A Non-standard Instance Checking for the Description Logic $\mathcal{ELH}$

Suwan Tongphu and Boontawee Suntisrivaraporn
Information, Computer and Communication Technology, Sirindhorn International Institute of Technology, Thammasat University, Bangkok, Thailand

Keywords: Similarity Measure, Description Logic, Degree of Membership, Instance Checking, Non-standard Reasoner.

Abstract: In Description Logics (DLs), an instance checking is regarded as one of the most important reasoning services involving individuals. Though the usability of the reasoner has been seemingly proven in many real-life applications, the classified results are merely a binary response, i.e. whether or not a given individual is an instance of a concept. As being a standard reasoning service, unsatisfying one among all sufficient conditions would basically lead to a negative conclusion. This work introduces a new method to enhance the capability of the instance checking in which the degree of membership could be unveiled though sufficient conditions are not completely satisfied. The proposed algorithm is developed based on the adoption of a homomorphism mapping.

1 INTRODUCTION

Representing knowledge in the form that can be utilized by computer agents, known as *Knowledge Representation*, is one challenge field in artificial intelligence. One common formalism is by using a family of the knowledge representation called *Description Logics* (DLs) (Baader et al., 2007). In DLs, the knowledge is structurally represented by means of concepts and their relationship. On the one hand, rich ontologies can be constituted using expressive DLs, i.e. an employment of the Web Ontology Language (OWL), which recently becomes a standard semantic web language recommended by W3C consortium. On the other hand, some other ontologies can alternatively be fformulated using lightweight ones which are sufficiently expressive for the domains and offer classification tractability, e.g. the use of extensions of the tractable DL $\mathcal{EL}$ in the renowned medical ontology (Schulz et al., 2009).

Among a variety of knowledge representation formalisms, main reasons making DLs distinct from others is their underlying reasoning services which makes implicit knowledge explicit. Apart from the most prominent *subsumption checking* service, which allows finding of subclass-superclass relationship, *instance checking* is one another readily available service, which checks whether a given individual is an instance of a certain concept. Serving as a standard service, a classical instance checking gives a positive response only when sufficient conditions are satisfied; that is, the missing of one of the required concept and/or role assertions consequently turns reasoning outputs negative without providing any beneficial clues no matter how rich the assertions are. This lack leads to an introduction of a non-standard instance checking service whose response is the degree of membership. In fact, the computation method is based on a structural homomorphism and is particularly the extension of our recent work (Suntisrivaraporn, 2013) on measuring similarity $\mathcal{EL}$ concepts. Hence, the idea is extended to ABox and thus the instance checking problem.

Given an ABox that fulfills all sufficient conditions of a concept description, the proposed algorithm basically produces the same result as that obtained from classical reasoners (i.e. both return 1 as the result). This reflects that, in common cases where sufficient conditions are fulfilled, both standard and the proposed non-standard algorithm behave in a similar manner. However, in a situation where not all concept conditions are satisfied. From a classical reasoning point of view, as previously mentioned, such the ABox is normally classified as irrelevant. In contrast to the classical reasoners, the proposed algorithm checks further to an existence of some commonality and subsequently computes a corresponding degree of membership which ranges between 0 and 1.

To be more illustrative, consider an application of visual object detection proposed in (Tongphu et al., 2012). In this work, the object of interest (i.e. car objects) is described by means of its composition (i.e.

Tongphu S. and Suntisrivaraporn B..
A Non-standard Instance Checking for the Description Logic ELH.
DOI: 10.5220/0005074900670074
In *Proceedings of the International Conference on Knowledge Engineering and Ontology Development* (KEOD-2014), pages 67-74
ISBN: 978-989-758-049-9

Table 1: Syntax and semantics of the Description Logic $\mathcal{ELH}$.

Name	Syntax	Semantics
top	$\top$	$\Delta^{\mathcal{I}}$
concept name	A	$A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$
conjunction	$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
existential restriction	$\exists r.C$	$\{x \in \Delta^{\mathcal{I}} \mid \exists y \in \Delta^{\mathcal{I}} : (x,y) \in r^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\}$
primitive concept def	$A \sqsubseteq D$	$A^{\mathcal{I}} \subseteq D^{\mathcal{I}}$
full concept def	$A \equiv C$	$A^{\mathcal{I}} = C^{\mathcal{I}}$
concept assertion	$C(x)$	$x^{\mathcal{I}} \in C^{\mathcal{I}}$
role assertion	$r(x,y)$	$(x^{\mathcal{I}}, y^{\mathcal{I}}) \in r^{\mathcal{I}}$

car parts) using an OWL ontology. During a testing stage, visual features extracted from an image are converted to ontological assertions. By using a classical instance checking service, the classification results would basically turn nagative when required car parts are not entirely detected. The proposed reasoning service, on the other hand, returns the degree of membership based on a shared commonality. This allows a certain cut-off threshold to be set up. The individual whose degree of membership is greater than the threshold can then be classified as a car instance.

The rest of this paper is organized in order. The background on the DL $\mathcal{ELH}$, the unfoldable TBoxes, and the $\mathcal{ELH}$ description tree are described in the next section. Section 3 and 4 introduce the notions of the $\mathcal{ELH}$ description graph of the assertion terminology and membership homomorphism for the instance checking problem, respectively. Section 5 and 6 describe related works and give conclusions of this work.

