Proceedings of the 21st International Conference on Principles of Knowledge Representation and Reasoning
Main Track

Extending Description Logics with Generic Concepts – the Tale of Two Semantics

Joshua Hirschbrunn[1], **Yevgeny Kazakov**[1]
[1]University of Ulm, Germany
{joshua.hirschbrunn, yevgeny.kazakov}@uni-ulm.de

Abstract

Description Logic (DL) ontologies often need to model similar properties for different concepts. Taking inspiration from generic classes in object-oriented programming, we introduce concept parameters to describe related concepts. For example, $\mathsf{LocalAnesthesia}[\mathsf{Eye}]$ and $\mathsf{LocalAnesthesia}[\mathsf{Knee}]$ can be used to describe the anesthesia of an eye or a knee, respectively. The main benefit of generic concepts is to be able to describe general properties, for example, that every local anesthesia is done by applying an anesthetic drug. We propose to use generic concepts, such as $\mathsf{LocalAnesthesia}[X]$ to define such properties, where a concept variable X can be replaced with suitable concepts. To capture the intended meaning of generic concepts, we define two semantics for this extension: the schema semantics, in which concept variables represent arbitrary concepts from a specific language, and the second-order semantics, in which variables represent arbitrary subsets of the domain. Generally, the second-order semantics gives more logical consequences, but the schema semantics allows a reduction to the classical DL reasoning. To combine the benefits of both semantics, we define a useful extension of the DL $\mathcal{EL}$, for which both semantics coincide, and a further restriction in which the entailment problem is decidable.

1 Introduction and Motivation

Complex ontologies often contain groups of structurally similar axioms. For example, one of the largest medical ontologies SNOMED CT[1] defines various medical procedures characterized by the body location in which they are applied and the medical substance used. For example, $\mathsf{DesensitizingTooth} \sqsubseteq \exists\mathsf{site}.\mathsf{ToothStructure} \sqcap \exists\mathsf{substance}.\mathsf{TopicalAnesthethic}$. When modeling such procedures in object-oriented languages, such as Java, one would typically use *generic classes* (see, e.g., (Garcia et al. 2003)) such as `Application<L,S>`, in which the *type parameter* `L` refers to the location of the application and `S` refers to the applied substance. These parameters can be used, e.g., as return types of functions like `getSite()` and `getSubstance()`. A concrete medical procedure, such as `DesensitizingTooth` can then be defined as `Application<Tooth,TopicalAnesthethic>`. Generic classes offer many advantages, such as code reusability, readability, and prevention of modeling errors.

[1]https://www.snomed.org/

In this paper, we look at how similar generic concepts can be defined in Description Logics (DLs). First, we allow atomic concepts to be *parameterized* with other concepts. For example, we can write $\mathsf{Heart}[\mathsf{Dog}]$ to represent the hearts of dogs or even $\mathsf{Heart}[\exists\mathsf{owns}.\mathsf{Dog}]$ or $\mathsf{Heart}[\mathsf{Owner}[\mathsf{Dog}]]$ to represent the hearts of dog owners. Second, we allow the use of *concept variables* in places of concepts to define the generic properties of (parameterized) concepts. For example, we can define the generic owner concept using an equivalence axiom $\mathsf{Owner}[X] \equiv \exists\mathsf{owns}.X$. Intuitively, here the concept variable X can be replaced with any other (parameterized) concept, obtaining more specific axioms such as $\mathsf{Owner}[\mathsf{Dog}] \equiv \exists\mathsf{owns}.\mathsf{Dog}$. We also allow the usage of concept variables without parameterized concepts, e.g., $\exists\mathsf{owns}.(X \sqcap \mathsf{Pet}) \sqsubseteq \exists\mathsf{feeds}.X$ as this axiom could anyway be expressed by axiom $\mathsf{Owner}[X \sqcap \mathsf{Pet}] \sqsubseteq \exists\mathsf{feeds}.X$.

One possibility for defining the semantics of our language extension is to regard parameterized concepts, such as $\mathsf{Heart}[C]$, as atomic concepts, interpreted independently for different values of C. Axioms with concept variables, such as $\mathsf{Owner}[X] \equiv \exists\mathsf{owns}.X$, can be regarded as *axiom schemata* that represent the set of axioms $\mathsf{Owner}[C] \equiv \exists\mathsf{owns}.C$ obtained after replacement of variables with concepts. We call the semantics defined in this way the *schema semantics*.

One advantage of the schema semantics for concept variables, is that any DL reasoning procedure could, in principle, be used for checking entailment in the corresponding generic DL extension, by systematically generating instances of axiom schemata and checking entailment from the resulting (increasing) sets of ordinary axioms (again, treating parametrized concepts as atomic concept names). This immediately implies *semi-decidability* of the schema extension and, in cases limited to a finite number of replacement concepts, also *decidability* and *complexity* results.

The schema semantics, however, also has disadvantages, one is, that the entailments depend on the choices of concepts used for variable replacements. Consider, for example, the axiom schema $\top \sqsubseteq \exists r.X$. If we allow only replacements of X with $\mathcal{EL}$ concepts C, the resulting set of axioms would be satisfiable, since there is a model (with one element) that interprets every $\mathcal{EL}$ concept (including every $\exists r.C$) by the whole domain. If, however, we additionally allow X to be replaced by $\bot$, i.e., by taking concepts from $\mathcal{EL}^{\bot}$, then, clearly, this schema becomes unsatisfiable. As a consequence, this axiom schema results in new $\mathcal{EL}$ logical conclusions (e.g., $A \sqsubseteq B$) when viewing it in the context of a larger language (e.g., $\mathcal{EL}^{\bot}$). In Section 3 we give a similar example where an $\mathcal{EL}$ schema results in new $\mathcal{EL}$ consequences when using $\mathcal{ALC}$ concept replacements, this time, without making axioms inconsistent.[2]

Another problem with schema semantics is the preservation of equivalence. Again, assume that we have an equivalence axiom $\mathsf{Owner}[\mathsf{Dog}] \equiv \exists \mathsf{owns}.\mathsf{Dog}$. Should then the concepts $\mathsf{Heart}[\exists \mathsf{owns}.\mathsf{Dog}]$ and $\mathsf{Heart}[\mathsf{Owner}[\mathsf{Dog}]]$ be equivalent as well? Intuitively, both concepts describe the same set of hearts of dog owners in two different ways. However, under the schema semantics, both concepts are regarded as different atomic concepts, which can have different interpretations. This example suggests that interpretations of parameterized concepts should depend on interpretations of parameters and not on their syntactic form.

To come up with an alternative semantics that overcomes these problems, once again, consider the generic axiom $\mathsf{Owner}[X] \equiv \exists \mathsf{owns}.X$. According to this definition, in every model of this axiom, the interpretation of $\mathsf{Owner}[C]$ should be obtained from the interpretation of C by applying some *function* – in this example the function that converts the interpretation of C to the interpretation of $\exists \mathsf{owns}.C$. Our general assumption now is that the same holds for *every* generic concept. E.g., we assume that concept $\mathsf{Heart}[X]$ is an abbreviation for some (unknown) concept (possibly in a very expressive DL) that uses the concept variable X. Semantically, this means that in every interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ there is a *function* $\mathsf{Heart}^{\mathcal{I}}$ that maps each subset of domain elements $M \subseteq \Delta^{\mathcal{I}}$ (potentially the interpretation of some concept C), to a subset $\mathsf{Heart}^{\mathcal{I}}(M) \subseteq \Delta^{\mathcal{I}}$ – the corresponding interpretation of the unknown concept in which X is replaced with C. Parameterized concepts can then be interpreted by applying this function, e.g., $(\mathsf{Heart}[\mathsf{Dog}])^{\mathcal{I}} = \mathsf{Heart}^{\mathcal{I}}(\mathsf{Dog}^{\mathcal{I}})$. This definition ensures that $(\mathsf{Heart}[C])^{\mathcal{I}} = (\mathsf{Heart}[D])^{\mathcal{I}}$ whenever $C^{\mathcal{I}} = D^{\mathcal{I}}$. The interpretation can also be extended to complex (generic) concepts, such as $\exists \mathsf{owns}.(X \sqcap \mathsf{Heart}[\exists \mathsf{owns}.X])$, which are, similarly, interpreted by functions from subsets of $\Delta^{\mathcal{I}}$ to subsets of $\Delta^{\mathcal{I}}$. Finally, an interpretation satisfies an axiom, such as, $\mathsf{Owner}[\mathsf{Owner}[X]] \sqsubseteq \mathsf{Owner}[X]$ if the values of the respective functions are included for every argument. Intuitively, the variable X ranges now not over possible concepts C of some language but over all subsets M of the domain, or, equivalently, all unary predicates $M(x)$. Hence, we refer to this semantics as the *second-order semantics*.

It is easy to see that the second-order semantics is *stronger* than the schema semantics. Indeed, if an axiom is satisfied under the second-order interpretation $\mathcal{I}$, then after replacement of variables X by concepts C, the resulting axioms are satisfied in the (classical) interpretation $\mathcal{J}$ that interprets (parameterized) concepts in the same way: $(\mathsf{Heart}[\mathsf{Dog}])^{\mathcal{J}} = (\mathsf{Heart}[\mathsf{Dog}])^{\mathcal{I}} = \mathsf{Heart}^{\mathcal{I}}(\mathsf{Dog}^{\mathcal{I}})$. The converse is not true: as mentioned, the axiom $\top \sqsubseteq \exists r.X$ is satisfiable for replacement of variables with $\mathcal{EL}$ concepts, but it is not satisfied in any second-order interpretation $\mathcal{I}$ because X can be replaced with the empty set $\emptyset = \bot^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$. Furthermore, whereas schema entailment can always be reduced to first-order entailment (from a possibly infinite set of formulas), there are axioms whose models under the second-order semantics cannot be expressed by first-order formulas. For example, the well-known Modal Logic (ML) *McKinsey axiom* $\Box\Diamond X \to \Diamond\Box X$ – which can be written in the generic extension of the DL $\mathcal{ALC}$ as $\forall r.\exists r.X \sqsubseteq \exists r.\forall r.X$ – cannot be translated into even an infinite set of first-order formulas that holds in exactly the same interpretations (of r) (Goldblatt 1975). Hence, even semi-decidability for this language under the second-order semantics is an open question.

To combine the advantages of the schema and the second-order semantics, in this paper, we are concerned with the question, of whether there are restricted forms of generic axioms for which second-order entailments are identical to schema entailments. We answer this question positively by defining a useful fragment of the generic extension of the DL $\mathcal{EL}$ for which both semantics coincide, and its further restriction, for which entailment under these (equivalent) semantics is decidable in EXPTIME and in PTIME (Section 6). Interestingly, the PTIME fragment can express several well-known DL features, such as role chain axioms, positive self restrictions, reflexive roles, and some forms of (local) role-value-maps from KL-ONE, without requiring additional DL constructors (see Section 7).

The main idea for proving these results, is to show that the standard $\mathcal{EL}$ canonical model (Baader, Brandt, and Lutz 2005) for axioms obtained by replacing concept variables with certain *relevant* $\mathcal{EL}$ concepts, can be extended to a second-order model of the generic axioms. To obtain the decidability and complexity result, we define a further restriction, in which the number of these relevant concepts is at most polynomial. Therefore, the number of the resulting schema instances is bounded by a function polynomial in the size of the ontology and exponential in the maximal number of different concept variables appearing in its axioms.

The results of this paper extend our previous results (Hirschbrunn and Kazakov 2023) in which we allowed concept variables but not parametrized concepts.

2 Related Works

Axiom schemata is certainly not a new concept. In fact, most logic languages, including PL, First-Order Logic (FOL), and MLs were originally defined *axiomatically* (see, e.g., (Blackburn, de Rijke, and Venema 2001)). In the context of DLs, axiom schemata are used to define Nominal Schemas (Krötzsch et al. 2011), in which (nominal) variables can be replaced with any individual appearing in the ontology. Axiom schemata with existentially-quantified concepts can also be used to define operations, such as interpolants or least-common subsumer (Colucci et al. 2010). To obtain decidability results, most works restrict the set of values that can

[2]To see why this could be a concern, imagine that the axiom schemata of Propositional Logic (PL) would obtain new logical consequences (in the language of PL) when allowing variables to be replaced by modal formulas.

be used for the replacement of variables so that the resulting set of (ordinary) axioms is *finite*. However, even for unrestricted axiom schemata, such as those found in PLs and MLs, it can often be shown that it is sufficient to replace variables with only finitely-many formulas found in (or built using) the entailment to be proved. For MLs, this *sub-formula property* usually follows from *cut-free sequent-style* calculi (see, e.g., (Fitting 2007)).

