledge base $\mathcal{K} = (\mathcal{S}, \mathcal{T}, \mathcal{A})$, we can compute its normal form by exhaustively applying the following transformation rules (where “rule application” means that the axiom on the left-hand side is replaced with the set of axioms on the right-hand side),

\[
\begin{align*}
\Diamond_\alpha [C(a)] &\rightarrow \{v \in s, \Box_\alpha [C(a)]\} & (11) \\
\Diamond_\alpha [R(a, b)] &\rightarrow \{v \in s, \Box_\alpha [R(a, b)]\} & (12) \\
\Diamond_\alpha [C \sqsubseteq D] &\rightarrow \{v \in s, \Box_\alpha [C \sqsubseteq D]\} & (13) \\
\Box_\alpha [A(a)] &\rightarrow \{\Box_\alpha [A(a)], \Box_\alpha [A \sqsubseteq C]\} & (14) \\
\Box_\alpha [B \sqcap \exists R.C] &\rightarrow \{\Box_\alpha [B \sqcap \exists R.A], \Box_\alpha [A \sqsubseteq C]\} & (15) \\
\Box_\alpha [B \sqcap C \sqsubseteq D] &\rightarrow \{\Box_\alpha [B \sqcap C \sqsubseteq D]\} & (16) \\
\Box_\alpha [C \sqcap \Box_\alpha D] &\rightarrow \{\Box_\alpha [C \sqcap \Box_\alpha A], \Box_\alpha [A \sqsubseteq D]\} & (17) \\
\Box_\alpha [B \sqcap C \sqsubseteq D] &\rightarrow \emptyset \text{ and } & (18) \\
\Box_\alpha [\exists R.C] &\rightarrow \{\Box_\alpha [C \sqsubseteq A], \Box_\alpha [R.A \sqsubseteq D]\} & (19) \\
\Box_\alpha [C \sqcap D \sqsubseteq E] &\rightarrow \{\Box_\alpha [C \sqsubseteq A], \Box_\alpha [A \sqsubseteq D \sqsubseteq E]\} & (20) \\
\Box_\alpha [\Box_\alpha C \sqsubseteq D] &\rightarrow \emptyset \text{ and } & (21) \\
\Box_\alpha [\Box_\alpha C \sqsubseteq D] &\rightarrow \{\Box_\alpha [C \sqsubseteq A], \Box_\alpha [A \sqsubseteq D]\} & (22) \\
\end{align*}
\]

Therein, $\bar{C}$ and $\bar{D}$ stand for complex concept terms not contained in $N_C \cup \{\top\}$, whereas each occurrence of $A$ on a right-hand side denotes the introduction of a fresh concept name; likewise, $v, u, v_0$, and $v_1$ denote the introduction of a fresh standpoint name. Rule (20) is applied modulo commutativity of $\sqcap$. Most of the transformation rules should be intuitive (for the first three, keep in mind that standpoints must be nonempty). A notable exception is Rule (22), which is crucial to remove boxes occurring with negative polarity. It draws some high-level inspiration from existing work on non-vacuous left-hand-side universal quantifiers in Horn DLs [Carra et al., 2014], yet the argument for its correctness requires a much more intricate model-theoretic construction and crucially hinges on “Hornness” of $\mathcal{K}$ and nonemptiness of standpoints. A careful analysis yields that the transformation has the desired semantic and computational properties.

**Lemma 1.** Every $\mathcal{S}_{\mathcal{FL}}$ knowledge base $\mathcal{K}$ can be transformed into a $\mathcal{S}_{\mathcal{FL}}$ knowledge base $\mathcal{K}'$ in normal form such that

- $\mathcal{K}'$ is a $\mathcal{S}_{\mathcal{FL}}$-conservative extension of $\mathcal{K}$,
- the size of $\mathcal{K}'$ is at most linear in the size of $\mathcal{K}$, and
- the transformation can be computed in PTIME.

While $\mathcal{K}'$ being a $\mathcal{S}_{\mathcal{FL}}$-conservative extension of $\mathcal{K}$ brings about various valuable properties, what matters for our purposes is that this implies equisatisfiability of $\mathcal{K}$ and $\mathcal{K}'$, thus we will not go into details about conservative extensions.

