An Expressive Sub-language of OWL 2 Full for Domain Meta-modeling

Zhenzhen Gu[1], Cungen Cao[1], and Songmao Zhang[2]

[1] Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
[2] Academy of Mathematics and Systems Sciences, Chinese Academy of Sciences, Beijing, China
guzhenzhen0720@163.com cgcao@ict.ac.cn smzhang@math.ac.cn

Abstract. With the motivation of reasoning with real-world, complex domain ontologies in the category of OWL 2 Full, in this paper, we study meta-modeling extension in $\mathcal{SROIQ}$, and propose an expressive sub-language of OWL 2 Full, called Hi($\mathcal{SROIQ}$), where the same names can be used as classes, roles and individuals simultaneously. For accessing meta-knowledge, meta-queries are introduced by allowing variables to occur in the class and role positions in conjunctive queries. In order to take advantage of the highly optimized reasoners for description logics, we provide a sound and complete way of reducing satisfiability checking and meta-query answering in Hi($\mathcal{SROIQ}$) to the corresponding reasoning tasks in $\mathcal{SROIQ}$. Based on this, we conclude that meta-modeling extension in $\mathcal{SROIQ}$ does not increase the complexity of reasoning.

1 Introduction

OWL 2 [1], as a de facto standard ontology language, consists of two expressive sub-languages OWL 2 Full and OWL 2 DL and three profiles. Among them, OWL 2 Full is the most expressive whereas its reasoning turns undecidable [5]. The distinctive feature of OWL 2 Full is meta-modeling, i.e., allowing the same names to have multiple uses. It is unfortunately the main reason that causes the undecidability of reasoning in the language. Meta-modeling, as an important knowledge representation mechanism, can be frequently spotted in real-world ontologies, including the commonsense ontologies OpenCyc and SUMO and the domain ontologies FMA, CHEBI, GO and NCI in life science [3]. In these ontologies, most of the names used as classes or roles are also used as individuals at the same time (as seen in Table 1), leading them to fall into the category of OWL 2 Full. Compared with other sub-languages, reasoning in OWL 2 Full has largely been unexplored, and there do not exist any reasoners tailored for OWL 2 Full. The gap between the meta-modeling requirements in reality and the few studies on reasoning in OWL 2 Full raises a challenge.

To cater for the need of meta-modeling, OWL 2 DL provides a technique [2] called punning which syntacticly allows names to have multiple uses while semantically treats the different uses of the same names as completely separate. As

[3] http://sw.opencyc.org/, https://www.bioontology.org/wiki/index.php/FMAInOwl, http://www.adampease.org/OP/, https://bioportal.bioontology.org/ontologies

Table 1. Statistics of some real-world ontologies where TBox, ABox, Cla, Rol and Ind respectively denote the sets of axioms, individual assertions, named classes, named roles and individuals, and || denotes the size of set.

Ontology	|TBox|	|ABox|	|Cla|	|Rol|	|Ind|	|Cla ∩ Ind|	|Rol ∩ Ind|
OpenCyc	282,613	2,699,372	119,962	26,832	1,077,732	116,842	26,829
SUMO	7,081	489,949	4,557	898	256,576	3,591	654
FMA	82,834	1,923,155	84,395	170	222,578	78,988	170
CHEBI	224,963	2,656,188	191,293	37	800,308	122,057	17
GO	154,270	622,341	149,026	217	227,050	47,483	177
NCI	180,890	1,305,846	239,504	177	119,115	118,941	173
OGG	70,232	1,068,010	70,127	133	70,557	69,688	105

a syntactic solution, punning would not infer any more entailments than OWL 2 DL. Besides punning, there exist works studying meta-modeling extension in description logics (DLs) based on different semantics and application motivations.

Among these works, [5–12] study meta-modeling extension by allowing names to have multiple uses and based on HiLog semantics [4] which takes the same way as the OWL 2 RDF-Based semantics (the de facto semantics of OWL 2 Full) [3] to interpret the multiple uses of names. Concretely, [5], [6] and [10] respectively discuss extending $\mathcal{SHOIQ}$ [19], $\mathcal{SHIQ}$ [20] and DL-Lite$_\mathcal{R}$ and provide the languages $\mathcal{SHOIQ}$ with meta-modeling, Hi($\mathcal{SHIQ}$) and Hi(OWL2QL). For $\mathcal{SHOIQ}$ with meta-modeling and Hi($\mathcal{SHIQ}$), complex role axioms and Self restriction which are usually used in biomedical ontologies are not supported. The expressivity of Hi(OWL2QL) is too limited to capture the complex knowledge described in actual complex ontologies. Complex axioms about roles can be captured by Hi(Horn-$\mathcal{SROIQ}$), Hi($\mathcal{SRIQ}$) and $\mathcal{HI}$($\mathcal{SROIQ}$) respectively proposed by [7], [8] and [12] and obtained by extending Horn-$\mathcal{SROIQ}$ [23], $\mathcal{SRIQ}$ [22] and $\mathcal{SROIQ}$ [24]. However, Hi($\mathcal{SRIQ}$), Hi(Horn-$\mathcal{SROIQ}$) and $\mathcal{HI}$($\mathcal{SROIQ}$) respectively do not support nominals, disjunctions and meta-modeling on roles which are heavily used in the KBs listed in Table 1. Moreover, meta-query answering is also discussed in these works except [5, 12].

On the other hand, works [13–18] study typed meta-modeling extension or extension based on Henkin semantics. Henkin semantics deals with higher-order structures via hierarchies of power sets. Under this semantics, for a KB $\mathcal{K}$, the relation (r) $\mathcal{K} \models a =_c b \Leftrightarrow \mathcal{K} \models a \approx b$ holds, where $=_c$ and $\approx$ respectively denote class equivalence and individual equivalence. However, in both HiLog semantics and OWL 2 Full RDF-Based Semantics, solely the ($\Leftarrow$) direction of (e) holds. Henkin semantics may make undesired conclusions be entailed. For example, consider the following knowledge (1)–(2) described in Linked Data:

$$\mathit{geosp{:}Country} =_c \mathit{geofr{:}Pays} \quad (1)$$

$$\mathit{rdfs{:}isDBy}(\mathit{geosp{:}Country}, \mathit{geospecies.owl}),\ \mathit{rdfs{:}isDBy}(\mathit{geofr{:}Pays}, \mathit{ontfr{:}geofr}) \quad (2)$$

$$\overline{\mathit{rdfs{:}isDBy}(\mathit{geofr{:}Pays}, \mathit{geospecies.owl}),\ \mathit{rdfs{:}isDBy}(\mathit{geosp{:}Country}, \mathit{ontfr{:}geofr})} \quad (c)$$

where *isDBy* is an abbreviation of *isDefinedBy*. By Henkin semantics, (1) implies *geosp:Country*$\approx$*geofr:Pays*. Then by (2), the assertions in (c) can be entailed. However, neither *geospecies.owl* defines *geofr:Pays*, nor *ontfr:geofr* defines *geosp:Country*. Thus these two conclusions are undesired. For the typed meta-modeling extension, names are attached with types or layer information (non-negative integers) with the intension to describe levels of classes and roles. Specifying axioms and assertions referring to names with different types, such as $A^t \sqsubseteq B^{t+2}$, is prohibited. Such kind of meta-modeling is rarely used in real-world ontologies. Besides, for all the works in this category, query answering which is crucial for knowledge sharing and reusing has never been discussed.