The idea of representing similar axioms in a general way has been used in Ontology Design Patterns (ODPs) (He, Zheng, and Lin 2015; Skjæveland et al. 2018; Kindermann et al. 2018; Krieg-Brückner, Mossakowski, and Neuhaus 2019; Kindermann, Parsia, and Sattler 2019; Skjæveland et al. 2017; Borgida et al. 2012). Somewhat similar to our schema semantics, here one uses concept variables to define axiom templates, which can be used to generate ordinary axioms for specific applications. The difference to our extension is that ODPs do not include proper generic concepts such as $\mathsf{Owner}[X]$ and are not endowed with model-theoretic semantics. Instead, ODPs are typically a kind of pre-processing step, where variables are replaced by concepts from fixed sets of candidates to produce a classical ontology that is then used as usual. This is even weaker than our schema semantics, as for the schema semantics we do not use a limited, application-dependent set of concepts to replace variables, but a large set of concepts independent of the specific use case, for example the set of all $\mathcal{EL}$ concepts.

3 Schema Semantics

We start by formally defining our extension of DLs with parameterized concepts and concept variables:

Definition 1 (Syntax)**.** *The* syntax *of DLs with parameterized concepts and concept variables consists of disjoint and countably infinite sets N_C of* concept names*, each with an assigned arity $\mathsf{ar}(A) \in \mathbb{N}$ ($A \in N_C$), N_R of* role names*, and N_X of* concept variables*. Given a base DL L that is a fragment of $\mathcal{SROIQ}$, such as $\mathcal{EL}$ and $\mathcal{ALC}$, we define by $\mathsf{L}\mathcal{X}$ its corresponding* extension with parameterized (atomic) concepts and concept variables*. Specifically, the set of* $\mathsf{L}\mathcal{X}$-concepts *is the smallest set containing concept variables $X \in N_X$,* concept atoms *$A[C_1, \ldots, C_n]$ where $A \in N_C$, $n = \mathsf{ar}(A)$ and C_i are $\mathsf{L}\mathcal{X}$-concepts ($1 \le i \le n$), and which is closed under the concept constructors of L. An* $\mathsf{L}\mathcal{X}$-ontology *is a (possibly infinite) set $\mathcal{K}$ of* $\mathsf{L}\mathcal{X}$-axioms*, which are constructed from $\mathsf{L}\mathcal{X}$-concepts using the axiom constructors of L.*

Let the expression ex be either a $\mathsf{L}\mathcal{X}$-concept, a $\mathsf{L}\mathcal{X}$-axiom, or a $\mathsf{L}\mathcal{X}$-ontology. We denote by $\mathsf{sub}(ex)$ (all) subconcepts of ex, i.e., substrings of the expression that are valid concepts (not to be confused with concepts that are subsumed by ex). For $\mathsf{L}\mathcal{X}$-concepts and $\mathsf{L}\mathcal{X}$-axioms, we split $\mathsf{sub}(ex)$ into $\mathsf{sub}^+(ex)$ and $\mathsf{sub}^-(ex)$ the set of concepts that occur positively*, respectively* negatively *in ex, i.e., they occur on the right side of the axiom under an even (odd) number of nested negations or on the left side under an odd (even) number of nested negations, $\mathsf{sub}(ex) = \mathsf{sub}^+(ex) \cup \mathsf{sub}^-(ex)$. We denote by $\mathsf{vars}(ex) = \mathsf{sub}(ex) \cap N_X$ the set of* concept variables occurring in *ex. We say that ex is* ground *if $\mathsf{vars}(ex) = \emptyset$. We denote by $\mathsf{args}(ex)$ the set of (complex) concepts* occurring as arguments of atomic concepts in *ex. Additionally, we define $\mathsf{args}^+(ex)$ and $\mathsf{args}^-(ex)$ analogous to $\mathsf{sub}^+(ex)$ and $\mathsf{sub}^-(ex)$.*

A (concept variable) substitution *is a partial mapping $\theta = [X_1/C_1, \ldots, X_n/C_n]$ that assigns concepts C_i to concept variables X_i ($1 \le i \le n$). We denote by $\theta(ex)$ the result of* applying the substitution *to ex, defined in the usual way.*

From now on we assume that DL concepts, axioms, and ontologies may contain concept variables and parameterized (atomic) concepts. If there are no variables in a concept, axiom, or ontology, we call it *ground*.

Example 1. *Consider the following three (non-ground) axioms:*

$$\alpha = A[X, Y] \equiv \exists r.X \sqcap \exists s.Y, \quad (1)$$

$$\beta = A[X, X] \sqsubseteq \exists t.X, \quad (2)$$

$$\gamma = A[X, Y] \sqsubseteq \exists t.(X \sqcup Y). \quad (3)$$

Axioms α, β belong to $\mathcal{ELX}$, whereas axiom γ belongs to $\mathcal{ALCX}$ due to the use of concept disjunction $\sqcup$. Further, $\mathsf{args}(\alpha) = \{X, Y\}$, $\mathsf{sub}^+(\beta) = \{\exists t.X, X\}$, $\mathsf{sub}^-(\gamma) = \{A[X, Y], X, Y\}$, and $\mathsf{vars}(\gamma) = \{X, Y\}$. Finally, for a (non-ground) substitution $\theta = [X/X \sqcup Y]$, we have:

$$\theta(\beta) = A[X \sqcup Y, X \sqcup Y] \sqsubseteq \exists t.(X \sqcup Y). \quad (4)$$

Intuitively the axioms can be thought of as *axiom schemata* representing all axioms obtained by replacing the concept variables X and Y with ordinary (ground) concepts. However, there are two problems: 1) it is unclear how to interpret the parameterized concept formally and 2) the choice of ground concepts to replace variables by is not obvious. One could replace variables by just atomic concepts, or only by concepts appearing in the given ontology, or by concepts that can be constructed in a particular DL. Clearly, each of these choices may result in different logical consequences and algorithmic properties of the resulting schema languages. To handle all such choices and the interpretation of parameterized concepts, we provide a general (parameterized) definition of a schema entailment.

Definition 2 (Schema Semantics)**.** *A* schema interpretation *for $\mathsf{L}\mathcal{X}$ is a pair $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$, where $\Delta^\mathcal{I}$ is a nonempty set called the* domain *of $\mathcal{I}$ and $\cdot$ an* interpretation function *that assigns to every ground $\mathsf{L}\mathcal{X}$-atom $A[C_1, \ldots, C_n]$ a subset of the domain $A[C_1, \ldots, C_n]^\mathcal{I} \subseteq \Delta^\mathcal{I}$ and to every $r \in N_R$ a relation $r^\mathcal{I} \subseteq \Delta^\mathcal{I} \times \Delta^\mathcal{I}$. The interpretation $\mathcal{I}$ is extended to ground $\mathsf{L}\mathcal{X}$-concepts and ground $\mathsf{L}\mathcal{X}$-axioms as usual. For a ground $\mathsf{L}\mathcal{X}$-axiom α and ground $\mathsf{L}\mathcal{X}$-ontology $\mathcal{K}$, we write $\mathcal{I} \models^* \alpha$ when α is satisfied in $\mathcal{I}$, $\mathcal{I} \models^* \mathcal{K}$ when $\mathcal{I}$ satisfies every axiom in $\mathcal{K}$, and $\mathcal{K} \models^* \alpha$ when $\mathcal{I} \models^* \mathcal{K}$ implies $\mathcal{I} \models^* \alpha$ for every schema interpretation $\mathcal{I}$.*

Let H be a (possibly infinite) set of ground $\mathsf{L}\mathcal{X}$-concepts called a concept base*. For an $\mathsf{L}\mathcal{X}$-axiom α, and $\mathsf{L}\mathcal{X}$-ontology $\mathcal{K}$, define by $\alpha_{\downarrow H} = \{\alpha[X_1/C_1, \ldots X_n/C_n] \mid X_i \in \mathsf{vars}(\alpha) \;\&\; C_i \in H\}$ and $\mathcal{K}_{\downarrow H} = \bigcup_{\alpha \in \mathcal{K}} \alpha_{\downarrow H}$ the set of H-*ground instances *of α and $\mathcal{K}$, respectively. For a schema interpretation $\mathcal{I}$, we write $\mathcal{I} \models^*_H \alpha$ and $\mathcal{I} \models^*_H \mathcal{K}$ if $\mathcal{I} \models^* \alpha_{\downarrow H}$ and $\mathcal{I} \models^* \mathcal{K}_{\downarrow H}$. We write $\mathcal{K} \models^*_H \alpha$ if*

$\mathcal{K}_{\downarrow H} \models^* \alpha_{\downarrow H}$ *and we say that* α *is a logical consequence of* $\mathcal{K}$ *under* schema entailment *for a concept base* H. *Finally, we write* $\alpha_{\downarrow \mathsf{L}}$, $\mathcal{K}_{\downarrow \mathsf{L}}$ *and* $\mathcal{K} \models^*_{\mathsf{L}} \alpha$ *instead of* $\alpha_{\downarrow H}$, $\mathcal{K}_{\downarrow H}$ *and* $\mathcal{K} \models^*_H \alpha$, *respectively, if* H *is the set of* all L*-concepts. We also sometimes use* $\mathcal{K} \models^*_{\mathsf{L}\mathcal{X}_\downarrow} \alpha$ *if* $\mathcal{K} \models^*_H \alpha$ *for the set* H *of all* ground $\mathsf{L}\mathcal{X}$*-concepts.*

Note that if the concept base H is *finite*, entailment $\mathcal{K} \models^*_H \alpha$ under the schema semantics for H in $\mathsf{L}\mathcal{X}$ can be reduced to the standard DL entailment in L by replacing each ground atom $A[C_1, \dots, C_n]$ in $\mathcal{K}_{\downarrow H}$ and $\alpha_{\downarrow H}$ with a distinguished atomic concept.

Example 2 (Example 1 Continued)**.** *Notice that* $\{\gamma\} \models^*_H \beta$ *for any concept base* H. *Indeed, take any* $\beta' = \theta(\beta) = A[C, C] \sqsubseteq \exists t.C \in \beta_{\downarrow H}$. *Then* $\gamma_{\downarrow H} \ni \gamma[X/C, Y/C] = A[C, C] \sqsubseteq \exists t.(C \sqcup C) \models \beta'$. *Also, notice that if* H *is closed under concept disjunctions (i.e.,* $C \in H$ *and* $D \in H$ *imply* $C \sqcup D \in H$*) then* $\{\alpha, \beta\} \models^*_H \gamma$. *Indeed, take any* $\gamma' = \theta(\gamma) = A[C, D] \sqsubseteq \exists t.(C \sqcup D) \in \gamma_{\downarrow H}$ *for* $\theta = [X/C, Y/D]$. *Since* H *is closed under disjunction, we have* $\{\alpha, \beta\}_{\downarrow H} \supseteq \{A[C, D] \equiv \exists r.C \sqcap \exists s.D, A[C \sqcup D, C \sqcup D] \equiv \exists r.(C \sqcup D) \sqcap \exists s.(C \sqcup D), A[C \sqcup D, C \sqcup D] \sqsubseteq \exists t.(C \sqcup D)\} \models \{\exists r.C \sqcap \exists s.D \sqsubseteq \exists t.(C \sqcup D), A[C, D] \sqsubseteq \exists t.(C \sqcup D)\} \models \gamma'$ *Consequently,* $\{\alpha, \beta\} \models^*_{\mathcal{ALCX}_\downarrow} \gamma$. *However, it can be shown that* $\{\alpha, \beta\} \not\models^*_{\mathcal{ELX}_\downarrow} \gamma$. *To prove this, consider the* $\mathcal{EL}$ *ontology:*

$$\mathcal{K} = \{\ L \sqsubseteq \exists r.B \sqcap \exists s.C, \quad \exists t.B \sqsubseteq D, \quad \exists t.C \sqsubseteq D\ \} \quad (5)$$

It is easy to see that $\mathcal{K} \cup \{\alpha, \gamma\}_{\downarrow \mathcal{ELX}_\downarrow} \supseteq \mathcal{K} \cup \{\alpha', \gamma'\} \models L \sqsubseteq D$ *for* $\alpha' = \alpha[X/B, Y/C] = A[B, C] \equiv \exists r.B \sqcap \exists r.C$, $\gamma' = \gamma[X/B, Y/C] = A[B, C] \sqsubseteq \exists t(B \sqcup C)$. *Hence,* $\mathcal{K} \cup \{\alpha, \gamma\} \models^*_{\mathcal{ELX}_\downarrow} L \sqsubseteq D$. *We show that* $\mathcal{K} \cup \{\alpha, \beta\} \not\models^*_{\mathcal{ELX}_\downarrow} L \sqsubseteq D$, *which, in particular, implies that* $\{\alpha, \beta\} \not\models^*_{\mathcal{ELX}_\downarrow} \gamma$, *for otherwise* $\mathcal{K} \cup \{\alpha, \beta\} \models^*_{\mathcal{ELX}_\downarrow} \mathcal{K} \cup \{\alpha, \gamma\} \models^*_{\mathcal{EL}} L \sqsubseteq D$.

