Similarity-based Relaxed Instance Queries in $\mathcal{EL}^{++}$

Andreas Ecke*

Theoretical Computer Science, TU Dresden
ecke@tcs.inf.tu-dresden.de

Abstract. Description Logic (DL) knowledge bases (KBs) allow to express knowledge about concepts and individuals in a formal way. This knowledge is typically crisp, i.e., an individual either is an instance of a given concept or it is not. However, in practice this is often too restrictive: when querying for instances, one may often also want to find suitable alternatives, i.e., individuals that are not instances of query concept, but could still be considered 'good enough'. Relaxed instance queries have been introduced to gradually relax this inference in a controlled way via the use of concept similarity measures (CSMs). So far, those algorithms only work for the DL $\mathcal{EL}$, which has limited expressive power. In this paper, we introduce a suitable CSM for $\mathcal{EL}^{++}$-concepts. $\mathcal{EL}^{++}$ adds nominals, role inclusion axioms, and concrete domains to $\mathcal{EL}$. We extend the algorithm to compute relaxed instance queries w.r.t. this new CSM, and thus to work for general $\mathcal{EL}^{++}$ KBs.

1 Introduction

Description Logics (DLs) are a family of knowledge representation formalisms widely used in AI to describe and reason about categories and objects (individuals) of an application domain [1]. Each DL has a set of concept constructors, that allow to build complex concepts to formalize those categories, and are used in axioms and assertions to define the relations between different concepts and individuals. The set of axioms and assertions that describe the terminological and the assertional knowledge of the application domain, respectively, are collected in the TBox and the ABox. Together, TBox and ABox form a DL knowledge base (KB).

The formal semantics of DLs allows for the definition of reasoning services, i.e., inferences that allow to compute implicit knowledge from that explicitly described in the KB. Standard reasoning services include consistency of a KB, subsumption tests between different concepts, and instance checking, which derives whether an individual is an instance of a concept or not. Those reasoning services have been implemented in many highly optimized DL systems. One DL

* Supported by the German Research Foundation (DFG) Graduiertenkolleg 1763 (QuantLA).

that is especially interesting in terms of reasoning is $\mathcal{EL}$; while quite restricted in the constructors it offers, all the standard inferences can be implemented using polynomial-time algorithms. $\mathcal{EL}$ has been extended to a maximal superset $\mathcal{EL}^{++}$ that still retains the favorable computational properties in [2, 3].

Since DLs are the underlying logics of the OWL ontology language and its profiles (including OWL 2 EL, which is based on $\mathcal{EL}^{++}$) standardized by the W3C, their usage has increased rapidly in many fields like the Semantic Web, biomedical ontologies and more. By now, there is a large collection of different KBs available written in those languages. However, for many applications, retrieving strict instances from these KBs is often too restrictive, as often one may want to find suitable alternatives, even in cases where no individual completely matches the query concept. Those alternatives may be individuals that are not strict instances, but fulfill most of the requirement and are thus quite similar to the query concept.

The reasoning services of relaxed instance queries has been introduced in [4]. This inference relaxes the instance retrieval problem to return more individuals by the use of concept similarity measure (CSM). Given a CSM and a threshold t, this inference will return all instances of concepts that have a similarity of at least t to the query concept. Algorithms to compute relaxed instances have been introduced for unfoldable and general $\mathcal{EL}$-TBoxes in [4, 5].

However, the limited expressiveness of $\mathcal{EL}$ is often a problem; especially for query answering, it is useful to be able to use concrete domains to describe quantitative aspects of individuals and use these for querying. For example, one can use this to describe the geographic location of objects, the bandwidth of servers or time points in measurement series and incorporate the similarity between these values to find relaxed instances. Similarly, other features of $\mathcal{EL}^{++}$, like nominals, that allow to refer to specific individuals in concepts, role inclusions, and domain and range restrictions can be very useful in practice.

In this paper, we will extend the problem of relaxed query answering to $\mathcal{EL}^{++}$. To do so, after formally defining $\mathcal{EL}^{++}$ and CSMs in Section 2, we introduce pseudo-interpretations in Section 3, which can than be used to define simulations and canonical models that correspond to the semantics of $\mathcal{EL}^{++}$. In Section 4 we define a parameterizable similarity measure on pointed pseudo-interpretations, which can be lifted to $\mathcal{EL}^{++}$-concepts using the canonical models. This CSM can then be used to query for relaxed instances in general $\mathcal{EL}^{++}$-KBs as shown in Section 5. We conclude the paper in Section 6.

2 Preliminaries

This section will give a brief introduction to the DL $\mathcal{EL}^{++}$ and define concept similarity measures and some of their properties.

2.1 The DL $\mathcal{EL}^{++}$

$\mathcal{EL}^{++}$ concepts are built from four countable, pairwise disjoint sets: The set N_C of concept names; the set N_R of role names; the set N_I of individual names; and

	syntax	usual semantics
concept name	A	$A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$
top concept	$\top$	$\top^{\mathcal{I}} = \Delta^{\mathcal{I}}$
bottom concept	$\bot$	$\bot^{\mathcal{I}} = \emptyset$
nominal	$\{o\}$	$\{o\}^{\mathcal{I}} = \{o^{\mathcal{I}}\}$
conjunction	$C \sqcap D$	$(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}$
existential restriction	$\exists r.C$	$(\exists r.C)^{\mathcal{I}} = \{d \in \Delta^{\mathcal{I}} \mid \exists e \in \Delta^{\mathcal{I}}: (d,e) \in r^{\mathcal{I}} \wedge e \in C^{\mathcal{I}}\}$
concrete domain	$p(f_1, \ldots, f_n)$	$p(f_1, \ldots, f_n)^{\mathcal{I}} = \{d \in \Delta^{\mathcal{I}} \mid (f_1^{\mathcal{I}}(d), \ldots, f_n^{\mathcal{I}}(d)) \in p^{\mathcal{D}}\}$
GCI	$C \sqsubseteq D$	$C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$
role inclusion	$r_1 \circ \cdots \circ r_n \sqsubseteq s$	$r_1^{\mathcal{I}} \circ \cdots \circ r_n^{\mathcal{I}} \subseteq s^{\mathcal{I}}$
domain restriction	$\mathsf{dom}(r) \sqsubseteq C$	$r^{\mathcal{I}} \subseteq C^{\mathcal{I}} \times \Delta^{\mathcal{I}}$
range restriction	$\mathsf{ran}(r) \sqsubseteq C$	$r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times C^{\mathcal{I}}$
concept assertion	$C(a)$	$a^{\mathcal{I}} \in C^{\mathcal{I}}$
role assertion	$r(a,b)$	$(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$

Table 1. Concept constructors, TBox axioms, and ABox assertions for $\mathcal{EL}^{++}$

the set N_F of feature names. Using the concept constructors given in the upper part of Table 1, these names are used to construct complex concept descriptions. The set of all $\mathcal{EL}^{++}$-concept descriptions is denoted with $\mathfrak{C}(\mathcal{EL}^{++})$.

