Inferring with Inconsistent OWL DL Ontology: a Multi-valued Logic Approach *

Yue Ma, Zuoquan Lin, Zhangang Lin

Department of Information Science, Peking University, Beijing, China
{mayue,lz,zglin}@is.pku.edu.cn

Abstract. Web ontology language OWL DL has two-valued model theory semantics so that ontologies expressed by it become trivial when contradictions occur. Based on classical description logic $\mathcal{SHOIN}(\mathcal{D})$, we propose the four-valued description logic $\mathcal{SHOIN}(\mathcal{D})4$ which has the ability to reason with inconsistencies. By transformation technic, we convert the reasoning problems of $\mathcal{SHOIN}(\mathcal{D})4$ to the counterparts of $\mathcal{SHOIN}(\mathcal{D})$. So $\mathcal{SHOIN}(\mathcal{D})4$ provides us with an approach to deal with contradictions by classical reasoning mechanism.

1 Introduction

The semantic web which is full of semantic information makes computers process information automatically. Kinds of standard semantic web languages provided by W3C[1], such as OWL DL and OWL Lite [1], are based on a rigorous logic basis — description logic which proves to be very useful for defining, integrating and maintaining ontologies [2]. Among the family of description logics is $\mathcal{SHOIN}(\mathcal{D})$ which is very close to OWL DL [3].

Description logics inherit the triviality from first order logic, that is, a single contradiction in the knowledge base leads to the only trivial logic consequence which includes everything. Therefore, a description logic knowledge base is ill when inconsistent. Considering a fragment of an ontology in medical treatment [4]: the one in surgical team does not belong to the team permitted to read patient's private record, while the one in urgency team does. We can express the knowledge by $\mathcal{SHOIN}(\mathcal{D})$ as follows:

$$SurgicalTeam \sqsubseteq \neg ReadPatientRecordTeam$$
$$UrgencyTeam \sqsubseteq ReadPatientRecordTeam$$

When we know the fact that $john$ belongs both to $SurgicalTeam$ and to $UrgencyTeam$, we find that there is a contradiction about whether $john$ is allowed to read patient's record. Under two-valued semantics, this knowledge base has no model so that anything can be deduced from it, even irrelative information like $Patient(john)$.

* This work is supported by NSFC(grant number 60496322).

[1] http://www.w3c.org

Decomposing the connections between information of being true and information of being false, thus yielding an extended semantics for ontology language, is the approach adopted in this paper to deal with inconsistencies. Actually, our work is based on multi-valued logic, whose truth value set is extended so that we can assign the contradiction to an additional truth value denoting contradiction. In the literature, the theory which describes contradiction but is nontrivial is called paraconsistent logic [5–7]. The underlying idea of this paper is Belnap's four-valued logic [8, 9] , which proves basic and important both in multi-valued logic and in para-consistent logic.

Terminological logic is an early version of description logic. Patel-Schneider [10] has proposed four-valued semantics for a terminological logic system which provides a tractable inclusion by the weaker inference ability of four-valued logic. In [10], *structural subsumption algorithm* is used to compute inclusion relation between classes (concepts), which is the first generation reasoning system of DLs and cannot treat complex constructors, such as disjunction ($\sqcup$), full negation ($\neg$), and full existence restriction ($\exists R.C$). Therefore, the language studied in [10] does not include these constructors which are important for OWL DL. Moreover, the semantics of inclusion has direct effect on complexity, thus defining more kinds of inclusion relations is difficult in [10]. In this paper, we propose a kind of four-valued semantics for all of these constructors as well as a complete algorithm for reasoning with $\mathcal{SHOIN}(\mathcal{D})4$ in a framework of two-valued $\mathcal{SHOIN}(\mathcal{D})$.

There are three main approaches to deal with inconsistent ontologies. The first is to reason with one(several) consistent subset(s) selected according to some principles, such as syntax/semantics relevance principle [11] and priority principle [4]. The second is to diagnose and repair contradictions when encountered. The third is through non-classical reasoning theory under new semantics. In this paper, we extend propositional four-valued semantics to ontology languages, thus forming $\mathcal{SHOIN}(\mathcal{D})4$ which is a four-valued version of $\mathcal{SHOIN}(\mathcal{D})$. The underlying idea is that we value the whole original theory instead of only choosing some sub-theory to take part in reasoning. However, ours is different from the third method in that we propose the decomposition of four-valued semantics to the two-valued, whereby existing reasoning systems for OWL DL remain useful for $\mathcal{SHOIN}(\mathcal{D})4$. $\mathcal{SHOIN}(\mathcal{D})4$ includes all the constructors of $\mathcal{SHOIN}(\mathcal{D})$ so that it can be used as an ontology language which is compatible with OWL DL but has the ability to deal with inconsistencies.