To prove that $\mathcal{K} \cup \{\alpha, \beta\} \not\models^*_{\mathcal{ELX}_\downarrow} L \sqsubseteq D$, *consider the interpretation* $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$ *defined by:* $\Delta^\mathcal{I} = \{a, b, c\}$, $L^\mathcal{I} = \{a\}$, $B^\mathcal{I} = \{b\}$, $C^\mathcal{I} = \{c\}$, $A[B, C]^\mathcal{I} = A[\top, C]^\mathcal{I} = A[B, \top]^\mathcal{I} = A[\top, \top]^\mathcal{I} = \{a\}$, $r^\mathcal{I} = \{\langle a, b\rangle\}$, $s^\mathcal{I} = \{\langle a, c\rangle\}$, $t^\mathcal{I} = \{\langle a, a\rangle\}$, *and* $E[E_1, \dots, E_n]^\mathcal{I} = h^\mathcal{I} = \emptyset$ *for all remaining concepts* $E[E_1, \dots, E_n]$ *and* $h \in N_R$. *Clearly,* $\mathcal{I} \models \mathcal{K}$ *and* $\mathcal{I} \not\models L \sqsubseteq D$. *It remains to prove that* $\mathcal{I} \models \{\alpha, \beta\}_{\downarrow \mathcal{ELX}_\downarrow}$. *Take any* $\alpha[X/E_1, Y/E_2] = A[E_1, E_2] \equiv \exists r.E_1 \sqcap \exists s.E_2 \in \alpha_{\downarrow \mathcal{ELX}_\downarrow}$. *Then either* $(\exists r.E_1 \sqcap \exists s.E_2)^\mathcal{I} = \emptyset$ *or and* $a \in (\exists r.E_1 \sqcap \exists s.E_2)^\mathcal{I}$ *and* $E_1 \in \{B, \top\}, E_2 \in \{C, \top\}$. *In the first case* $A[E_1, E_2]^\mathcal{I} = \emptyset$ *and in the latter* $A[B, C]^\mathcal{I} = A[\top, C]^\mathcal{I} = A[B, \top]^\mathcal{I} = A[\top, \top]^\mathcal{I} = \{a\}$. *Therefore* $\mathcal{I} \models \alpha_{\downarrow \mathcal{ELX}_\downarrow}$. *Next, take any* $\beta[X/F] = A[F, F] \sqsubseteq \exists t.F \in \beta_{\downarrow \mathcal{ELX}_\downarrow}$ *and any* $d \in A[F, F]^\mathcal{I}$. *Then* $d = a$ *and* $F = \top$ *by definition of* $\mathcal{I}$. *Then* $d = a \in (\exists t.\top)^\mathcal{I} = (\exists t.F)^\mathcal{I}$ *as required.*

Combining the above observations, we obtain: $\mathcal{K} \cup \{\alpha, \beta\} \models^*_{\mathcal{ALCX}_\downarrow} \mathcal{K} \cup \{\alpha, \gamma\} \models^*_{\mathcal{ELX}_\downarrow} \mathcal{K} \cup \{\alpha, \gamma\} \models^*_{\mathcal{ELX}_\downarrow} A \sqsubseteq D$, *however,* $\mathcal{K} \cup \{\alpha, \beta\} \not\models^*_{\mathcal{ELX}_\downarrow} A \sqsubseteq D$.

4 Second-Order Semantics

Example 2 shows that, for the schema semantics, an ontology formulated in one DL may have different conclusions, even in the same DL, when the concept base is extended to a larger language. This goes against the usual understanding of logic, as this means that the consequences of an ontology are not determined by the ontology alone. To mitigate this problem, we consider the second-order semantics that is independent of a concept base.

Definition 3 (Second-Order Semantics)**.** *A (second-order) interpretation for a* $\mathsf{L}\mathcal{X}$ *is a pair* $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$, *where* $\Delta^\mathcal{I}$ *is a nonempty set called the* domain *of* $\mathcal{I}$ *and* $\cdot^\mathcal{I}$ *is an* interpretation function, *that assigns to every* $A \in N_C$ *with arity* $n = \mathsf{ar}(A)$ *a function* $A^\mathcal{I} : (2^{\Delta^\mathcal{I}})^n \to 2^{\Delta^\mathcal{I}}$ *and to every* $r \in N_R$ *a relation* $r^\mathcal{I} \subseteq \Delta^\mathcal{I} \times \Delta^\mathcal{I}$. *A* valuation *for* $\mathcal{I}$ *(also called a variable assignment) is a mapping* η *that assigns to every variable* $X \in N_X$ *a subset* $\eta(X) \subseteq \Delta^\mathcal{I}$.

The interpretation of $\mathsf{L}\mathcal{X}$*-concepts* $C^{\mathcal{I},\eta} \subseteq \Delta^\mathcal{I}$ *is recursively defined by* $X^{\mathcal{I},\eta} = \eta(X)$ *for* $X \in N_X$, $A[C_1, \dots, C_n]^{\mathcal{I},\eta} = A^\mathcal{I}(C_1^{\mathcal{I},\eta}, \dots, C_n^{\mathcal{I},\eta})$, *and is extended to other* $\mathsf{L}\mathcal{X}$*-concepts in the usual way.* Satisfaction of axioms $\mathcal{I} \models^2_\eta \alpha$ under $\mathcal{I}$ and η *is determined from interpretation of* $\mathsf{L}\mathcal{X}$*-concepts in* α *in the standard way. For example,* $\mathcal{I} \models^2_\eta C \sqsubseteq D$ *iff* $C^{\mathcal{I},\eta} \subseteq D^{\mathcal{I},\eta}$. *We write* $\mathcal{I} \models^2 \alpha$ *if* $\mathcal{I} \models^2_\eta \alpha$ *for every* valuation η. *Finally, for an ontology* $\mathcal{K}$, *we write* $\mathcal{I} \models^2 \mathcal{K}$ *if* $\mathcal{I} \models^2 \beta$ *for every* $\beta \in \mathcal{K}$, *and we write* $\mathcal{K} \models^2 \alpha$ *if* $\mathcal{I} \models^2 \mathcal{K}$ *implies* $\mathcal{I} \models^2 \alpha$.

It is easy to show that second-order entailments are closed under substitutions:

Lemma 1. $\mathcal{I} \models^2 \alpha$ *implies* $\mathcal{I} \models^2 \theta(\alpha)$ *for every* $\mathsf{L}\mathcal{X}$*-axiom* α, *and every concept substitution* θ.

Proof. To show that $\mathcal{I} \models^2 \theta(\alpha)$, take any valuation η. We need to prove that $\mathcal{I} \models^2_\eta \theta(\alpha)$. Define another valuation μ by setting $\mu(X) := \theta(X)^{\mathcal{I},\eta}$ for every $X \in N_X$. By induction over $\mathsf{L}\mathcal{X}$-constructors, it is easy to show that $C^{\mathcal{I},\mu} = \theta(C)^{\mathcal{I},\eta}$ for every $\mathsf{L}\mathcal{X}$-concept C and $\mathcal{I} \models^2_\eta \theta(\beta)$ iff $\mathcal{I} \models^2_\mu \beta$ for every $\mathsf{L}\mathcal{X}$-axiom β. Now, since $\mathcal{I} \models^2 \alpha$, we obtain $\mathcal{I} \models^2_\mu \alpha$ and thus $\mathcal{I} \models^2_\eta \theta(\alpha)$. Since η was arbitrary, this proves that $\mathcal{I} \models^2 \theta(\alpha)$. ☐

As discussed in Section 1, the second-order semantics is stronger than the schema semantics:

Lemma 2. *Let* $\mathcal{K}$ *be an* $\mathsf{L}\mathcal{X}$*-ontology,* α *a ground* $\mathsf{L}\mathcal{X}$*-axiom, and* H *a set of ground* $\mathsf{L}\mathcal{X}$*-concepts. Then* $\mathcal{K} \models^*_H \alpha$ *implies* $\mathcal{K} \models^2 \alpha$.

Proof. To show $\mathcal{K} \models^2 \alpha$, take any second-order interpretation $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$ such that $\mathcal{I} \models^2 \mathcal{K}$. We prove that $\mathcal{I} \models^2 \alpha$. For this, define a schema interpretation $\mathcal{J} = (\Delta^\mathcal{J}, \cdot^\mathcal{J})$ with $\Delta^\mathcal{J} = \Delta^\mathcal{I}$, $A[D_1, \dots, D_n]^\mathcal{J} := A[D_1, \dots, D_n]^\mathcal{I}$ for every $A \in N_C$, $n = \mathsf{ar}(A)$ and D_i ground $\mathsf{L}\mathcal{X}$-concepts, and $r^\mathcal{J} = r^\mathcal{I}$ for every $r \in N_R$. Note that this definition implies that $D^\mathcal{J} = D^\mathcal{I}$ for every ground $\mathsf{L}\mathcal{X}$-concept since the extension of interpretation under concept constructors is defined in $\mathcal{I}$ and $\mathcal{J}$ in the same way. Likewise, $\mathcal{I} \models^2 \beta$ iff $\mathcal{J} \models^* \beta$ for every ground $\mathsf{L}\mathcal{X}$-axiom β.

We claim that $\mathcal{J} \models^*_H \mathcal{K}$. Indeed, take any $\beta \in \mathcal{K}_{\downarrow H}$. Then $\beta = \theta(\gamma)$ for some $\gamma \in \mathcal{K}$ and substitution θ. Since $\mathcal{I} \models^2 \mathcal{K}$, we have $\mathcal{I} \models^2 \gamma$. Then by Lemma 1, $\mathcal{I} \models^2 \beta$.

Since β is ground, we have $\mathcal{J} \models^* \beta$. Since $\beta \in \mathcal{K}_{\downarrow H}$ was arbitrary, this proves that $\mathcal{J} \models^*_H \mathcal{K}$. Since $\mathcal{K} \models^*_H \alpha$, we obtain $\mathcal{J} \models^*_H \alpha$. Since α is ground, we have $\mathcal{J} \models^* \alpha$, and so $\mathcal{I} \models^2 \alpha$. Since $\mathcal{I} \models^2 \mathcal{K}$ was arbitrary, this proves that $\mathcal{K} \models^2 \alpha$. □

Note that Lemma 2 cannot be extended to non-ground α. Indeed, take $\mathcal{K} = \{A[C] \sqsubseteq B[C]\}$, $\alpha = A[X] \sqsubseteq B[X]$ and $H = \{C\}$. Clearly, $\mathcal{K} \models^*_H \alpha$, but $\mathcal{K} \not\models^2 \alpha$.

Let us consider some examples for the DL $\mathcal{ELX}$, which have more second-order entailments than entailments using the schema semantics for which all ground $\mathcal{ELX}$ concepts are included in the concept base. These examples will help us to determine the restrictions for the use of concept variables, under which both semantics coincide.

First, we give a minor modification of the example with axiom $\top \sqsubseteq \exists r.X$ discussed in Section 1, which does not result in an inconsistent ontology under the second-order semantics.