2 BACKGROUND

In the knowledge representation using the family of DLs, the *concept descriptions* of the $\mathcal{ELH}$ regarded as the lightweight DL can be built from a set of *primitive concept names* CN, a set of *role names* RN, and a set of constructors shown in the first part of Table 1. A finite set of *terminological axioms* of the form shown in the second part of Table 1 is called an $\mathcal{ELH}$ *terminology* or *TBox*. The TBox is said to be *unfoldable* if it contains at most one definition for each concept name, and it is *acyclic* (i.e. there is no direct or indirect definition refers to the concept itself). For the rest of this paper, we denote by $\mathcal{T}$ an unfoldable TBox.

Let A,B be concept names, and C,D be arbitrary concept descriptions. For the sake of simplicity, we denote by $\mathsf{CN}^{\mathsf{def}}$ and $\mathsf{CN}^{\mathsf{pri}}$, the set of *defined concepts* (i.e. the concept that appears on the left-hand side of a concept description) and the set of *primitive concepts* (i.e. the concept that only appears on the right-hand side of a concept description).

$$\mathsf{CN} = \mathsf{CN}^{\mathsf{pri}} \cup \mathsf{CN}^{\mathsf{def}}$$

Let x and y be individuals, and Ind be a set of *individual names*. An $\mathcal{ELH}$ *ABox* $\mathcal{A}$ is a finite set of *assertions* shown in the third part of Table 1. A knowledge base $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ comprises of $\mathcal{T}$ and $\mathcal{A}$.

Like all other DLs, the semantics of $\mathcal{ELH}$ is defined by means of interpretations. An interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ consists of *interpretation domain* $\Delta^{\mathcal{I}}$ and *interpretation function* $\cdot^{\mathcal{I}}$. The interpretation function maps every concept name $A \in \mathsf{CN}$ to a subset $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$, every role name $r \in \mathsf{RN}$ to a binary relation $r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$, and every individual $x \in \mathsf{Ind}$ to an element $x^{\mathcal{I}} \in \Delta^{\mathcal{I}}$. The last column of Table 1 depicts the semantics for $\mathcal{ELH}$ constructors, terminological axioms, and assertions, respectively. An interpretation $\mathcal{I}$ is called a model of the knowledge base $\mathcal{K}$ if it satisfies every axiom in $\mathcal{T}$ and every assertion in $\mathcal{A}$, i.e. conditions in the semantics column of Table 1 are fulfilled. Figure 1 depicts a knowledge base about family constructed by using DL $\mathcal{ELH}$.

$$\begin{aligned}
\mathsf{Woman} &\equiv \mathsf{Female} \sqcap \mathsf{Person}\\
\mathsf{Mother} &\equiv \mathsf{Woman} \sqcap \exists \mathsf{child}.\mathsf{Person}\\
\mathsf{GrandMother} &\equiv \mathsf{Woman} \sqcap \exists \mathsf{child}.(\mathsf{Person} \sqcap \exists \mathsf{child}.\mathsf{Person})\\
\mathsf{Syster} &\equiv \mathsf{Woman} \sqcap \exists \mathsf{sibling}.\mathsf{Person}\\
\mathsf{Aunt} &\equiv \mathsf{Woman} \sqcap \exists \mathsf{sibling}.(\mathsf{Person} \sqcap \exists \mathsf{child}.\mathsf{Person})\\
\mathsf{Man} &\equiv \mathsf{Male} \sqcap \mathsf{Person}\\
\mathsf{Father} &\equiv \mathsf{Man} \sqcap \exists \mathsf{child}.\mathsf{Person}\\
\mathsf{GrandFather} &\equiv \mathsf{Man} \sqcap \exists \mathsf{child}.(\mathsf{Person} \sqcap \exists \mathsf{child}.\mathsf{Person})\\
\mathsf{Brother} &\equiv \mathsf{Man} \sqcap \exists \mathsf{sibling}.\mathsf{Person}\\
\mathsf{Uncle} &\equiv \mathsf{Man} \sqcap \exists \mathsf{sibling}.(\mathsf{Person} \sqcap \exists \mathsf{child}.\mathsf{Person})
\end{aligned}$$

$$\mathsf{Father}(a), \mathsf{GrandMother}(b), \mathsf{sibling}(a,b), \mathsf{sibling}(b,a)$$

Figure 1: Knowledge base of family ($\mathcal{K}_{\mathsf{family}}$). The terminological box ($\mathcal{T}_{\mathsf{family}}$) and the examples of assertional box ($\mathcal{A}_{\mathsf{family}}$) are shown in the upper part and lower part, respectively.