When formulating a knowledge domain in terms of DLs, one expresses all classes of interest as (possibly complex) $\mathcal{EL}^{++}$-concepts, and possible relations between those classes as roles. The general knowledge about the classes can then be formalized using the axioms given in the middle part of Table 1, while the knowledge about specific objects can be expressed using concept and role assertions of the form $C(a)$ and $r(a,b)$. The axioms and assertions are collected in the TBox and ABox, respectively, which together form a knowledge base (KB).

$\mathcal{EL}^{++}$ allows the use of p-admissible concrete domains. Such a concrete domain $\mathcal{D} = (\Delta^{\mathcal{D}}, P^{\mathcal{D}})$ consist of a set of concrete values $\Delta^{\mathcal{D}}$ and a set of predicates $p \in P$, each associated with an arity $n > 0$ and an extension $p^{\mathcal{D}} \subseteq (\Delta^{\mathcal{D}})^n$. Features connect objects described by the DL to elements of the concrete domain. For example, using the concrete domain Q with $\Delta^Q = \mathbb{Q}$ the set of rational numbers and predicates $P = \{=, \geq_p, =_p\}$ for $p \in \mathbb{Q}$ with the obvious meanings, one can express that adults are persons that are at least 18 using $\mathsf{Adult} \sqsubseteq \mathsf{Person} \sqcap \geq_{18}(\mathsf{age})$ or that Anna is 171 cm tall and her age is the same as her shoe size: $(=_{171}(\mathsf{height}) \sqcap =(\mathsf{age}, \mathsf{shoeSize}))(\mathsf{anna})$.

P-admissible concrete domains in $\mathcal{EL}^{++}$ only allow for limited expressiveness, in order to retain tractability. Besides the obvious requirement that satisfiability and implication in these concrete domains must be decidable in polynomial time, there are two other changes when compared to general concrete domains: First, as there are no abstract features, predicates can only compare features of a single element. This means that $\mathcal{EL}^{++}$ does not allow to express

$\mathsf{Person} \sqsubseteq {<}(\mathsf{age}, \mathsf{mother} \circ \mathsf{age}) \sqcap {<}(\mathsf{age}, \mathsf{father} \circ \mathsf{age})$, i.e., that every person is younger than her parents. Second, the concrete domains need to be convex, i.e., if a set of predicates implies the disjunction of some predicates, then it must also imply one of the disjuncts. This is a rather big restriction, but there still exist useful p-admissible concrete domains, like those given in [2], which allow to refer to rational numbers and strings. Indeed, we argue that for our purpose, relaxed instance queries, for any set of concrete values $\Delta^{\mathcal{D}}$, even a single unary predicate $=_d$ for $d \in \Delta^{\mathcal{D}}$ to attach concrete values to individuals is useful.

Example 1. Consider the concrete domain G to represent geographic coordinates as a pair of latitude and longitude, with $\Delta^G = [-90, 90] \times [-180, 180] \subseteq \mathbb{R} \times \mathbb{R}$ and the unary predicates $=_p$ for $p \in \Delta^G$. This allows a service provider to describe the location of all its branch offices in the ABox using assertions like $(=_{(51.026,13.723)}(\mathsf{location}))(\mathsf{office1})$. If we construct the similarity measure used for relaxing the queries in such a way that it assigns larger similarities to locations closer together, an instance query which includes the predicate $=_l(\mathit{location})$ for the location of the user will try to find the closest branch offices that also match the rest of the query. Indeed, one could also construct a similarity measure that returns similarity 0 for locations more than a set distance away, allowing the user to specify the maximum distance. Thus, while the concrete domain itself is extremely inexpressive, it allows the relaxed instance queries to include the distance between locations in its similarity evaluation.

Note that in this paper we restrict to a single concrete domain $\mathcal{D}$, but it is easy to generalize the similarity measure and the algorithms to compute relaxed instance queries to handle multiple concrete domains at once. Also note that it is possible to remove domain and range restrictions from the KB without changing its semantics [3]. To do so, we can replace every domain restrictions $\mathit{dom}(r) \sqsubseteq C$ with $\exists r.\top \sqsubseteq C$ and for any range restrictions $\mathit{ran}(r) \sqsubseteq C$, we replace all $\exists r.D$ occurring in the KB with $\exists r.(C \sqcap D)$ and for any role assertion $r(a, b)$ we add $D(b)$. Thus, in the remainder of this paper, we assume that KBs do not contain any domain or range restrictions.

The semantics of $\mathcal{EL}^{++}$-concepts is given by means of interpretations. An interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ is a tuple consisting of a domain $\Delta^{\mathcal{I}}$ and an interpretation function $\cdot^{\mathcal{I}}$ that assigns to each concept name $C \in N_C$ a subset $C^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ of the the domain, to each role name $r \in N_R$ a binary relation $r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$, to each individual name $a \in N_I$ an element $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$, and to each feature name $f \in N_F$ a partial function $f^{\mathcal{I}} : \Delta^{\mathcal{I}} \nrightarrow \Delta^{\mathcal{D}}$. The interpretations can be extended to complex $\mathcal{EL}$-concepts as shown in the last column of Table 1.