Example 3. *Consider the ontology $\mathcal{K} = \{A \sqsubseteq \exists r.X\}$. It is easy to see that $\mathcal{K} \models^2 \exists r.A \sqsubseteq A$. Indeed, assume that $\mathcal{I} \models^2 \mathcal{K}$, and take any valuation η such that $\eta(X) = \emptyset$. Then $A^{\mathcal{I}} = A^{\mathcal{I},\eta} \subseteq (\exists r.X)^{\mathcal{I},\eta} = \emptyset$. Hence $(\exists r.A)^{\mathcal{I}} = \emptyset \subseteq A^{\mathcal{I}}$. At the same time $\mathcal{K} \not\models^*_{\mathcal{ELX}_\downarrow} \exists r.A \sqsubseteq A$. Indeed, consider the interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ with $\Delta^{\mathcal{I}} = \{a, b\}$, $A^{\mathcal{I}} = \{a\}$, $F[C_1, \dots, C_n]^{\mathcal{I}} = \{a, b\}$ for every $F[C_1, \dots, C_n] \neq A$, and $r^{\mathcal{I}} = \{\langle a, a\rangle, \langle b, a\rangle\}$ for every $r \in N_R$. By structural induction, it is easy to show that $a \in C^{\mathcal{I}}$ for every ground $\mathcal{ELX}$ concept C, hence $\mathcal{I} \models A \sqsubseteq \exists r.C$. Therefore, $\mathcal{I} \models^*_{\mathcal{ELX}_\downarrow} \mathcal{K}$. However, since $(\exists r.A)^{\mathcal{I}} = \{a, b\} \not\subseteq \{a\} = A^{\mathcal{I}}$, we obtain $\mathcal{K} \not\models^*_{\mathcal{ELX}_\downarrow} \exists r.A \sqsubseteq A$.*

Example 3 can be generalized to many other $\mathcal{ELX}$ axioms $C \sqsubseteq D$ for which there exists a concept variable X appearing in D but not in C. In this case, $\mathcal{I} \models^2 C \sqsubseteq D$ implies that $C^{\mathcal{I},\eta} = \emptyset$ for every valuation η because for the extension η' of η with $\eta'(X) = \emptyset$, we obtain $C^{\mathcal{I},\eta} = C^{\mathcal{I},\eta'} \subseteq D^{\mathcal{I},\eta'} = \emptyset$.

Now, take any $\mathcal{I} \models^2 C \sqsubseteq D$. Then $C^{\mathcal{I},\eta} = \emptyset$ for every valuation η. Then $\theta(C)^{\mathcal{I}} = \emptyset$ for every concept variable substitution θ. Then for every ground $\mathcal{ELX}$ concept E such that $\theta(C) \in \mathsf{sub}(E)$ we have $E^{\mathcal{I}} = \emptyset$ as well. Thus, $\{C \sqsubseteq D\} \models^2 E \sqsubseteq F$ for every F. If the schema semantics preserves all these entailments then, in particular, all such concepts E (containing instances of C) must be equivalent. This can happen only in some trivial cases, e.g., when the $\mathcal{K}$ contains an axiom of the form $X \sqsubseteq Y$, which implies that all concepts are equivalent. To ensure that the semantics coincide in non-trivial cases, it, therefore, makes sense to require that all variables that are present on the right side of a concept inclusion axioms are also present on the left side. Axioms that fulfill this requirement we called *range restricted* axioms.

Example 4. *Consider the $\mathcal{ELX}$ ontology $\mathcal{K} = \{A \equiv B, C \sqsubseteq F[A, B], F[X, X] \sqsubseteq D\}$. It is easy to see that $\mathcal{K} \models^2 C \sqsubseteq D$ but $\mathcal{K} \not\models^*_{\mathcal{ELX}_\downarrow} C \sqsubseteq D$.*

Example 4 presents another situation when the schema semantics gives fewer consequences than the second-order semantics. The problem in this example is that the variable X occurs twice on the left side of an axiom, which prevents the axiom $F[A, B] \sqsubseteq D$ from being constructed in the schema entailment, even though $F[A, B] \equiv F[A, A] \equiv F[B, B]$ in second-order semantics.

Example 5. *Consider the following two (non-ground) axioms:*

$$\alpha = A[X, X] \sqsubseteq \exists t.X, \tag{6}$$

$$\beta = A[X, Y] \sqsubseteq \exists t.(X \sqcup Y). \tag{7}$$

Example 5 shows that this is also an issue in the absence of atomic concepts with arity non-zero. This is a simplified form of Example 2. For the same argument as in this example, for α (6) and an $\mathcal{EL}$ ontology $\mathcal{K}$ from (5), we have $\mathcal{K} \cup \{\alpha\} \not\models^*_{\mathcal{ELX}_\downarrow} A \sqsubseteq D$, however, since $\{\alpha\} \models^*_{\mathcal{ALCX}_\downarrow} \beta$ (7) and $\mathcal{K} \cup \{\beta\} \models^*_{\mathcal{ELX}_\downarrow} A \sqsubseteq D$, we have $\mathcal{K} \cup \{\alpha\} \models^2 A \sqsubseteq D$.

The problem with α in this example is that the variable X occurs twice on the left side of the axiom, which makes this axiom equivalent to β under the second-order semantics, and, consequently, being able to express an axiom with a concept disjunction $\exists r.C \sqcap \exists s.D \sqsubseteq \exists t.(C \sqcup D)$, which otherwise could not be expressed by ordinary $\mathcal{ELX}$ axioms, i.e., under the schema semantics.

To prevent such cases for our fragment of $\mathcal{ELX}$, it, therefore, makes sense to prohibit the occurrence of the same variable twice on the left side of an axiom. Concepts in which each variable occurs at most once, we call *linear*.

Example 6. *Consider the ground $\mathcal{ELX}$ ontology $\mathcal{K} = \{A \equiv B, C \sqsubseteq F[A], F[B] \sqsubseteq D\}$. It is easy to see that $\mathcal{K} \models^2 C \sqsubseteq D$. Indeed, take any $\mathcal{I} \models^2 \mathcal{K}$ and $d \in C^{\mathcal{I}}$. Then, because $A^{\mathcal{I}} = B^{\mathcal{I}}$, $d \in F[A]^{\mathcal{I}} = F^{\mathcal{I}}(A^{\mathcal{I}}) = F^{\mathcal{I}}(B^{\mathcal{I}}) = F[B]^{\mathcal{I}}$ and then because $\mathcal{I} \models F[B] \sqsubseteq D$, $d \in D^{\mathcal{I}}$.*

On the other hand, $\mathcal{K} \not\models^*_{\mathcal{ELX}_\downarrow} C \sqsubseteq D$, because for schema entailment, we see parameterized concepts as names of new, independent atomic concepts, which means that there is a $\mathcal{I} \models K$ such that $F[A]^{\mathcal{I}} \neq F[B]^{\mathcal{I}}$.

Therefore, for schema entailment and entailment using the second-order semantic to be equivalent, we need to prevent cases such as in Example 6. To do this, we need to either disallow $C \sqsubseteq F[A]$ or $F[B] \sqsubseteq D$. If we return to generics in programming languages, we find that indeed $F[B] \sqsubseteq D$ is usually disallowed. Consider e.g. Java, here we can define `class Text extends ArrayList<String>`, but not `class ArrayList<String> extends Text`. Therefore, we disallow using parameterized concepts on the right side of axioms *if* they contain an argument that is not a simple variable. That is, $F[X] \sqsubseteq D$ is still allowed, corresponding to `class MyArrayList<T> extends MyList<T>` in Java. Concepts containing only parameterized concepts with simple variables as arguments, we call *pure*.

The next example motivates our last restriction.

Example 7. *Consider the $\mathcal{ELX}$ ontology $\mathcal{K} = \{\exists r.X \sqsubseteq \exists s.(X \sqcap A)\}$. It is easy to see that $\mathcal{K} \models^2 \exists r.\top \sqsubseteq \exists r.A$. Indeed, take any $\mathcal{I} \models^2 \mathcal{K}$ and $d \in (\exists r.\top)^{\mathcal{I}}$. Then there exists $d' \in \Delta^{\mathcal{I}}$ such that $\langle d, d'\rangle \in r^{\mathcal{I}}$. Take any valuation*

η with $\eta(X) = \{d'\}$. Since $\langle d, d'\rangle \in r^{\mathcal{I}}$, we have $d \in (\exists r.X)^{\mathcal{I},\eta}$. Since $\mathcal{I} \models^2_\eta \mathcal{K}$, we have $d \in (\exists s.(X \sqcap A))^{\mathcal{I},\eta}$. In particular, $\emptyset \neq (X \sqcap A)^{\mathcal{I},\eta} = \{d'\} \cap A^{\mathcal{I}}$. Hence $d' \in A^{\mathcal{I}}$. Consequently, $d \in (\exists r.A)^{\mathcal{I}}$.

On the other hand, $\mathcal{K} \not\models^*_{\mathcal{ELX}_\downarrow} \exists r.\top \sqsubseteq \exists r.A$, as evidenced by the counter-model $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ with $\Delta^{\mathcal{I}} = \{a, b\}$, $A^{\mathcal{I}} = \{a\}$, $r^{\mathcal{I}} = \{\langle a, b\rangle\}$, $s^{\mathcal{I}} = \{\langle a, a\rangle\}$, and $E[C_1, \dots, C_n]^{\mathcal{I}} = \emptyset$, $h^{\mathcal{I}} = \emptyset$ for any remaining $E[C_1, \dots, C_n]$ and $h \in N_R$. To show that $\mathcal{I} \models^*_{\mathcal{ELX}_\downarrow} \mathcal{K}$, we prove that $\mathcal{I} \models \exists r.C \sqsubseteq \exists s.(C \sqcap A)$ for every ground $\mathcal{ELX}$ concept C. For this, take any $d \in (\exists r.C)^{\mathcal{I}}$. By definition of $r^{\mathcal{I}}$, $d = a$ and $b \in C^{\mathcal{I}}$. Then C can be only a conjunction of concept $\top$. Hence $a \in C$. Hence $d = a \in (\exists s.C)^{\mathcal{I}}$. Thus $\mathcal{I} \models \mathcal{K}$. Since, $a \in (\exists r.\top)^{\mathcal{I}}$ but $(\exists r.A)^{\mathcal{I}} = \emptyset$, we proved that $\mathcal{K} \not\models^*_{\mathcal{ELX}_\downarrow} \exists r.\top \sqsubseteq \exists r.A$.

Note that under the second-order semantics, the axiom $\exists r.X \sqsubseteq \exists s.(X \sqcap A)$ in $\mathcal{K}$ from Example 7 implies two properties: (1) that r is a *subrole* of s ($r \sqsubseteq s$), which is equivalent to axiom $\exists r.X \sqsubseteq \exists s.X$, and (2) that A is a *range* of the role r ($ran(r) \sqsubseteq A$), which is due to the fact that for any element d' such that $\langle d, d'\rangle \in r^{\mathcal{I}}$ this axiom holds for $X = \{d'\}$. As was shown in the example, the schema semantics cannot capture the second kind of properties. In fact, an extension of $\mathcal{EL}$ with both (complex) role inclusions and range restrictions becomes undecidable (Baader, Lutz, and Brandt 2008)), which could explain why the schema semantics cannot characterize consequences in this extension. The same problem also occurs if single variables occur as part of arguments of parameterized concepts. The easiest way to see this is by simply adapting Example 7. Consider the ontology $\mathcal{K} = \{\exists r.X \sqsubseteq B[X \sqcap A], B[X] \sqsubseteq \exists s.X\}$, clearly this is equivalent to the ontology in Example 7 and has the same issues. To prevent situations like the one in Example 7, we require that variables in the right side of axioms appear only directly under existential restrictions. We generalize a related notion of *safe nominals* (Kazakov, Kroetzsch, and Simancik 2012) to define this restriction:

Definition 4 (Safe Concept)**.** *A $\mathcal{ELX}$ concept G is called* safe *(for concept variables) if variables only occur in the form of $\exists r.X$ or directly as arguments of atomic concepts, i.e. safe concepts are defined by the grammar:*

$$G^{(i)} ::= \top \mid F[D_1, \dots, D_n] \mid \exists r.X \mid \exists r.G \mid G^1 \sqcap G^2$$

where D_i is either a safe concept or a variable $X \in N_X$.