We assume without loss of generality that an $\mathcal{ELH}$ concept C can be represented using the following form ((Suntisrivaraporn, 2013)):

Algorithm 1: $\mathcal{ELH}$ description tree construction.

function build-tree$(\mathcal{P}_C, \mathcal{E}_C)$

Create a new tree $\mathfrak{T}$
Create a new vertex $v \in V$
$\ell(v) \leftarrow \mathcal{P}_C$
for each $\exists r.C' \in \mathcal{E}_C$ **do**
  build-child-node$(v, r, \mathcal{P}_{C'}, \mathcal{E}_{C'})$
return $\mathfrak{T}$

function build-child-node$(v, r, \mathcal{P}_C, \mathcal{E}_C)$

Create a new vertex $w \in V$
$\ell(w) \leftarrow \mathcal{P}_C$
Add a new edge $(v, \mathcal{R}_r, w)$ to E
for each $\exists s.C' \in \mathcal{E}_C$ **do**
  build-child-node$(w, s, \mathcal{P}_{C'}, \mathcal{E}_{C'})$

$$P_1 \sqcap \cdots \sqcap P_m \sqcap \exists r_1.C_1 \sqcap \cdots \sqcap \exists r_n.C_n$$

where $P_i \in \mathsf{CN}^{\mathsf{pri}}$, $r_j \in \mathsf{RN}$, and C_j are concept descriptions, for $1 \leq i \leq m$ and $1 \leq j \leq n$. To be more understandable, consider the concept Aunt defined in $\mathcal{T}_{\mathsf{family}}$, the following shows its equivalent expanded form.

$$\mathsf{Female} \sqcap \mathsf{Person} \sqcap \exists \mathsf{sibling}.(\mathsf{Person} \sqcap \exists \mathsf{child}.\mathsf{Person})$$

For convenience, we denote by $\mathcal{P}_C$ the set of top-level primitive concepts $\{P_1, \ldots, P_m\}$ and $\mathcal{E}_C$ the set of top-level existential restrictions $\{\exists r_1.C_1, \ldots, \exists r_n.C_n\}$. To handle a role hierarchy, we denote by $\mathcal{R}_r = \{s | r \sqsubseteq^* s\}$ where $*$ is a transitive closure, the set of role expansion w.r.t. r.

We define the $\mathcal{ELH}$ description tree of C w.r.t. the unfoldable TBox by $\mathfrak{T}_C = (V, E, rt, \ell)$ where V is a set of nodes, $E \subseteq V \times 2^{\mathsf{RN}^{\mathsf{pri}}} \times V$ is a set of labeled edges, rt is a root, and $\ell : V \to 2^{\mathsf{CN}^{\mathsf{pri}}}$ is a node labeling function. Algorithmically, $\mathfrak{T}_C$ can be constructed using Algorithm 1. Figure 2 (left and right) shows an example of the $\mathcal{ELH}$ description tree for the concept Aunt, written $\mathfrak{T}_{\mathsf{Aunt}}$.

3 REASONING ABOUT INDIVIDUALS

Given a knowledge base $\mathcal{K} = (\mathcal{T}, \mathcal{A})$, an individual x and a concept C, the *instance checking* problem consists on deciding whether the concept assertion $C(x)$ is satisfied in every model of $\mathcal{K}$, in symbols $\mathcal{K} \models C(x)$, i.e. $x^{\mathcal{I}} \in C^{\mathcal{I}}$ for every model $\mathcal{I}$ of $\mathcal{K}$.

Let $\mathsf{Ind}(\mathcal{A})$ denote the set of individuals in $\mathcal{A}$. In order to enable an investigation for a membership, a representation of $\mathcal{A}$ is transformed into an $\mathcal{ELH}$ description graph $\mathcal{G}(\mathcal{A}) = (V, E, \ell)$ where V denotes a set of nodes, $E \subseteq V \times 2^{\mathsf{RN}^{\mathsf{pri}}} \times V$ is a set of labeled edges, and $\ell : V \to 2^{\mathsf{CN}^{\mathsf{pri}}}$ is a node labeling function. Algorithm 2 shows a process of the $\mathcal{ELH}$ description graph $\mathcal{G}(\mathcal{A})$ construction. Intuitively, for each individual x defined in $\mathcal{A}$, a corresponding node v_x is introduced and added to the graph $\mathcal{G}(\mathcal{A})$. For each $C(x) \in \mathcal{A}$, v_x is augmented by all successors of the root of $\mathfrak{T}_C$. The outgoing edge that links v_x to v_y where $r(x,y) \in \mathcal{A}$ is then added.

Algorithm 2: $\mathcal{ELH}$ description graph construction.

function build-graph$(\mathcal{A})$

Create a new graph $\mathcal{G} = (V, E, \ell)$
for each $x \in \mathsf{Ind}(\mathcal{A})$ **do**
  Add v_x to V
for each $C(x) \in \mathcal{A}$ **do**
  $\ell(v_x) \leftarrow \mathcal{P}_C$
  $V \leftarrow V \cup (V_{\mathfrak{T}_C} \setminus \{rt\})$
  for each $(v, \mathcal{R}_r, u) \in E_{\mathfrak{T}_C}$ **do**
    if $v \neq rt$ **then**
      Add $(v, \mathcal{R}_r, u)$ to E
    else
      Add $(v_x, \mathcal{R}_r, u)$ to E
for each $r(x,y) \in \mathcal{A}$ **do**
  Add $(v_x, \mathcal{R}_r, v_y)$ to E
return $\mathcal{G}$

Definition 1 (Homomorphism)**.** Let $\mathfrak{T}$ and $\mathfrak{T}'$ be two $\mathcal{ELH}$ description trees as defined above. There exists a homomorphism h from $\mathfrak{T}$ to $\mathfrak{T}'$ written $h : \mathfrak{T} \to \mathfrak{T}'$ iff the following conditions are satisfied:

1. $\ell(v) \subseteq \ell'(h(v))$.
2. For each successor w of v in $\mathfrak{T}$, $h(w)$ is a successor of $h(v)$ with $(v, \mathcal{R}_r, w) \in E$, $(h(v), \mathcal{R}_s, h(w)) \in E'$, and $\mathcal{R}_r \subseteq \mathcal{R}_s$.