Instead of viewing an interpretation $\mathcal{I}$ as a tuple of functions that assign subsets, binary relations and elements of $\Delta^{\mathcal{I}}$ to the elements of N_C, N_R and N_I, and partial functions to the elements of N_F, one can also view it as a tuple of functions

$$\mathcal{I} : (\Delta^{\mathcal{I}} \to \mathcal{P}(N_C), \Delta^{\mathcal{I}} \to \mathcal{P}(N_R \times \Delta^{\mathcal{I}}), \Delta^{\mathcal{I}} \to \mathcal{P}(N_I), \Delta^{\mathcal{I}} \to N_F \nrightarrow \Delta^{\mathcal{D}})$$

from the domain $\Delta^{\mathcal{I}}$ to a subset of N_C (the concept names that this element is an instance of), to a binary relation between N_R and $\Delta^{\mathcal{I}}$ (the successors of the

element), to a subset of N_I (the individual names that map to this element), and to a partial function mapping feature names to concrete values, respectively. If we require that each individual name occurs only once, this definition is equivalent to the usual one.

2.2 Concept similarity measures

Given a $\mathcal{EL}^{++}$-KB $\mathcal{K}$, a concept similarity measure (CSM) is a function $\sim_{\mathcal{K}} : \mathfrak{C}(\mathcal{EL}^{++}) \times \mathfrak{C}(\mathcal{EL}^{++}) \to [0,1]$ such that $C \sim_{\mathcal{K}} C = 1$ for all concepts C. Intuitively, a value $C \sim_{\mathcal{K}} D = 0$ means that the concepts C and D are totally dissimilar w.r.t. $\mathcal{K}$, while a value of 1 indicates total similarity. We often simply write $\sim$ instead of $\sim_{\mathcal{K}}$ if the KB $\mathcal{K}$ is clear from the context.

In [6] a set of properties for CSMs is defined. We extend the definition of these properties to the case of general TBoxes.

Definition 1. *A CSM* $\sim_{\mathcal{K}} : \mathfrak{C}(\mathcal{EL}^{++}) \times \mathfrak{C}(\mathcal{EL}^{++}) \to [0,1]$ *is:*
symmetric iff $C \sim_{\mathcal{K}} D = D \sim_{\mathcal{K}} C$*;*
equivalence invariant iff for all $C \equiv_{\mathcal{K}} D$ *it holds that* $C \sim_{\mathcal{K}} E = D \sim_{\mathcal{K}} E$*;*
equivalence closed iff $C \equiv_{\mathcal{K}} D \iff C \sim_{\mathcal{K}} D = 1$*;*
bounded iff the existence of $E \not\equiv_{\mathcal{K}} \top$ *with* $C \sqsubseteq_{\mathcal{K}} E$ *and* $D \sqsubseteq_{\mathcal{K}} E$ *implies* $C \sim_{\mathcal{K}} D > 0$*; and*
dissimilar closed iff $C, D \not\equiv_{\mathcal{K}} \top$ *and there is no* $E \not\equiv_{\mathcal{K}} \top$ *with* $C \sqsubseteq_{\mathcal{K}} E$ *and* $D \sqsubseteq_{\mathcal{K}} E$ *implies* $C \sim_{\mathcal{K}} D = 0$*.*

These formally defined properties make CSMs more predictable for users. The measures in [5–7] fulfill most of these properties. The measures from [5, 6] are additionally parameterizable, which allows users to calibrate the measure to fit their expectations. In our setting these parameterizable CSMs enable users to specify which features of query concepts should be relaxed.

3 Pseudo-interpretations

Unlike in $\mathcal{EL}$ without concrete domains, the definition of interpretations for $\mathcal{EL}^{++}$ given in the last section does not admit canonical models. For example, in the concrete domain of the rational numbers $Q = (\mathbb{Q}, P^Q)$ introduced before, a concept like $>_0(f)$ will have infinitely many models (one for each positive rational number) without any of them being preferable and therefore canonical. One way to avoid this problem and ensure the existence of canonical models is to only consider *pseudo-interpretations*.

These pseudo-interpretations differ from the usual interpretations in just the fourth component: Instead of assigning each domain element a partial function from the feature names to concrete elements, we simply assign to each element directly a subset of the set of all predicates of $\mathcal{D}$ over the feature names, denoted with $\mathrm{Pred}^{\mathcal{D}}(N_F)$. In that way, each pseudo-interpretation corresponds to a set of usual interpretations, namely all those whose concrete elements assigned to the feature names of a domain element satisfy all the predicates mapped to the domain element by the pseudo-interpretation.

Definition 2. *A* pseudo-interpretation $\mathcal{J} = (\Delta^\mathcal{J}, f_C^\mathcal{J}, f_R^\mathcal{J}, f_I^\mathcal{J}, f_F^\mathcal{J})$ *consists of an interpretation domain* $\Delta^\mathcal{J}$ *and the interpretation functions* $f_C^\mathcal{J} : \Delta^\mathcal{J} \to \mathcal{P}(N_C)$, $f_R^\mathcal{J} : \Delta^\mathcal{J} \to \mathcal{P}(N_R \times \Delta^\mathcal{J})$, $f_I^\mathcal{J} : \Delta^\mathcal{J} \to \mathcal{P}(N_I)$, *and* $f_F^\mathcal{J} : \Delta^\mathcal{J} \to \mathcal{P}(Pred^\mathcal{D}(N_F))$, *such that for each* $a \in N_I$ *there exists exactly one* $d \in \Delta^\mathcal{J}$ *with* $a \in f_I^\mathcal{J}(d)$, *and the conjunction*

$$conj((\mathcal{J}, d)) = \bigwedge_{p(f_1,\ldots,f_n) \in f_F^\mathcal{J}(d)} p(f_1, \ldots, f_n)$$

is satisfiable in $\mathcal{D}$ *for any* $d \in \Delta^\mathcal{J}$.