In the rest, we first briefly review description logic $\mathcal{SHOIN}(\mathcal{D})$ and Belnap's four-valued $\mathcal{FOUR}$. Then we describe $\mathcal{SHOIN}(\mathcal{D})4$ in details in section 3, and prove its inference can be reduced to that of $\mathcal{SHOIN}(\mathcal{D})$ in section 4. At last, we conclude this paper, compare it with related work, and point out our future work.

2 Description Logic and Four-valued Logic

2.1 OWL DL and Description Logic $\mathcal{SHOIN}(\mathcal{D})$

OWL DL is a subset of ontology web language OWL that has close relation with $\mathcal{SHOIN}(\mathcal{D})$. The main semantic relationship for OWL DL is entailment between pairs of OWL ontologies. An ontology O_1 entails an ontology O_2, written $O_1 \models O_2$, if and only if all interpretations that satisfy O_1 also satisfy O_2 [1]. Moreover, the OWL DL entailment can be transformed into $\mathcal{SHOIN}(\mathcal{D})$ knowledge base (un)satisfiability [3].

Generally, a description logic system includes: the set of concept and role constructors, inclusion assertions in TBox, fact assertions in ABox, and reasoning mechanism on TBox and ABox. The semantics of $\mathcal{SHOIN}(\mathcal{D})$ is given by means of an interpretation $I = (\Delta^I, \cdot^I)$ consisting of a non-empty domain Δ^I, disjoint from the datatype (or concrete) domain Δ^I_D, and a mapping $\cdot^I$, which interprets atomic and complex concepts, roles, and nominals according to Table 1 [3]. All the axiom forms contained in TBox and ABox of $\mathcal{SHOIN}(\mathcal{D})$ are also shown in Table 1. An interpretation satisfies a knowledge base K iff it satisfies each axiom in K; K is satisfiable (unsatisfiable) iff there exists (does not exist) such an interpretation.

Table 1. Syntax and Semantics of $\mathcal{SHOIN}(\mathcal{D})$

Constructor Name	Syntax	Semantics
atomic concept A	A	$A^I \subseteq \Delta^I$
datatypes D	D	$D^{\mathrm{D}} \subseteq \Delta^I_{\mathrm{D}}$
abstract role R_A	R	$R^I \subseteq \Delta^I \times \Delta^I$
datatype role R_D	U	$U^I \subseteq \Delta^I \times \Delta^I_{\mathrm{D}}$
individuals I	o	$o^I \in \Delta^I$
data values	v	$v^I = v^{\mathrm{D}}$
inverse role	R^-	$(R^-)^I \subseteq \Delta^I \times \Delta^I$
top concept	$\top$	Δ^I
bottom concept	$\bot$	$\emptyset$
conjunction	$C_1 \sqcap C_2$	$C^I \cap D^I$
disjunction	$C_1 \sqcup C_2$	$C^I \cup D^I$
negation	$\neg C$	$\Delta^I \setminus C^I$
oneOf	$\{o_1, ...\}$	$\{o_1^1, ...\}$
exists restriction	$\exists R.C$	$\{x \mid \exists y, (x,y) \in R^I \wedge y \in C^I\}$
value restriction	$\forall R.C$	$\{x \mid \forall y, (x,y) \in R^I \rightarrow y \in C^I\}$
atleast restriction	$\geq n.R$	$\{x \mid card(\{y.(x,y) \in R^I\}) \geq n\}$
atmost restriction	$\leq n.R$	$\{x \mid card(\{y.(x,y) \in R^I\}) \leq n\}$
datatype exists	$\exists U.D$	$\{x \mid \exists y, (x,y) \in U^I \wedge y \in D^I\}$
datatype value	$\forall U.D$	$\{x \mid \forall y, (x,y) \in U^I \rightarrow y \in D^I\}$
datatype atleast	$\geq n.U$	$\{x \mid card(\{y.(x,y) \in U^I\}) \geq n\}$
datatype atmost	$\leq n.U$	$\{x \mid card(\{y.(x,y) \in U^I\}) \leq n\}$
datatype oneOf	$\{v_1, ...\}$	$\{v_1^I, ...\}$
Axiom Name	**Syntax**	**Semantics**
concept inclusion	$C_1 \sqsubseteq C_2$	$C_1^I \subseteq C_2^I$
object role inclusion	$R_1 \sqsubseteq R_2$	$R_1^I \subseteq R_2^I$
object role transitivity	Trans(R)	$R^I = (R^I)^+$
datatype role inclusion	$U_1 \sqsubseteq U_2$	$U_1^I \subseteq U_2^I$
individual inclusion	$a : C$	$a^I \in C^I$
individual equality	$a = b$	$a^I = b^I$
individual inequality	$a \neq b$	$a^I \neq b^I$