Consider $\mathcal{A}_{\mathsf{family}}$ w.r.t. $\mathcal{K}_{\mathsf{family}}$, the corresponding $\mathcal{ELH}$ description graph for $\mathcal{A}_{\mathsf{family}}$ can be constructed using Algorithm 2. Figure 2 illustrates an existence of a homomorphism that maps the root of $\mathfrak{T}_{\mathsf{Aunt}}$ to b in $\mathcal{G}(\mathcal{A}_{\mathsf{family}})$ and a failed attempt to find a homomorphism that maps the root of $\mathfrak{T}_{\mathsf{Aunt}}$ to a in $\mathcal{G}(\mathcal{A}_{\mathsf{family}})$. Though the failed mapping does not satisfy the homomorphism conditions, there still exists some commonality shared between the corresponding nodes and edges (e.g. both are person and have sibling); that is, though not being considered as the instance of the concept, inductively it exhibits some degree of membership.

Proposition 2 shows the characterization of an instance checking problem by means of an existence of

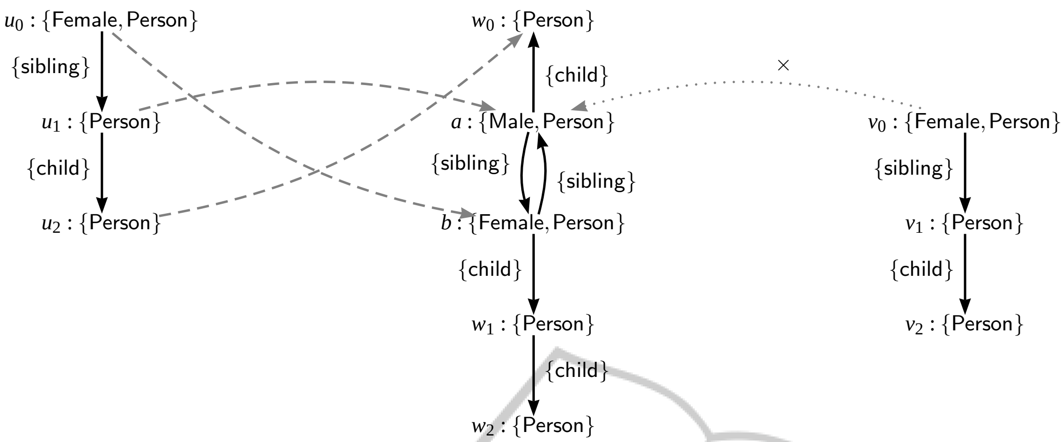

Figure 2: A homomorphism that maps the root of $\mathfrak{T}_{\mathsf{Aunt}}$ (left) to b in $\mathcal{G}(\mathcal{A}_{\mathsf{family}})$ (middle). A failed attempt to map the root of $\mathfrak{T}_{\mathsf{Aunt}}$ (right) to a in $\mathcal{G}(\mathcal{A}_{\mathsf{family}})$.

a homomorphism. It shall turn out that this proposition can be a generalization for the DL $\mathcal{ELH}$.

Proposition 2. *(Baader, 2003) Let $\mathcal{T}$ be an $\mathcal{EL}$-TBox, $\mathcal{A}$ an $\mathcal{EL}$-ABox, C a concept in $\mathcal{T}$ and x an individual occurring in $\mathcal{A}$. Then, the following are equivalent:*

1. $(\mathcal{T},\mathcal{A}) \models C(x)$
2. *There is a homomorphism from the root of $\mathfrak{T}_C$ to x in $\mathcal{G}(\mathcal{A})$*

4 MEMBERSHIP HOMOMORPHISM DEGREE

In addition to the method for computing the membership homomorphism degree originally introduced in (Suntisrivaraporn, 2013), this work follows the idia with an extension to handle role hierarchy axioms.

Let C be $\mathcal{ELH}$ unfolded concept descriptions, $\mathcal{P}_C$, $\mathcal{E}_C$ be as defined in the previous section, $\mathfrak{T}_C$ be the corresponding $\mathcal{ELH}$ description tree, $\mathcal{R}_r$ and $\mathcal{R}_s$ be sets of roles w.r.t. the role expansions of r and s, respectively. For convenience, let $\mathsf{edge}(v)$ represents the set of edges from the vertex v, i.e. $\mathsf{edge}(v) = \{(\mathcal{R}_r, w) \mid (v, \mathcal{R}_r, w) \in E\}$. Then, the degree of having a membership homomorphism from $rt \in \mathfrak{T}_C$ to $v \in V_{\mathcal{G}(\mathcal{A})}$ is defined as follows:

Definition 3 (Membership Homomorphism Degree)**.** Let $\mathbf{T}^{\mathcal{T}}$ be the set of all $\mathcal{ELH}$ description trees from TBox $\mathcal{T}$ and $V_{\mathcal{G}(\mathcal{A})}$ be the set of all vertices in the description graph from ABox $\mathcal{A}$. The *membership homomorphism degree function* $\mathsf{mh} : \mathbf{T}^{\mathcal{T}} \times V_{\mathcal{G}(\mathcal{A})} \to [0,1]$ is inductively defined as follows:

$$\mathsf{mh}(\mathfrak{T}_C, v \in V_{\mathcal{G}}) := \mu \cdot \mathsf{p\text{-}mh}(\mathcal{P}_C, \ell(v)) + (1-\mu) \cdot \mathsf{e\text{-}set\text{-}mh}(\mathcal{E}_C, \mathsf{edge}(v)), \quad (1)$$

where $0 \leq \mu \leq 1$;

$$\mathsf{p\text{-}mh}(\mathcal{P}_C, \ell(v)) := \begin{cases} 1 & \text{if } \mathcal{P}_C = \emptyset \\ \frac{|\mathcal{P}_C \cap \ell(v)|}{|\mathcal{P}_C|} & \text{otherwise,} \end{cases} \quad (2)$$

where $|\cdot|$ represents a set cardinality;

$$\mathsf{e\text{-}set\text{-}mh}(\mathcal{E}_C, E) := \sum_{\varepsilon \in \mathcal{E}_C} \frac{max\{\mathsf{e\text{-}mh}(\varepsilon, e) : e \in E\}}{|\mathcal{E}_C|}, \quad (3)$$

where ε is an existential restriction, e is an edge, and $E \subseteq E_{\mathcal{G}(\mathcal{A})}$ is a set of outgoing edges; and

$$\mathsf{e\text{-}mh}(\exists r.X, (\mathcal{R}_s, w)) := \gamma(\nu + (1-\nu) \cdot \mathsf{mh}(\mathfrak{T}_X, w)) \quad (4)$$

where $\gamma = \frac{|\mathcal{R}_r \cap \mathcal{R}_s|}{|\mathcal{R}_r|}$ and $0 \leq \nu < 1$.

The meaning of the parameters μ and ν are similar to those defined in (Suntisrivaraporn, 2013) and set to $\frac{|\mathcal{P}_C|}{|\mathcal{P}_C \cup \mathcal{E}_C|}$ and 0.4, respectively. The value of γ in Formula 4 is a proportion of a set of common roles against all those respect to r. For the case where $\gamma = 0$, this means there is no commonality between two given roles r and s, i.e. further computations for the degrees of membership among their successors should be omitted. If $0 < \gamma \leq 1$, this reveals that there exists some commonality. However, in the case where $\gamma = 1$, both r and s are totally similar and thus considered logically equivalent.

Algorithm 3: $\mathcal{ELH}$ similarity measure.

function $\mathsf{mh}(\mathfrak{T}_C, v \in V_{\mathcal{G}(\mathcal{A})})$

$i \leftarrow \mu \cdot \mathsf{p\text{-}mh}(\mathcal{P}_C, \ell(v)) + (1-\mu)\mathsf{e\text{-}set\text{-}mh}(\mathcal{E}_C, \mathsf{edge}(v))$
return i

function $\mathsf{p\text{-}mh}(\mathcal{P}_C, \ell(v))$

if $\mathcal{P}_C \leftarrow \emptyset$ **then**
  return 1
else
  return $\frac{|\mathcal{P}_C \cap \ell(v)|}{|\mathcal{P}_C|}$

function $\mathsf{e\text{-}set\text{-}mh}(\mathcal{E}_C, E \subseteq E_{\mathcal{G}(\mathcal{A})})$

$sum \leftarrow 0$
for each $\varepsilon \in \mathcal{E}_C$ **do**
  $max \leftarrow 0$
  for each $e \in E$ **do**
    if $\mathsf{e\text{-}mh}(\varepsilon, e) > max$ **then**
      $max \leftarrow \mathsf{e\text{-}mh}(\varepsilon, e)$
  $sum \leftarrow sum + \frac{max}{|\mathcal{E}_C|}$
return sum

function $\mathsf{e\text{-}mh}(\exists r.X, (\mathcal{R}_s, w))$

if $\gamma = 0$ **then**
  return 0
else
  return $\gamma(\nu + (1-\nu) \cdot \mathsf{mh}(\mathfrak{T}_X, w))$

Proposition 4. *Providing a description graph $\mathcal{G}(\mathcal{A})$ w.r.t. $\mathcal{A}$ and an $\mathcal{ELH}$ description tree $\mathfrak{T}_C$ of a concept C in an unfoldable $\mathcal{ELH}$ TBox $\mathcal{T}$, the followings are equivalent:*

1. $(\mathcal{T}, \mathcal{A}) \models C(x)$
2. $\mathsf{mh}(\mathfrak{T}_C, v_x \in \mathcal{G}(\mathcal{A})) = 1$

