Towards Approximative Most Specific Concepts by Completion for $\mathcal{EL}$ with Subjective Probabilities

Rafael Peñaloza and Anni-Yasmin Turhan

TU Dresden, Germany, email: *last name*@tcs.inf.tu-dresden.de

1 Introduction

In Description Logics the inference *most specific concept* (msc) constructs a concept description that generalizes an individual into a concept description. For the Description Logic $\mathcal{EL}$ the msc needs not exist [1], if computed with respect to general $\mathcal{EL}$-TBoxes. However, it is still possible to find a concept description that is the msc up to a fixed role-depth. In this paper we present a practical approach for computing the role-depth bounded msc, based on the polynomial-time completion algorithm for $\mathcal{EL}$. We extend this method to a simple probabilistic variant of $\mathcal{EL}$ that can express subjective probabilities and that was recently introduced in [6]. The probabilistic DL that we use, called Prob-$\mathcal{EL}_c^{01}$, allows only a fairly limited use of uncertainty. More precisely, it is only possible to express that a concept *may* hold ($P_{>0}C$), or that it holds *almost surely* ($P_{=1}C$). Despite its limited expressivity, this logic is interesting due to its nice algorithmic properties; as shown in [6], subsumption can be decided in polynomial time and instance checking can be performed in polynomial time as well.

Many practical applications that need to represent probabilistic information, such as medical applications or context-aware applications, need to characterize that observations only hold with certain probability. Furthermore, these applications face the problem that information from different sources does not coincide or that different diagnoses yield differing results. These applications need to "integrate" differing observations for the same state of affairs. A way to determine what the different information sources agree upon is to represent this information as ABox individuals and to find a common generalization of these individuals. A description of such a generalization of a group of ABox individuals can be obtained by applying the so-called *bottom-up approach* for constructing knowledge bases [4]. In this approach a set of individuals is generalized into a single concept description by first generating the msc of each concept and then apply the least common subsumer (lcs) to the set of obtained concept descriptions to extract their commonalities.

The second step, i.e., a computation procedure for the approximative lcs has been investigated for $\mathcal{EL}$ and Prob-$\mathcal{EL}_c^{01}$ in [8]. In this paper we present a similar procedure for the msc. We devise a practical algorithm for computing the msc up to a certain role-depth for $\mathcal{EL}$ and Prob-$\mathcal{EL}_c^{01}$. The so-called k-msc obtained by the algorithm is still a generalization of the input, but not necessarily the least one – in this sense it is only an approximation of the msc. Moreover, our algorithms are based upon the completion algorithms for $\mathcal{EL}$ and Prob-$\mathcal{EL}_c^{01}$ and thus can be easily implemented on top of reasoners of these DLs. Due to space limitations the proofs can be found in [7].

2 $\mathcal{EL}$ and Prob-$\mathcal{EL}$

We introduce the DL $\mathcal{EL}$ and its probabilistic variant Prob-$\mathcal{EL}_c^{01}$. Let N_I, N_C and N_R be disjoint sets of *individual-*, *concept-* and *role names*, respectively. *Prob-$\mathcal{EL}_c^{01}$-concept descriptions* are built using the syntax rule

$$C ::= \top \mid A \mid C \sqcap D \mid \exists r.C \mid P_{>0}C \mid P_{=1}C,$$

where $A \in N_C$, and $r \in N_R$. $\mathcal{EL}$-concept descriptions are Prob-$\mathcal{EL}_c^{01}$-concept description that do not contain the constructors $P_{>0}$ or $P_{=1}$.

A *knowledge base* $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ consists of a TBox $\mathcal{T}$ and an ABox $\mathcal{A}$. An $\mathcal{EL}$- (Prob-$\mathcal{EL}_c^{01}$-)TBox is a finite set of *concept inclusions* (CIs) of the form $C \sqsubseteq D$, where C, D are $\mathcal{EL}$- (Prob-$\mathcal{EL}_c^{01}$-)concept descriptions. An $\mathcal{EL}$-ABox is a set of assertions of the form $C(a), r(a,b)$, where C is an $\mathcal{EL}$-concept description, $r \in N_R$, and $a, b \in N_I$. A Prob-$\mathcal{EL}_c^{01}$-ABox is a set of assertions of the form $C(a), r(a,b), P_{>0}r(a,b), P_{=1}r(a,b)$, where C is a Prob-$\mathcal{EL}_c^{01}$-concept description, $r \in N_R$, and $a, b \in N_I$.

The semantics of $\mathcal{EL}$ is defined by means of interpretations $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$ consisting of a non-empty *domain* $\Delta^\mathcal{I}$ and an *interpretation function* $\cdot^\mathcal{I}$ that assigns binary relations on $\Delta^\mathcal{I}$ to role names, subsets of $\Delta^\mathcal{I}$ to concepts and elements of $\Delta^\mathcal{I}$ to individual names. For a more detailed description of this semantics, see [3].

An interpretation $\mathcal{I}$ *satisfies* a concept inclusion $C \sqsubseteq D$, denoted as $\mathcal{I} \models C \sqsubseteq D$ if $C^\mathcal{I} \subseteq D^\mathcal{I}$; it *satisfies* an assertion $C(a)$ ($r(a,b)$), denoted as $\mathcal{I} \models C(a)$ ($\mathcal{I} \models r(a,b)$) if $a^\mathcal{I} \in C^\mathcal{I}$ ($(a^\mathcal{I}, b^\mathcal{I}) \in r^\mathcal{I}$). It is a *model* of a knowledge base $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ if it satisfies all CIs in $\mathcal{T}$ and all assertions in $\mathcal{A}$.