Pseudo-interpretations can be used exactly as usual interpretations, with the exception that it does not interpret feature names itself; however, it does interpret predicates of the concrete domain:

$$\begin{aligned} A^\mathcal{J} &= \{d \in \Delta^\mathcal{J} \mid A \in f_C^\mathcal{J}(d)\} \\ r^\mathcal{J} &= \{(d, e) \in \Delta^\mathcal{J} \times \Delta^\mathcal{J} \mid (r, e) \in f_R^\mathcal{J}(d)\} \\ a^\mathcal{J} &= d \iff a \in f_I^\mathcal{J}(d) \\ p(f_1, \ldots, f_n)^\mathcal{J} &= \{d \in \Delta^\mathcal{J} \mid \mathcal{D} \models \text{conj}((\mathcal{J}, d)) \Rightarrow p(f_1, \ldots, f_n)\} \end{aligned}$$

Any other concept constructors, axioms, and assertions can then be interpreted as given in Table 1. We say that a pseudo-interpretation $\mathcal{J}$ is a model a KB $\mathcal{K}$, if it satisfies all axioms and assertions in $\mathcal{K}$. This is the case if and only if all corresponding usual interpretations are models of $\mathcal{K}$.

We call a pair $(\mathcal{J}, d)$ consisting of a pseudo-interpretation $\mathcal{J}$ and an element $d \in \Delta^\mathcal{J}$ a *pointed pseudo-interpretation* and denote the set of all pointed pseudo-interpretations as $\mathfrak{P}$. We sometimes use $f_C(p)$ (and similarly for f_R, f_I and f_F) instead of $f_C^\mathcal{J}(d)$ for $p = (\mathcal{J}, d)$.

3.1 Simulations

Simulations allow the characterization of elements of interpretations w.r.t. the concepts they are instance of. To extend the simulation relation between interpretations w.r.t. $\mathcal{EL}$ given in [8] to pseudo-interpretations w.r.t. $\mathcal{EL}^{++}$, we observe the following:

- role inclusions, range and domain restrictions are not concept constructors, and thus do not matter for the set of concepts that an element of a pseudo-interpretation is instance of;
- the bottom concept $\bot$ can not occur in pseudo interpretations;
- nominals allow to use individual names in concepts, and thus simulations need to preserve individuals; and
- for concrete domains, simulations need to preserve the valuations that satisfy the elements, which can be formalized using implications between the predicate sets of pointed pseudo-interpretations.

Thus, we can define a simulation relation for $\mathcal{EL}^{++}$ as follows:

Definition 3. *Let $\mathcal{J}_1$ and $\mathcal{J}_2$ be pseudo-interpretations. A relation $S \subseteq \Delta^{\mathcal{J}_1} \times \Delta^{\mathcal{J}_2}$ is a* simulation *between $\mathcal{J}_1$ and $\mathcal{J}_2$, if the following conditions hold:*

1. *For all $(d, e) \in S$ and $A \in N_C$, if $d \in A^{\mathcal{J}_1}$ then $e \in A^{\mathcal{J}_2}$.*
2. *For all $(d, e) \in S$, $r \in N_R$ and $(d, d') \in r^{\mathcal{J}_1}$, there is an $(e, e') \in r^{\mathcal{J}_2}$ with $(d', e') \in S$.*
3. *For all $(d, e) \in S$ and $a \in N_I$, if $d = a^{\mathcal{J}_1}$ then $e = a^{\mathcal{J}_2}$.*
4. *For all $(d, e) \in S$, we have that $\mathcal{D} \models \mathit{conj}((\mathcal{J}_2, e)) \Rightarrow \mathit{conj}((\mathcal{J}_1, d))$.*

Given two pointed pseudo-interpretations $p = (\mathcal{J}_1, d)$ and $q = (\mathcal{J}_2, e)$, we say that p simulates q (denoted $p \lesssim q$), if there exists a simulation $S \subseteq \Delta^{\mathcal{J}_1} \times \Delta^{\mathcal{J}_2}$ between $\mathcal{J}_1$ and $\mathcal{J}_2$ with $(d, e) \in S$. p and q are equisimilar (denoted $p \simeq q$), if $p \lesssim q$ and $q \lesssim p$.

This definition of simulations is reasonable, as it corresponds with the set of concepts that the elements in the simulation are instances of. Indeed, we can extent the following result from [8] to simulations of pseudo-interpretations in $\mathcal{EL}^{++}$:

Theorem 1. *Let p and q be pointed pseudo-interpretations, then:*

1. *$p \lesssim q$ iff $\mathfrak{C}(p) \subseteq \mathfrak{C}(q)$, and*
2. *$p \simeq q$ iff $\mathfrak{C}(p) = \mathfrak{C}(q)$.*

3.2 Canonical models

Next, we need to define canonical models for $\mathcal{EL}^{++}$. For these, the additional axioms like role inclusions *are* important. However, if the concept C contains the bottom concept $\bot$, it must be equivalent to $\bot$, and thus can not be instance of any element in an interpretation – in particular, it does not have a canonical model. Thus, by requiring that C is satisfiable w.r.t. $\mathcal{K}$, we do not have to worry about $\bot$ at all.

Since individuals can be part of concepts via nominals, we need to take care of the case that 2 individuals are equivalent, e.g. by the GCI $\{a\} \sqsubseteq \{b\}$. In this case, we cannot create two elements in the canonical interpretation for the two concepts $\{a\}$ and $\{b\}$, since this would not yield a model of the TBox anymore. Instead, we need to take one representative for all equivalence classes of concepts that are subsumed by the same individual:

$$[C] = \{D \in \mathfrak{C}(\mathcal{EL}^{++}) \mid \exists a \in N_I \colon \mathcal{K} \models C \sqsubseteq \{a\} \wedge \mathcal{K} \models D \sqsubseteq \{a\}\}$$

Finally, we need the notion $\mathrm{Sig}(X)$ of the signature of X, i.e., the set of all concept, role, individual and feature names occurring in X, and the notion $\mathrm{sub}(C)$ and $\mathrm{sub}(\mathcal{K})$ of the set of all sub-concepts of C and all sub-concepts of concepts occurring in $\mathcal{K}$, respectively. Then, we can define canonical models as follows:

Definition 4. *Let $\mathcal{K}$ be a satisfiable $\mathcal{EL}^{++}$-KB and $C \in \mathfrak{C}(\mathcal{EL}^{++})$ be an $\mathcal{EL}^{++}$-concept with $C \not\equiv_{\mathcal{K}} \bot$. The* canonical model $\mathcal{J}_{C,\mathcal{K}} = (\Delta^{\mathcal{J}_{C,\mathcal{K}}}, f_C, f_R, f_I, f_F)$ of C *w.r.t. $\mathcal{K}$ is a pseudo-interpretations defined as follows:*

– $\Delta^{\mathcal{J}_{C,\mathcal{K}}} = \{d_{[C]}\} \cup \{d_{[\{a\}]} \mid a \in (Sig(\mathcal{K}) \cup Sig(C)) \cap N_I\} \cup \{d_{[D]} \mid \exists r.D \in sub(C) \cup sub(\mathcal{K})\}$,

and for all $d_{[D]}$ *in* $\Delta^{\mathcal{J}_{C,\mathcal{K}}}$*:*

– $f_C(d_{[D]}) = \{A \in N_C \mid \mathcal{K} \models D \sqsubseteq A\}$,
– $f_R(d_{[D]}) = \{(r, d_{[E]}) \in N_R \times \Delta^{\mathcal{J}_{C,\mathcal{K}}} \mid \mathcal{K} \models D \sqsubseteq \exists r.E\}$,
– $f_I(d_{[D]}) = \{a \in N_I \mid \mathcal{K} \models D \sqsubseteq \{a\}\}$, *and*
– $f_F(d_{[D]}) = \{p(f_1, \ldots, f_n) \in Pred^{\mathcal{D}}(N_F) \mid \mathcal{K} \models D \sqsubseteq p(f_1, \ldots, f_n)\}$.

It can be shown that the canonical model $\mathcal{J}_{C,\mathcal{K}}$ is indeed a model of the KB $\mathcal{K}$, and its elements $d_{[D]}$ are instances of the corresponding concept D, for all $d_{[D]} \in \Delta^{\mathcal{J}_{C,\mathcal{K}}}$.

Lemma 1. *Let* $\mathcal{K}$ *be a satisfiable* $\mathcal{EL}^{++}$*-KB and* C, D *be* $\mathcal{EL}^{++}$*-concepts with* $C \not\equiv_{\mathcal{K}} \bot$*. Then:*

1. *if* $d_{[D]} \in \Delta^{\mathcal{J}_{C,\mathcal{K}}}$*, then* $d_{[D]} \in D^{\mathcal{J}_{C,\mathcal{K}}}$*, and*
2. $\mathcal{J}_{C,\mathcal{K}} \models \mathcal{K}$.

Finally, it can be shown that the canonical model is indeed 'canonical', i.e., it can simulate all other models (and is thus least w.r.t. $\lesssim$):

Theorem 2. *Let* $\mathcal{K}$ *be a satisfiable* $\mathcal{EL}^{++}$*-KB and* C, D *be* $\mathcal{EL}^{++}$*-concepts with* $C \not\equiv_{\mathcal{K}} \bot$*. Then:*

1. *for all pseudo-models* $\mathcal{J}$ *of* $\mathcal{K}$ *and all elements* $d \in \Delta^{\mathcal{J}}$ *it holds* $d \in C^{\mathcal{J}}$ *iff* $(\mathcal{J}_{C,\mathcal{K}}, d_{[C]}) \lesssim (\mathcal{J}, d)$,
2. *for all pseudo-models* $\mathcal{J}$ *of* $\mathcal{K}$*, all individuals* a *occurring in* $\mathcal{K}$*, and all elements* $d \in \Delta^{\mathcal{J}}$ *it holds* $d = a^{\mathcal{J}}$ *iff* $(\mathcal{J}_K, d_{[\{a\}]}) \lesssim (\mathcal{J}, d)$*, and*
3. $C \sqsubseteq_{\mathcal{K}} D$ *iff* $d_{[C]} \in D^{\mathcal{J}_{C,\mathcal{K}}}$ *iff* $(\mathcal{J}_{D,\mathcal{K}}, d_{[D]}) \lesssim (\mathcal{J}_{C,\mathcal{K}}, d_{[C]})$.

Those results, besides being needed to prove formal properties of the similarity measure, show that canonical models are reasonably defined.

4 A Concept Similarity Measure for $\mathcal{EL}^{++}$

Similarly to [5], we will define the CSM via a similarity measure on pointed pseudo-interpretations, by translating the concepts into interpretations by taking their canonical model. To define the similarity measure on pointed pseudo-interpretations, we need a few basic ingredients:

– a primitive measure $\sim_{prim}$: $N_C \times N_C \cup N_R \times N_R \cup N_I \times N_I \to [0, 1]$ that assigns a similarity value to each pair of concept names, role names, and individual names,
– a weighting function $g : N_C \cup N_R \cup N_I \cup N_F \to \mathbb{R}_{>0}$, which allows more important features of interpretations to contribute more to the final similarity values than others,
– a similarity measure $\sim_{\mathcal{D}}$: $\Delta^{\mathcal{D}} \times \Delta^{\mathcal{D}} \to [0, 1]$ on the concrete domain,
– a discounting factor $w \in (0, 1)$, and a concrete domain factor $c > 0$.

We will extend the concrete similarity measure $\sim_{\mathcal{D}}$ to handle undefined values, i.e., $\sim_{\mathcal{D}}$: $(\Delta^{\mathcal{D}} \cup \{\bot\}) \times (\Delta^{\mathcal{D}} \cup \{\bot\}) \to [0,1]$ by setting $\bot \sim_{\mathcal{D}} d = d \sim_{\mathcal{D}} \bot = 0$ for $d \in \Delta^{\mathcal{D}}$ and $\bot \sim_{\mathcal{D}} \bot = 1$. This can be further extended to similarity on valuations, i.e., partial functions $u, v : N_F \nrightarrow \Delta^{\mathcal{D}}$, by computing the weighted average of the similarity values for all features:

$$u \sim_{\mathcal{D}} v = \frac{\sum_{f \in \mathrm{dom}(u) \cup \mathrm{dom}(v)} g(f) \cdot \mathrm{sim}_{\mathcal{D}}(u(f), v(f))}{\sum_{f \in \mathrm{dom}(u) \cup \mathrm{dom}(v)} g(f)}.$$