In order to reason with the actual, complex OWL 2 Full ontologies, in this paper we study meta-modeling extension in the expressive DL $\mathcal{SROIQ}$ by allowing all the names to have multiple uses without any restrictions and based on HiLog semantics. The contribution can be summarized as follows. (1) We define a sub-language of OWL 2 Full, called Hi($\mathcal{SROIQ}$), and for accessing meta-knowledge, meta-queries are introduced by allowing variables to occur in the class and role positions in conjunctive queries. (2) We provide a sound and complete way of reducing satisfiability checking and meta-query answering in Hi($\mathcal{SROIQ}$) to the corresponding reasoning tasks in $\mathcal{SROIQ}$ via renaming and materialization. (3) We prove that meta-modeling extension in $\mathcal{SROIQ}$ does not increase the complexity of reasoning. As a result, the real-world and complex OWL 2 Full ontologies, such as those in Table 1, can be captured by Hi($\mathcal{SROIQ}$), and by reasoning reduction, highly optimized DL reasoners can take effect to reason with Hi($\mathcal{SROIQ}$) KBs. *All the proofs are presented in the supplementary file*[4].

Compared with [12], besides meta-modeling on roles and meta-query answering, we provide a different way of reasoning reduction without increasing the size of the original KBs, however the price is to consider more than one DL KB when reasoning with and querying a Hi(SROIQ) KB. Based on the results in [18], [12] provides a reasoning reduction procedure by adding extra axioms and assertions to the original KB. Besides increasing the sizes of the KBs, this approach cannot be applied to reduce reasoning in Hi($\mathcal{L}$) to the corresponding reasoning in $\mathcal{L}$, where $\mathcal{L}$ is a sub-language of $\mathcal{SROIQ}$ without nominals, $\neg$ and $\forall$.

2 The definition of Hi($\mathcal{SROIQ}$) and meta-queries

2.1 The syntax of Hi($\mathcal{SROIQ}$) and meta-queries

Hi($\mathcal{SROIQ}$) is defined based on $\mathcal{SROIQ}$. Different to $\mathcal{SROIQ}$, Hi($\mathcal{SROIQ}$) has only one name set $\mathbf{N}$ for classes, roles and individuals. This means that the same names in $\mathbf{N}$ can be used as classes, roles and individuals at the same time. The syntax of Hi($\mathcal{SROIQ}$) is illustrated in the following definitions.

Definition 1. *A Hi($\mathcal{SROIQ}$) role is either a role name $P \in \boldsymbol{N}$ or its inverse P^-. In order not to consider the roles in the form of P^{--}, we define $P^{--} = P$. A Hi($\mathcal{SROIQ}$) RBox is a finite set of role axioms with the following forms:*

$$S_1 \cdots S_n \sqsubseteq_r R, \quad \mathsf{Dis}(S_1, S_2), \quad \mathtt{Ref}(R), \quad \mathtt{Irr}(R)$$

where S_1–S_n and R are roles. The Hi($\mathcal{SROIQ}$) classes C and D are inductively defined as follows:

[4] https://github.com/Lucy321456/SupplementaryFile/blob/master/DL19-proofs.pdf

$$C, D ::= A \mid \{o\} \mid \exists S.\mathsf{Self} \mid \neg C \mid C \sqcap D \mid C \sqcup D \mid \exists R.C \mid \forall R.C \mid \ \geq nS.C \mid \ \leq nS.C$$

where $A \in \boldsymbol{N}$ and $o \in \boldsymbol{N}$ are respectively called class name and individual, n is a non-negative integer, R and S are roles. A Hi($\mathcal{SROIQ}$) TBox $\mathcal{T}$ is a finite set of class inclusion axioms in the form of $C \sqsubseteq_c D$ where C and D are classes. A Hi($\mathcal{SROIQ}$) ABox $\mathcal{A}$ is a finite set of individual assertions with the forms:

$$C(a), \quad R(a,b), \quad \neg S(a,b), \quad a \approx b, \quad a \not\approx b$$

where C is a class, both R and S are roles, and $a, b \in \boldsymbol{N}$ are called individuals.

Hi($\mathcal{SROIQ}$) does not separate the names for classes, roles and individuals, thus the symbols $\sqsubseteq_c$ and $\sqsubseteq_r$ are used to distinguish between class inclusion axioms and role inclusion axioms. For simplicity, we use $a =_l b$ to abbreviate these two axioms $a \sqsubseteq_l b$ and $b \sqsubseteq_l a$, where $l \in \{c, r\}$. For decidability, the *regular role hierarchy* and *simple role* restrictions defined in $\mathcal{SROIQ}$ [24] are adopted.

Definition 2. *For a Hi($\mathcal{SROIQ}$) RBox $\mathcal{R}$, we say $\mathcal{R}$ is regular if there exists a partial order (an irreflexive and transitive relation) $\prec$ on $\boldsymbol{N} \cup \{n^- | n \in \boldsymbol{N}\}$ such that: (1) $S \prec R$ iff $S^- \prec R$, for every two roles S and R; and (2) each role inclusion axiom (axioms with $\sqsubseteq_r$) in $\mathcal{R}$ takes one of the following forms:*

$$RR \sqsubseteq_r R, \quad R^- \sqsubseteq_r R, \quad S_1 \cdots S_n \sqsubseteq_r R, \quad RS_1 \cdots S_n \sqsubseteq_r R, \quad S_1 \cdots S_n R \sqsubseteq_r R$$

where R is a role and $S_i \prec R$ for each $1 \leq i \leq n$. Given a Hi($\mathcal{SROIQ}$) RBox $\mathcal{R}$, the set of simple roles is inductively defined as following. A role R is simple, if (1) R is a name and does not occur on the right hand of any role inclusion axioms in $\mathcal{R}$; (2) R^- is simple; or (3) each role inclusion axiom containing R at the right hand has the form $S \sqsubseteq_r R$ and S is a simple role.

By Definitions 1–2, Hi($\mathcal{SROIQ}$) KBs are illustrated in the definition below.

Definition 3. *A Hi($\mathcal{SROIQ}$) KB $\mathcal{K} = (\mathcal{R}, \mathcal{T}, \mathcal{A})$ is a tuple where $\mathcal{R}$, $\mathcal{T}$ and $\mathcal{A}$ are respectively regular RBox, TBox and ABox, and all the roles (R/S) occurring in the role positions of the following expressions are simple roles.*

$$\geq nR.C, \quad \leq nR.C, \quad \exists R.\mathsf{Self}, \quad \mathsf{Dis}(R,S), \quad \mathsf{Irr}(R), \quad \neg R(a,b)$$

We use $\mathsf{ind}(\mathcal{K})$ to denote the set of all the names used as individuals in $\mathcal{A}$ or $\mathcal{T}$, and use $\mathsf{nSR}(\mathcal{K})$ to denote the set of all the non-simple roles w.r.t. $\mathcal{R}$.

The following example illustrates a Hi($\mathcal{SROIQ}$) KB about the knowledge of football teams described in the OpenCyc ontology.