Proof. ($1 \Longrightarrow 2$) By Proposition 2, $C(x) \in \mathcal{A}$ then there exists a homomorphism mapping $rt \in \mathfrak{T}_C$ to $v_x \in V_{\mathcal{G}(\mathcal{A})}$. For the induction base case where the depth of $\mathfrak{T}_C$ is zero (i.e. $\mathfrak{T}_C$ contains only one node), by Definition 3, this inductively implies that $\ell(rt) \subseteq \ell(v_x)$, such that $\mu = 1$, and $\mathsf{p\text{-}mh}(\mathcal{P}_C, \ell(v_x)) = 1$ and as a consequence $\mathsf{mh}(\mathfrak{T}_C, v_x \in V_{\mathcal{G}(\mathcal{A})}) = 1$. For the induction step where the depth of $\mathfrak{T}_C$ is nonzero, for every $v \in V_{\mathfrak{T}_C}$ there exists $h(v) \in V_{\mathcal{G}(\mathcal{A})}$ such that $\ell(v) \subseteq \ell(h(v))$ (i.e. $\mathsf{p\text{-}mh}(\cdot,\cdot) = 1$) and for every $(v, \mathcal{R}_r, w) \in E_{\mathfrak{T}_C}$ there exists $(h(v), \mathcal{R}_s, h(w)) \in E_{\mathcal{G}(\mathcal{A})}$ where w and $h(w)$ are successors of v and $h(v)$, respectively, such that $\mathcal{R}_r \subseteq \mathcal{R}_s$ (i.e. $\mathsf{e\text{-}set\text{-}mh}(\cdot,\cdot) = 1$). Hence, $\mathsf{mh}(\mathfrak{T}_C, v_x \in V_{\mathcal{G}(\mathcal{A})}) = 1$.

($2 \Longrightarrow 1$) By Definition 3, $\mathsf{mh}(\mathfrak{T}_C, v_x \in V_{\mathcal{G}(\mathcal{A})}) = 1$ means $\mathsf{mh}(\cdot,\cdot) = 1$ and $\mathsf{e\text{-}set\text{-}mh}(\cdot,\cdot) = 1$) (in case that the tree has child nodes) such that two conditions of a homomorphism defined in Definition 1 are satisfied and by Proposition 2, $C(x) \in \mathcal{A}$.

The membership homomorphism function unveils a numerical value measuring the degree membership of an individual in an ABox description graph against a compared concept description tree. Intuitively, this infers the degree of membership which suggests how close an individual is an instance of a concept. Therefore, we define the degree of membership of the individual x to the concept C as the numerical value obtained from $\mathsf{mh}(\mathfrak{T}_C, v_x \in V_{\mathcal{G}(\mathcal{A})})$.

Example 5. To illustrate how the algorithm works, consider the expanded description tree $\mathfrak{T}_{\mathsf{Aunt}}$ and the expanded description graph $\mathcal{G}(\mathcal{A}_{\mathsf{family}})$ shown in Figure 2, using μ and ν as previously described, the degrees of membership of the individual a to the concept Aunt can be computed using Algorithm 3. The following shows computation steps:

$$\begin{aligned}
&\mathsf{mh}(\mathfrak{T}_{\mathsf{Aunt}}, v_a \in V_{\mathcal{G}}) \\
&:= \tfrac{2}{3}\mathsf{p\text{-}mh}(\mathcal{P}_{\mathsf{Aunt}}, \ell(v_a)) \\
&\quad + \tfrac{1}{3}\mathsf{e\text{-}set\text{-}mh}(\mathcal{E}_{\mathsf{Aunt}}, \mathsf{edge}(v_a)) \\
&:= \tfrac{2}{3}(\tfrac{1}{2}) + \tfrac{1}{3}(\gamma(\nu + (1-\nu) \\
&\quad \cdot \mathsf{mh}(\mathfrak{T}_{\mathsf{Person} \sqcap \exists \mathsf{ch.Person}}, v_b))) \\
&:= \tfrac{2}{3}(\tfrac{1}{2}) + \tfrac{1}{3}(1(\tfrac{2}{5} + \tfrac{3}{5}\mathsf{mh}(\mathfrak{T}_{\mathsf{Person} \sqcap \exists \mathsf{ch.Person}}, v_b)))
\end{aligned}$$

// where $\mathsf{mh}(\mathfrak{T}_{\mathsf{Person} \sqcap \exists \mathsf{ch.Person}}, v_b)$ yields 1;
// see belows

$$\begin{aligned}
&:= \tfrac{2}{3}(\tfrac{1}{2}) + \tfrac{1}{3}(1(\tfrac{2}{5} + \tfrac{3}{5}(1))) \\
&:= \tfrac{2}{3} := 0.667
\end{aligned}$$

The computation for the sub-description corresponding to v_1 and b in Figure 2 is as follows:

$$\begin{aligned}
&\mathsf{mh}(\mathfrak{T}_{\mathsf{Person} \sqcap \exists \mathsf{ch.Person}}, v_b \in V_{\mathcal{G}}) \\
&:= \tfrac{2}{3}\mathsf{p\text{-}mh}(\mathcal{P}_{\mathsf{Person} \sqcap \exists \mathsf{ch.Person}}, \ell(v_b)) \\
&\quad + \tfrac{1}{3}\mathsf{e\text{-}set\text{-}mh}(\mathcal{E}_{\mathsf{Person} \sqcap \exists \mathsf{ch.Person}}, \mathsf{edge}(v_b)) \\
&:= \tfrac{1}{2}(\tfrac{1}{2}) + \tfrac{1}{2}\mathsf{e\text{-}set\text{-}mh}(\mathcal{E}_{\mathsf{Person} \sqcap \exists \mathsf{ch.Person}}, \mathsf{edge}(v_b))
\end{aligned}$$