### 3 A Tableau Algorithm for Standpoint $\mathcal{EL}$

We present a PTIME tableau decision algorithm for $\mathcal{S}_{\mathcal{EL}}$. Complexity-optimal tableau algorithms have been proposed for description logics with modal operators applied to concepts and axioms such as $\mathcal{K}_{\mathcal{A}_{\mathcal{EL}}}$ [Lutz et al., 2002], which is known to be in NEXPTIME. Our case cannot be treated in the same way, as we need to take greater care to show tractability in the end. Lutz et al. [2002] build a “modal-model” from a tree of “quasi-worlds”, which is not as easily applicable in our case, so we follow a dual approach: we will build a quasi-model from a completion graph of (quasi) domain elements, where each of the latter is associated to a constraint system with assembled information regarding one individual’s specifics in each precisification. We begin with some definitions.

Given a $\mathcal{S}_{\mathcal{EL}}$ knowledge base $\mathcal{K}$, denote by

- $\mathcal{ST}_\mathcal{K}$ the elements of $N_2$ occurring in $\mathcal{K}$ together with $\ast$,
- $\mathcal{IN}_{\mathcal{K}}$ the set of all individual names occurring in $\mathcal{K}$,
- $\mathcal{BC}_\mathcal{K}$ (basic concepts) the concept names used in $\mathcal{K}$, plus $\top$,
- $\mathcal{C}_\mathcal{K}$ the set of concept terms used in $\mathcal{K}$ (with $\mathcal{BC}_\mathcal{K} \subseteq \mathcal{C}_\mathcal{K}$),
- $\mathcal{SF}_\mathcal{K}$ the set of subformulas of $\mathcal{K}$, consisting of all axioms of $\mathcal{K}$ with and without their outer standpoint modality.

A constraint for $\mathcal{K}$ is of the form $(x : C), (x : a), (x : \phi)$, or $(x : s)$, where $x$ is a variable, $C \in \mathcal{C}_\mathcal{K}$ a concept, $a \in \mathcal{IN}_{\mathcal{K}}$ an individual, $\phi \in \mathcal{SF}_\mathcal{K}$ a formula, and $s \in \mathcal{ST}_\mathcal{K}$ a standpoint name. Constraint systems are finite sets of constraints.

**Definition 2 ((Initial) Constraint System for $\mathcal{K}$).** The initial constraint system for $\mathcal{K}$, called $\mathcal{S}^0_\mathcal{K}$, is the set

$$\{x_2 : \ast, x_3 : \top, x_5 : \phi, x_6 : s \mid \phi \in \mathcal{K}, s \in \mathcal{ST}_\mathcal{K}\}$$

A constraint system for $\mathcal{K}$ is a finite set $\mathcal{S}$ of constraints for $\mathcal{K}$ such that $\mathcal{S}^0_\mathcal{K} \subseteq \mathcal{S}$ and $\{x : \ast, x : \top\} \subseteq \mathcal{S}$ for each $x$ in $\mathcal{S}$. For a variable $x$, let $\mathcal{S}_{\mathcal{K}}(x) = \{s \mid (x : s) \in \mathcal{S} \}$ be the standpoint signature of $x$ in $\mathcal{S}$.

Intuitively, each constraint system $\mathcal{S}$ produced by the algorithm corresponds to a domain element $\varepsilon \in \Delta$ and each variable $x$ in $\mathcal{S}$ corresponds to some precisification $\pi$.

---

\[^3\text{For better legibility, we will sometimes omit the parentheses.}\]
Moreover, each constraint \( x : X \) in \( S \) encodes information of \( \in \) in \( \pi \). Namely, \( X \) may be an axiom that holds in \( \pi \), a standpoint that contains \( \pi \), or a concept expression of which \( \in \) is an instance in \( \pi \). Initialising one variable per standpoint in the initial constraint system guarantees non-empty standpoints.

A constraint system is complete iff it satisfies every local completion rule from Figure 3. Local completion rules operate on constraint systems, while global rules involve more than one constraint system and operate on completion graphs.

**Definition 3 (Completion Graph).** An element label is a set \( L \) of triples of the form \((C, s, x)\), where \( C \in \mathbb{B}_K \) is a concept, \( s \in \mathcal{S}_K \) is a set of standpoints, and \( x \) is a variable.

A quasi-role for a set \( L \) is a tuple \((\varepsilon, v, \varepsilon', v', R)\) where \( v \) and \( v' \) are variables, \( \varepsilon, \varepsilon' \in \Delta \), and \( R \) is a role name in \( K \).