2.2 Bilattice and Four-valued Logic

For a given $Domain$, $(\{< P, N >\}, \leq_k, \leq_t)$ constructs a bilattice space [12], where P and N are subsets of $Domain$ which stand for the information set of being true and of being false, respectively; and where the two partial orders $\leq_k$ and $\leq_t$ reflect differences in the amount of *truth* and the amount of *information*, respectively. The logical operators on the bilattice are defined as follows:

- Negation($\neg$) on direction $\leq_t$: $\neg < P, N > = < N, P >$
- Lower bound($\wedge$) and upper bound($\vee$) on direction $\leq_t$:

$$< P_1, N_1 > \wedge < P_2, N_2 >=< P_1 \cap P_2, N_1 \cup N_2 >$$

$$< P_1, N_1 > \vee < P_2, N_2 >=< P_1 \cup P_2, N_1 \cap N_2 >$$

The negation as well as lower and upper bounds on direction $\leq_k$ are also defined by Fitting in [12], but the above is enough for this paper since we only consider logic constructors in truth direction $\leq_t$.

Belnap's four-valued logic $\mathcal{FOUR}$ [8, 9, 13], whose truth value set is $FOUR = \{t, f, \top, \bot\}$ (also written as $\{t\}$, $\{f\}$,$\{t, f\}$, and $\emptyset$, respectively), is a special bilattice logic. The designated set of $\mathcal{FOUR}$ is $\{t, \top\}$ and three kinds of implications of it are material implication($\mapsto$), internal implication($\supset$) and strong implication($\rightarrow$) defined as follows [13, 14]:

$$\varphi \mapsto \psi \stackrel{\text{def}}{=} \neg\varphi \vee \psi.$$

$$\varphi \supset \psi \stackrel{\text{def}}{=} \begin{cases} \psi & \text{if } \varphi \in \{t, \top\}, \\ t & \text{if } \varphi \in \{f, \bot\}. \end{cases}$$

$$\varphi \rightarrow \psi \stackrel{\text{def}}{=} (\varphi \supset \psi) \wedge (\neg\psi \supset \neg\varphi).$$

$$\varphi \leftrightarrow \psi \stackrel{\text{def}}{=} (\varphi \rightarrow \psi) \wedge (\psi \rightarrow \varphi).$$

Note that, exception could occur for material implication, while it is not the case for internal and strong implications, since $\varphi \mapsto \psi = \top$ still holds when $\varphi = \top$ and $\phi \in \{f, \bot\}$. Intuitively, it is the contradictions in the precondition that bring exceptions for material implication, that is, material implication tolerates the situation that the conclusion is not true (valuing f or $\bot$) when we have information asserting the truth of the precondition (valuing $\top$ which includes truth information). For the other two implications, conclusions must be true when the preconditions are true. Therefore, internal and strong implications cannot characterize exceptions. Furthermore, when we lack the information about precondition, i.e. its truth value is $\bot$, the conclusion of material implication must value t or $\top$ which means it has information of being true; the conclusion of strong implication should value f or $\bot$, which means we lack information of it being true; the conclusion of internal implication accepts any truth value of $FOUR$. However, internal implication corresponds to the basic consequence of the four-valued logic as the following proposition says:

Proposition 1 *[14]*

- $\Gamma, \psi \models^4 \phi, \Delta$ *iff* $\Gamma \models^4 \psi \supset \phi, \Delta$.
- *If* $\Gamma \models^4 \psi, \Gamma \models^4 \psi \supset \phi$, *then* $\Gamma \models^4 \phi$.