The semantics of Prob-$\mathcal{EL}_c^{01}$ generalizes the semantics of $\mathcal{EL}$. A *probabilistic interpretation* is of the form

$$\mathcal{I} = (\Delta^\mathcal{I}, W, (\mathcal{I}_w)_{w \in W}, \mu),$$

where $\Delta^\mathcal{I}$ is the (non-empty) *domain*, W is a set of *worlds*, μ is a discrete probability distribution on W, and for each world $w \in W$, $\mathcal{I}_w$ is a classical $\mathcal{EL}$ interpretation with domain $\Delta^\mathcal{I}$, where $a^{\mathcal{I}_w} = a^{\mathcal{I}_{w'}}$ for all $a \in N_I$, $w, w' \in W$. The probability that a given element of the domain $d \in \Delta^\mathcal{I}$ belongs to the interpretation of a concept name A is

$$p_d^\mathcal{I}(A) := \mu(\{w \in W \mid d \in A^{\mathcal{I}_w}\}).$$

The functions $\mathcal{I}_w$ and $p_d^\mathcal{I}$ are extended to complex concepts in the usual way for the classical $\mathcal{EL}$ constructors, where the extension to the new constructors P_* is defined as

$$(P_{>0}C)^{\mathcal{I}_w} := \{d \in \Delta^\mathcal{I} \mid p_d^\mathcal{I}(C) > 0\}, \qquad (P_{=1}C)^{\mathcal{I}_w} := \{d \in \Delta^\mathcal{I} \mid p_d^\mathcal{I}(C) = 1\}.$$

A probabilistic interpretation $\mathcal{I}$ *satisfies* a concept inclusion $C \sqsubseteq D$, denoted as $\mathcal{I} \models C \sqsubseteq D$ if for every $w \in W$ it holds that $C^{\mathcal{I}_w} \subseteq D^{\mathcal{I}_w}$. It is a *model* of a TBox $\mathcal{T}$ if it satisfies all concept inclusions in $\mathcal{T}$. Let C, D be two Prob-$\mathcal{EL}_c^{01}$ concepts and $\mathcal{T}$ a TBox. We say that C is *subsumed* by D w.r.t. $\mathcal{T}$ ($C \sqsubseteq_\mathcal{T} D$) if for every model $\mathcal{I}$ of $\mathcal{T}$ it holds that $\mathcal{I} \models C \sqsubseteq D$. The probabilistic interpretation $\mathcal{I}$ *satisfies* the assertion $P_{>0}r(a,b)$ if $\mu(\{w \in W \mid I_w \models r(a,b)\}) > 0$, and analogously for $P_{=1}r(a,b)$. $\mathcal{I}$ *satisfies* the ABox $\mathcal{A}$ if there is a $w \in W$ such that $\mathcal{I}_w \models \mathcal{A}$.

Finally, an individual $a \in N_I$ is an *instance* of a concept description C w.r.t. $\mathcal{K}$ ($\mathcal{K} \models C(a)$) if $\mathcal{I} \models C(a)$ for all models $\mathcal{I}$ of $\mathcal{K}$. The *ABox realization problem* is to compute for each individual a in $\mathcal{A}$ the set of named concepts from $\mathcal{K}$ that have a as an instance and that are least (w.r.t. $\sqsubseteq$). In this paper we are interested in computing most specific concepts.

Definition 1 (most specific concept). *Let $\mathcal{L}$ be a DL, $\mathcal{K}$ = ($\mathcal{T}$, $\mathcal{A}$) be a $\mathcal{L}$-KB. The* most specific concept *(msc) of an individual a from $\mathcal{A}$ is the $\mathcal{L}$-concept description C s. t.*

1. $\mathcal{K} \models C(a)$*, and*
2. *for each $\mathcal{L}$-concept description D holds: $\mathcal{K} \models D(a)$ implies $C \sqsubseteq_{\mathcal{T}} D$.*

The msc depends on the DL in use. For the DLs with conjunction as concept constructor the msc is, if it exists, unique up to equivalence. Thus it is justified to speak of *the* msc.

3 Completion-based Instance Checking Algorithms

Now we briefly sketch the completion algorithms for instance checking in $\mathcal{EL}$ [2] and Prob-$\mathcal{EL}_c^{01}$ [6].

3.1 Completion Algorithms for $\mathcal{EL}$

Assume we want to test for an $\mathcal{EL}$-KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ whether $\mathcal{K} \models D(a)$ holds. The completion algorithm first augments the knowledge base by introducing a concept name for the complex concept description D from the instance check, i.e., it sets $\mathcal{K} = (\mathcal{T} \cup \{A_q \equiv D\}, \mathcal{A})$, where A_q is a new concept name not appearing in $\mathcal{K}$. The instance checking algorithm for $\mathcal{EL}$ works on normalized knowledge bases. The normalization is done in two steps: first the ABox is transformed into a simple ABox. An ABox is a *simple ABox*, if it only contains concept names in concept assertions. An $\mathcal{EL}$-ABox $\mathcal{A}$ can be transformed into a simple ABox by first replacing each complex assertion $C(A)$ in $\mathcal{A}$ by $A(a)$ with a fresh name A and, second, introduce $A \equiv C$ in the TBox.