A completion graph for \( K \) is a tuple \( \mathcal{G}_C = (\Delta, \mathcal{S}, L, R) \), with \( \Delta \) a non-empty set of elements; \( L \) a map from \( \Delta \) into constraint systems; \( \mathcal{S} \) a map from \( \Delta \) into element labels; and \( R \) a set of quasi-roles such that

- for all \((\varepsilon, v, \varepsilon', v', R) \in R \), \((v : s) \in S(\varepsilon) \) iff \((v' : s) \in S(\varepsilon') \);
- if \((C, s, x) \in L(\varepsilon) \), then \( \{x : C\} \cup \{x : s | s \in s\} \subseteq S(\varepsilon) \).

For convenience of presentation, we use the shortcut \( \mathcal{S}(\varepsilon)(v) \) for \( S(\varepsilon(v)) \) and for any \( \mathcal{G}_C = (\Delta, \mathcal{S}, L, R) \), we will refer to all \( \varepsilon \in \Delta \) simply as elements of \( \mathcal{G}_C \).

\( \mathcal{G}_C \) is said to be locally complete iff for every element \( \varepsilon \) in \( \mathcal{G}_C \), \( \mathcal{S}(\varepsilon) \) is complete, and we call \( \mathcal{G}_C \) globally complete iff it is locally complete and no global completion rule (see Figure 3) is applicable to \( \mathcal{G}_C \) as a whole.

Intuitively, the next definition poses some global requirements for \( \mathcal{G}_C \) to warrant its eligibility as a model-substitute.

**Definition 4 (Coherence).** Let \( \mathcal{G}_C = (\Delta, \mathcal{S}, L, R) \) be a completion graph for \( K \). \( \mathcal{G}_C \) is called coherent iff

- for each \( a \in \mathbb{I}_K \) there is a unique element \( \varepsilon_a \in \Delta \) such that \((\varepsilon, a) \in S(\varepsilon_a) \) for all variables \( v \) in \( S(\varepsilon_a) \);
- for each \( \varepsilon, \varepsilon' \in \Delta \) and each variable \( v \) contained in \( S(\varepsilon) \), \( S(\varepsilon') \) contains some \( v' \) such that \( \mathcal{S}(\varepsilon)(v) = \mathcal{S}(\varepsilon')(v') \), and
- if \((v : \phi) \in S(\varepsilon) \) and \( \mathcal{S}(\varepsilon)(v) = \mathcal{S}(\varepsilon')(v') \), then \((v' : \phi) \in S(\varepsilon') \).

As usual in tableaux, inconsistencies emerge as clashes.

**Definition 5 (Clash).** A clash is a constraint of the form \((x : \bot) \). A completion graph \( \mathcal{G}_C \) is said to contain a clash iff \( S(\varepsilon) \) does for some \( \varepsilon \) in \( \mathcal{G}_C \). Constraint systems or completion graphs not containing clashes are called clash-free.

### 3.1 The Algorithm

To decide whether a given \( \mathbb{S}_{\mathbb{E}_C} \) knowledge base \( K \) in normal form is satisfiable, we form the initial completion graph \( \mathcal{G}_K \) with \( R = \emptyset \) and \( \Delta \) consisting of an element \( \varepsilon_T \) with \( L(\varepsilon_T) = \emptyset \) and \( S(\varepsilon_T) = S_0 \), and for every \( a \in \mathbb{I}_K \) an element \( \varepsilon_a \) with \( L(\varepsilon_a) = \emptyset \) and \( S(\varepsilon_a) = S_0 \cup \{(x : a) | (x : T) \in S_0\} \).

After that, we repeatedly apply the local and global completion rules from Figure 3, where \( LL \) rules have the highest priority, followed by \( LC \), \( GN \), and \( GG \) rules, in that order. After each rule application, we check if \( \mathcal{G}_C \) contains a clash and terminate with answer “unsatisfiable” should this be the case. If we arrive at a clash-free \( \mathcal{G}_C \) with no more rules applicable, the algorithm terminates and returns “satisfiable”.

The first table in Figure 2 illustrates the generated tableau for the \( \mathbb{S}_{\mathbb{E}_C} \) knowledge base \( K = \{T \subseteq \emptyset, C \subseteq \exists R.D\} \).