5 When Semantics Coincide

In this section, we prove that the restrictions on the use of concept variables and parameterized concepts as discussed in Section 4 are sufficient to guarantee that the (ground) logical consequences under the schema semantics and the second-order semantics coincide. Towards this goal, we define a fragment $\mathcal{ELX}_{F1}$ of $\mathcal{ELX}$ that satisfies these restrictions:

Definition 5 ($\mathcal{ELX}_{F1}$)**.** *A $\mathcal{ELX}$ axiom $\beta = E \sqsubseteq G$ is in the fragment $\mathcal{ELX}_{F1}$, if*

- *β is range restricted, i.e. $\mathsf{vars}(G) \subseteq \mathsf{vars}(E)$,*
- *E is linear, i.e. E does not contain any variable twice,*
- *E is pure, i.e. $\mathsf{args}(E) \subseteq N_X$, and*
- *G is safe (cf. Definition 4).*

Our next goal is to prove that $\mathcal{K} \models^2 \alpha$ implies $\mathcal{K} \models^*_H \alpha$ for every $\mathcal{ELX}_{F1}$ ontology $\mathcal{K}$ and certain $\mathcal{ELX}$-axioms α, provided that H contains all "relevant" ground $\mathcal{ELX}$-concepts determined by $\mathcal{K}$ and α. We prove this implication by extending the well-known $\mathcal{EL}$ canonical model construction (Baader, Brandt, and Lutz 2005; Kazakov, Krötzsch, and Simancik 2014) to $\mathcal{ELX}$. Usually, canonical models are defined using consequences of the ontology derived by certain inference rules, however, in our case, we define them using consequences under the schema semantics.

Definition 6 (Canonical Interpretation)**.** *Let $\mathcal{K}$ be an $\mathcal{ELX}$ ontology and H a nonempty concept base. The* canonical interpretation *(w.r.t. $\mathcal{K}$ and H) is a second-order interpretation $\mathcal{I} = \mathcal{I}(\mathcal{K}, H) = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ where $\Delta^{\mathcal{I}} = \{x_C \mid C \in H\}$, $F^{\mathcal{I}}(M_1, \dots, M_n) = \{x_C \in \Delta^{\mathcal{I}} \mid \exists x_{D_i} \in M_i : \mathcal{K} \models^*_H C \sqsubseteq F[D_1, \dots, D_n]\}$ for $F \in N_C$, and $r^{\mathcal{I}} = \{\langle x_C, x_D\rangle \in \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \mid \mathcal{K} \models^*_H C \sqsubseteq \exists r.D\}$ for $r \in N_R$.*

Our next goal is to show that the defined interpretation is indeed a model of $\mathcal{K}$: $\mathcal{I} \models^2 \mathcal{K}$. To achieve this, we characterize the interpretation of $\mathcal{ELX}$-concepts in terms of schema entailment from $\mathcal{K}$, similar to how interpretations of atomic concepts were defined in $\mathcal{I}$. For this, we need to be able to turn valuations into substitutions by ground concepts.

Definition 7. *Let $\mathcal{I}$ be a canonical interpretation w.r.t. an $\mathcal{ELX}$ ontology $\mathcal{K}$ and a concept base H, and η a valuation for $\mathcal{I}$. An η-substitution is any substitution $\theta = [X_1/D_1, \dots, X_n/D_n]$ such that $x_{D_i} \in \eta(X_i)$ $(1 \leq i \leq n)$.*

Lemma 3. *Let $\mathcal{I}$ be the canonical interpretation w.r.t. an $\mathcal{ELX}$ ontology $\mathcal{K}$ and a concept base H, η a valuation, E a linear pure $\mathcal{ELX}$-concept, and $x_C \in E^{\mathcal{I},\eta}$. Then $\mathcal{K} \models^*_H C \sqsubseteq \theta(E)$ for some η-substitution θ.*

Proof. The proof is by induction over the definition of linear pure $\mathcal{ELX}$ concept E:

- $E = \top$. Then $\mathcal{K} \models^*_H C \sqsubseteq \theta(E)$ holds for every C and θ.
- $E = F[X_1, \dots, X_n]$, $F \in N_C$. Then by definition of $F^{\mathcal{I}}$, $x_C \in E^{\mathcal{I},\eta} = F^{\mathcal{I}}(\eta(X_1), \dots, \eta(X_n))$ iff $\mathcal{K} \models^*_H C \sqsubseteq F[D_1, \dots, D_n]$ for some $D_i \in H$ with $x_{D_i} \in \eta(X_i)$ $(1 \leq i \leq n)$. Then, because every variable occurs at most once in $F[X_1, \dots, X_n]$ (linear), $\theta = [X_1/D_1, \dots, X_n/D_n]$ is an η-substitution and $\mathcal{K} \models^*_H C \sqsubseteq \theta(E)$.
- $E = X \in N_X$. Then $x_C \in E^{\mathcal{I},\eta} = \eta(X)$. Then $\mathcal{K} \models^*_H C \sqsubseteq \theta(X)$ holds for the η-substitution $\theta = [X/C]$.
- $E = \exists r.L$. Then $x_C \in E^{\mathcal{I},\eta}$ implies $\langle x_C, x_D\rangle \in r^{\mathcal{I}}$ for some $x_D \in L^{\mathcal{I},\eta}$. Then $\mathcal{K} \models^*_H C \sqsubseteq \exists r.D$ by definition of $r^{\mathcal{I}}$. If $L = X \in N_X$ then $L^{\mathcal{I},\eta} = \eta(X)$, in which case $\mathcal{K} \models^*_H C \sqsubseteq \exists r.\theta(X)$ for the η-substitution $\theta = [X/D]$. Otherwise, L is linear and pure, and since $x_D \in L^{\mathcal{I},\eta}$, by induction hypothesis, $\mathcal{K} \models^*_H D \sqsubseteq \theta(L)$ for some η-substitution θ. Combined with $\mathcal{K} \models^*_H C \sqsubseteq \exists r.D$, we obtain $\mathcal{K} \models^*_H C \sqsubseteq \exists r.\theta(L) = C \sqsubseteq \theta(E)$.

- $E = E_1 \sqcap E_2$. Then $x_C \in E^{\mathcal{I},\eta} = E_1^{\mathcal{I},\eta} \cap E_2^{\mathcal{I},\eta}$. Since E is linear and pure then so are E_1 and E_2. So, by induction hypothesis, $\mathcal{K} \models^*_H C \sqsubseteq \theta_1(E_1)$ and $\mathcal{K} \models^*_H C \sqsubseteq \theta_2(E_2)$ for some η-substitutions θ_1 and θ_2. Furthermore, since E is linear, E_1 and E_2 do not share variables. Hence, there exists an η-substitution θ such that $\theta(E_1) = \theta_1(E_1)$ and $\theta(E_2) = \theta_2(E_2)$. Consequently, $\mathcal{K} \models^*_H C \sqsubseteq \theta_2(E_2) \sqcap \theta_2(E_2) = C \sqsubseteq \theta(E)$. □

Lemma 3 characterizes interpretations of concepts on the left-hand side of $\mathcal{ELX}_{F1}$-axioms. We now provide the converse characterization of concepts appearing on the right-hand side. This time, however, we need to make additional assumptions about the content of the concept base H.

Lemma 4. *Let $\mathcal{I}$ be the canonical interpretation w.r.t. an $\mathcal{ELX}$ ontology $\mathcal{K}$ and a concept base H, η a valuation, G a safe $\mathcal{ELX}$-concept, and θ an η-substitution such that $\mathcal{K} \models^*_H C \sqsubseteq \theta(G)$ for some $C \in H$, and $\theta(L) \in H$ for every $L \in$ args(G) and $\exists r.L \in$ sub(G). Then $x_C \in G^{\mathcal{I},\eta}$.*

Proof. The proof is by induction over the definition of safe $\mathcal{ELX}$ concept G (see Definition 4):

- $G = \top$. Then $x_C \in G^{\mathcal{I},\eta} = \Delta^{\mathcal{I}}$.
- $G = F[L_1, \ldots L_n]$, $F \in N_C$. Take any i with $1 \leq i \leq n$. Since $L_i \in$ args(G), we have $D_i := \theta(L_i) \in H$. If $L_i = X \in N_X$ then $x_{D_i} \in \eta(X) = L_i^{\mathcal{I},\eta}$ since θ is an η-substitution. Otherwise, L_i is safe since G is safe. Since, trivially, $\mathcal{K} \models^*_H D_i \sqsubseteq \theta(L_i)$, we obtain $x_{D_i} \in L_i^{\mathcal{I},\eta}$ by induction hypothesis. Finally, since $\mathcal{K} \models^*_H C \sqsubseteq F(D_1, \ldots, D_n)$ and $x_{D_i} \in L_i^{\mathcal{I},\eta}$ $(1 \leq i \leq n)$, by definition of $F^{\mathcal{I}}$ we obtain $x_C \in F^{\mathcal{I}}(L_1^{\mathcal{I},\eta}, \ldots, L_n^{\mathcal{I},\eta}) = G^{\mathcal{I},\eta}$.
- $G = \exists r.X$. Since θ is an η-substitution, $\theta(X) = D$ for some $x_D \in \eta(X)$, and since $\mathcal{K} \models^*_H C \sqsubseteq \theta(G) = C \sqsubseteq \exists r.D$, by definition of $r^{\mathcal{I}}$, we obtain $\langle x_C, x_D \rangle \in r^{\mathcal{I}}$. Consequently, $x_C \in (\exists r.X)^{\mathcal{I},\eta} = G^{\mathcal{I},\eta}$.
- $G = \exists r.L$ where L is safe. Then $D := \theta(L) \in H$. Since, trivially, $\mathcal{K} \models^*_H D \sqsubseteq \theta(L)$, we obtain $x_D \in L^{\mathcal{I},\eta}$ by induction hypothesis. Since $\mathcal{K} \models^*_H C \sqsubseteq \theta(G) = C \sqsubseteq \exists r.D$, we obtain $\langle x_C, x_D \rangle \in r^{\mathcal{I}}$ by definition of $r^{\mathcal{I}}$. Hence $x_C \in (\exists r.L)^{\mathcal{I},\eta}$.
- $G = G_1 \sqcap G_2$. From $\mathcal{K} \models^*_H C \sqsubseteq \theta(G)$, we obtain $\mathcal{K} \models^*_H C \sqsubseteq \theta(G_1)$ and $\mathcal{K} \models^*_H C \sqsubseteq \theta(G_2)$. Since G_1 and G_2 are safe, by induction hypothesis, $x_C \in G_1^{\mathcal{I},\eta}$ and $x_C \in G_2^{\mathcal{I},\eta}$. Hence $x_C \in (G_1 \sqcap G_2)^{\mathcal{I},\eta} = G^{\mathcal{I},\eta}$. □

The restriction to safe concepts in Lemma 4 is necessary, as the following example shows:

Example 8. *Consider $H = \{A, B\} \subseteq N_C$ and $\mathcal{K} = \{A \sqsubseteq B\}$. Then the canonical interpretation $\mathcal{I}$ for $\mathcal{K}$ and H has domain $\Delta^{\mathcal{I}} = \{x_A, x_B\}$ and assigns $A^{\mathcal{I}} = \{x_A, x_B\}$, $B^{\mathcal{I}} = \{x_B\}$. Now take $E = X \in N_X$ and define $\eta(X) = \{x_B\}$. Then $\theta(X) = B$ is an η-substitution and $\mathcal{K} \models^*_H A \sqsubseteq \theta(E)$. However, $x_A \notin E^{\mathcal{I},\eta} = \eta(X) = \{x_B\}$.*

We are now ready to harvest the fruits of our characterization of canonical interpretations. We first show that the canonical interpretation satisfies every axiom from $\mathcal{K}$, provided that H contains relevant concepts from such axioms.