// where $\mathsf{e\text{-}set\text{-}mh}(\mathcal{E}_{\mathsf{Person} \sqcap \exists \mathsf{ch.Person}}, \mathsf{edge}(v_b))$
// yields 1; see belows

$$:= \tfrac{1}{2}(1) + \tfrac{1}{2}(1) := 1$$

The computation for the sub-description corresponding with $\varepsilon = \exists\mathsf{child.Person}$ and $e = (\{\mathsf{child}\}, v_{w_1})$ is as follows:

$$\begin{aligned}
&\mathsf{e\text{-}mh}(\varepsilon, e) \\
&\quad := \gamma(\nu + (1-\nu)\mathsf{mh}(\mathfrak{T}_{\mathsf{Person}}, v_{w_1} \in V_{\mathcal{G}})) \\
&\quad := 1(\tfrac{2}{5} + \tfrac{3}{5}\mathsf{mh}(\mathfrak{T}_{\mathsf{Person}}, v_{w_1} \in V_{\mathcal{G}})) \\
&\quad := \tfrac{2}{5} + \tfrac{3}{5}(1) := 1
\end{aligned}$$

The computation for the alternative sub-description corresponding with $\varepsilon = \exists\mathsf{child.Person}$ and $e = (\{\mathsf{sibling}\}, v_a)$ is, however, equal to 0 since $\gamma = 0$. That is $\max(\mathsf{e\text{-}mh}(\varepsilon, e)) = 1$.

Table 2: The degrees of membership of all concepts in $\mathcal{T}_{\mathsf{family}}$ to the individual *a* and *b*.

Concept Names	Degrees of membership	
	a	*b*
Woman	0.5	1.0
Mother	0.667	1.0
GrandMother	0.567	1.0
Sister	0.667	1.0
Aunt	0.667	1.0
Man	1.0	0.5
Father	1.0	0.667
GrandFather	0.9	0.667
Brother	1.0	0.667
Uncle	1.0	0.667

By applying the same computation approach to the rest of all defined concepts w.r.t. $\mathcal{T}_{\mathsf{family}}$, Table 2 shows the degrees of membership of the individual *a* and *b* to all defined concepts.

Providing that *a* is an instance of Father, and *a* is a sibling of *b*, the degrees of membership, obtained through the proposed algorithm together with Proposition 4, reveal that *a* is also an instance of Man, Brother, and Uncle. Likewise, *b* is not only an instance of GrandMother but also an instance of Woman, Mother, Sister, and Aunt.

Apart from a crisp response, the proposed service is yet capable of inductively unveiling the degrees of membership though the two stated conditions of being homomorphism are not completely satisfied. For example, consider the degrees of membership of *a* to the concept GrandFather and GrandMother. Though *a* is not considered as an instance of either concepts in view of classical instance checking, intuitively, it is reasonable to argue that *a* is more similar to GrandFather than GrandMother (see e.g. the degrees of membership of 0.9 and 0.567, respectively).

5 RELATED WORKS

In DLs, prominent reasoning services conern concepts are concept subsumption and concept satisfiability; whereas those concerning individuals are instance checking and instance retrieval. Oftentimes, instance checking algorithms have obtained from adaptation of existing algorithms for concept subsumption and satisfiability (Baader et al., 2003; Baader and Sattler, 2001). In a sense, studying on concept similarity measures is a natural approach to solving instance checking problem.

Measuring degrees of membership as well as similarity in DLs have been intensively studied in the past few decades with a number of great attempts. The computation methods can be, however, broadly categorized into two major approaches. One is simply focused on a structural distance (Batet et al., 2010; Schickel-Zuber and Faltings, 2007; Blanchard et al., 2005; Passant, 2010) which normally ignores the semantics. The other try to semantically analyze the relationship among concepts and to inductively compute the degree of membership based on the defined description itself. Our approach is in the second category. The following describes major related papers.

Stuckenschmidt adopts the algorithm (Stuckenschmidt, 2009) originally introduced by Champin et al. (Champin and Solnon, 2003). The algorithm measures a similarity between concept instances by analyzing the degrees of commonality between the graphs of two concept instances. The proposed algorithm however ignores a deliberation of a subsumption relation. Hence, instances of different concepts are always identified as dissimilar.

Amato et al. (d'Amato et al., 2006) propose a method measuring a semantic similarity between concepts and instances. In this work, the degrees of membership are based on a counting of concept membership (i.e. a counting for instances of concepts). The estimation is then inductively computed using the *k*-Nearest Neighbor (*k*-NN) method. One disadvantage of this method is an undecidable concept membership could be possibly found, i.e. the individual cannot be determined whether it is an instance of a certain concept.

Bianchini et al. (Bianchini et al., 2005) propose a hybrid method that combines a deductive matching method with constraints (Li and Horrocks, 2003) and a semantic-based matching method. The degree of similarity is numerical measured with a big range of similarity coefficient which turns the analysis of a similarity measure among a number of concepts becomes a difficult task.

A probabilistic variant of description logic has been introduced in (Fagin et al., 1990) and partially implemented in the Pronto system (Klinov, 2008). Instead of merely stating crisp axioms and assertions, the probabilistic inference engine Pronto allows an existence of a probabilistic TBox and ABox. One obvious drawback of Pronto is that all sufficient conditions of a concept description must be satisfied. Unlike Pronto, our algorithm requires neither a fulfillment of concept conditions nor a probabilistic assertion.