Example 1. Consider the Hi($\mathcal{SROIQ}$) KB $\mathcal{K}$ consisting of the axioms and individual assertions (1)–(3) below described in the OpenCyc ontology. In $\mathcal{K}$, the names *FootballTeam* and *Football_team* are used as both classes and individuals.

$$\mathit{SportsTeam} \sqsubseteq_c \neg \mathit{AllStarTeam}, \ \mathit{Football\_team} \sqsubseteq_c \mathit{SportsTeam} \quad (1)$$

$$\mathit{FootballTeam} \approx \mathit{Football\_team} \quad (2)$$

$$\mathit{SportsTeamTypeBySport}(\mathit{Football\_team}), \ \mathit{FootballTeam}(\mathit{BarcelonaDragons}) \quad (3)$$

Let $\mathbf{V}$ be a set of variables so that $\mathbf{V} \cap \mathbf{N} = \emptyset$. The syntax of meta-queries is illustrated in the definition below.

Definition 4. *A query atom takes the form of $x(y)$ or $x(y,z)$ where $x, y, z \in$ $\boldsymbol{N} \cup \boldsymbol{V}$. A meta-query Q is an expression of the form: $\alpha_1 \wedge \cdots \wedge \alpha_n \rightarrow q(\boldsymbol{x})$ where α_1–α_n are query atoms, $\boldsymbol{x}$ is a tuple of elements in $\boldsymbol{N} \cup \boldsymbol{V}$ and each variable in $\boldsymbol{x}$ occurs in some α_i. We define $\mathsf{body}(Q) = \cup_{i=1}^{n}\{\alpha_i\}$ and $\mathsf{head}(Q) = \boldsymbol{x}$.*

For meta-query Q, a variable x is called a distinguished variable of Q if x occurs in $\mathsf{head}(Q)$, and x is called a non-distinguished variable of Q if x solely occurs in $\mathsf{body}(Q)$. Moreover, x is called a class variable (role variable) of Q if Q contains an atom $x(y)$ ($x(y,z)$). If Q does not contain any class or role variables then it becomes a conjunctive query. By allowing variables to occur in the class and role positions, schema knowledge and data can be queried in a uniform way. For example, the query asking for the types of the team *BarcelonaDragons* can be formally represented as the following meta-query:

$$?x(\mathit{BarcelonaDragons}) \wedge \mathit{SportsTeamTypeBySport}(?x) \rightarrow q(?x)$$

2.2 The semantics of Hi($\mathcal{SROIQ}$) and meta-queries

Hi($\mathcal{SROIQ}$) and meta-queries are interpreted by the ν-semantics [5] which is based on HiLog semantics.

Syntax	Semantics	Syntax	Semantics
P	$\mathfrak{R}^{\mathcal{V}}(P^{\mathcal{V}})$	$C \sqcup D$	$\mathfrak{C}^{\mathcal{V}}(C) \cup \mathfrak{C}^{\mathcal{V}}(D)$
P^{-}	$\{(y,x) \mid (x,y) \in \mathfrak{R}^{\mathcal{V}}(P^{\mathcal{V}})\}$	$C \sqcap D$	$\mathfrak{C}^{\mathcal{V}}(C) \cap \mathfrak{C}^{\mathcal{V}}(D)$
ω	$\mathfrak{R}^{\mathcal{V}}(R_1) \circ \cdots \circ \mathfrak{R}^{\mathcal{V}}(R_1)$	$C \sqsubseteq_c D$	$\mathfrak{C}^{\mathcal{V}}(C) \subseteq \mathfrak{C}^{\mathcal{V}}(D)$
A	$\mathfrak{C}^{\mathcal{V}}(A^{\mathcal{V}})$	$\omega \sqsubseteq_r R$	$\mathfrak{R}^{\mathcal{V}}(R_1 \cdots R_n) \subseteq \mathfrak{R}^{\mathcal{V}}(R)$
$\{a\}$	$\{a^{\mathcal{V}}\}$	$\mathsf{Dis}(S,R)$	$\mathfrak{R}^{\mathcal{V}}(S) \cap \mathfrak{R}^{\mathcal{V}}(R) = \emptyset$
$\exists S.\mathsf{Self}$	$\{x \mid (x,x) \in \mathfrak{R}^{\mathcal{V}}(R)\}$	$\mathsf{Ref}(S)$	$\{(x,x) \mid x \in \Delta^{\mathcal{V}}\} \subseteq \mathfrak{R}^{\mathcal{V}}(S)$
$\neg C$	$\Delta^{\mathcal{V}} - \mathfrak{C}^{\mathcal{V}}(C)$	$\mathsf{Irr}(S)$	$\{(x,x) \mid x \in \Delta^{\mathcal{V}}\} \cap \mathfrak{R}^{\mathcal{V}}(S) = \emptyset$
$\geq nS.D$	$\{x \mid \lvert\{y \mid (x,y) \in \mathfrak{R}^{\mathcal{V}}(S) \wedge y \in \mathfrak{C}^{\mathcal{V}}(D)\}\rvert \geq n\}$	$C(a)$	$a^{\mathcal{V}} \in \mathfrak{C}^{\mathcal{V}}(C)$
$\leq nS.D$	$\{x \mid \lvert\{y \mid (x,y) \in \mathfrak{R}^{\mathcal{V}}(S) \wedge y \in \mathfrak{C}^{\mathcal{V}}(D)\}\rvert \leq n\}$	$R(a,b)$	$(a^{\mathcal{V}}, b^{\mathcal{V}}) \in \mathfrak{R}^{\mathcal{V}}(R)$
$\exists R.C$	$\{x \mid \exists y.(x,y) \in \mathfrak{R}^{\mathcal{V}}(R) \wedge y \in \mathfrak{C}^{\mathcal{V}}(C)\}$	$\neg S(a,b)$	$(a^{\mathcal{V}}, b^{\mathcal{V}}) \notin \mathfrak{R}^{\mathcal{V}}(S)$
$\forall R.C$	$\{x \mid \forall y.(x,y) \in \mathfrak{R}^{\mathcal{V}}(R) \rightarrow y \in \mathfrak{C}^{\mathcal{V}}(C)\}$	$a \approx b, a \not\approx b$	$a^{\mathcal{V}} = b^{\mathcal{V}}, a^{\mathcal{V}} \neq b^{\mathcal{V}}$

Fig. 1. Interpretation of Hi($\mathcal{SROIQ}$) roles, classes, axioms and assertions w.r.t. an interpretation $\mathcal{V}$, where $A, P \in \mathbf{N}$, $\circ$ denotes binary relation composition, $\omega = R_1 \cdots R_n$.