Corollary 1. *Let $\mathcal{I}$ be the canonical interpretation w.r.t. $\mathcal{K}$ and H, and $\beta = E \sqsubseteq G \in \mathcal{K}$ an $\mathcal{ELX}_{F1}$ axiom such $L_{\downarrow H} \subseteq H$ for every $\exists r.L \in$ sub(G) and every $L \in$ args(G). Then $\mathcal{I} \models^2 \beta$.*

Proof. Take any valuation η and any $x_C \in E^{\mathcal{I},\eta}$. We need to prove that $x_C \in G^{\mathcal{I},\eta}$. Since E is linear and pure, by Lemma 3, $\mathcal{K} \models^*_H C \sqsubseteq \theta(E)$ for some η-substitution θ. Since $\beta \in \mathcal{K}$ is range restricted, $\theta(\beta) = \theta(E) \sqsubseteq \theta(G) \in \mathcal{K}_{\downarrow H}$, therefore $\mathcal{K} \models^*_H C \sqsubseteq \theta(G)$. Since G is safe and $\theta(L) \in L_{\downarrow H} \subseteq H$ for every $\exists r.L \in$ sub(G) and every $L \in$ args(G), by Lemma 4, $x_C \in G^{\mathcal{I},\eta}$, as required. □

Corollary 1 means that we can construct a canonical model $\mathcal{I}$ of any $\mathcal{K}$ satisfying our syntactic restrictions. Now if $\mathcal{K} \models^2 \alpha$, this implies that $\mathcal{I} \models^2 \alpha$. Next, we show that, in this case, $\mathcal{K} \models^*_H \alpha$, that is, the second-order entailment $\mathcal{K} \models^2 \alpha$ is characterized by the schema entailment for a suitable H. As in the case of axioms in $\mathcal{K}$, we also need to apply syntactic restrictions to α, however, this time the conditions for concepts on the left and on the right must be swapped.

Corollary 2. *Let $\mathcal{I}$ be the canonical interpretation w.r.t. $\mathcal{K}$ and H, and $\alpha = G \sqsubseteq E$ an $\mathcal{ELX}$ axiom such that G is safe, $G_{\downarrow H} \subseteq H$ and $L_{\downarrow H} \subseteq H$ for every $\exists r.L \in$ sub(G) and $L \in$ args(G), and E is linear and pure. Then $\mathcal{I} \models^2 \alpha$ implies $\mathcal{K} \models^*_H \alpha$.*

Proof. Take any substitution θ such that $\theta(X) \in H$ for $X \in$ vars(α). We need to show that $\mathcal{K} \models^*_H \theta(\alpha)$. Since $G_{\downarrow H} \subseteq H$, we have $C := \theta(G) \in H$. Trivially, $\mathcal{K} \models^*_H C \sqsubseteq \theta(G)$. Now take any valuation η such that $\eta(X) = \{x_D\}$ whenever $\theta(X) = D \in H$. Note that θ is an η-substitution. Since G is safe and $L_{\downarrow H} \subseteq H$ for every $\exists r.L \in$ sub(G) and $L \in$ args(G), by Lemma 4, $x_C \in G^{\mathcal{I},\eta}$. Since $\mathcal{I} \models^2 \alpha$, we have $x_C \in G^{\mathcal{I},\eta} \subseteq E^{\mathcal{I},\eta}$. Since E is linear and pure, by Lemma 3, $\mathcal{K} \models^*_H C \sqsubseteq \sigma(E)$ for some η-substitution σ. Since $\eta(X) = \{x_D\}$ for every $X \in$ vars(α) with $D = \theta(X)$, and both θ and σ are η-substitutions, we obtain $\sigma(\alpha) = \theta(\alpha)$. Consequently, $\mathcal{K} \models^*_H C \sqsubseteq \sigma(E) = \theta(G) \sqsubseteq \theta(E) = \theta(\alpha)$. □

By combining Corollaries 1 and 2 for the given $\mathcal{ELX}_{F1}$ ontology $\mathcal{K}$ and an $\mathcal{EL}$ axiom α, we can define the *smallest* concept base H such that the canonical interpretation $\mathcal{I}$ w.r.t. $\mathcal{K}$ and H satisfies all axioms $\beta \in \mathcal{K}$ under the second-order semantics and entails α only if $\mathcal{K} \models^*_H \alpha$. Note that conditions $L_{\downarrow H} \subseteq H$ and $G_{\downarrow H} \subseteq H$ in Corollaries 1 and 2 are recursive over H. Therefore, the required concept base H is defined as a fixed point limit for this condition.

Definition 8. *Let $\mathcal{K}$ be a $\mathcal{ELX}$ ontology and $\alpha = G \sqsubseteq E$ an $\mathcal{ELX}$ axiom. Let $H^0 = \{\top\}$, and $H^{i+1} = H^i \cup G_{\downarrow H^i} \cup \bigcup\{L_{\downarrow H^i} \mid \exists r.L \in$ sub$^+(\mathcal{K}) \cup$ sub$(G)\} \cup \bigcup\{L_{\downarrow H^i} \mid L \in$ args$^+(\mathcal{K}) \cup$ args$(G)\}$ for $i \geq 0$. We call $H^\infty = \bigcup_{i \geq 0} H^i$ the* expansion base *for $\mathcal{K}$ w.r.t α.*

We show that the expansion base H^∞ is indeed a fixed point of the required conditions:

Lemma 5. *Let H^∞ be the expansion base for $\mathcal{K}$ w.r.t. $\alpha = G \sqsubseteq E$. Then $G_{\downarrow H^\infty} \subseteq H^\infty$ and $L_{\downarrow H^\infty} \subseteq H^\infty$ for every*

$\exists r.L \in \textsf{sub}^+(\mathcal{K}) \cup \textsf{sub}(G)$ *and every* $L \in \textsf{args}^+(\mathcal{K}) \cup \textsf{args}(G)$.

Proof. Take any $L \in \{G\} \cup \textsf{args}^+(\mathcal{K}) \cup \textsf{args}(G) \cup \{L \mid \exists r.L \in \textsf{sub}^+(\mathcal{K}) \cup \textsf{sub}(G)\}$ and any $D \in L_{\downarrow H^\infty}$. Then $D = L[X_1/C_1, \ldots, X_n/C_n]$ for $\{X_1, \ldots, X_n\} = \textsf{vars}(L)$ and $\{C_1, \ldots, C_n\} \subseteq H^\infty = \bigcup_{i \geq 0} H^i$. Since $H^i \subseteq H^{i+1}$ for every $i \geq 0$, then $\{C_1, \ldots, C_n\} \subseteq H^i$ for some $i \geq 0$. Then $D \in L_{\downarrow H^i} \subseteq H^{i+1} \subseteq H^\infty$. □

By combining Corollaries 1 and 2 and Lemma 5, we now prove the following result:

Theorem 1. *Let $\mathcal{K}$ be an $\mathcal{ELX}_{F1}$ ontology and $\alpha = G \sqsubseteq E$ an $\mathcal{ELX}$ axiom such that G is safe and E is linear and pure. Let H^∞ be the expansion base for $\mathcal{K}$ and α. Then $\mathcal{K} \models^2 \alpha$ implies $\mathcal{K} \models^*_{H^\infty} \alpha$.*

Proof. By Definition 8, H^∞ is nonempty. Let $\mathcal{I}$ be the canonical interpretation w.r.t. $\mathcal{K}$ and H^∞. By Lemma 5, the conditions of Corollary 1 are satisfied for $H = H^\infty$, and every $\beta \in \mathcal{K}$. This implies $\mathcal{I} \models^2 \mathcal{K}$. Then $\mathcal{K} \models^2 \alpha$ implies $\mathcal{I} \models^2 \alpha$. Likewise, by Lemma 5, the conditions of Corollary 2 are satisfied for $H = H^\infty$ and α. Hence, $\mathcal{I} \models^2 \alpha$ implies $\mathcal{K} \models^*_{H^\infty} \alpha$, as required. □

By combining Theorem 1 with Lemma 2, we obtain:

Corollary 3. *Let $\mathcal{K}$ be a $\mathcal{ELX}_{F1}$ ontology and $\alpha = G \sqsubseteq E$ a ground $\mathcal{ELX}$ axiom with $\textsf{args}^+(\alpha) = \emptyset$. Then $\mathcal{K} \models^2 \alpha$ iff $\mathcal{K} \models^*_{\mathcal{ELX}_\downarrow} \alpha$ iff $\mathcal{K} \models^*_{H^\infty} \alpha$.*

Proof. Since α is ground, $\mathcal{K} \models^*_{H^\infty} \alpha$ implies $\mathcal{K} \models^*_{\mathcal{ELX}_\downarrow} \alpha$, which, by Lemma 2, implies $\mathcal{K} \models^2 \alpha$. Suppose that $\alpha = G \sqsubseteq E$. Since α is ground, then G is safe and E is linear, and since $\textsf{args}^+(\alpha) = \emptyset$, E is pure. Hence, by Theorem 1, $\mathcal{K} \models^2 \alpha$ implies $\mathcal{K} \models^*_{H^\infty} \alpha$. □

6 Decidability

Because the schema semantics and the second-order semantics coincide for $\mathcal{ELX}_{F1}$, we immediately obtain semi-decidability of the entailment (of ground axioms) for the latter. In general, entailment in $\mathcal{ELX}_{F1}$ is still *undecidable* because $\mathcal{ELX}_{F1}$ can express (unrestricted) role-value-maps:

Definition 9 (Role-Value-Maps)**.** *A* role-value-map *(Baader 2003) is an axiom of the form $r_1 \circ \cdots \circ r_m \sqsubseteq s_1 \circ \cdots \circ s_n$ with $m, n \geq 1$, $r_i, s_j \in N_R$ ($1 \leq i \leq m$, $1 \leq j \leq n$). The* interpretation of role-value-maps *is defined by: $\mathcal{I} \models r_1 \circ \cdots \circ r_m \sqsubseteq s_1 \circ \cdots \circ s_n$ iff $r_1^\mathcal{I} \circ \cdots \circ r_m^\mathcal{I} \subseteq s_1^\mathcal{I} \circ \cdots \circ s_n^\mathcal{I}$, where $\circ$ is the usual composition of binary relations.*

Lemma 6. *For every interpretation $\mathcal{I}$ it holds $\mathcal{I} \models^2 \exists r_1.\exists r_2.\ldots\exists r_m.X \sqsubseteq \exists s_1.\exists s_2.\ldots\exists s_n.X$ iff $r_1^\mathcal{I} \circ \cdots \circ r_m^\mathcal{I} \subseteq s_1^\mathcal{I} \circ \cdots \circ s_n^\mathcal{I}$.*

Proof. ($\Rightarrow$): Assume that $\mathcal{I} \models^2 \exists r_1.\exists r_2.\ldots\exists r_m.X \sqsubseteq \exists s_1.\exists s_2.\ldots\exists s_n.X$. Then we need to show that $\forall x, z \in \Delta^\mathcal{I} : (\exists y_1, \ldots, y_{m-1} \in \Delta^\mathcal{I} : \langle x, y_1\rangle \in r_1^\mathcal{I}, \langle y_1, y_2\rangle \in r_2^\mathcal{I}, \ldots, \langle y_{m-1}, z\rangle \in r_m^\mathcal{I}) \Rightarrow (\exists y_1, \ldots, y_{n-1} \in \Delta^\mathcal{I} : \langle x, y_1\rangle \in s_1^\mathcal{I}, \langle y_1, y_2\rangle \in s_2^\mathcal{I}, \ldots, \langle y_{n-1}, z\rangle \in s_n^\mathcal{I})$. Take any $x, z \in \Delta^\mathcal{I}$ such that $\exists y_1, \ldots, y_{m-1} \in \Delta^\mathcal{I} : \langle x, y_1\rangle \in r_1^\mathcal{I}, \langle y_1, y_2\rangle \in r_2^\mathcal{I}, \ldots, \langle y_{m-1}, z\rangle \in r_m^\mathcal{I}$ Let η be a valuation such that $\eta(X) = \{z\}$. Then $x \in (\exists r_1.\exists r_2.\ldots\exists r_m.X)^{\mathcal{I},\eta}$ and it follows that $x \in (\exists s_1.\exists s_2.\ldots\exists s_n.X)^{\mathcal{I},\eta}$. Then $\exists y_1, \ldots, y_{n-1} \in \Delta^\mathcal{I} : \langle x, y_1\rangle \in s_1^\mathcal{I}, \langle y_1, y_2\rangle \in s_2^\mathcal{I}, \ldots, \langle y_{n-1}, z\rangle \in s_n^\mathcal{I}$.