Finally, we can define the similarity of conjunctions of predicates on the concrete domain using a similar construction to the Hausdorff metric, where the valuations u, v are restricted to those feature names occurring in $f_F(p)$ or $f_F(q)$:

$$\mathrm{sim}_{\mathcal{D}}(p,q) = \min\left(\inf_{u \models \mathrm{conj}(p)} \sup_{v \models \mathrm{conj}(q)} u \sim_{\mathcal{D}} v, \inf_{u \models \mathrm{conj}(q)} \sup_{v \models \mathrm{conj}(p)} u \sim_{\mathcal{D}} v \right)$$

All other things, i.e., concept names, successors, and individual names, can be compared directly. For this, we introduce a new role $r_\top$ and a new individual $a_\top$, in case that a pointed pseudo-interpretation does not have any successors or individuals, similarly to how $\top$ is used for concept names. Then we can define for a pointed pseudo-interpretation p:

- $\mathrm{CN}(p) = \begin{cases} f_C(p) & \text{if } f_C(p) \neq \emptyset \\ \{\top\} & \text{otherwise} \end{cases}$, the set of concept names of p,
- $\mathrm{SC}(p) = \begin{cases} f_R(p) & \text{if } f_R(p) \neq \emptyset \\ \{(r_\top, d)\} & \text{otherwise} \end{cases}$, the set of successors of p,
- $\mathrm{IN}(p) = \begin{cases} f_I(p) & \text{if } f_I(p) \neq \emptyset \\ \{a_\top\} & \text{otherwise} \end{cases}$, the set of individuals of p.

To compare how similar two pointed pseudo-interpretations are for these aspects, we use pairings. A pairing $P \subseteq X \times Y$ between sets X and Y is a total binary relation, where totality means that all elements $x \in X$ and $y \in Y$ occur in some some tuple of P. For two pointed pseudo-interpretations $p = (\mathcal{J}_1, d)$ and $q = (\mathcal{J}_2, e)$, we are mainly interested in the following pairings:

- $P_C(p,q) \subseteq \mathcal{P}((\mathrm{CN}(p) \times \mathrm{CN}(q)) \setminus \{(\top, \top)\})$ is the set of all concept name pairings on the concepts that p and q are instance of.
- $P_S(p,q) \subseteq \mathcal{P}((\mathrm{SC}(p) \times \mathrm{SC}(q)) \setminus \{((r_\top, d), (r_\top, e))\})$ is the set of all successor pairings of p and q.
- $P_I(p,q) \subseteq \mathcal{P}((\mathrm{IN}(p) \times \mathrm{IN}(q)) \setminus \{(a_\top, a_\top)\})$ is the set of all individual pairings of p and q.

Using these pairings, we can finally define the similarity measure $\sim_i$ for pointed pseudo-interpretations. It works by averaging over the weighted similarity of the pairs in the best concept name, successor, and individual pairings. The similarity between pairs of successors is computed recursively. If at least one of the pointed interpretations contain any predicates, the similarity between these predicates as defined before is added, weighted with the concrete domain factor c.

$$p \sim_i q = \max_{\substack{p_C \in P_C(p,q) \\ p_S \in P_S(p,q) \\ p_I \in P_I(p,q)}} \frac{\mathrm{sim}_C(p_C) + \mathrm{sim}_S(p_S) + \mathrm{sim}_I(p_I) + \mathrm{sim}_F(p,q)}{\sum_{(A,B)\in p_C} g(A,B) + \sum_{((r,d),(s,e))\in p_S} g(r,s) + \sum_{(a,b)\in p_I} g(a,b) + g_F(p,q)}$$

with:

$$\mathrm{sim}_C(p_C) = \sum_{(A,B)\in p_C} g(A,B)(A \sim_{\mathrm{prim}} B),$$

$$\mathrm{sim}_S(p_S) = \sum_{((r,d),(s,e))\in p_S} g(r,s)(r \sim_{\mathrm{prim}} s)(w + (1-w)(\mathcal{J}_1, d) \sim_i (\mathcal{J}_2, e)),$$

$$\mathrm{sim}_I(p_I) = \sum_{(a,b)\in p_I} g(a,b)(a \sim_{\mathrm{prim}} b),$$

$$\mathrm{sim}_F(p,q) = g_F(p,q) \cdot \mathrm{sim}_{\mathcal{D}}(p,q),$$

$$g_F(p,q) = \begin{cases} c & \text{if } f_F(p) \neq \emptyset \vee f_F(q) \neq \emptyset \\ 0 & \text{otherwise} \end{cases}.$$

Since $\sim_i$ can be seen as a contraction mapping on the similarity values between all elements of $\mathcal{J}_1$ and $\mathcal{J}_2$, the Banach fixed-point theorem will yield the following result:

Theorem 3. *$\sim_i$ is well-defined, i.e., $p \sim_i q$ has a unique solution.*

This definition of $\sim_i$ is not equivalence invariant and equisimulation closed. In order to regain these properties, we need to normalize the pointed pseudo-interpretations before evaluating $\sim_i$. We say that an pseudo-interpretation $\mathcal{J}$ is in *normal form* if there are no elements $a, b, c \in \Delta^{\mathcal{J}}$ with $\{(a,b),(a,c)\} \in r^{\mathcal{J}}$ and $(\mathcal{J}, b) \lesssim (\mathcal{J}, c)$, i.e., no node has two successor nodes for the same role name that are in a simulation relation.
Any pseudo-interpretation $\mathcal{J}$ can be transformed into normal form as follows:

1. remove all edges $(a,b) \in r^{\mathcal{J}}$ from $\mathcal{J}$, for which there exists an edge $(a,c) \in r^{\mathcal{J}}$ such that $(\mathcal{J}, b) \lesssim (\mathcal{J}, c)$ but not $(\mathcal{J}, b) \simeq (\mathcal{J}, c)$;
2. for all edges $(a, b_0) \in r^{\mathcal{J}}$, check if there are other edges $(a, b_i) \in r^{\mathcal{J}}$, $i > 0$, with $(\mathcal{J}, b_0) \simeq (\mathcal{J}, b_i)$ and choose one representative b_j; then remove all other edges (a, b_i), $i \neq j$, from $r^{\mathcal{J}}$.