Definition 5. *A ν-interpretation $\mathcal{V} = (\Delta^{\mathcal{V}}, \cdot^{\mathcal{V}}, \mathfrak{C}^{\mathcal{V}}, \mathfrak{R}^{\mathcal{V}})$ is a tuple where $\Delta^{\mathcal{V}}$ is a non-empty set, $\cdot^{\mathcal{V}}$, $\mathfrak{C}^{\mathcal{V}}$ and $\mathfrak{R}^{\mathcal{V}}$ are functions such that (1) $\cdot^{\mathcal{V}}$ maps each name in $\boldsymbol{N}$ to an element in $\Delta^{\mathcal{V}}$; (2) $\mathfrak{C}^{\mathcal{V}}$ maps each element in $\Delta^{\mathcal{V}}$ to a subset of $\Delta^{\mathcal{V}}$; and (3) $\mathfrak{R}^{\mathcal{V}}$ maps each element in $\Delta^{\mathcal{V}}$ to a subset of $(\Delta^{\mathcal{V}})^2$. The interpretation of constructors, axioms and individual assertions is illustrated in Fig. 1. For a Hi($\mathcal{SROIQ}$) KB $\mathcal{K}$, $\mathcal{V}$ is called a ν-model of $\mathcal{K}$ if $\mathcal{V}$ satisfies all the axioms and assertions in $\mathcal{K}$. The ν-satisfiability and ν-entailment ($\models_\nu$) are defined as usual.*

Different to the classic semantics of DLs, under ν-semantics, each name is mapped into a domain element and each domain element is associated with a class extension and a role extension. The motivation is to guarantee that if two names a and b are mapped into the same domain element, i.e., they denote the same semantic entity, then they should have the same class and role extension, i.e., $\mathfrak{C}^{\mathcal{V}}(a^{\mathcal{V}})=\mathfrak{C}^{\mathcal{V}}(b^{\mathcal{V}})$ and $\mathfrak{R}^{\mathcal{V}}(a^{\mathcal{V}})=\mathfrak{R}^{\mathcal{V}}(b^{\mathcal{V}})$. Such distinguished feature is crucial in enabling Hi($\mathcal{SROIQ}$) KBs to entail more conclusions compared with punning. However, this feature may cause Hi($\mathcal{SROIQ}$) KBs to imply the equivalences between non-simple roles and simple roles, as shown in the example below.

Example 2. Consider the KB consisting of the following axioms and assertions:

$$RP_1 \sqsubseteq_r P_1,\ P_1P_2 \sqsubseteq_r P_2,\ P_2P_3 \sqsubseteq_r P_3,\ P_3S \sqsubseteq_r S,\ SS \sqsubseteq_r S\ \top \sqsubseteq_c \geq 5R.\top,\ R \approx S$$

Obviously, R is a simple role and S is a non-simple role. Under ν-semantics, the individual assertion $R \approx S$ implies that R and S are equivalent roles, i.e., $R =_r S$ holds, as $\mathfrak{R}^{\mathcal{V}}(R^{\mathcal{V}})=\mathfrak{R}^{\mathcal{V}}(S^{\mathcal{V}})$ holds for each ν-model $\mathcal{V}$ of this KB.

Such implied equivalences between simple-roles and non-simple roles can lead to (1) transitive roles (e.g., $SS \sqsubseteq_r S$) being used in number restrictions ($\geq nS.C$ or $\leq nS.C$), and (2) role hierarchies containing cyclic dependencies. (1) and (2) are well-known for causing undecidability of reasoning [25, 26]. Thus we further adopt the unique non-simple role assumption (UNSRA) defined in our previous paper [7] to forbid a KB from implying the undesired equivalences among roles.

Definition 6. *A Hi($\mathcal{SROIQ}$) KB $\mathcal{K} = (\mathcal{R}, \mathcal{T}, \mathcal{A})$ adopts the UNSRA if for each $(a, b) \in \mathsf{ind}(\mathcal{K})^2$, if a or b is a non-simple role w.r.t. $\mathcal{R}$ then $a \not\approx b \in \mathcal{A}$.*

For decidability, we only consider the KBs adopting UNSRA. For a tuple $\boldsymbol{u}$, we use $|\boldsymbol{u}|$ and $\boldsymbol{u}[i]$ to denote the length and the i-th element of $\boldsymbol{u}$, respectively. For a tuple $\boldsymbol{x}$ with length $|\boldsymbol{u}|$, we use $[\boldsymbol{x}/\boldsymbol{u}]$ to denote a substitution. For a KB, tuple or query O, we use $O[\boldsymbol{x}/\boldsymbol{u}]$ to denote the result of replacing each occurrence of $\boldsymbol{x}[i]$ in O with $\boldsymbol{u}[i]$, for $1 \leq i \leq |\boldsymbol{x}|$. For a query Q and tuple $\boldsymbol{u}$ with length $|\mathsf{head}(Q)|$, we use $Q(\boldsymbol{u})$ to denote $Q[\mathsf{head}(Q)/\boldsymbol{u}]$. The semantics of meta-queries is illustrated in the definition below.

Definition 7. *For a meta-query Q and ν-interpretation $\mathcal{V}$, a binding π of Q over $\mathcal{V}$ is a function that maps each name a in Q to $a^{\mathcal{V}}$ and each variable in Q to an element in $\Delta^{\mathcal{V}}$. We write $\mathcal{V}, \pi \models_\nu Q$ if $\pi(y) \in \mathfrak{C}^{\mathcal{V}}(\pi(x))$ for each $x(y) \in \mathsf{body}(Q)$ and $(\pi(y), \pi(z)) \in \mathfrak{R}^{\mathcal{V}}(\pi(x))$ for each $x(y, z) \in \mathsf{body}(Q)$. We write $\mathcal{V} \models_\nu Q$ if there exists a binding π of Q over $\mathcal{V}$ such that $\mathcal{V}, \pi \models_\nu Q$. For a Hi($\mathcal{SROIQ}$) KB $\mathcal{K}$, a tuple $\boldsymbol{u}$ consisting of names in $\mathcal{K}$ and with length $\mathsf{head}(Q)$ is called a certain answer of Q over $\mathcal{K}$ if for each ν-model $\mathcal{V}$ of $\mathcal{K}$, $\mathcal{V} \models_\nu Q(\boldsymbol{u})$ holds and for $1 \leq i \leq |\boldsymbol{u}|$, if $\mathsf{head}(Q)[i] \in \boldsymbol{N}$ then $\boldsymbol{u}[i]^{\mathcal{V}} = \mathsf{head}(Q)[i]^{\mathcal{V}}$ holds. We use $\mathsf{answer}_\nu(Q, \mathcal{K})$ to denote the set of all the certain answers of Q over $\mathcal{K}$.*

3 Satisfiability checking in Hi($\mathcal{SROIQ}$)

Here, we present the way of reducing ν-satisfiability checking in Hi($\mathcal{SROIQ}$) to satisfiability checking in $\mathcal{SROIQ}$ with the aid of punning. Before that, we first show the translation from Hi($\mathcal{SROIQ}$) KBs to $\mathcal{SROIQ}$ KBs via renaming.