($\Leftarrow$): Assume that $\forall x, z \in \Delta^\mathcal{I} : (\exists y_1, \ldots, y_{m-1} \in \Delta^\mathcal{I} : \langle x, y_1\rangle \in r_1^\mathcal{I}, \langle y_1, y_2\rangle \in r_2^\mathcal{I}, \ldots, \langle y_{m-1}, z\rangle \in r_m^\mathcal{I}) \Rightarrow (\exists y_1, \ldots, y_{n-1} \in \Delta^\mathcal{I} : \langle x, y_1\rangle \in s_1^\mathcal{I}, \langle y_1, y_2\rangle \in s_2^\mathcal{I}, \ldots, \langle y_{n-1}, z\rangle \in s_n^\mathcal{I})$. We need to show that $\mathcal{I} \models^2 \exists r_1.\exists r_2.\ldots\exists r_m.X \sqsubseteq \exists s_1.\exists s_2.\ldots\exists s_n.X$. Take any x and η then if $x \in (\exists r_1.\exists r_2.\ldots\exists r_m.X)^{\mathcal{I},\eta}$ then $\exists y_1, \ldots, y_{m-1}, z \in \Delta^\mathcal{I} : \langle x, y_1\rangle \in r_1^\mathcal{I}, \langle y_1, y_2\rangle \in r_2^\mathcal{I}, \ldots, \langle y_{m-1}, z\rangle \in r_m^\mathcal{I}, z \in \eta(X)$. Then from the role-value-map axiom it follows that $\exists y_1, \ldots, y_{n-1} \in \Delta^\mathcal{I} : \langle x, y_1\rangle \in s_1^\mathcal{I}, \langle y_1, y_2\rangle \in s_2^\mathcal{I}, \ldots, \langle y_{n-1}, z\rangle \in s_n^\mathcal{I}$. Then $x \in (\exists s_1.\exists s_2.\ldots\exists s_n.X)^{\mathcal{I},\eta}$. As x and η were chosen arbitrarily, this proves the proposition. □

Theorem 2. *Axiom entailment in $\mathcal{ELX}_{F1}$ is undecidable.*

Proof. Theorem 2 follows directly from Lemma 6 and the fact that axiom entailment is undecidable in $\mathcal{EL}$ extended with role-value-maps (Baader 2003). □

The reason for this undecidability is the deep nesting of concept variables on the right side of axioms under existential restrictions. Such nested variables result in infinite expansion base H^∞ as the following example shows:

Example 9. *Consider the $\mathcal{ELX}_{F1}$ ontology $\mathcal{K} = \{X \sqsubseteq \exists r.\exists r.X\}$ and $\alpha = A \sqsubseteq B$. Then according to Definition 8, we have $H^1 = \{A\}$, $\mathcal{K}^1 = \{A \sqsubseteq \exists r.\exists r.A\}$, $H^2 = \{A, \exists r.A\}$, $\mathcal{K}^2 = \{A \sqsubseteq \exists r.\exists r.A, \exists r.A \sqsubseteq \exists r.\exists r.\exists r.A\}$, $H^3 = \{A, \exists r.A, \exists r.\exists r.A\}$, etc.*

The same problem also occurs for nested parameterized concepts such as $A[A[X]]$. If we restrict $\mathcal{ELX}_{F1}$ so that variables on the right side do not appear under nested existential restrictions or parameterized atomic concepts, we can show that the expansion $\mathcal{K}^\infty$ of the ontology is, in fact, exponential in the size of $\mathcal{K}$, which gives us EXPTIME decidability of the (schema and second-order) entailment.

Definition 10 ($\mathcal{ELX}_{F2}$)**.** *An $\mathcal{ELX}_{F2}$ axiom is an $\mathcal{ELX}_{F1}$ axiom α such that for every $\exists r.L \in \textsf{sub}^+(\alpha)$ and every $L \in \textsf{args}^+(\alpha)$, either $L = X \in N_X$ or $\textsf{vars}(L) = \emptyset$.*

Lemma 7. *Let $\mathcal{K}$ be an $\mathcal{ELX}_{F2}$ ontology, $\alpha = G \sqsubseteq E$ a ground $\mathcal{ELX}$ axiom and H^∞ the expansion base for $\mathcal{K}$ and α. Then $H^\infty = \{\top, G\} \cup \{L \mid (\exists r.L \in \textsf{sub}^+(\mathcal{K}) \cup \textsf{sub}(G)$ or $L \in \textsf{args}^+(\mathcal{K}) \cup \textsf{args}(G))$ & $\textsf{vars}(L) = \emptyset\}$.*

Proof. By Definition 8, $H^{i+1} = H^i \cup \Delta^i$ where $\Delta^i = G_{\downarrow H^i} \cup \bigcup\{L_{\downarrow H^i} \mid \exists r.L \in \textsf{sub}^+(\mathcal{K}) \cup \textsf{sub}(G)\} \cup \bigcup\{L_{\downarrow H^i} \mid L \in \textsf{args}^+(\mathcal{K}) \cup \textsf{args}(G)\}$ for $i \geq 0$. By Definition 10 for $\exists r.L \in \textsf{sub}^+(\mathcal{K})$ and $L \in \textsf{args}^+(\mathcal{K})$ we have $L = X \in N_X$ or $\textsf{vars}(L) = \emptyset$. If $L = X$ then $L_{\downarrow H^i} = X_{\downarrow H^i} = H^i$. If $\textsf{vars}(L) = \emptyset$ then $L_{\downarrow H^i} = \{L\}$. Since α is ground, $G_{\downarrow H^i} = \{G\}$ and $L_{\downarrow H^i} = \{L\}$ for every $\exists r.L \in \textsf{sub}(G)$ and $L \in \textsf{args}(G)$. Then $\Delta^i = \Delta := \{G\} \cup \{L \mid (\exists r.L \in \textsf{sub}^+(\mathcal{K}) \cup \textsf{sub}(G)$ or $L \in \textsf{args}^+(\mathcal{K}) \cup \textsf{args}(G))$ & $\textsf{vars}(L) = \emptyset\}$. Hence $H^\infty =$

$\bigcup_{i\geq 0} H^i = H^0 \cup \Delta = \{\top, G\} \cup \{L \mid (\exists r.L \in \mathsf{sub}^+(\mathcal{K}) \cup \mathsf{sub}(G)$ or $L \in \mathsf{args}^+(\mathcal{K}) \cup \mathsf{args}(G))\,\&\,\mathsf{vars}(L) = \emptyset\}$. □

Since H^∞ is a subset of (ground) concepts appearing in $\mathcal{K}$ and α, we obtain:

Theorem 3. *Let $\mathcal{K}$ be a $\mathcal{ELX}_{F2}$ ontology, α a ground $\mathcal{ELX}$ axiom with $\mathsf{args}^+(\alpha) = \emptyset$, and V the maximal number of different concept variables in axioms of $\mathcal{K}$. Then the entailment $\mathcal{K} \models^2 \alpha$ is decidable in polynomial time in the size of $\mathcal{K}$ and α and exponential time in V.*

Proof. By Corollary 3, $\mathcal{K} \models^2 \alpha$ iff $\mathcal{K} \models^*_{H^\infty} \alpha$. Since α is ground, $\mathcal{K} \models^*_{H^\infty} \alpha$ iff $\mathcal{K}_{\downarrow H^\infty} \models^* \alpha$. The latter can be reduced to classical $\mathcal{EL}$ entailment checking, which is polynomially decidable (Baader, Brandt, and Lutz 2005).

By Lemma 7, $H^\infty \subseteq \{\top\} \cup \mathsf{sub}(\mathcal{K}) \cup \mathsf{sub}(\alpha)$, so its size is linear in the size of $\mathcal{K}$ and α: $\|H^\infty\| \leq \|\mathcal{K}\| + \|\alpha\| + 1$. The size of $\mathcal{K}_{\downarrow H^\infty}$ can be bounded by $\|\mathcal{K}_{\downarrow H^\infty}\| \leq \|\mathcal{K}\| \cdot \|H^\infty\|^V$. Thus, the size of $\mathcal{K}_{\downarrow H^\infty}$ is bounded by a function, which is polynomial in the size of $\mathcal{K}$ and α and exponential in V. □

7 Discussion of Syntactical Restrictions

As the fragments $\mathcal{ELX}_{F1}$ and $\mathcal{ELX}_{F2}$ are obtained by imposing a large number of syntactic restrictions, it is natural to ask, what the resulting fragment is actually capable of. In this section, we look at the expressive power of these fragments.

We start by showing, that $\mathcal{ELX}_{F2}$ expresses many features from polynomial extensions of $\mathcal{EL}$ that require special constructors. First, we can express role chain axioms from $\mathcal{EL}^{++}$ (Baader, Brandt, and Lutz 2005), which are restricted forms of role-value-maps (cf. Definition 9) with one role on the right-hand-side. For example $\mathsf{father} \circ \mathsf{father} \sqsubseteq \mathsf{grandfather}$ is equivalent to $\exists\mathsf{father}.\exists\mathsf{father}.X \sqsubseteq \exists\mathsf{grandfather}.X$. Second, we can express self restrictions (Horrocks, Kutz, and Sattler 2006) on the right side of axioms. For example, $\mathsf{GreatApes} \sqsubseteq \exists\mathsf{recognize}.\mathsf{Self}$ is equivalent to $\mathsf{GreatApes} \sqcap X \sqsubseteq \exists\mathsf{recognize}.X$. Third, we can express positive occurrences of (local) *role-value-map concepts* (see (Donini 2003)). For example, $\mathsf{Male} \sqsubseteq (\mathsf{isParentOf} \subseteq \mathsf{isFatherOf})$ is equivalent to $\mathsf{Male} \sqcap \exists\mathsf{isParentOf}.X \sqsubseteq \exists\mathsf{isFatherOf}.X$. Actually, *all* these examples are special cases of $\mathcal{ELX}_{F2}$ axioms of the form $C_0 \sqcap \exists r_1.(C_1 \sqcap \exists r_2.(C_2 \cdots \sqcap \exists r_n.(C_n \sqcap X)\dots)) \sqsubseteq \exists s.X$, $(n \geq 0)$.

Many motivating examples from Section 1 can be expressed in $\mathcal{ELX}_{F2}$. For example, we can define a generic concept $\mathsf{Application}[X, Y] \equiv \mathsf{Procedure} \sqcap \exists\mathsf{site}.X \sqcap \exists\mathsf{substance}.Y$ and use it to define $\mathsf{DesensitizingTooth} \sqsubseteq \mathsf{Application}[\mathsf{ToothStructure}, \mathsf{TopicalAnesthetic}]$. If we additionally define $\mathsf{LocalAnesthesia} \equiv \mathsf{Procedure} \sqcap \exists\mathsf{substance}.\mathsf{LocalAnesthetic}$ and $\mathsf{TopicalAnesthetic} \sqsubseteq \mathsf{LocalAnesthetic}$, we obtain the logical consequence $\mathsf{DesensitizingTooth} \sqsubseteq \mathsf{LocalAnesthesia}$. If we define a generic version of this concept $\mathsf{LocalAnesthesia}[X] \equiv \mathsf{LocalAnesthesia} \sqcap \exists\mathsf{site}.X$, using the schema semantics we can obtain the conclusion $\mathsf{DesensitizingTooth} \sqsubseteq \mathsf{LocalAnesthesia}[\mathsf{ToothStructure}]$, however, since the right-hand side of this entailment is not a pure concept (see Definition 5), by Theorem 1 we are not guaranteed to obtain all such entailments for the reasons discussed in Example 6.

Note that due to the purity condition, we also cannot use ground parameterized concepts in equivalences, such as $\mathsf{DogOwner} \equiv \mathsf{Owner}[\mathsf{Dog}]$, or, in the left-hand sides of axioms, such as $\mathsf{Owner}[\mathsf{Dog}] \sqsubseteq \mathsf{Human}$. However, in cases when generic concepts are defined using equivalences, such as $\mathsf{Owner}[X] \equiv \exists\mathsf{owns}.X$, we can lift this restriction because the problematic axioms can be *rewritten* to $\mathcal{ELX}_{F1}$; in our example: $\mathsf{DogOwner} \equiv \exists\mathsf{owns}.\mathsf{Dog}$ and $\exists\mathsf{owns}.\mathsf{Dog} \sqsubseteq \mathsf{Human}$. The axiom $\mathsf{Owner}[X \sqcap \mathsf{Pet}] \sqsubseteq \exists\mathsf{feeds}.X$ mentioned in Section 1 can be fixed in the same way. Using the same approach for the generic concepts of the previous paragraph, we could upgrade the definition of $\mathsf{DesensitizingTooth}$: $\mathsf{DesensitizingTooth} \equiv \mathsf{Application}[\mathsf{ToothStructure}, \mathsf{TopicalAnesthetic}]$ and even check entailments of $A \sqsubseteq \mathsf{LocalAnesthesia}[B]$ for any atomic concepts A and B.