After this step the TBox is normalized. For a concept description C let $\mathsf{CN}(C)$ denote the set of all concept names and $\mathsf{RN}(C)$ denote the set of all role names that appear in C. The *signature of a concept description* C (denoted $\mathsf{sig}(C)$) is $\mathsf{CN}(C) \cup \mathsf{RN}(C)$. Similarly, the set of concept (role) names that appear in a TBox are denoted by $\mathsf{CN}(\mathcal{T})$ ($\mathsf{RN}(\mathcal{T})$). The *signature of a TBox* $\mathcal{T}$ (denoted $\mathsf{sig}(\mathcal{T})$) is $\mathsf{CN}(\mathcal{T}) \cup \mathsf{RN}(\mathcal{T})$. The *signature of an ABox* $\mathcal{A}$ (denoted $\mathsf{sig}(\mathcal{A})$) is the set of concept (role / individual) names $\mathsf{CN}(\mathcal{A})$ ($\mathsf{RN}(\mathcal{A})$/$\mathsf{IN}(\mathcal{A})$ resp.) that appear in $\mathcal{A}$. The signature of a KB $\mathcal{K}$ = ($\mathcal{T}$, $\mathcal{A}$) (denoted $\mathsf{sig}(\mathcal{K})$) is $\mathsf{sig}(\mathcal{T}) \cup \mathsf{sig}(\mathcal{A})$.

Now, an $\mathcal{EL}$-TBox $\mathcal{T}$ is in *normal form* if all concept axioms have one of the following forms, where $C_1, C_2 \in \mathsf{sig}(\mathcal{T})$ and $D \in \mathsf{sig}(\mathcal{T}) \cup \{\bot\}$:

$$C_1 \sqsubseteq D, \quad C_1 \sqcap C_2 \sqsubseteq D, \quad C_1 \sqsubseteq \exists r.C_2 \quad \text{or} \quad \exists r.C_1 \sqsubseteq D.$$

Any $\mathcal{EL}$-TBox can be transformed into normal form by introducing new concept names and by applying the normalization rules displayed in Figure 1 exhaustively. These rules replace the GCI on the left-hand side of the rules with the set of GCIs on the right-hand

$$\begin{array}{ll}
\textbf{NF1} & C \sqcap \hat{D} \sqsubseteq E \longrightarrow \{\, \hat{D} \sqsubseteq A, C \sqcap A \sqsubseteq E \,\} \\
\textbf{NF2} & \exists r.\hat{C} \sqsubseteq D \longrightarrow \{\, \hat{C} \sqsubseteq A, \exists r.A \sqsubseteq D \,\} \\
\textbf{NF3} & \hat{C} \sqsubseteq \hat{D} \longrightarrow \{\, \hat{C} \sqsubseteq A, A \sqsubseteq \hat{D} \,\} \\
\textbf{NF4} & B \sqsubseteq \exists r.\hat{C} \longrightarrow \{\, B \sqsubseteq \exists r.A, A \sqsubseteq \hat{C} \,\} \\
\textbf{NF5} & B \sqsubseteq C \sqcap D \longrightarrow \{\, B \sqsubseteq C, B \sqsubseteq D \,\}
\end{array}$$

where $\hat{C}, \hat{D} \notin \mathsf{BC}_{\mathcal{T}}$ and A is a new concept name.

Fig. 1. $\mathcal{EL}$ normalization rules (from [2])

side. Clearly, for a KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ the signature of $\mathcal{A}$ may be changed only during the first of the two normalization steps and the signature of $\mathcal{T}$ may be extended during both of them. The normalization of the KB can be done in linear time.

The completion algorithm for instance checking is based on the one for classifying $\mathcal{EL}$-TBoxes introduced in [2]. The completion algorithm constructs a representation of the minimal model of $\mathcal{K}$. Let $\mathcal{K}$ =($\mathcal{T}$, $\mathcal{A}$) be a normalized $\mathcal{EL}$-KB, i.e., with a simple ABox $\mathcal{A}$ and a TBox $\mathcal{T}$ in normal form. The completion algorithm works on four kinds of *completion sets*: $S(a), S(a,r), S(C)$ and $S(C,r)$ for each $a \in \mathsf{IN}(\mathcal{A})$, $C \in \mathsf{CN}(\mathcal{K})$ and $r \in \mathsf{RN}(\mathcal{K})$. The completion sets contain concept names from $\mathsf{CN}(\mathcal{K})$. Intuitively, the completion rules make implicit subsumption and instance relationships explicit in the following sense:

- $D \in S(C)$ implies that $C \sqsubseteq_{\mathcal{T}} D$,
- $D \in S(C,r)$ implies that $C \sqsubseteq_{\mathcal{T}} \exists r.D$.
- $D \in S(a)$ implies that a is an instance of D w.r.t. $\mathcal{K}$,
- $D \in S(a,r)$ implies that a is an instance of $\exists r.D$ w.r.t. $\mathcal{K}$.