### 3.2 Quasi-Models and Quasi-Satisfiability

This section sketches how special structures, called (dual) quasi-models can serve as proxies for proper \( \mathbb{S}_{\mathbb{E}_C} \) models.

**Definition 6 (Run, Quasi-model).** Let \( \mathcal{G}_C = (\Delta, \mathcal{S}, L, R) \) be a completion graph. A run \( r \) in \( \mathcal{G}_C \) is a function mapping each element \( \varepsilon \) in \( \Delta \) to a variable of \( S(\varepsilon) \), such that

1. If \((r(\varepsilon) : s) \in S(\varepsilon) \), then \((r(\varepsilon') : s) \in S(\varepsilon') \)
   - for all \( \varepsilon, \varepsilon' \in \Delta \) and \( s \in \mathcal{S}_K \).

2. If \((\varepsilon, v, \varepsilon', v', R) \in R \) and \( r(\varepsilon) = v \), then \( r(\varepsilon') = v' \), and

3. If \((r(\varepsilon) : \exists R.C) \in S(\varepsilon) \), there exists some \( \varepsilon' \) in \( \Delta \) with
   - \((\varepsilon, r(\varepsilon), \varepsilon', r(\varepsilon'), R) \in R \) and \( (r(\varepsilon') : C) \in S(\varepsilon') \).

A quasi-model of \( K \) is a tuple \( Q = (\Delta, \mathcal{S}, L, R, \Gamma) \) where \((\Delta, \mathcal{S}, L, R, \Gamma) \) is a globally complete, coherent and clash-free completion graph for \( K \), and \( \Gamma \) a set of runs in \((\Delta, \mathcal{S}, L, R) \) such that for every \( \varepsilon \) in \( \Delta \) and variable \( v \) in \( S(\varepsilon) \), there is a run \( r \) in \( \Gamma \) such that \( r(\varepsilon) = v \). \( K \) is called quasi-satisfiable iff \( K \) has a quasi-model.

In a nutshell, runs serve the purpose of lining up “compatible” variables, one from each individual constraint system, in a way that precisifications can be constructed (cf. Figure 2: in \( Q \), a “compatible” set of runs over \( CG \) is displayed using dotted lines). With these notions in place, we can establish the desired result.

**Theorem 2.** A \( \mathbb{S}_{\mathbb{E}_C} \) knowledge base \( K \) is satisfiable iff it is quasi-satisfiable.
Local labelling (LL) rule:

\[ R_\text{LL} : \text{If } \{ x : s, x' : s' \} \subseteq S \text{ but } (x : s') \not\in S, \text{ then set } S := S \cup \{ x : s' \}. \]

Local content (LC) rules:

\[ R_\text{LC} : \text{If } \{ x : C, x : D \} \subseteq S, \text{ and } C \cap D \subseteq \mathbb{C}_K \text{, then set } S := S \cup \{ x : C \cap D \}. \]

Global non-generating (GN) rules:

\[ R_\text{GN} : \]

- If \( (x : C) \in S(e), (x' : x, x, R) \in R, \) then set \( S(e') := S(e') \cup \{ x' : R.C \}. \)
- If \( (x, x', x, R) \in S(e), \) and \( \langle x, x', x, R \rangle \not\in R, \) then set \( S(e') := S(e') \cup \{ x, x', x, R \}. \)

Global generating (GG) rule:

\[ R_\text{GG} : \]

- If \( (x : R.C) \in S(e), \) then \( S(e') := S(e') \cup \{ x : C, x', x'' \} \) is bounded by the number of elements in each completion graph, i.e. at most \( \|K\| \).

3.3 Polytyme Termination and Correctness

Next, we give an overview of our argument why our algorithm runs in polynomial time with respect to \( \|K\| \), the size of its input \( K \). We observe that the number of potential elements of any completion graph \( CG \) constructed by our algorithm is bounded by \( 3 \|K\|^2 \). We also account that the number of \( (\|K\|^2) \) of domain elements of any completion graph \( CG \) constructed by our algorithm is bounded by \( 3 \|K\|^2 \).