The following counterexamples show that material and strong implications don't have the above property:

- $\{\psi, \neg\psi, \neg\phi\} \models^4 \psi \mapsto \phi$, but $\{\psi, \neg\psi, \neg\phi\} \not\models^4 \phi$.
- $\{\psi, \phi, \neg\phi\} \models^4 \phi$, but $\{\phi, \neg\phi\} \not\models^4 \psi \rightarrow \phi$.

Strong implication characterizes a class of stricter implication relationship: on one hand, when there is information of being true about the precondition, its conclusion must have information of being true; on the other hand, when there is information of being false about the conclusion, its precondition must have information of being false. The following proposition shows that the four-valued equality between two formulas can be defined through strong implication instead of material and internal implications.

Proposition 2 *[14] For every schemata* $\Theta, \psi \leftrightarrow \phi \models^4 \Theta(\psi) \leftrightarrow \Theta(\phi)$.

3 Four-valued Description Logic $\mathcal{SHOIN}(\mathcal{D})4$

3.1 Syntax

The inclusion axiom in $TBox$ characterizes human's exact knowledge of concept classifications. For example, $surgeon \sqsubseteq doctor$ means "whenever an instance is a surgeon, he/she must be a doctor". Salem et al. [4] declare that there are three types of information in a knowledge base (KB for short): facts, assertions without exception, and assertions with exception. Without distinguishing different information, contradictions easily occur in KB. Consider the following KB: (1) Generally, the person who is not a stuff of the hospital is not allowed to check patient's record; (2) However, the person who is doing temporary study practices in the hospital is generally allowed to do so. For this KB, some graduate of a medicine college may become an exception of the first axiom of KB — that is, although he/she is not a stuff of the hospital, he/she has the permission to read patient's record. The inclusion described in $\mathcal{SHOIN}(\mathcal{D})$ is exact knowledge without exceptions.

The concept constructors and fact axioms in $\mathcal{SHOIN}(\mathcal{D})4$ are the same as those in $\mathcal{SHOIN}(\mathcal{D})$. In addition, three kinds of inclusion axioms, denoted by $C \mapsto D$, $C \sqsubset D$, and $C \rightarrow D$ called material inclusion, internal inclusion and strong inclusion respectively, are defined in $\mathcal{SHOIN}(\mathcal{D})4$. These three inclusions are corresponding to the three implications in four-valued logic $\mathcal{FOUR}$. The first allows exceptions, and the other two do not. These three subsumptions help us to describe various class hierarchies. The exactnesses expressed by them increase one by one. For example, "$Bird \mapsto Fly$" means that birds can fly with exceptions, that is, there may be some bird which cannot fly; "$Bird \sqsubset Fly$" means that every bird must can fly. Note that, if we have some information indicates that some bird cannot fly, this implication still cannot tell us whether it is not a bird; "$Bird \rightarrow Fly$" means that an instance can fly whenever it is known to be a bird. Moreover, it can not be a bird if we know it cannot fly. Similarly, three kinds of inclusion axioms both of object and of datatype roles are defined in $\mathcal{SHOIN}(\mathcal{D})4$.

Knowledge with different exactness surely exists in human mind. So does it in the Semantic Web. However, all standard ontology languages for semantic web don't consider it. The main goal of this paper is to propose $\mathcal{SHOIN}(\mathcal{D})4$ which provides us with a way to characterize them.

3.2 Semantics

Generally speaking, there are four situations describing whether an individual is an instance of a concept: we surely know it is an instance of the concept; we surely know it is not an instance of the concept; we neither know it is an instance of the concept nor not (the situation of lacking information); or we have data indicating both it is an instance of the concept and not (the contradictory situation). So we define the semantic of $\mathcal{SHOIN}(\mathcal{D})4$ concepts by bilattice in this subsection.

For any given domain Δ and a concept C, we assign C an extended truth value $< P, N >$, where P is the subset of Δ that supports C to be true and N is the subset of Δ that supports C to be false. Cancelling the requirements $P \cap N = \emptyset$ and $P \cup N = \Delta$ in classical semantic conditions of $\mathcal{SHOIN}(\mathcal{D})$, an extended semantics forms and we will show that inconsistencies and uncertainty can be properly handled under this semantics.