$\mathsf{S}_{\mathcal{K}}$ denotes the set of all completion sets of a normalized $\mathcal{K}$. The completion sets are initialized for each $a \in \mathsf{IN}(\mathcal{A})$ and each $C \in \mathsf{CN}(\mathcal{K})$ as follows:

- $S(C) := \{C, \top\}$ for each $C \in \mathsf{CN}(\mathcal{K})$,
- $S(C,r) := \emptyset$ for each $r \in \mathsf{RN}(\mathcal{K})$,
- $S(a) := \{C \in \mathsf{CN}(\mathcal{A}) \mid C(a)$ appears in $\mathcal{A}\} \cup \{\top\}$, and
- $S(a,r) := \{b \in \mathsf{IN}(\mathcal{A}) \mid r(a,b)$ appears in $\mathcal{A}\}$ for each $r \in \mathsf{RN}(\mathcal{K})$.

Then these sets are extended by applying the completion rules shown in Figure 2 until no more rule applies. In these rules X and Y can refer to concept or individual names, while C, C_1, C_2 and D are concept names and r is a role name. After the completion has terminated, the following relations hold between an individual a, a role r and named concepts A and B:

- subsumption relation between A and B from $\mathcal{K}$ holds iff $B \in S(A)$
- instance relation between a and B from $\mathcal{K}$ holds iff $B \in S(a)$,

which has been shown in [2]. To decide the initial query: $\mathcal{K} \models D(a)$, one has to test now, whether A_q appears in $S(a)$. In fact, instance queries for all individuals and all named concepts from the KB can be answered now; the completion algorithm does not only perform one instance check, but complete ABox realization. The completion algorithm runs in polynomial time in size of the knowledge base.

CR1 If $C \in S(X)$, $C \sqsubseteq D \in \mathcal{T}$, and $D \notin S(X)$
then $S(X) := S(X) \cup \{D\}$

CR2 If $C_1, C_2 \in S(X)$, $C_1 \sqcap C_2 \sqsubseteq D \in \mathcal{T}$, and $D \notin S(X)$
then $S(X) := S(X) \cup \{D\}$

CR3 If $C \in S(X)$, $C \sqsubseteq \exists r.D \in \mathcal{T}$, and $D \notin S(X, r)$
then $S(X, r) := S(X, r) \cup \{D\}$

CR4 If $Y \in S(X, r)$, $C \in S(Y)$, $\exists r.C \sqsubseteq D \in \mathcal{T}$, and
$D \notin S(X)$ then $S(X) := S(X) \cup \{D\}$

Fig. 2. $\mathcal{EL}$ completion rules

3.2 Completion Algorithms for Prob-$\mathcal{EL}$

To describe the completion algorithm for Prob-$\mathcal{EL}$, we need the notion of basic concepts. The set $\mathsf{BC}_{\mathcal{T}}$ of Prob-$\mathcal{EL}_c^{01}$ *basic concepts* for a KB $\mathcal{K}$ is the smallest set that contains (1) $\top$, (2) all concept names used in $\mathcal{K}$, and (3) all concepts of the form $P_* A$, where A is a concept name in $\mathcal{K}$. A Prob-$\mathcal{EL}_c^{01}$-TBox $\mathcal{T}$ is in *normal form* if all its axioms are of one of the following forms

$$C \sqsubseteq D, \quad C_1 \sqcap C_2 \sqsubseteq D, \quad C \sqsubseteq \exists r.A, \quad \exists r.A \sqsubseteq D,$$

where $C, C_1, C_2, D \in \mathsf{BC}_{\mathcal{T}}$ and A is a concept name. The normalization rules in Figure 1 can also be used to transform a Prob-$\mathcal{EL}_c^{01}$-TBox into this extended notion of normal form. We further assume that for all assertions $C(a)$ in the ABox $\mathcal{A}$, C is a concept name. We denote as $\mathcal{P}_0^{\mathcal{T}}$, $\mathcal{P}_1^{\mathcal{T}}$ and $\mathcal{R}_0^{\mathcal{T}}$ the set of all concepts of the form $P_{>0}A$, $P_{=1}A$, and $P_{>0}r(a, b)$ respectively, occurring in a normalized knowledge base $\mathcal{K}$.

The completion algorithm for Prob-$\mathcal{EL}_c^{01}$ follows the same idea as the algorithm for $\mathcal{EL}$, but uses several completion sets to deal with the information of what needs to be satisfied in the different worlds of a model. We define the set of worlds $V := \{0, \varepsilon, 1\} \cup \mathcal{P}_0^{\mathcal{T}} \cup \mathcal{R}_0^{\mathcal{T}}$, where the probability distribution μ assigns a probability of 0 to the world 0, and the uniform probability $1/(|V| - 1)$ to all other worlds. For each individual name a, concept name A, role name r and world v, we store the completion sets $S_0(A, v), S_\varepsilon(A, v), S_0(A, r, v)$, $S_\varepsilon(A, r, v)$, $S(a, v)$, and $S(a, r, v)$.