When the number of variables used in any single \( S(\varepsilon) \) is bounded by \( 2 \|K\|^2 \) and the number of constraints in \( S(\varepsilon) \) by \( 2 \|K\|^3 \) (10). Now, the number of applications of \( R_\text{LL} \) is bounded by the number of elements in each completion graph, i.e. at most \( 3 \|K\|^2 \) in view of (1). Since the rules \( R_\text{LC}, R_\text{GN}, R_\text{GG}, R_\text{GG} \) and \( R_\text{GG} \) produce one or more new constraints in an element, the number of applications of such rules per element is bounded by \( 2 \|K\|^2 \), as the total number of rule applications is bounded by the result applications per element multiplied by the bound on elements, together with the bound on \( R_\text{LL} \), which gives us 

\( \|K\|^2(3) + 2\|K\|^2(3) + 3\|K\|^2 \leq (27\|K\|^6). \)

3.4 Proving completeness

We now ready to establish correctness of our decision algorithm, by showing its soundness and completeness. For both directions, Theorem 2 will come in handy. As usual, the soundness part of our argument is the easier one.

\[ \square \]

3.5 Proving completeness requires significantly more work. We make use of a notion that, intuitively, formalizes the idea that a completion graph \( CG \) under development is “in sync” with a quasi-model \( Q \) of the same knowledge base, where \( Q \) can be conceived as a model-theoretic “upper bound” of \( CG \).

Definition 7 (\( \mathbb{Q} \)-compatibility). Let \( K \) be a \( \mathbb{S}\mathcal{L} \) knowledge base and \( Q = \langle \Delta, \mathcal{S}, L, \mathcal{R}, \mathcal{G} \rangle \) be a quasi-model for \( K \). A completion graph \( CG = \langle \Delta, \mathcal{S}, L, \mathcal{R}, \mathcal{G} \rangle \) of \( K \) is called \( \mathbb{Q} \)-compatible if there is a total relation map \( \mu \subseteq \Delta \times \mathcal{G} \) where

- for all \( g \in \Delta \) and \( e \in \Delta \), if both \( L(\varepsilon) \) and \( \{ a : (x : a) \in \mathcal{S}(\varepsilon) \} \), then \( (g, \varepsilon) \in \mu. \)

\[ \mu(g, \varepsilon)(x) \text{ is a surjective function } \mu(g, \varepsilon) \text{ from } \mathcal{S}(\varepsilon) \text{ to the variables } \mathcal{S}(\varepsilon) \text{ such that } \]

- \( \mu(g, \varepsilon)(x) : \mathcal{S}(\varepsilon) \text{ implies } (v : s) \in \mathcal{S}(\varepsilon) \)
- \( \mu(g, \varepsilon)(x) : \mathcal{S}(\varepsilon) \text{ implies } (v : \Phi) \in \mathcal{S}(\varepsilon) \)
- \( \mu(g, \varepsilon)(x) : \mathcal{S}(\varepsilon) \text{ implies } (g, x, g', x') \in \mathcal{R} \) for some \( (g, \varepsilon), (g', \varepsilon') \in \mu \) with \( \mu(g, \varepsilon)(x) = x \) and \( \mu(g', \varepsilon')(y') = x'. \)

\[ \square \]
With this definition, we can establish two important insights:

- The tableau algorithm’s initial completion graph \( CG \) is \( Q \)-compatible for any quasimodel \( Q \) of \( K \).
- Applications of tableau rules preserve \( Q \)-compatibility. This entails that the completion graph maintained in the algorithm will be \( Q \)-compatible at all times, thus also upon termination. We exploit this insight to show completeness.

**Theorem 5 (Completeness).** If a \( S_{EL} \) knowledge base \( K \) is satisfiable, the tableau algorithm will construct a globally complete, coherent, and clash-free completion graph for \( K \).

**Proof.** If \( K \) is satisfiable then by Theorem 2, there is a quasimodel \( Q \) for \( K \). According to Theorem 3, we can obtain a globally complete completion graph \( CG \) after polynomially many applications of the tableau rules, which, as just discussed, is \( Q \)-compatible. It must thus also be clash-free, because otherwise there were an element \( g \) and variable \( x \), with \((x: \bot) \in S(g)\), and thus there is \((g, \varepsilon) \in \mu \) and \( \mu_{x, \varepsilon} \) such that \((\mu_{x, \varepsilon}(x): \bot) \in S^Q(\varepsilon)\), which is a contradiction because \( Q \) is a quasi-model. It is not hard to show that \( CG \) is also coherent, whence we can conclude that \( CG \) is a globally complete, coherent, and clash-free completion graph for \( K \). ∎

Together with the well-known \( \text{PTIME} \)-hardness of the satisfiability problem in (standpoint-free) \( EL \), we have therefore established \( \text{PTIME} \)-completeness of \( S_{EL} \) and exhibited a worst-case optimal algorithm for it.