Finally, we can define the CSM $\sim_c$ for concept descriptions w.r.t. an $\mathcal{EL}^{++}$ KB $\mathcal{K}$ as follows:

$$C \sim_c D = (\mathcal{J}'_{C,\mathcal{K}}, d_{[C]}) \sim_i (\mathcal{J}'_{D,\mathcal{K}}, d_{[D]}),$$

where $\mathcal{J}'_{C,\mathcal{K}}$ and $\mathcal{J}'_{D,\mathcal{K}}$ are the normalized canonical models of C and D w.r.t. $\mathcal{K}$. If C or D are equivalent to $\bot$, they do not have a canonical model. In this case, we set $C \sim_c \bot = \bot \sim_c D = 0$ for $C, D \not\equiv_{\mathcal{K}} \bot$. $\sim_c$ has all of the properties given in Definition 1:

Theorem 4. *The CSM $\sim_c$ is symmetric, bounded, dissimilar closed, equivalence invariant, and equivalence closed.*

5 Relaxed instance queries w.r.t. $\sim_c$

We established in [5] that in order to compute the maximal similarity between a query concept C and all concepts that an individual a is instance of, it is enough to check all *generalized concepts* of the msc of a, or in terms of $\sim_i$: It is enough to compute the maximal similarity $(\mathcal{J}'_{C,\mathcal{K}}, d_{[C]}) \sim_i q$ for all pointed interpretations $(\mathcal{J}_{\top,\mathcal{K}}, d_{[\{a\}]}) \lesssim q$. This can also be achieved by using $(\mathcal{J}_{\top,\mathcal{K}}, d_{[\{a\}]})$ directly in the computation of the $\sim_i$ and allowing to generalize this pointed pseudo-interpretations, i.e. finding the best subsets of f_C, f_R, and f_I, and taking the best set of predicates that follow from f_F.

Since f_C, f_R and f_I are finite, finding the best subsets is always possible by checking all of them. However, there can be infinitely many predicate sets following from f_F. Note that in order to maximize $\mathrm{sim}_{\mathrm{conc}}(p, q)$, generalizing q can always increase the left part of $\mathrm{sim}_{\mathrm{conc}}(p, q)$, $\inf_{u \models \mathrm{conj}(p)} \sup_{v \models \mathrm{conj}(q)} \mathrm{sim}_{\mathcal{D}}(u, v)$, to a value of 1, by simply taking the empty set of predicates (which has all valuations as model), but it can never increase the right part. Thus, the maximal value for $\mathrm{sim}_{\mathrm{conc}}(p, q)$ that can be achieved by generalizing q is simply $\inf_{u \models \mathrm{conj}(q)} \sup_{v \models \mathrm{conj}(p)} \mathrm{sim}_{\mathcal{D}}(u, v)$.

Procedure: maxsim $(\mathcal{J}_1, \mathcal{J}_2, \sim_{\mathrm{prim}}, \sim_{\mathcal{D}}, g, w, c)$

Input: $\mathcal{J}_1, \mathcal{J}_2$: finite pseudo-interpretations; $\sim_{\mathrm{prim}}$: primitive measure; $\sim_{\mathcal{D}}$: similarity measure on $\mathcal{D}$; g: weighting function; $w \in (0, 1)$: discount factor; $c > 0$: concrete domain factor

Output: maximal similarities between $p = (\mathcal{J}_1, a)$ and all generalizations of $q = (\mathcal{J}_2, b)$

1: $\mathrm{msim}_0(d, e) \leftarrow 0$ for all $d \in \Delta^{\mathcal{J}_1}$ and $e \in \Delta^{\mathcal{J}_2}$
2: **for** $i \leftarrow 1, 2, 3, \ldots$ **do**
3: **for all** $d \in \Delta^{\mathcal{J}_1}$ and $e \in \Delta^{\mathcal{J}_2}$ **do**
4: $\mathrm{msim}_i(d, e) \leftarrow \max_{\substack{S_{\mathrm{CN}} \subseteq \mathrm{CN}(e) \\ S_{\mathrm{SC}} \subseteq \mathrm{SC}(e) \\ S_{\mathrm{IN}} \subseteq \mathrm{IN}(e)}} \Big(\max_{\substack{p_C \subseteq \mathrm{CN}(d) \times S_{\mathrm{CN}} \\ p_S \subseteq \mathrm{SC}(d) \times S_{\mathrm{SC}} \\ p_I \subseteq \mathrm{IN}(d) \times S_{\mathrm{IN}}}} \mathsf{similarity}(p_C, p_S, p_I, d, e, i) \Big)$
5: **end for**
6: **end for**

Procedure: similarity (p_C, p_S, p_I, d, e, i)

1: $\mathrm{sim}(p_C) \leftarrow \sum_{(A,B) \in p_C} g(A, B)(A \sim_{\mathrm{prim}} B)$
2: $\mathrm{sim}(p_S) \leftarrow \sum_{((r,p'),(s,q')) \in p_S} g(r, s)(r \sim_{\mathrm{prim}} s)\big((1 - w) + w \cdot \mathrm{msim}_{i-1}(p', q)\big)$
3: $\mathrm{sim}(p_I) \leftarrow \sum_{(a,b) \in p_I} g(a, b)(a \sim_{\mathrm{prim}} b)$
4: $g_F \leftarrow c$ if $f_F(p) \neq \emptyset \vee f_F(q) \neq \emptyset$; $\quad g_F \leftarrow 0$ otherwise
5: **return** $\dfrac{\mathrm{sim}(p_C) + \mathrm{sim}(p_S) + \mathrm{sim}(p_I) + g_F(p, q) \cdot \big(\inf_{u \models \mathrm{conj}(\mathcal{J}_2, e)} \sup_{v \models \mathrm{conj}(\mathcal{J}_1, d)} u \sim_{\mathcal{D}} v \big)}{\sum_{(A,B) \in p_C} g(A, B) + \sum_{((r,d),(s,e)) \in p_S} g(r, s) + \sum_{(a,b) \in p_I} g(a, b) + g_F(p, q)}$

Fig. 1. Algorithm to compute the maximal similarities between all elements of the finite pseudo interpretation $\mathcal{J}_1$ and all generalizations of the finite pseudo interpretation $\mathcal{J}_2$.