The algorithm initializes the sets as follows for every $A \in \mathsf{BC}_{\mathcal{T}}, r \in \mathsf{RN}(\mathcal{K})$, and $a \in \mathsf{IN}(\mathcal{A})$:

- $S_0(A, 0) = \{\top, A\}$ and $S_0(A, v) = \{\top\}$ for all $v \in V \setminus \{0\}$,
- $S_\varepsilon(A, \varepsilon) = \{\top, A\}$ and $S_\varepsilon(A, v) = \{\top\}$ for all $v \in V \setminus \{\varepsilon\}$,
- $S(a, 0) = \{\top\} \cup \{A \mid A(a) \in \mathcal{A}\}$, $S(a, v) = \{\top\}$ for all $v \neq 0$,
- $S_0(A, r, v) = S_\varepsilon(A, r, v) = \emptyset$ for all $v \in V$, $S(a, r, v) = \emptyset$ for $v \neq 0$,
- $S(a, r, 0) = \{b \in \mathsf{IN}(\mathcal{A}) \mid r(a, b) \in \mathcal{A}\}$.

These sets are then extended by exhaustively applying the rules shown in Figure 3, where X ranges over $\mathsf{BC}_{\mathcal{T}} \cup \mathsf{IN}(\mathcal{A})$, $S_*(X, v)$ stands for $S(X, v)$ if X is an individual and for $S_0(X, v), S_\varepsilon(X, v)$ if $X \in \mathsf{BC}_{\mathcal{T}}$, and $\gamma : V \to \{0, \varepsilon\}$ is defined by $\gamma(0) = 0$, and $\gamma(v) = \varepsilon$ for all $v \in V \setminus \{0\}$.

PR1 If $C' \in S_*(X,v)$, $C' \sqsubseteq D \in \mathcal{T}$, and $D \notin S_*(X,v)$
then $S_*(X,v) := S_*(X,v) \cup \{D\}$

PR2 If $C_1, C_2 \in S_*(X,v)$, $C_1 \sqcap C_2 \sqsubseteq D \in \mathcal{T}$, and $D \notin S_*(X,v)$
then $S_*(X,v) := S_*(X,v) \cup \{D\}$

PR3 If $C' \in S_*(X,v)$, $C' \sqsubseteq \exists r.D \in \mathcal{T}$, and $D \notin S_*(X,r,v)$
then $S_*(X,r,v) := S_*(X,r,v) \cup \{D\}$

PR4 If $D \in S_*(X,r,v)$, $D' \in S_{\gamma(v)}(D,\gamma(v))$, $\exists r.D' \sqsubseteq E \in \mathcal{T}$,
and $E \notin S_*(X,v)$ then $S_*(X,v) := S_*(X,v) \cup \{E\}$

PR5 If $P_{>0}A \in S_*(X,v)$, and $A \notin S_*(X,P_{>0}A)$
then $S_*(X,P_{>0}A) := S_*(X,P_{>0}A) \cup \{A\}$

PR6 If $P_{=1}A \in S_*(X,v)$, $v \neq 0$, and $A \notin S_*(X,v)$
then $S_*(X,v) := S_*(X,v) \cup \{A\}$

PR7 If $A \in S_*(X,v)$, $v \neq 0$, $P_{>0}A \in \mathcal{P}_0^{\mathcal{T}}$, and $P_{>0}A \notin S_*(X,v')$
then $S_*(X,v') := S_*(X,v') \cup \{P_{>0}A\}$

PR8 If $A \in S_*(X,1)$, $P_{=1}A \in \mathcal{P}_1^{\mathcal{T}}$, and $P_{=1}A \notin S_*(X,v)$
then $S_*(X,v) := S_*(X,v) \cup \{P_{=1}A\}$

PR9 If $r(a,b) \in \mathcal{A}, C \in S(b,0), \exists r.C \sqsubseteq D \in \mathcal{T}$,
and $D \notin S(a,0)$ then $S(a,0) := S(a,0) \cup \{D\}$

PR10 If $P_{>0}r(a,b) \in \mathcal{A}$, $C \in S(b, P_{>0}r(a,b)), \exists r.C \sqsubseteq D \in \mathcal{T}$,
and $D \notin S(a, P_{>0}r(a,b))$
then $S(a, P_{>0}r(a,b)) := S(a, P_{>0}r(a,b)) \cup \{D\}$

PR11 If $P_{=1}r(a,b) \in \mathcal{A}$, $C \in S(b,v)$ with $v \neq 0$, $\exists r.C \sqsubseteq D \in \mathcal{T}$
and $D \notin S(a,v)$ then $S(a,v) := S(a,v) \cup \{D\}$

Fig. 3. Prob-$\mathcal{EL}_c^{01}$ completion rules

This algorithm terminates in polynomial time. After termination, the completion sets store all the information necessary to decide subsumption of concept names, as well as checking whether an individual is an instance of a given concept name [6]. For the former decision, it holds that for every pair A, B of concept names: $B \in S_0(A,0)$ iff $A \sqsubseteq_{\mathcal{K}} B$. In the case of instance checking, we have that $\mathcal{K} \models A(a)$ iff $A \in S(a,0)$.

4 Computing the k-MSC using Completion

The msc was first investigated for $\mathcal{EL}$-concept descriptions and w.r.t. unfoldable TBoxes and possibly cyclic ABoxes in [5]. It was shown that the msc does not need to exists for cyclic ABoxes. Consider the ABox $\mathcal{A} = \{r(a,a), C(a)\}$. The msc of a is then

$$C \sqcap \exists r.(C \sqcap \exists r.(C \sqcap \exists r.(C \sqcap \cdots$$

and cannot be expressed by a finite concept description. For cyclic TBoxes it has been shown in [1] that the msc does not need to exists even if the ABox is acyclic.