### 4 Intractable Extensions

While the shown tractability of reasoning in \( S_{EL} \) is good news, one might ask if one could include more modelling features or relax certain side conditions and still preserve tractability. This section shows that tractability can be easily lost (at least under standard complexity-theoretic assumptions).

#### 4.1 Empty Standpoints

While it may make sense on a philosophical level, one might wonder whether the constraint that \( \sigma(s) \) needs to be nonempty for every \( s \in ST_c \) has an impact on tractability. In fact, dropping this constraint, obtaining a logic \( S^0_{EL} \) with the same syntax but modified semantics, would increase expressivity (standpoint non-emptiness could still be enforced in \( S_{EL} \) by asserting \( \top \subseteq \Phi^T \) for every \( s \in ST_c \)). However, satisfiability in \( S^0_{EL} \) turns out to be \( \text{NP} \)-hard, even when disallowing usage of concept and role names entirely. The key insight is that both \( \Phi^T \) and its negation \( \square \bot \) can be expressed as \( S^0_{EL} \) concepts gives rise to the following reduction from \( 3\text{SAT} \): Assume an instance \( \phi = \bigvee C_1 \land \ldots \land \bigvee C_n \) of \( 3\text{SAT} \) containing \( n \) clauses (i.e., disjunctions of literals) \( \bigvee C_j \) over the propositional variables \( P = \{ p_1, \ldots, p_k \} \). We note that \( \phi \) is equivalent to \(( \bigwedge C_1 \rightarrow \text{false} ) \land \ldots \land ( \bigwedge C_n \rightarrow \text{false} ) \), where \( \overline{C}_i \) is obtained from \( C_j \) by replacing every literal by its negated version. Let now \( \{ s_1, \ldots, s_k \} \) be a set of standpoint names and, for any literal \( \ell \) over \( P \), define

\[
L_\ell = \begin{cases}
\Phi^T & \text{if } \ell = p_i,
\square \bot & \text{if } \ell = \overline{p}_i.
\end{cases}
\]

Then, \( \phi \) is satisfiable iff the following \( S^0_{EL} \) knowledge base is:

\[
K_\phi = \{ L_\ell \cap L_{\ell'} \cap L_{\ell''} \subseteq \bot \mid (\ell, \ell', \ell'') = \overline{C}_j, 1 \leq j \leq n \}.
\]

#### 4.2 Rigid Roles

\( S_{EL} \) allows knowledge engineers to globally enforce rigidity of specific concepts through axioms of the shape \( A \sqsubseteq \square A \). (This is in contrast to e.g. \( K_a \text{ALC} \), where rigidity of concepts can only be expressed relative to a given formula.) In a similar manner, rigidity of roles (i.e., the interpretation of certain distinguished roles being the same throughout all precisifications) would represent a desirable modelling feature. Other modal extensions of DLs have easily been shown to even become undecidable when this feature is permitted, but as \( S_{EL} \) uses a much simplified semantics on the modal dimension, these results do not carry over to \( S_{EL} \). Yet, we will show that just the presence of one distinguished rigid role \( R \) causes \( S_{EL} \) to become intractable as satisfiability turns \( \text{coNP} \)-hard.

To demonstrate this, we reduce \( 3\text{SAT} \) to \( \text{KB} \) unsatisfiability. As above, assume an instance \( \phi = \bigvee C_1 \land \ldots \land \bigvee C_n \) of \( 3\text{SAT} \) over propositional variables \( P = \{ p_1, \ldots, p_k \} \). Then \( \phi \) is satisfiable iff the following \( S_{EL} \) TBox is unsatisfiable (with all axioms instantiated for \( 1 \leq i \leq k \)):

\[
\begin{align*}
T & \subseteq ST. L_0 \\
L_{i-1} & \sqsubseteq R.\square_s (L_i \cap T_{p_i}) \\
L_{i-1} & \sqsubseteq R.\square_s (L_i \cap T_{\overline{p}_i}) \\
L_k & \sqsubseteq q. S \\
\exists R.(T_{p_i} \sqsubseteq S) & \subseteq (T_{p_i} \sqsubseteq S) \\
\exists R.(T_{\overline{p}_i} \sqsubseteq S) & \subseteq (T_{\overline{p}_i} \sqsubseteq S) \\
L_0 \cap T_\ell & \sqsubseteq T_{C_j} \quad \text{for all } \ell \in C_j \\
T_{C_j} & \cap \ldots \cap T_{C_n} \subseteq \bot
\end{align*}
\]