Let $\mathbf{C}$ and $\mathbf{R}$ be the sets of names for $\mathcal{SROIQ}$ classes and roles, respectively. For simplicity, we suppose $\mathbf{N}$ is the set of names for $\mathcal{SROIQ}$ individuals. Let $\boldsymbol{v}_c$

and $\boldsymbol{v}_r$ be two bijective functions that respectively map each name in $\mathbf{N}$ to an unique name in $\mathbf{C}$ and an unique name in $\mathbf{R}$. The translation of Hi($\mathcal{SROIQ}$) classes, roles, axioms and assertions realized by functions τ_c, τ_r and τ is illustrated in Fig. 2. For a Hi($\mathcal{SROIQ}$) KB $\mathcal{K}$, we use $\tau_{\mathsf{dl}}(\mathcal{K})$ to denote the KB obtained by replacing each axiom (assertion) α in $\mathcal{K}$ with $\tau(\alpha)$. The soundness of punning can be guaranteed by the proposition below.

f	α	$f(\alpha)$
τ_r	$P,\ P^-$	$\boldsymbol{v}_r(P),\ \boldsymbol{v}_r(P)^-$
	$R_1 \cdots R_n$	$\tau_r(R_1) \cdots \tau_r(R_n)$
τ_c	$A,\ \{o\}$	$\boldsymbol{v}_c(A),\ \{o\}$
	$\exists S.\mathsf{Self}$	$\exists \tau_r(S).\mathsf{Self}$
	$\neg C$	$\neg \tau_c(C)$
	$\geq nS.D$	$\geq n\tau_r(S).\tau_c(D)$
	$\leq nS.D$	$\leq n\tau_r(S).\tau_c(D)$
	$\exists R.C$	$\exists \tau_r(R).\tau_c(C)$
	$\forall R.C$	$\forall \tau_r(R).\tau_c(C)$

f	α	$f(\alpha)$
τ_c	$C \sqcup D$	$\tau_c(C) \sqcup \tau_c(D)$
	$C \sqcap D$	$\tau_c(C) \sqcap \tau_c(D)$
τ	$C \sqsubseteq_c D$	$\tau_c(C) \sqsubseteq \tau_c(D)$
	$R_1 \cdots R_n \sqsubseteq_r R$	$\tau_r(R_1) \cdots \tau_r(R_n) \sqsubseteq \tau_r(R)$
	$\mathsf{Dis}(S, R)$	$\mathsf{Dis}(\tau_r(S), \tau_r(R))$
	$\mathsf{Ref}(S),\ \mathsf{Irr}(S)$	$\mathsf{Ref}(\tau_r(S)),\ \mathsf{Irr}(\tau_r(S))$
	$C(a),\ \neg S(a,b)$	$\tau_c(C)(a),\ \neg \tau_r(S)(a,b)$
	$R(a,b)$	$\tau_r(R)(a,b)$
	$a \approx b,\ a \not\approx b$	$a \approx b,\ a \not\approx b$

Fig. 2. Definition of functions τ_c, τ_r and τ where $P, A, o, a, b \in \mathbf{N}$, R_1–R_n and S are Hi($\mathcal{SROIQ}$) roles, and C and D are Hi($\mathcal{SROIQ}$) classes.

Proposition 1. *If Hi($\mathcal{SROIQ}$) KB $\mathcal{K}$ is ν-satisfiable then $\tau_{\mathsf{dl}}(\mathcal{K})$ is satisfiable.*

However, by punning technique alone, completeness cannot be ensured.

Example 3. Consider the Hi($\mathcal{SROIQ}$) KB $\mathcal{K}$ consisting of the following axiom and individual assertions described in the OpenCyc KB.

$$PrimeMinister\_HeadOfGovernment \sqsubseteq_c \neg Prime\_minister \quad (1)$$
$$PrimeMinister\_HeadOfGovernment \approx Prime\_minister \quad (2)$$
$$PrimeMinister\_HeadOfGovernment(MargaretThatcher) \quad (3)$$

Obviously, $\tau_{\mathsf{dl}}(\mathcal{K})$ is satisfiable. However, $\mathcal{K}$ is not ν-satisfiable, since under ν-semantics, assertion (2) implies the axiom $PrimeMinister\_HeadOfGovernment =_c Prime\_minister$ which contradicts with the axiom (1) and assertion (3).

The incompleteness is caused by the fact that under ν-semantics, the behaviors of names used as individuals will affect the same names used as classes or roles, i.e., if a KB implies the assertion $a \approx b$ then the axioms $a =_c b$ and $a =_r b$ are also implied, whereas under punning, such impact of individuals on classes and roles do not exist anymore. Inspired by the results in [5, 7], next we present an approach of materializing such impact in the original KB in order to obtain completeness. Before this, we first illustrate a sufficient and necessary condition under which ν-semantics and punning coincide in terms of KB satisfiability.

Lemma 1. *For a Hi($\mathcal{SROIQ}$) KB $\mathcal{K} = (\mathcal{R}, \mathcal{T}, \mathcal{A})$, if $a \not\approx b \in \mathcal{A}$ for each $(a,b) \in \mathsf{ind}(\mathcal{K})^2$ and $a \neq b$, then $\mathcal{K}$ is ν-satisfiable iff $\tau_{\mathsf{dl}}(\mathcal{K})$ is satisfiable.*

Lemma 1 indicates that if all the individuals of $\mathcal{K}$ are pairwise nonequivalent then punning and ν-semantics are coincide in terms of satisfiability checking. Based on Lemma 1, the basic idea of materializing the impacts of individuals on classes and roles is to first guess the equivalence between individuals implied by a KB, and then for the guessed equivalent individuals, in order not to increase the size of the original KB, we choose to replace them with the same representative in the KB rather than add extra class and role equivalent axioms. Note that the guess of the equivalence between individuals rather than detecting such equivalence by DL Reasers like [7] is caused by the "uncertainty" of individual equivalence entailment resulted from number restrictions ($\leq n$).

Example 4. Consider the KB $\mathcal{K}$ consisting of the following axioms and assertions:

$$A \sqsubseteq_c \leq 2R.B,\ a \sqsubseteq_c \neg b,\ a \sqsubseteq_c \neg c,\ b \sqsubseteq_c \neg c,\ a(b),\ b(a),\ c(b) \quad (1)$$
$$A(a),\ R(a,b),\ R(a,c),\ R(a,a),\ B(a),\ B(b),\ B(c) \quad (2)$$

$\mathcal{K}$ and $\tau_{\mathsf{dl}}(\mathcal{K})$ imply $a \approx b$, $a \approx c$ or $b \approx c$. However, which can be entailed cannot be guaranteed by the current knowledge. So we need to try all the possibilities of the equivalences among the individuals in $\{a, b, c\}$ to check whether there exists one possibility that makes the materialized KB be satisfiable under punning.

For a function f, we use $\mathsf{dom}(f)$ and $\mathsf{ran}(f)$ to denote the domain and range of f, respectively. For a KB, query or tuple O, we use Of to denote the result of replacing each occurrence of a in O with $f(a)$ for each $a \in \mathsf{dom}(f)$. Next, we formalize the materialization and reduction technique in detail.

Definition 8. *For a Hi($\mathcal{SROIQ}$) KB $\mathcal{K}$, a function E is called a candidate individual equivalence replacing function (CIERF) of $\mathcal{K}$ if* $\mathsf{dom}(E) = \mathsf{ran}(E) = \mathsf{ind}(\mathcal{K}) - \mathsf{nSR}(\mathcal{K})$ *and* $E(E(a)) = E(a)$ *for each* $a \in \mathsf{dom}(E)$.