To avoid infinite nestings in presence of cyclic ABoxes it was proposed in [5] to limit the role-depth of the concept description to be computed. This limitation yields an approximation of the msc, which is still a concept description with the input individual as an instance, but it does not need to be the least one (w.r.t. $\sqsubseteq$) with this property. We follow this idea to compute approximative msc also in presence of general TBoxes.

The *role-depth* of a concept description C (denoted $rd(C)$) is the maximal number of nested quantifiers of C. Now we can define the msc with limited role-depth for $\mathcal{EL}$.

Definition 2 (role-depth bounded $\mathcal{EL}$-msc). *Let $\mathcal{K}$ =($\mathcal{T}$, $\mathcal{A}$) be an $\mathcal{EL}$-KB and a an individual in $\mathcal{A}$ and $k \in \mathbb{N}$. Then the $\mathcal{EL}$-concept description C is the* role-depth bounded $\mathcal{EL}$-most specific concept *of a w.r.t. $\mathcal{K}$ and role-depth k (written k-$msc_{\mathcal{K}}(a)$) iff*

1. $rd(C) \leq k$,
2. $\mathcal{K} \models C(a)$, *and*
3. *for all $\mathcal{EL}$-concept descriptions E with $rd(E) \leq k$ holds: $\mathcal{K} \models E(a)$ implies $C \sqsubseteq_{\mathcal{T}} E$.*

Please note that in case the exact msc has a role-depth less than k the role-depth bounded msc is the exact msc.

4.1 Computing the k-msc in $\mathcal{EL}$ by completion

The computation of the msc relies on a characterization of the instance relation. While in earlier works this was given by homomorphism [5] or simulations [1] between graph representations of the knowledge base and the concept in question, we use the completion algorithm as such a characterization. Furthermore, we construct the msc by traversing the completion sets to "collect" the msc. More precisely, the set of completion sets encodes a graph structure, where the sets $S(X)$ are the nodes and the sets $S(X,r)$ encode the edges. Traversing this graph structure, one can construct an $\mathcal{EL}$-concept. To obtain a finite concept in the presence of cyclic ABoxes or TBoxes one has to limit the role-depth of the concept to be obtained.

Definition 3 (traversal concept). *Let $\mathcal{K}$ be an $\mathcal{EL}$-KB, $\mathcal{K}''$ be its normalized form, $\mathsf{S}_{\mathcal{K}}$ the completion set obtained from $\mathcal{K}$ and $k \in \mathbb{N}$. Then the* traversal concept of a named concept *A (denoted k-$\mathsf{C}_{\mathsf{S}_{\mathcal{K}}}(A)$) with $\mathsf{sig}(A) \subseteq \mathsf{sig}(\mathcal{K}'')$ is the concept obtained from executing the procedure call* traversal-concept-c*(A, $\mathsf{S}_{\mathcal{K}}$, k) shown in Algorithm 1.*

The traversal concept of an individual *a (denoted k-$\mathsf{C}_{\mathsf{S}_{\mathcal{K}}}(a)$) with $a \subseteq \mathsf{sig}(\mathcal{K})$ is the concept description obtained from executing the procedure call* traversal-concept-i*(a, $\mathsf{S}_{\mathcal{K}}$, k) shown in Algorithm 1.*

The idea is that the traversal concept of an individual yields its msc. However, the traversal concept contains names from $\mathsf{sig}(\mathcal{K}'') \setminus \mathsf{sig}(\mathcal{K})$, i.e., concept names that were introduced during normalization – we call this kind of concept names *normalization names* in the following. The returned msc should be formulated w.r.t. the signature of the original KB, thus the normalization names need to be removed or replaced.

Algorithm 1 Computation of a role-depth bounded $\mathcal{EL}$-msc.

Procedure k-msc $(a, \mathcal{K}, k)$
Input: a: individual from $\mathcal{K}$; $\mathcal{K}$ =($\mathcal{T}$, $\mathcal{A}$) an $\mathcal{EL}$-KB; $k \in \mathbb{N}$
Output: role-depth bounded $\mathcal{EL}$-msc of a w.r.t. $\mathcal{K}$ and k.
1: $(\mathcal{T}', \mathcal{A}')$:= simplify-ABox($\mathcal{T}$, $\mathcal{A}$)
2: $\mathcal{K}''$:= (normalize($\mathcal{T}'$), $\mathcal{A}'$)
3: $\mathsf{S}_\mathcal{K}$:= apply-completion-rules($\mathcal{K}$)
4: **return** Remove-normalization-names (traversal-concept-i$(a, \mathsf{S}_\mathcal{K}, k)$)