The intuition behind this is to construct a “decision tree” where \( R \) acts as child relation and which (thanks to \( R \)’s rigidity) is synchronized across all precisifications. The leaves of this tree correspond to all possible truth assignments for \( p_1, \ldots, p_k \). Then we make sure that for every leaf, there exists a precisification, where this leaf is “selected” (indicated by the non-rigid concept \( S \)). This “selection marker” \( S \) is propagated from children to parents, taking along information on the truth assignments. Finally, when all the information on the path leading to the selected node has been accumulated in the root node, it is checked if this information corresponds to an assignment satisfying \( \phi \). If so, a “global inconsistency” is triggered. Figure 4 provides a small example.

![Figure 4: Standpoint structure, witnessing the unsatisfiability of the propositional formula \( \phi = (\neg p_1 \lor p_2) \land (p_1) \land (\neg p_2) \). All arrows indicate \( R \); role \( T \) is omitted for better readability. The concept names displayed in the top row hold throughout all precisifications.](image-url)
4.3 Nominal Concepts

Nominal concepts are a modelling feature widely used in ontology languages. For an individual \( o \), the nominal concept \( \{o\} \) refers to the singleton set \( \{o\} \). Let \( \mathcal{EL} \) denote \( \mathcal{EL} \) extended by nominal concepts. Several formalisms subsuming \( \mathcal{EL} \), including OWL 2 EL, are known to allow for tractable reasoning [Baader et al., 2005; Krötzsch, 2010]. However, in the presence of standpoints, nominals prove to be detrimental for the reasoning complexity: satisfiability of \( \mathcal{S}_{\mathcal{EL}} \) TBoxes using just one nominal concept \( \{o\} \) turns out to be \( \text{PTIME} \)-hard and thus definitely harder than for \( \mathcal{S}_{\mathcal{EL}} \). This can be shown by a \( \text{PTIME} \) reduction of satisfiability for Horn-\( \mathcal{ALC} \) TBoxes (which is known to be \( \text{PTIME} \)-complete [Krötzsch et al., 2013]) to satisfiability of \( \mathcal{S}_{\mathcal{EL}} \) TBoxes with just one standpoint (the global one) and one nominal concept \( \{o\} \). To this end, recall that any Horn-\( \mathcal{ALC} \) TBox can be normalised in \( \text{PTIME} \) to consist of only axioms of the following shapes:

\[
A \sqsubseteq B \quad A \sqcap B \sqsubseteq C \quad \exists R.A \sqsubseteq B \quad A \sqsubseteq \forall R.B
\]

where \( A, B, C \) can be concept names, \( T \), or \( \bot \). From a normalised Horn-\( \mathcal{ALC} \) TBox \( \mathcal{T} \), we obtain the target \( \mathcal{S}_{\mathcal{EL}} \) TBox \( \mathcal{T}' \) by (i) declaring every original concept name as rigid via the axiom \( A \sqsubseteq \Box A \) as well as (ii) replacing every axiom of the shape \( A \sqsubseteq \forall R.B \) by the axiom

\[
A \sqsubseteq \Diamond_{\text{src}}((\exists \text{src} \{o\}) \sqcap (\exists R.(B \sqcap \exists \text{tgt} \{o\})))
\]

(introducing two fresh role names \( \text{src} \) and \( \text{tgt} \), and replacing every axiom of the shape \( A \sqsubseteq \forall R.B \) by the two axioms

\[
A \sqcap \exists R.\top \sqsubseteq (\exists \text{src} \{o\}) \sqcap (\exists R.(B \sqcap \exists \text{tgt} \{o\})) \quad \text{and} \quad \exists \text{tgt.} \cdot \tilde{B} \sqsubseteq B,
\]

introducing a (non-rigid) copy \( \tilde{A} \) for every original concept name \( A \). With this polytime translation, satisfiability of the Horn-\( \mathcal{ALC} \) TBox \( \mathcal{T} \) and the \( \mathcal{S}_{\mathcal{EL}} \) TBox \( \mathcal{T}' \) coincide.