For brevity, we first define positive projecting operator and negative projecting operator as follows:

Definition 1 *$proj^+(\cdot)$ and $proj^-(\cdot)$ are respectively positive projecting operator and negative projecting operator on bilattice space $(\{< P, N >\}, \leq_k, \leq_t)$, such that for any $< P, N >$,*

$$proj^+(< P, N >) = P;$$

$$proj^-(< P, N >) = N.$$

Definition 2 *A four-valued interpretation $I = (\Delta^I, \cdot^I)$ of $\mathcal{SHOIN}(\mathcal{D})4$ includes an object domain Δ^I, a datatype domain Δ^I_D, and a function $\cdot^I$ which satisfies all the interpretation requirements as shown in Table 2. (In Table 2, $\wedge$ and $\vee$ are the lower and upper bound of bilattice on direction $\leq_t$, respectively. $\sharp$ stands for set cardinality.)*

The following definition indicates why we use the name "four-valued interpretation" in definition 2.

Definition 3 *For any given instance $a, b \in \Delta^I$, concept name C and object/datatype role name R:*

- $C^I(a) = t$, *iff* $a^I \in proj^+(C^I)$ *and* $a^I \notin proj^-(C^I)$;
- $C^I(a) = f$, *iff* $a^I \notin proj^+(C^I)$ *and* $a^I \in proj^-(C^I)$;
- $C^I(a) = \top$, *iff* $a^I \in proj^+(C^I)$ *and* $a^I \in proj^-(C^I)$;
- $C^I(a) = \bot$, *iff* $a^I \notin proj^+(C^I)$ *and* $a^I \notin proj^-(C^I)$.
- $R^I(a, b) = \top$, *iff* $(a^I, b^I) \in proj^+(R^I)$ *and* $(a^I, b^I) \in proj^-(R^I)$;
- $R^I(a, b) = f$, *iff* $(a^I, b^I) \notin proj^+(R^I)$ *and* $(a^I, b^I) \in proj^-(R^I)$;

Table 2. Syntax and Semantics of $\mathcal{SHOIN}(\mathcal{D})4$

Constructor Syntax	Semantics
A	$A^I = < P, N >$, where $P, N \subseteq \Delta^I$
D	$D^{\mathrm{D}} \subseteq \Delta^I_{\mathrm{D}}$
R	$R^I = < P_1 \times P_2, N_1 \times N_2 >$, where $P_i, N_i \subseteq \Delta^I$ for $i = 1, 2$
U	$U^I = < P_1 \times P_2, N_1 \times N_2 >$, $P_i \in \Delta^I, N_i \subseteq \Delta^I_D$ for $i = 1, 2$
o	$o^I \in \Delta^I$
v	$v^I = v^{\mathrm{D}}$
R^-	$(R^-)^I = (R^I)^-$
$\top$	$< \Delta^I, \emptyset >$
$\bot$	$< \emptyset, \Delta^I >$
$C_1 \sqcap C_2$	$C^I \wedge D^I$
$C_1 \sqcup C_2$	$C^I \vee D^I$
$\neg C$	$(\neg C)^I = < N, P >$,$C^I = < P, N >$
$\{o_1, ...\}$	$< \{o^I_1, ...\}, N >$
$\exists R.C$	$< \{x \mid \exists y, (x, y) \in proj^+(R^I) \wedge y \in proj^+(C^I)\}$, $\{x \mid \forall y, (x, y) \in proj^+(R^I) \Rightarrow y \in proj^-(C^I)\} >$
$\forall R.C$	$< \{x \mid \forall y, (x, y) \in proj^+(R^I) \Rightarrow y \in proj^+(C^I)\}$, $\{x \mid \exists y, (x, y) \in proj^+(R^I) \wedge y \in proj^-(C^I)\} >$
$\geq n.R$	$< \{x \mid \#(y.(x, y) \in proj^+(R^I)) \geq n\}$, $\{x \mid \#(y.(x, y) \notin proj^-(R^I)) < n\} >$
$\leq n.R$	$< \{x \mid \#(y.(x, y) \notin proj^-(R^I)) \leq n\}$, $\{x \mid \#(y.(x, y) \in proj^+(R^I)) > n\} >$
$\exists U.D$	$< \{x \mid \exists y, (x, y) \in proj^+(U^I) \wedge y \in D^I\}$, $\{x \mid \forall y, (x, y) \in proj^-(U^I) \Rightarrow y \in D^I\} >$
$\forall U.D$	$< \{x \mid \forall y, (x, y) \in proj^+(U^I) \Rightarrow y \in D^I\}$, $\{x \mid \exists y, (x, y) \in proj^-(U^I) \wedge y \in D^I\} >$
$\geq n.U$	$< \{x \mid \#(y.(x, y) \in proj^+(U^I)) \geq n\}$, $\{x \mid \#(y.(x, y) \notin proj^-(U^I)) < n\} >$
$\leq n.U$	$< \{x \mid \#(y.(x, y) \notin proj^-(U^I)) \leq n\}$, $\{x \mid \#(y.(x, y) \in proj^+(U^I)) > n\} >$
oneOf $\{v_1, ...\}$	$\{v^I_1, ...\}$