Procedure traversal-concept-i (a, S, k)
Input: a: individual name from $\mathcal{K}$; S: set of completion sets; $k \in \mathbb{N}$
Output: role-depth traversal concept (w.r.t. $\mathcal{K}$) and k.
1: **if** $k = 0$ **then return** $\bigsqcap_{A \,\in\, S(a)} A$
2: **else return** $\bigsqcap_{A \,\in\, \mathsf{S}(a)} A \,\sqcap$
$$\bigsqcap_{r \in \mathsf{RN}(\mathcal{K}'')} \; \bigsqcap_{A \,\in\, \mathsf{CN}(\mathcal{K}'') \cap S(a,r)} \exists r.\ \text{traversal-concept-c}\ (A, \mathsf{S}, k-1) \ \sqcap$$
$$\bigsqcap_{r \in \mathsf{RN}(\mathcal{K}'')} \; \bigsqcap_{b \,\in\, \mathsf{IN}(\mathcal{K}'') \cap S(a,r)} \exists r.\ \text{traversal-concept-i}\ (b, \mathsf{S}, k-1)$$
3: **end if**

Procedure traversal-concept-c (A, S, k)
Input: A: concept name from $\mathcal{K}''$; S: set of completion sets; $k \in \mathbb{N}$
Output: role-depth bounded traversal concept.
1: **if** $k = 0$ **then return** $\bigsqcap_{B \in S(A)} B$
2: **else return** $\bigsqcap_{B \in S(A)} B \ \sqcap \bigsqcap_{r \in \mathsf{RN}(\mathcal{K}'')} \; \bigsqcap_{B \in S(A,r)} \exists r.\text{traversal-concept-c}\ (B, \mathsf{S}, k-1)$
3: **end if**

Lemma 1. *Let $\mathcal{K}$ be an $\mathcal{EL}$-KB, $\mathcal{K}''$ its normalized version, $\mathsf{S}_\mathcal{K}$ be the set of completion sets obtained for $\mathcal{K}$, $k \in \mathbb{N}$ a natural number and $a \in \mathsf{IN}(\mathcal{K})$. Furthermore let $C =$ k-$\mathsf{C}_{\mathsf{S}_\mathcal{K}}(a)$ and $\widehat{C}$ be obtained from C by removing the normalization names. Then*

$$\mathcal{K}'' \models C(a) \textit{ iff } \mathcal{K} \models \widehat{C}(a).$$

This lemma guarantees that removing the normalization names from the traversal concept preserves the instance relationships. Intuitively, this lemma holds since the construction of the traversal concept conjoins exhaustively all named subsumers and all subsuming existential restrictions to a normalization name up to the role-depth bound. Thus removing the normalization name does not change the extension of the conjunction. The proof can be found in [7]. We are now ready to devise a computation algorithm for the role-depth bounded msc: procedure k-msc as displayed in Algorithm 1.

The procedure k-msc has an individual a from a knowledge base $\mathcal{K}$, the knowledge base $\mathcal{K}$ itself and number k for the role depth-bound as parameter. It first performs the two normalization steps on $\mathcal{K}$, then applies the completion rules from Figure 2 to the normalized KB $\mathcal{K}''$ and stores the set of completion sets in $\mathsf{S}_\mathcal{K}$. Afterwards it computes the traversal-concept of a from $\mathsf{S}_\mathcal{K}$ w.r.t. role-depth bound k. In a post-processing step it applies Remove-normalization-names to the traversal concept.

Obviously, the concept description returned from the procedure k-msc has a role-depth less or equal to k. Thus the first condition of Definition 2 is fulfilled. We prove next that the concept description obtained from k-msc fulfills the second condition from Definition 2.

Lemma 2. *Let $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ be an $\mathcal{EL}$-KB and a an individual in $\mathcal{A}$ and $k \in \mathbb{N}$. If $C = \mathsf{k\text{-}msc}(a, \mathcal{K}, k)$, then $\mathcal{K} \models C(a)$.*

The claim can be shown by induction on k. Each name in C is from a completion set of (1) an individual or (2) a concept, which is connected via existential restrictions to an individual. The full proof can be found in [7].

Lemma 3. *Let $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ be an $\mathcal{EL}$-KB and a an individual in $\mathcal{A}$ and $k \in \mathbb{N}$. If $C = \mathsf{k\text{-}msc}(a, \mathcal{K}, k)$, then for all $\mathcal{EL}$-concept descriptions E with $rd(E) \leq k$ holds: $\mathcal{K} \models E(a)$ implies $C \sqsubseteq_{\mathcal{T}} E$.*

Again, the full proof can be found in [7]. The two lemmas yield the correctness of the overall procedure.

Theorem 1. *Let $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ be an $\mathcal{EL}$-KB and a an individual in $\mathcal{A}$ and $k \in \mathbb{N}$. Then $\mathsf{k\text{-}msc}(a, \mathcal{K}, k) \equiv k\text{-}msc_{\mathcal{K}}(a)$.*

The $k\text{-}msc$ can grow exponential in the size of the knowledge base.