The work on a similarity measure for the DL $\mathcal{ELH}$ proposed by Lehman and Turhan (Lehmann and Turhan, 2012) introduces a new similarity framework that allows tuning of various parameters. Our approach is similar to this work.

6 CONCLUSION AND FUTURE WORKS

This work presents an attempt to measure the degree of membership of an individual to a compared concept. The capability of the proposed reasoning services is devised to handle the case where necessary conditions are not completely satisfied, but there exists some commonality. The instance checking problem is, thus, rather resolved by means of the numerical degree of membership. The usability of the proposed algorithm is demonstrated through the well-known terminology of family. The examples simply depict a common case of the individuals that could possibly be found in such the assertional terminology.

As being speculated as common steps for future works, it would be beneficial to extend the proposed method to be supported on more expressive DLs as well as to increase a capability of handling the concepts w.r.t. general TBoxes (i.e. cyclic TBoxes).

It is to be mentioned that with a pre-processing of a concept description expansion, the complexity of the algorithm is polynomially bounded and directly variant to the depth of a concept description tree and an ABox description graph. However, the expansion process itself can be dramatically grown in an exponential time. Fortunately, this can be handled through a representation of an entire TBox as a forest of inter-dependent $\mathcal{ELH}$ description trees.

REFERENCES

Baader, F. (2003). The instance problem and the most specific concept in the description logic $\mathcal{EL}$ w.r.t. terminological cycles with descriptive semantics. In Gnter, A., Kruse, R., and Neumann, B., editors, *KI*, volume 2821 of *Lecture Notes in Computer Science*, pages 64–78. Springer.

Baader, F., Calvanese, D., McGuinness, D., Nardi, D., and Patel-Schneider, P., editors (2007). *The Description Logic Handbook: Theory, Implementation and Applications*. Cambridge University Press, second edition.

Baader, F., Hladik, J., Lutz, C., and Wolter, F. (2003). From tableaux to automata for description logics. *Fundamenta Informaticae*, 57(2-4):247–279.

Baader, F. and Sattler, U. (2001). An overview of tableau algorithms for description logics. *Studia Logica*, 69(1):5–40.

Batet, M., Valls, A., and Gibert, K. (2010). A distance function to assess the similarity of words using ontologies. *XV Congreso Espaol sobre Tecnologas y Lgica Fuzzy (ESTYLF)*, 10:561–566.

Bianchini, D., Antonellis, V. D., and Melchiori, M. (2005). Capability matching and similarity reasoning in service discovery. In Missikoff, M. and 0001, A. D. N., editors, *EMOI-INTEROP*, volume 160 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Blanchard, E., Harzallah, M., Briand, H., and Kuntz, P. (2005). A typology of ontology-based semantic measures. In *Proceedings of the Open Interop Workshop on Enterprise Modelling and Ontologies for Interoperability*.

Champin, P. A. and Solnon, C. (2003). Measuring the similarity of labeled graphs. In Ashley, K. D. and Bridge, D. G., editors, *5th International Conference On Case-Based Reasoning (ICCBR 2003)*, LNAI, pages 80–95. Springer.

d'Amato, C., Fanizzi, N., and Esposito, F. (2006). Reasoning by analogy in description logics through instance-based learning. In *Semantic Web Applications and Perspectives*, volume 201 of *CEUR Workshop Proceedings*.

Fagin, R., Halpern, J. Y., and Megiddo, N. (1990). A logic for reasoning about probabilities. *Information and Computation.*, 87(1-2):78–128.

Klinov, P. (2008). Pronto: A non-monotonic probabilistic description logic reasoner. In *Proceedings of European Semantic Web Conference (ESWC)*, pages 822–826.

Lehmann, K. and Turhan, A.-Y. (2012). A framework for semantic-based similarity measures for $\mathcal{ELH}$ -concepts. In del Cerro, L. F., Herzig, A., and Mengin, J., editors, *JELIA*, volume 7519 of *Lecture Notes in Computer Science*, pages 307–319. Springer.

Li, L. and Horrocks, I. (2003). A software framework for matchmaking based on semantic web technology. In *12th international conference on World Wide Web*, pages 331–339. ACM Press.

Passant, A. (2010). Measuring semantic distance on linking data and using it for resources recommendations. In *AAAI Spring Symposium: Linked Data Meets Artificial Intelligence*.

Schickel-Zuber, V. and Faltings, B. (2007). Oss: A semantic similarity function based on hierarchical ontologies. In *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, IJCAI'07, pages 551–556, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Schulz, S., Suntisrivaraporn, B., Baader, F., and Boeker, M. (2009). SNOMED reaching its adolescence: Ontologists' and logicians' health check. *International Journal of Medical Informatics*, 78(Supplement 1):S86–S94.

Stuckenschmidt, H. (2009). A semantic similarity measure for ontology-based information. In *Flexible Query Answering Systems*, pages 406–417. Springer.

Suntisrivaraporn, B. (2013). A similarity measure for the description logic $\mathcal{EL}$ with unfoldable terminologies. In *International Conference on Intelligent Networking and Collaborative Systems (INCoS-13)*, pages 408–413.

Tongphu, S., Suntisrivaraporn, B., Uyyanonvara, B., and Dailey, M. N. (2012). Ontology-based object recognition of car sides. In *9th International Conference on Electrical Engineering/Electronics Computer Telecommunications and Information Technology (ECTI)*.