- $R^I(a, b) = t$, *iff* $(a^I, b^I) \in proj^+(R^I)$ *and* $(a^I, b^I) \notin proj^-(R^I)$;
- $R^I(a, b) = \bot$, *iff* $(a^I, b^I) \notin proj^+(R^I)$ *and* $(a^I, b^I) \notin proj^-(R^I)$;

where, $t, f, \top, \bot$ *are truth values of four-valued logic.*

The semantics of material inclusion axioms, internal inclusion axioms, and strong inclusion axioms, as shown in Table 3, are corresponding to the sematics of material implication, internal implication, and strong implication in four-valued logic, respectively.

A four-valued interpretation I satisfies a $\mathcal{SHOIN}(\mathcal{D})4$ knowledge base $\mathcal{K}$ iff it satisfies each axiom in $\mathcal{K}$. $\mathcal{K}$ is satisfiable (unsatisfiable) iff there exists (does not exist) such an interpretation.

Table 3. Syntax and Semantics of axioms in $\mathcal{SHOIN}(\mathcal{D})4$

Axiom Name	Syntax	Semantics
concept material inclusion	$C_1 \mapsto C_2$	$\Delta^I \setminus proj^-(C_1^I) \subseteq proj^+(C_2^I)$
concept internal inclusion	$C_1 \sqsubset C_2$	$proj^+(C_1^I) \subseteq proj^+(C_2^I)$
concept strong inclusion	$C_1 \rightarrow C_2$	$proj^+(C_1^I) \subseteq proj^+(C_2^I)$ and $proj^-(C_2^I) \subseteq proj^-(C_1^I)$
object role material inclusion	$R_1 \mapsto R_2$	$\Delta^I \times \Delta^I \setminus proj^+(R_1^I) \subseteq proj^+(R_2^I)$
object role internal inclusion	$R_1 \sqsubset R_2$	$proj^+(R_1^I) \subseteq proj^+(R_2^I)$
object role strong inclusion	$R_1 \rightarrow R_2$	$proj^+(R_1^I) \subseteq proj^+(R_2^I)$ and $proj^-(R_2^I) \subseteq proj^-(R_1^I)$
datatype role material inclusion	$U_1 \mapsto U_2$	$\Delta^I \times \Delta_D^I \setminus proj^+(U_1^I) \subseteq proj^+(U_2^I)$
datatype role internal inclusion	$U_1 \sqsubset U_2$	$proj^+(U_1^I) \subseteq proj^+(U_2^I)$
datatype role strong inclusion	$U_1 \rightarrow U_2$	$proj^+(U_1^I) \subseteq proj^+(U_2^I)$ and $proj^-(U_2^I) \subseteq proj^-(U_1^I)$
object role transitivity	Trans(R)	$R^I = (R^I)^+$
individual inclusion	$a : C$	$a^I \in proj^+(C^I)$
individual equality	$a = b$	$a^I = b^I$
individual inequality	$a \neq b$	$a^I \neq b^I$

For an interpretation $I = (\Delta^I, \Delta_D^I, \cdot^I)$ of a $\mathcal{SHOIN}(\mathcal{D})4$ ontology, the semantics of an object concept C in it is some element, say $< P_0, N_0 >$, of the bilattice space $(< P, N >, \leq_t, \leq_k)$ which is formed based on Δ^I (i.e., $P, N \in \Delta^I$). If we restrict that $P_0 \cap N_0 = \emptyset$ and $P_0 \cup N_0 = \Delta^I$, then it is the classical two-valued semantics of C. The situation is the same for object role name and datatype role name. Therefore, the semantics of $\mathcal{SHOIN}(\mathcal{D})4$ is an extension of that of $\mathcal{SHOIN}(\mathcal{D})$.