4.2 Most specific concept in Prob-$\mathcal{EL}_c^{01}$

In order to compute the msc, we simply accumulate all concepts to which the individual a belongs, given the information in the completion sets. This process needs to be done recursively in order to account for both, the successors of a explicitly encoded in the ABox, and the nesting of existential restrictions masked by normalization names. In the following we use the abbreviation $S^{>0}(a, r) := \bigcup_{v \in V \setminus \{0\}} S(a, r, v)$. We then define $\mathsf{traversal\text{-}concept\text{-}i}(a, \mathsf{S}, k)$ as

$$\begin{aligned}\bigsqcap_{B \in S(a,0)} B \sqcap \bigsqcap_{r \in \mathsf{RN}(\mathcal{K}'')} \Big(&\bigsqcap_{r(a,b) \in \mathcal{K}''} \exists r.\mathsf{traversal\text{-}concept\text{-}i}(b, \mathsf{S}, k-1) \sqcap \\ &\bigsqcap_{B \in \mathsf{CN}(\mathcal{K}'') \cap S(a,r,0)} \exists r.\mathsf{traversal\text{-}concept\text{-}c}(B, \mathsf{S}, k-1) \sqcap \\ &\bigsqcap_{B \in \mathsf{CN}(\mathcal{K}'') \cap S(a,r,1)} P_{=1}(\exists r.\mathsf{traversal\text{-}concept\text{-}c}(B, \mathsf{S}, k-1)) \sqcap \\ &\bigsqcap_{B \in \mathsf{CN}(\mathcal{K}'') \cap S^{>0}(a,r)} P_{>0}(\exists r.\mathsf{traversal\text{-}concept\text{-}c}(B, \mathsf{S}, k-1))\Big),\end{aligned}$$

where $\mathsf{traversal\text{-}concept\text{-}c}(B, \mathsf{S}, k+1)$ is

$$\begin{aligned}\bigsqcap_{C \in S_0(B,0)} B \sqcap \bigsqcap_{r \in \mathsf{RN}} \Big(&\bigsqcap_{C \in S_0(B,r,0)} \exists r.\mathsf{traversal\text{-}concept\text{-}c}(C, \mathsf{S}, k) \sqcap \\ &\bigsqcap_{C \in S_0(B,r,1)} P_{=1}(\exists r.\mathsf{traversal\text{-}concept\text{-}c}(C, \mathsf{S}, k)) \sqcap \\ &\bigsqcap_{C \in S_0^{>0}(B,r)} P_{>0}(\exists r.\mathsf{traversal\text{-}concept\text{-}c}(C, \mathsf{S}, k))\Big)\end{aligned}$$

and traversal-concept-c$(B, \mathsf{S}, 0) = \bigsqcap_{C \in S_0(B,0)} B$. Once the traversal concept has been computed, it is possible to remove all normalization names preserving the instance relation, which gives us the msc in the original signature of $\mathcal{K}$. The proof can be found in [7].

Theorem 2. *Let $\mathcal{K}$ a Prob-$\mathcal{EL}_c^{01}$-knowledge base, $a \in \mathsf{IN}(\mathcal{A})$, and $k \in \mathbb{N}$; then* Remove-normalization-names(traversal-concept-i$(a, \mathsf{S}, k)) \equiv k\text{-}msc_{\mathcal{K}}(a)$.

5 Conclusions

In this paper we have presented a practical method for computing the role-depth bounded msc of $\mathcal{EL}$ concepts w.r.t. a general TBox. Our approach is based on the completion sets that are computed during realization of a KB. Thus, any of the available implementations of the $\mathcal{EL}$ completion algorithm can be easily extended to an implementation of the (approximative) msc computation algorithm. We also showed that the same idea can be adapted for the computation of the msc in the probabilistic DL Prob-$\mathcal{EL}_c^{01}$.

Together with the completion-based computation of role-depth limited (least) common subsumers given in [8] these results complete the bottom-up approach for general $\mathcal{EL}$- and Prob-$\mathcal{EL}_c^{01}$-KBs. This approach yields a practical method to compute commonalities for differing observations regarding individuals. To the best of our knowledge this has not been investigated for DLs that can express uncertainty.

References

1. F. Baader. Least common subsumers and most specific concepts in a description logic with existential restrictions and terminological cycles. In G. Gottlob and T. Walsh, editors, *Proc. of IJCAI'03*, pages 325–330. Morgan Kaufmann, 2003.
2. F. Baader, S. Brandt, and C. Lutz. Pushing the $\mathcal{EL}$ envelope. In *Proc. of IJCAI'05*, Edinburgh, UK, 2005. Morgan-Kaufmann Publishers.
3. F. Baader, D. Calvanese, D. McGuinness, D. Nardi, and P.F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation, and Applications*. CUP., 2003.
4. F. Baader, R. Küsters, and R. Molitor. Computing least common subsumer in description logics with existential restrictions. In T. Dean, editor, *Proc. of IJCAI'99*, pages 96–101, Stockholm, Sweden, 1999. Morgan Kaufmann.
5. R. Küsters and R. Molitor. Approximating most specific concepts in description logics with existential restrictions. *AI Communications*, 15(1):47–59, 2002.
6. C. Lutz and L. Schröder. Probabilistic description logics for subjective probabilities. In F. Lin and U. Sattler, editors, *Proc. of KR'10*, 2010.
7. R. Peñaloza and A.-Y. Turhan. Completion-based computation of most specific concepts with limited role-depth for $\mathcal{EL}$ and prob-$\mathcal{EL}^{01}$. LTCS-Report LTCS-10-03, Chair f. Autom. Theory, Inst. f. Theor. C. S. TU Dresden, Germany, 2010.
8. R. Peñaloza and A.-Y. Turhan. Role-depth bounded least common subsumers by completion for $\mathcal{EL}$- and Prob-$\mathcal{EL}$-TBoxes. In V. Haarslev, D. Toman, and G. Weddell, editors, *Proc. of the 2010 Description Logic Workshop (DL'10)*, 2010.