Violations of other syntactic restrictions of $\mathcal{ELX}_{F1}$ could also be fixed by applying similar transformations. For example, the axiom $\mathsf{Child}[X] \equiv \exists\mathsf{hasFather}.X \sqcap \exists\mathsf{hasMother}.X$ is not linear. We could, however, potentially rewrite the ontology to $\mathcal{ELX}_{F1}$, by replacing this definition with a more general (linear) definition $\mathsf{Child}[X, Y] \equiv \exists\mathsf{hasFather}.X \sqcap \exists\mathsf{hasMother}.Y$. For the result to be in $\mathcal{ELX}_{F1}$, we need to make sure that the subsequent replacement of $\mathsf{Child}[C]$ with $\mathsf{Child}[C, C]$ in other places does not violate the linearity restrictions. This means that the usage of $\mathsf{Child}[C]$ on the left-hand side must not contain variables; instead C has to be ground, like, e.g., in the axiom $\mathsf{Child}[\mathsf{Dog}] \sqsubseteq \mathsf{Dog}$. This situation is reminiscent of syntactic restrictions in other DLs. For example, concept disjunctions can be allowed in the left-hand sides of $\mathcal{EL}$ axioms without any change in the expressive power.

8 Conclusions and Outlook

To handle large sets of similar axioms in ontologies more effectively, we draw inspiration from generics, a standard feature of modern object-oriented programming languages. In this paper, we introduce concept variables and parameterized concepts as an extension of existing DLs. This allows us to leverage the advantages of generics in programming for DLs: the definition of a (generic) concept can be reused in multiple places, reducing the need to copy and modify complex concept formulations. This approach promotes more modular ontology development and helps avoid errors that can arise from refactoring axioms.

While type inference for generic classes in compilers is typically defined using syntactic rules, our extension of DL is grounded in a robust model-theoretic semantics based on Second-Order Logic. This provides a strong foundation for reasoning with generics but also leads to high algorithmic complexity. To restore first-order expressiveness and decidability, we investigated the relationship between the second-order semantics and the schema semantics, which can be reduced to classical DL reasoning. To combine the advantages of both semantics, we identified a fragment of the extension of $\mathcal{EL}$ where the conclusions entailed by both semantics coincide, allowing us to translate well-behaved second-order reasoning into classical $\mathcal{EL}$ reasoning. Additionally, we demonstrated that with further restrictions on this fragment, decidability of reasoning can be achieved.

By Theorem 3, the complexity of checking entailment in our second fragment $\mathcal{ELX}_{F2}$ is in EXPTIME. The source of the exponential complexity is the number of different variables that appear in axioms of the ontology because the variables can be replaced with concepts from H^∞ independently from each other. Note that if the number of variables is bounded by a constant, by Theorem 3 the complexity of reasoning reduces to PTIME. As in existing ontologies the size of axioms is small compared to the size of the ontology, it is reasonable to assume that the number of different variables would indeed be similarly small. In terms of these complexity results, our fragment $\mathcal{ELX}_{F2}$ behaves similarly to Datalog. Note that we can express any Datalog rule such as $T(x,y) \wedge T(y,z) \rightarrow T(x,z)$ in $\mathcal{ELX}$ simply by converting predicates to parameterized concepts and replacing the Boolean connectives accordingly like $T[X,Y] \sqcap T[Y,Z] \sqsubseteq T[X,Z]$. However, this translation does not generally satisfy the restrictions of $\mathcal{ELX}_{F1}$, particularly, the linearity condition as seen in this example. Therefore, we could not easily use this reduction to prove that the complexity bound in Theorem 3 is tight.

In this paper, we focused on situations where the introduced semantics coincide, but it may also be possible to reduce second-order entailment to schema entailment even when they do not coincide. For instance, to compute the missing consequences from Examples 4 and 6, it seems sufficient to perform replacements of generic parameters under (stated or entailed) equivalences. For example, if we obtain the equivalence $A \equiv B$, we can add an axiom $F[\ldots A \ldots] \equiv F[\ldots B \ldots]$ for every parameterized ground concept containing A or B. In future work, we plan to lift some of the imposed syntactic restrictions using this approach. Another research direction is to extend generic parameters with bounds – commonly used in object-oriented programming – which prevent replacing variables with arbitrary values. For example, an axiom $\mathsf{Owner}[X \sqsubseteq \mathsf{Pet}] \sqsubseteq \exists \mathsf{feeds}.X$ would allow the replacement of concept variables X only with (entailed) sub-concepts of Pet, such as Dog, but not, say, Car.

In summary, the findings of this paper demonstrate that extending DLs with generic concepts is both feasible and useful. This extension can be achieved while maintaining decidability, provided certain reasonable restrictions are applied. With the help of abstract, generic concepts, we can describe concepts in a more modular and reusable way. Additionally, the introduction of concept variables enables the expression of facts in novel ways, without relying on special constructors. This paves the way for further expansions of DLs that aim to generalize axioms across a broad spectrum of specific cases.

A Acronyms

DL Description Logic
PL Propositional Logic
FOL First-Order Logic
ODP Ontology Design Pattern
ML Modal Logic

References

Baader, F.; Brandt, S.; and Lutz, C. 2005. Pushing the EL envelope. In Kaelbling, L. P., and Saffiotti, A., eds., *IJCAI-05, Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence, Edinburgh, Scotland, UK, July 30 - August 5, 2005*, 364–369. Professional Book Center.

Baader, F.; Lutz, C.; and Brandt, S. 2008. Pushing the EL envelope further. In Clark, K., and Patel-Schneider, P. F., eds., *Proceedings of the Fourth OWLED Workshop on OWL: Experiences and Directions, Washington, DC, USA, 1-2 April 2008*, volume 496 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Baader, F. 2003. Restricted role-value-maps in a description logic with existential restrictions and terminological cycles. In Calvanese, D.; Giacomo, G. D.; and Franconi, E., eds., *Proceedings of the 2003 International Workshop on Description Logics (DL2003), Rome, Italy September 5-7, 2003*, volume 81 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Blackburn, P.; de Rijke, M.; and Venema, Y. 2001. *Modal Logic*, volume 53 of *Cambridge Tracts in Theoretical Computer Science*. Cambridge University Press.

Borgida, A.; Horkoff, J.; Mylopoulos, J.; and Rosati, R. 2012. Experiences in mapping the business intelligence model to description logics, and the case for parametric concepts. In Kazakov, Y.; Lembo, D.; and Wolter, F., eds., *Proceedings of the 2012 International Workshop on Description Logics, DL-2012, Rome, Italy, June 7-10, 2012*, volume 846 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Colucci, S.; Noia, T. D.; Sciascio, E. D.; Donini, F. M.; and Ragone, A. 2010. Second-order description logics: Semantics, motivation, and a calculus. In Haarslev, V.; Toman, D.; and Weddell, G. E., eds., *Proceedings of the 23rd International Workshop on Description Logics (DL 2010), Waterloo, Ontario, Canada, May 4-7, 2010*, volume 573 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Donini, F. M. 2003. Complexity of reasoning. In Baader, F.; Calvanese, D.; McGuinness, D. L.; Nardi, D.; and Patel-Schneider, P. F., eds., *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press. 96–136.

Fitting, M. 2007. Modal proof theory. In Blackburn, P.; van Benthem, J. F. A. K.; and Wolter, F., eds., *Handbook of Modal Logic*, volume 3 of *Studies in logic and practical reasoning*. North-Holland. 85–138.

Garcia, R.; Jarvi, J.; Lumsdaine, A.; Siek, J. G.; and Willcock, J. 2003. A comparative study of language support for generic programming. In *Proceedings of the 18th annual ACM SIGPLAN conference on Object-oriented programing, systems, languages, and applications*, 115–134.

Goldblatt, R. 1975. First-order definability in modal logic. *J. Symb. Log.* 40(1):35–40.

He, Y.; Zheng, J.; and Lin, Y. 2015. Ontorat: Automatic generation and editing of ontology terms. In Couto, F. M., and Hastings, J., eds., *Proceedings of the International Con-*

ference on Biomedical Ontology, ICBO 2015, Lisbon, Portugal, July 27-30, 2015, volume 1515 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Hirschbrunn, J., and Kazakov, Y. 2023. Description Logics Go Second-Order - Extending EL with Universally Quantified Concepts. In Kutz, O.; Lutz, C.; and Ozaki, A., eds., *Proceedings of the 36th International Workshop on Description Logics (DL 2023)*, volume 3515 of *CEUR Workshop Proceedings*. Rhodes, Greece: CEUR. ISSN: 1613-0073.

Horrocks, I.; Kutz, O.; and Sattler, U. 2006. The even more irresistible SROIQ. In Doherty, P.; Mylopoulos, J.; and Welty, C. A., eds., *Proceedings, Tenth International Conference on Principles of Knowledge Representation and Reasoning, Lake District of the United Kingdom, June 2-5, 2006*, 57–67. AAAI Press.

Kazakov, Y.; Kroetzsch, M.; and Simancik, F. 2012. Practical reasoning with nominals in the EL family of description logics. In Brewka, G.; Eiter, T.; and McIlraith, S. A., eds., *Principles of Knowledge Representation and Reasoning: Proceedings of the Thirteenth International Conference, KR 2012, Rome, Italy, June 10-14, 2012*. AAAI Press.

Kazakov, Y.; Krötzsch, M.; and Simancik, F. 2014. The incredible ELK - from polynomial procedures to efficient reasoning with EL ontologies. *J. Autom. Reason.* 53(1):1–61.

Kindermann, C.; Lupp, D. P.; Sattler, U.; and Thorstensen, E. 2018. Generating ontologies from templates: A rule-based approach for capturing regularity. In Ortiz, M., and Schneider, T., eds., *Proceedings of the 31st International Workshop on Description Logics co-located with 16th International Conference on Principles of Knowledge Representation and Reasoning (KR 2018), Tempe, Arizona, US, October 27th - to - 29th, 2018*, volume 2211 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Kindermann, C.; Parsia, B.; and Sattler, U. 2019. Comparing approaches for capturing repetitive structures in ontology design patterns. In Janowicz, K.; Krisnadhi, A. A.; Poveda-Villalón, M.; Hammar, K.; and Shimizu, C., eds., *Proceedings of the 10th Workshop on Ontology Design and Patterns (WOP 2019) co-located with 18th International Semantic Web Conference (ISWC 2019), Auckland, New Zealand, October 27, 2019*, volume 2459 of *CEUR Workshop Proceedings*, 17–31. CEUR-WS.org.

Krieg-Brückner, B.; Mossakowski, T.; and Neuhaus, F. 2019. Generic ontology design patterns at work. In Barton, A.; Seppälä, S.; and Porello, D., eds., *Proceedings of the Joint Ontology Workshops 2019 Episode V: The Styrian Autumn of Ontology, Graz, Austria, September 23-25, 2019*, volume 2518 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Krötzsch, M.; Maier, F.; Krisnadhi, A.; and Hitzler, P. 2011. A better uncle for OWL: nominal schemas for integrating rules and ontologies. In Srinivasan, S.; Ramamritham, K.; Kumar, A.; Ravindra, M. P.; Bertino, E.; and Kumar, R., eds., *Proceedings of the 20th International Conference on World Wide Web, WWW 2011, Hyderabad, India, March 28 - April 1, 2011*, 645–654. ACM.

Skjæveland, M. G.; Forssell, H.; Klüwer, J. W.; Lupp, D. P.; Thorstensen, E.; and Waaler, A. 2017. Reasonable ontology templates: APIs for OWL. In Nikitina, N.; Song, D.; Fokoue, A.; and Haase, P., eds., *Proceedings of the ISWC 2017 Posters & Demonstrations and Industry Tracks co-located with 16th International Semantic Web Conference (ISWC 2017), Vienna, Austria, October 23rd - to - 25th, 2017*, volume 1963 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Skjæveland, M. G.; Lupp, D. P.; Karlsen, L. H.; and Forssell, H. 2018. Practical ontology pattern instantiation, discovery, and maintenance with reasonable ontology templates. In Vrandecic, D.; Bontcheva, K.; Suárez-Figueroa, M. C.; Presutti, V.; Celino, I.; Sabou, M.; Kaffee, L.; and Simperl, E., eds., *The Semantic Web - ISWC 2018 - 17th International Semantic Web Conference, Monterey, CA, USA, October 8-12, 2018, Proceedings, Part I*, volume 11136 of *Lecture Notes in Computer Science*, 477–494. Springer.