Procedure: relaxed-instances $(Q, \mathcal{K}, t, \sim_{\text{prim}}, \sim_{\mathcal{D}}, g, w, c)$
Input: Q: $\mathcal{EL}$-concept; $\mathcal{K} = (\mathcal{T}, \mathcal{A})$: $\mathcal{EL}^{++}$-KB; $t \in [0, 1]$: threshold; $\sim_{\text{prim}}$: primitive measure; $\sim_{\mathcal{D}}$: concrete measure; g: weighting function; $w \in (0, 1)$: discounting factor; $c > 0$: concrete factor
Output: individuals $a \in Relax_t^{\sim_c}(Q)$

1: compute canonical models $\mathcal{I}_{Q,\mathcal{T}}$ and $\mathcal{I}_{\mathcal{K}}$
2: $\text{maxsim}(d, e) \leftarrow \mathsf{maxsim}(\mathcal{J}_{Q,\mathcal{K}}, \mathcal{J}_{\top,\mathcal{K}}, \sim_{\text{prim}}, \sim_{\mathcal{D}}, g, w, c)$
3: **return** $\{a \in N_I \cap \text{Sig}(\mathcal{K}) \mid \text{maxsim}(d_{[Q]}, d_{[\{a\}]}) > t\}$

Fig. 2. Algorithm to compute all relaxed instances of a query concept Q w.r.t. a knowledge base $\mathcal{K}$ and threshold t.

The algorithm to compute the maximal similarities between elements of a pseudo-interpretation $\mathcal{J}_1$ and all generalizations of elements of a pseudo-interpretation $\mathcal{J}_2$ is shown in Figure 1. Using this, the algorithm to actually compute all relaxed instances of a query concept Q w.r.t. $\sim_c$ and an $\mathcal{EL}^{++}$-KB $\mathcal{K}$ is conceptually quite easy, as it only needs to compute the maximal similarities between Q and the individuals $a \in \mathcal{K}$ and check whether they are larger than t. The algorithm is depicted in Figure 2.

The maxsim_i values computed in the algorithm monotonically converge from below to the maximal similarities between generalized concepts of the individuals and the query concept. Thus, the algorithm is sound and complete in the following sense:

Theorem 5. *Let $\sim_c$ be the CSM derived from $\sim_i$ with the primitive measure $\sim_{prim}$, concrete measure $\sim_{\mathcal{D}}$ and factor c, weighting function g and discounting factor w. Then the algorithm* relaxed-instances *is sound and complete:*

1. Soundness*: If $a \in$ relaxed-instances$(Q, \mathcal{K}, t, \sim_{prim}, \sim_{\mathcal{D}}, g, w, c)$ for a number n of iterations, then $a \in Relax_t^{\sim_c}(Q)$.*
2. Completeness*: If $a \in Relax_t^{\sim_c}(Q)$, then there exists an $i \in \mathbb{N}$ such that for $n \geq i$ iterations it holds that $a \in$ relaxed-instances$(Q, \mathcal{K}, t, \sim_{prim}, \sim_{\mathcal{D}}, g, w, c)$.*

Note the the number of of iterations i needed in the completeness part of Theorem 5 is not bounded. However, since the algorithm converges quite fast, this should not be a problem in most practical applications.

6 Conclusions

In this paper we extended the concepts similarity measure for general TBoxes introduced in [5] to the DL $\mathcal{EL}^{++}$. Since concrete domains do not allow do define canonical models for standard interpretations in $\mathcal{EL}^{++}$, we defined pseudo-interpretations, which correspond to a set of standard interpretations. This is used to define a similarity measure on pointed pseudo-interpretations, which is extended to concept descriptions w.r.t. a KB via the canonical models. We use the proposed CSM for relaxed instance querying of $\mathcal{EL}^{++}$ KBs and give an algorithm that computes all relaxed instances.

References

1. Baader, F., Calvanese, D., McGuinness, D.L., Nardi, D., Patel-Schneider, P.F., eds.: The Description Logic Handbook: Theory, Implementation, and Applications. Cambridge University Press, New York, NY, USA (2003)
2. Baader, F., Brandt, S., Lutz, C.: Pushing the $\mathcal{EL}$ envelope. In: Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence IJCAI-05, Edinburgh, UK, Morgan-Kaufmann Publishers (2005)
3. Baader, F., Brandt, S., Lutz, C.: Pushing the el envelope further. In Clark, K., Patel-Schneider, P.F., eds.: In Proceedings of the OWLED 2008 DC Workshop on OWL: Experiences and Directions. (2008)
4. Ecke, A., Peñaloza, R., Turhan, A.Y.: Towards instance query answering for concepts relaxed by similarity measures. In Godo, L., Prade, H., Qi, G., eds.: Workshop on Weighted Logics for AI (in conjunction with IJCAI'13), Beijing, China (2013)
5. Ecke, A., Peñaloza, R., Turhan, A.Y.: Answering instance queries relaxed by concept similarity. In: Proceedings of the Fourteenth International Conference on Principles of Knowledge Representation and Reasoning (KR'14), Vienna, Austria, AAAI Press (2014) To appear.
6. Lehmann, K., Turhan, A.Y.: A framework for semantic-based similarity measures for $\mathcal{ELH}$-concepts. In del Cerro, L.F., Herzig, A., Mengin, J., eds.: Proc. of the 13th European Conf. on Logics in A.I. (JELIA 2012). Lecture Notes In Artificial Intelligence, Springer (2012) 307–319
7. Suntisrivaraporn, B.: A similarity measure for the description logic $\mathcal{EL}$ with unfoldable terminologies. In: 5th International Conference on Intelligent Networking and Collaborative Systems (INCoS). (2013) 408–413
8. Lutz, C., Wolter, F.: Deciding inseparability and conservative extensions in the description logic $\mathcal{EL}$. Journal of Symbolic Computation **45**(2) (2010) 194–228