In Definition 8, $E(E(a)) = E(a)$ requires that the representatives for the guessed equivalent individuals will not be replaced by other individuals. Moreover, both $\mathsf{dom}(E)$ and $\mathsf{ran}(E)$ do not contain non-simple roles, since (1) the *UNSRA* requires that non-simple roles are nonequivalent with other individuals and (2) replacing non-simple roles with other individuals may lead to the resultant KB do not satisfy the *simple role restriction.* One guess of the equivalences between the individuals implied by $\mathcal{K}$ actually corresponds to a partition of $\mathsf{ind}(\mathcal{K}) - \mathsf{nSR}(\mathcal{K})$. So a CIERF E of $\mathcal{K}$ can be obtained by first computing a partition $\mathcal{P}$ of $\mathsf{ind}(\mathcal{K}) - \mathsf{nSR}(\mathcal{K})$, and then for each $\mathrm{U} \in \mathcal{P}$, choosing a representative $a \in \mathrm{U}$ and setting $E(b) = a$ for each $b \in \mathrm{U}$. The way of materializing the guessed individual equivalence relations is shown in the definition below.

Definition 9. *For a Hi($\mathcal{SROIQ}$) KB $\mathcal{K}$ and a CIERF E of $\mathcal{K}$, we use $[\mathcal{K}E]$ to denote the KB obtained by first computing $\mathcal{K}E$, and then adding $a \not\approx b$ to $\mathcal{K}E$ for each $(a, b) \in \{E(c) | c \in \mathsf{dom}(E)\}^2$ and $a \neq b$.*

The concrete way of reducing ν-satisfiability checking in Hi($\mathcal{SROIQ}$) to satisfiability checking in $\mathcal{SROIQ}$ with the aid of punning and the materialization technique is shown in the following theorem.

Theorem 1. *A Hi($\mathcal{SROIQ}$) KB $\mathcal{K}$ is ν-satisfiable iff there exists a CIERF E of $\mathcal{K}$ such that $\tau_{\mathsf{dl}}([\mathcal{K}E])$ is satisfiable.*

Note that for a partition $\mathcal{P}$ of the set $\mathsf{ind}(\mathcal{K})-\mathsf{nSR}(\mathcal{K})$, choosing different representative elements for the sets in $\mathcal{P}$ will generate different CIERFs of $\mathcal{K}$, thus yielding different materialized KBs. Take the KB $\mathcal{K}$ in Example 3 as an example. For the following partition $\mathcal{P}$ of $\mathsf{ind}(\mathcal{K})-\mathsf{nSR}(\mathcal{K})$:

$$\{\{\mathit{PrimeMinister\_HeadOfGovernment}, \mathit{Prime\_minister}, \mathit{MargaretThatcher}\}\}$$

totally three CIERFs E_1–E_3 can be generated by choosing different representives for the sole equivalence class in $\mathcal{P}$. Nevertheless, as shown in the lemma below, different CIERFs corresponding to the same partition do not affect the result of satisfiability checking, i.e., $[\mathcal{K}E_1]$–$[\mathcal{K}E_3]$ are pairwise equisatisfiable.

Lemma 2. *For a Hi($\mathcal{SROIQ}$) KB $\mathcal{K}$ and two CIERFs E_1 and E_2 of $\mathcal{K}$ such that for each $(a,b) \in (\mathsf{ind}(\mathcal{K})-\mathsf{nSR}(\mathcal{K}))^2$ and $a \neq b$, $E_1(a) = E_1(b)$ holds iff $E_2(a) = E_2(b)$ holds, then $\tau_{\mathsf{dl}}([\mathcal{K}E_1])$ is satisfiable iff $\tau_{\mathsf{dl}}([\mathcal{K}E_2])$ is satisfiable.*

Lemma 2 implies that for each partition $\mathcal{P}$ of $\mathsf{ind}(\mathcal{K})-\mathsf{nSR}(\mathcal{K})$, considering one CIERF is sufficient. Thus for checking the ν-satisfiability of $\mathcal{K}$, no more than $2^{|\mathsf{ind}(\mathcal{K})-\mathsf{nSR}(\mathcal{K})|^2}$ $\mathcal{SROIQ}$ KBs need to be considered. Then based on [13], we can further obtain the complexity of ν-satisfiability checking in Hi($\mathcal{SROIQ}$).

Theorem 2. *Checking the ν-satisfiability of a Hi($\mathcal{SROIQ}$) KB $\mathcal{K}$ can be done in* N2ExpTime *w.r.t. the size of $\mathcal{K}$.*

Thus meta-modeling extension in $\mathcal{SROIQ}$ does not increase the complexity of reasoning. However, Theorem 1 indicates that checking the ν-satisfiability of a Hi($\mathcal{SROIQ}$) KB $\mathcal{K}$ may need to consider as many as an exponential size of $\mathcal{SROIQ}$ KBs w.r.t. $|\mathsf{ind}(\mathcal{K})|$. Therefore, we further provide a condition to reduce the number of $\mathcal{SROIQ}$ KBs needed to be considered.

Lemma 3. *For a CIERF E of a Hi($\mathcal{SROIQ}$) KB $\mathcal{K} = (\mathcal{R}, \mathcal{T}, \mathcal{A})$, if there exist $(a,b) \in (\mathsf{ind}(\mathcal{K})-\mathsf{nSR}(\mathcal{K}))^2$ satisfying that $E(a)=E(b)$ and $\tau_{\mathsf{dl}}(\mathcal{K}) \models a \not\approx b$, or that $E(a) \neq E(b)$ and $\tau_{\mathsf{dl}}(\mathcal{K}) \models a \approx b$, then $\tau_{\mathsf{dl}}([\mathcal{K}E])$ is not satisfiable.*

Example 5. Consider the KB $\mathcal{K}$ in Example 3 again. Lemmas 2–3 indicate that for a partition $\mathcal{P}$ of $\mathsf{ind}(\mathcal{K})-\mathsf{nSR}(\mathcal{K})$, if there does not exist $U \in \mathcal{P}$ such that $\{\mathit{PrimeMinister\_HeadOfGovernment}, \mathit{Prime\_minister}\} \subseteq U$, then for each CIERF E generated from $\mathcal{P}$, $\tau_{\mathsf{dl}}([\mathcal{K}E])$ is not satisfiable. Three out of five partitions of $\mathsf{ind}(\mathcal{K}) - \mathsf{nSR}(\mathcal{K})$ satisfy this condition. Thus, checking the ν-satisfiability of $\mathcal{K}$ solely needs to consider the satisfiability of two $\mathcal{SROIQ}$ KBs rather than five.

4 Meta-query answering in Hi($\mathcal{SROIQ}$)

Here, we first present the way of reducing conjunctive query (CQ) answering in Hi($\mathcal{SROIQ}$) to CQ answering in $\mathcal{SROIQ}$, then we show that meta-query answering in Hi($\mathcal{SROIQ}$) can be captured by CQ answering.

For the reduction of CQ answering, a naive solution is to use punning. For a CQ Q, we use $\tau_{\mathsf{dl}}(Q)$ to denote the result of replacing each atom $A(x)$ in Q with $\tau_c(A)(x)$ and each $P(x,y)$ in Q with $\tau_r(P)(x,y)$. The soundness of punning in terms of CQ answering is shown in the proposition below.

Proposition 2. *For a ν-satisfiable Hi($\mathcal{SROIQ}$) KB $\mathcal{K}$,* $\mathsf{answer}(\tau_{\mathsf{dl}}(Q), \tau_{\mathsf{dl}}(\mathcal{K})) \subseteq \mathsf{answer}_\nu(Q, \mathcal{K})$ *holds for each CQ Q*[5].

Nevertheless, similarly to ν-satisfiability checking, by punning technique alone, the completeness of CQ answering cannot be ensured.