The intuition behind this is to emulate the DL-model of \( \mathcal{T} \) by a standpoint structure where every role connection gets assigned its own precification, wherein the nominal acts as “witness” for this connection and is linked to source and target element by dedicated roles. Then the “forward transfer” of information by axioms of the shape \( A \sqsubseteq \forall R.B \) can be realized using the nominal element as “proxy” (cf. Figure 5).

5 Conclusion and Future Work

In this paper, we introduced Standpoint \( \mathcal{EL} \), a new lightweight member of the emerging family of standpoint logics. We described the new modelling and reasoning capabilities it brings to large-scale ontology management and established a \( \text{PTIME} \) (and thus worst-case optimal) tableau-based decision procedure for standard reasoning tasks. We also demonstrated that certain extensions of \( \mathcal{S}_{\mathcal{EL}} \), which would be desirable from an expressivity point of view, inevitably come with a loss of tractability (sometimes under the assumption \( P \not= NP \)).

Yet, several modelling features can be accommodated into \( \mathcal{S}_{\mathcal{EL}} \) without endangering tractability. For instance, from a practical perspective, it appears very desirable and advantageous modelling-wise, if not just single axioms, but whole axiom sets (up to whole knowledge bases) could be preceded by standpoint modalities. By definition, an axiom of the type \( \Box_{\text{src}} K \) can be equivalently rewritten into the axiom set \( \{\Box_{\text{src}} \phi \mid \phi \in K\} \). While something alike is not immediately possible for axioms of the type \( \Diamond_{\text{src}} K \), our normalization rule for diamond-precended axioms can be lifted and thus \( \Diamond_{\text{src}} K \) can be rewritten to \( \Box_{\text{src}} K \) (and further to \( \{\Box_{\text{src}} \phi \mid \phi \in K\} \) upon introducing a fresh standpoint name \( s' \) and asserting \( s' \models s \). Thus standpoint-modality-annotated knowledge bases come essentially free for in \( \mathcal{S}_{\mathcal{EL}} \). In fact, we already made tacit use of this modelling feature in Axiom 9 and Axiom 10 of our initial example.

Moreover, we are confident that, as opposed to nominal concepts, other modelling features of OWL 2 EL can be added to \( \mathcal{S}_{\mathcal{EL}} \) without harming tractability. These include complex role inclusions (also called role-chain axioms) such as \( \text{FindingSite} :: \text{PartOf} 
\text{FindingSite} \) and the self-concept as in \( \text{ApoptoticCell} :: \text{DestroysSelf} \).

Beyond exploring the tractability boundaries, potential next research endeavours include to investigate diverse feasible strategies for developing a \( \mathcal{S}_{\mathcal{EL}} \) reasoner. Options worth pursuing toward this goal include

- to implement our tableau algorithm from scratch or by modifying existing open-source tableaux systems,
- to design a deduction calculus over normalised axioms that can be translated into a datalog program, akin to the approach of Krötzsch [2010], then utilizing a state-of-the-art datalog engine like VLog [Urbani et al., 2014], or
- to find a reduction to reasoning in standpoint-free (\( \text{PTIME} \) extensions of) \( \mathcal{EL} \) that is supported by existing reasoners (such as ELP [Kazakov et al., 2014]).

With reasoners in place, appropriate experiments can be conducted to assess practical feasibility and scalability.

In addition to the \( \mathcal{EL} \) family, further popular and computationally lightweight formalisms exist, such as the tractable profiles OWL 2 RL and OWL 2 QL [Motik et al., 2009]. It would be interesting to investigate options to extend these by standpoint reasoning without sacrificing tractability. More generally, we intend to research the effect of adding standpoints to KR languages – light- or heavyweight – in terms of computational properties and expressivity as well as avenues for implementing efficient reasoners for them. Beyond the large and versatile family of description logics, a worthwhile target for these efforts would be existential rule languages.
Acknowledgements

This work was supported by funding from BMBF within project KIMEDS (grant no. GW0552B), by the European Research Council through the ERC Consolidator Grant DeciGUT (771779), and by the Federal Ministry of Education and Research of Germany and the Sächsische Staatsministerium für Wissenschaft Kultur und Tourismus through the Center of Excellence for AI research ScaDS.AI.

References