Example 6. Consider the KB $\mathcal{K}$ in Example 1 again. $\mathcal{K}$ is ν-satisfiable. For the query $Q: \mathit{Football\_team}(?x) \rightarrow q(?x)$, $\mathsf{answer}_\nu(Q, \mathcal{K}) = \{(\mathit{BarcelonaDragons})\}$ holds, since $\mathit{Football\_team} \approx \mathit{FootballTeam}$ implies the axiom $\mathit{Football\_team} =_c \mathit{FootballTeam}$. However, $\mathsf{answer}(\tau_{\mathsf{dl}}(Q), \tau_{\mathsf{dl}}(\mathcal{K})) = \emptyset$, as the class equivalence between *Football_team* and *FootballTeam* is not implied by $\mathcal{K}$ under punning.

The incompleteness is also caused by the impacts of individuals on classes and roles. Thus for completeness, the basic idea is to materialize such impacts. The lemma below indicates that the materialization technique designed for satisfiability checking can also be adopted to obtain the completeness of CQ answering.

Lemma 4. *For a ν-satisfiable Hi($\mathcal{SROIQ}$) KB $\mathcal{K} = (\mathcal{R}, \mathcal{T}, \mathcal{A})$, if $a \not\approx b \in \mathcal{A}$ for each $(a, b) \in \mathsf{ind}(\mathcal{K})^2$ and $a \neq b$, then* $\mathsf{answer}_\nu(Q, \mathcal{K}) = \mathsf{answer}(\tau_{\mathsf{dl}}(Q), \tau_{\mathsf{dl}}(\mathcal{K}))$ *holds for each CQ Q.*

By trying each possibility of materializing the impacts of individuals on classes and roles and then intersecting the answers obtained, we can obtain all the certain answers of CQs over ν-satisfiable KBs, shown in the theorem below.

Theorem 3. *For a ν-satisfiable Hi($\mathcal{SROIQ}$) KB $\mathcal{K}$ and conjunctive query Q, let $\mathcal{E}$ be the set of all the CIERFs E satisfying that $\tau_{\mathsf{dl}}([\mathcal{K}E])$ is satisfiable, then:*

$$\mathsf{answer}_\nu(Q, \mathcal{K}) = \cap_{E \in \mathcal{E}} \{\boldsymbol{u} \mid \boldsymbol{u} \in \boldsymbol{N}^{|\mathsf{head}(Q)|} \text{ and } \boldsymbol{u}E \in \mathsf{answer}(\tau_{\mathsf{dl}}(QE), \tau_{\mathsf{dl}}([\mathcal{K}E]))\}$$

As mentioned earlier, choosing different representatives for the sets in a partition of $\mathsf{ind}(\mathcal{K}) - \mathsf{nSR}(\mathcal{K})$ will generate different CIERFs of $\mathcal{K}$ and thus different materialized KBs. However, this does not affect the result of CQ answering.

Lemma 5. *For a ν-satisfiable Hi($\mathcal{SROIQ}$) KB $\mathcal{K}$, let E_1 and E_2 be two CIERFs of $\mathcal{K}$ so that $\tau_{\mathsf{dl}}(\mathcal{K}E_1)$ and $\tau_{\mathsf{dl}}(\mathcal{K}E_1)$ are satisfiable and for each $(a, b) \in (\mathsf{ind}(\mathcal{K}) - \mathsf{nSR}(\mathcal{K}))^2$ and $a \neq b$, $E_1(a) = E_1(b)$ iff $E_2(a) = E_2(b)$. Then for each CQ Q:*

$$\mathsf{answer}(\tau_{\mathsf{dl}}(QE_1), \tau_{\mathsf{dl}}([\mathcal{K}E_1])) = \mathsf{answer}(\tau_{\mathsf{dl}}(QE_2), \tau_{\mathsf{dl}}([\mathcal{K}E_2]))$$

Lemma 5 implies that for a partition of $\mathsf{ind}(\mathcal{K}) - \mathsf{nSR}(\mathcal{K})$, considering one CIERF is sufficient for CQ answering. Thus answering a CQ over $\mathcal{K}$ requires no more than $2^{|\mathsf{ind}(\mathcal{K}) - \mathsf{nSR}(\mathcal{K})|^2}$ DL KBs to be considered. Therefor we can further obtain the complexity of CQ answering in Hi($\mathcal{SROIQ}$).

Theorem 4. $\mathsf{answer}_\nu(Q, \mathcal{K})$ *can be obtained in* N2ExpTime *w.r.t. the total size of a Hi($\mathcal{SROIQ}$) KB $\mathcal{K}$ and a CQ Q without non-distinguished variables.*

The decidability of CQ answering in $\mathcal{SROIQ}$ is currently unknown [28]. Thus in Theorem 4, just the CQs without non-distinguished variables are considered. Next, we illustrate the way of answering meta-queries in Hi($\mathcal{SROIQ}$).

[5] For a $\mathcal{SROIQ}$ KB $\mathcal{O}$ and CQ Q, we use $\mathsf{answer}(Q, \mathcal{O})$ to denote the set of all the certain answers of Q over $\mathcal{O}$

For a meta-query Q and Hi($\mathcal{SROIQ}$) KB $\mathcal{K}$, a CRV-Binding ξ of Q over $\mathcal{K}$ is a function that maps each class (resp. role) variable of Q to a name occurring in $\mathcal{K}$. The way of answering the meta-queries without non-distinguished variables by materialization is shown in the theorem below.

Theorem 5. *For a ν-satisfiable Hi($\mathcal{SROIQ}$) KB $\mathcal{K}$ and meta-query Q without non-distinguished variables, let $\mathcal{B}$ be the set of all the CRV-Bindings of Q over $\mathcal{K}$. Then* $\mathsf{answer}_\nu(Q,\mathcal{K}) = \cup_{\xi\in\mathcal{B}}\mathsf{answer}_\nu(Q\xi,\mathcal{K})$ *holds.*

As shown in the example below, if we allow meta-queries to contain non-distinguished variables, the completeness cannot be guaranteed anymore by class and role variable materialization.

Example 7. Consider the ν-satisfiable Hi($\mathcal{SROIQ}$) KB $\mathcal{K}$ and meta-query Q:

$$\mathcal{K} = (\emptyset, \{A \sqsubseteq_c \exists P.B \sqcup \exists P.C\}, \{A(a)\})$$
$$Q\colon A(?x) \wedge P(?x, ?z) \wedge ?c(?z) \rightarrow q(?x)$$

Obviously, $\mathsf{answer}_\nu(Q,\mathcal{K}) = \{(a)\}$, as for each ν-model $\mathcal{V}$ of $\mathcal{K}$, $a^{\mathcal{V}} \in \mathfrak{C}^{\mathcal{V}}(A^{\mathcal{V}})$ holds and there exists $A' \in \{B, C\}$ satisfying that $a^{\mathcal{V}} \in \mathfrak{C}^{\mathcal{V}}(\exists P.A')$. However, for each CRV-binding ξ of Q over $\mathcal{K}$, $\mathsf{answer}_\nu(Q,\mathcal{K}) = \emptyset$, as there does not exist any name A' in $\mathcal{K}$ satisfying that for each ν-model $\mathcal{V}$ of $\mathcal{K}$, $a^{\mathcal{V}} \in \mathfrak{C}^{\mathcal{V}}(\exists P.A')$ holds.

By Theorems 3-5, we can further obtain the complexity of answering the meta-queries without non-distinguished variables over Hi($\mathcal{SROIQ}$) KBs.

Theorem 6. $\mathsf{answer}_\nu(Q,\mathcal{K})$ *can be obtained in* N2ExpTime *w.r.t. the total size of a Hi($\mathcal{SROIQ}$) KB $\mathcal{K}$ and meta-query Q without non-distinguished variables.*

5 Discussion and conclusions

In order to consume the real-world and complex OWL 2 Full ontologies, in this paper, we propose an expressive sub-language of OWL 2 Full, called Hi($\mathcal{SROIQ}$), and provide a sound and complete way of realizing satisfiability checking and query answering in Hi($\mathcal{SROIQ}$) as well as prove the complexity of reasoning in Hi($\mathcal{SROIQ}$). Our future work will mainly focus on optimizing the procedure of reasoning in Hi($\mathcal{SROIQ}$). As Theorems 1 and 3 indicate, reasoning with a Hi($\mathcal{SROIQ}$) KB may need to consider an exponential number of $\mathcal{SROIQ}$ KBs w.r.t. the number of individuals of this KB. Heuristics shall be devised to reduce the number of such $\mathcal{SROIQ}$ KBs. On the other hand, identifying fragments $\mathcal{L}$ of Hi($\mathcal{SROIQ}$) so that solely one DL KB needs to be considered when reasoning and querying an $\mathcal{L}$ KB would be an interesting direction to explore. Besides, based on the work [29] which concentrates on optimizing SPARQL query answering in $\mathcal{SHOIQ}$, designing heuristics to optimize the procedure of meta-query answering in Hi($\mathcal{SROIQ}$) is also valuable and significant.

Acknowledgments

We thank the reviewers for their valuable suggestions and comments.This work has been supported by the National Key Research and Development Program of China under grants 2017YFC1700300, 2017YFB1002300 and 2016YFB1000902.

References

1. Hitzler, P., Krötzsch, M., Parsia, B., Patel-Schneider, P. F., Rudolph, S.: OWL 2 Web Ontology Language Primer (Second Edition), W3C Recommendation 11 December 2012.
2. Golbreich, C., Wallace, E K.: OWL 2 Web Ontology Language New Features and Rationale (Second Edition). W3C Recommendation 11 December 2012.
3. Schneider, M.: OWL 2 Web Ontology Language RDF-Based Semantics (Second Edition). W3C Recommendation 11 December 2012.
4. Chen, W., Kifer, M., Warren, D S.: HILOG: a foundation for higher-order logic programming. J. of Logic Programming, 15(3):187-230, 1993.
5. Motik, B.: On the Properties of Metamodeling in OWL. Journal of Logic and Computation, 17(4):617-637 (2007).
6. De Giacomo, G., Lenzerini, M., Rosati, R.: Higher-Order Description Logics for Domain Metamodeling. In: 25th Conference on Artificial Intelligence (2011).
7. Gu, Z.: Meta-modeling extension of Horn-SROIQ and Query Answering. In: 28th International Workshop on Description Logics (2016).
8. Gu, Z., Zhang, S.: Querying Large and Expressive Biomedical Ontologies. In: 17th IEEE International Conference on High Performance Computing and Communications (2015).
9. Gu, Z., Zhang, S.: The more irresistible Hi(SRIQ) for meta-modeling and meta-query answering. Frontiers of Computer Science, 12 (5): 1029-1031 (2018).
10. Lenzerini, M., Lepore, L., Poggi, A.: Answering Metaqueries over Hi(OWL 2 QL) Ontologies. In: IJCAI (2016).
11. Lenzerini, M., Lepore, L., Poggi, A.: Practical higher-order query answering over Hi(DL-Lite$_R$) knowledge bases. In: 27th International Workshop on Description Logics (2014).
12. Kubincová, P., Kl'uka, J., Homola, M.: Towards Expressive Metamodelling with Instantiation. In: 28th International Workshop on Description Logics DL (2015).
13. Pan, J. Z., Horrocks, I.: OWL FA: A Metamodeling Extension of OWL DL. In: 15th International Conference on World Wide Web (2006).
14. Homola, M., Kl'uka, J., Svátek, V., Vacura, M.: Typed higher-order variant of SROIQ - why not?. In: 27th International Workshop on Description Logics (2014).
15. Homola, M., Kl'uka, J., Svátek, V., Vacura, M.: Towards typed higher-order description logics. In: 26th International Workshop on Description Logics (2013).
16. Motz, R., Rohrer, E., Severi, P.: Reasoning for ALCQ extended with a flexible meta-modeling hierarchy. In: 4th Joint International Semantic Technology Conference (2014).
17. Motz, R., Rohrer, E., Severi, P.: The description logic SHIQ with a flexible meta-modeling hierarchy. Web Semantics: Science, Services and Agents on the World Wide Web, 35: 214-234 (2015).
18. Glimm, B., Rudolph, S., Völker, J.: Integrated metamodeling and diagnosis in OWL 2. In: 9th International Semantic Web Conference (2010).
19. Horrocks, I., Sattler, U.: Atableau decision procedure for SHOIQ. Journal of Automated Reasoning, 39 (3): 249-276 (2007).
20. Horrocks, I., Sattler, U., Tobies, S.: Reasoning with individuals for the description logic SHIQ. In: 17th Conference on Automated Deduction (2000).
21. Calvanese, D., De Giacomo, G., Lembo, D., Lenzerini, M., Rosati, R.: Tractable reasoning and efficient query answering in description logics: The DL-Lite family. Journal of automated reasoning, 39 (3): 385-429, 2007.

22. Horrocks, I., Kutz, O., Sattler, U.: The irresistible $\mathcal{SRIQ}$. In OWL: Experiences and Directions (2005).
23. Ortiz, M., Rudolph, S., Šimkus, M.: Query Answering in the Horn Fragments of the Description Logics SHOIQ and SROIQ. In: 22th International Joint Conference on Artificial Intelligence (2011).
24. Horrocks, I., Kutz, O., Sattler, U.: The Even More Irresistible $\mathcal{SROIQ}$. In: 10th International Conference on Principles of Knowledge Representation and Reasoning (2006).
25. Horrocks, I., Sattler, U., Tobies, S.: Practical Reasoning for Expressive Description Logics. In: 6th International Conference on Logic for Programming and Automated Reasoning (1999).
26. Horrocks, I., Sattler, U.: Decidability of SHIQ with complex role inclusion axioms. Artificial Intelligence, 160 (1-2): 79-104 (2004).
27. Kazakov, Y.: $\mathcal{RIQ}$ and $\mathcal{SROIQ}$ are Harder than $\mathcal{SHOIQ}$. In: 17th International Conference on Principles of Knowledge Representation and Reasoning (2008).
28. Motik, B., Grau, B. C., Horrocks, I., Wu, Z., Fokoue, A., Lutz, C.: OWL 2 Web Ontology Language Profiles (Second Edition). W3C Recommendation 11 December 2012.
29. Kollia, I., Glimm, B.: Optimizing SPARQL Query Answering over OWL Ontologies. Journal of Artificial Intelligence Research, 48 (2013): 253-303, 2013.