In the following two propositions, we show that the semantics defined above has the similar intuition as the classical two-valued semantics.

Proposition 3 *Let C, D be concepts. For any $\mathcal{SHOIN}(\mathcal{D})4$ interpretation I,*

$$(C \sqcap \top)^I = C^I, (C \sqcup \top)^I = \top^I,$$

$$(C \sqcap \bot)^I = \bot^I, (C \sqcup \bot)^I = C^I.$$

Proof. For any given interpretation $I = (\Delta^I, \cdot^I) \top^I =< \Delta^I, \emptyset >, \bot^I =< \emptyset, \Delta^I >$. Without loss of generalitysuppose $C^I =< P, N >$. By definition 2

$$(C \sqcap \top)^I = < P \cap \Delta^I, N \cup \emptyset > = < P, N > = C^I$$
$$(C \sqcup \top)^I = < P \cup \Delta^I, N \cap \emptyset > = < \Delta^I, \emptyset > = \top^I$$
$$(C \sqcap \bot)^I = < P \cap \emptyset, N \cup \Delta^I > = < \emptyset, \Delta^I > = \bot^I$$
$$(C \sqcup \bot)^I = < P \cup \emptyset, N \cap \Delta^I > = < P, N > = C^I. \square$$

Proposition 4 *Let C, D be concepts, R be an object role name or a datatype role name. For any $\mathcal{SHOIN(D)}4$ interpretation I,*

$$(\neg\neg C)^I = C^I, (\neg\top)^I = \bot^I, (\neg\bot)^I = \top^I,$$
$$(\neg(C \sqcup D))^I = (\neg C \sqcap \neg D)^I, (\neg(C \sqcap D))^I = (\neg C \sqcup \neg D)^I,$$
$$(\neg(\forall R.C))^I = (\exists R.\neg C)^I, (\neg(\exists R.C))^I = (\forall R.\neg C)^I,$$
$$(\neg(\geq n.R))^I = (< n.R)^I, (\neg(\leq n.R))^I = (> n.R)^I.$$

Proof. For any interpretation $I = (\Delta^I, \cdot^I)$,$\top^I =< \Delta^I, \emptyset >, \bot^I =< \emptyset, \Delta^I >$ Without loss of generalitysuppose $C^I =< P, N >, D^I =< P', N' >$. By definition 2, the first three formulae hold obviously. Since

$$(\neg(C \sqcup D))^I = \neg < P \cup P', N \cap N' >=< N \cap N', P \cup P' >,$$
$$(\neg C \sqcap \neg D)^I =< N, P > \wedge < N', P' >=< N \cap N', P \cup P' > .$$

$(\neg(C \sqcup D))^I = (\neg C \sqcap \neg D)^I$.

$(\neg(C \sqcap D))^I = (\neg C \sqcup \neg D)^I$ follows in the same way.

Note that $proj^+(C^I) = proj^-((\neg C)^I) = P, proj^-(C^I) = proj^+((\neg C)^I) = N$. By definition 2

$$\begin{aligned}(\neg(\forall R.C))^I = \neg &< \{x \mid \forall y, (x, y) \in proj^+(R^I) \Rightarrow y \in proj^+(C^I)\},\\ &\{x \mid \exists y, (x, y) \in proj^+(R^I) \wedge y \in proj^-(C^I)\} >\\ =&< \{x \mid \exists y, (x, y) \in proj^+(R^I) \wedge y \in proj^-(C^I)\},\\ &\{x \mid \forall y, (x, y) \in proj^+(R^I) \Rightarrow y \in proj^+(C^I)\} >\\ =&< \{x \mid \exists y, (x, y) \in proj^+(R^I) \wedge y \in proj^+((\neg C)^I)\},\\ &\{x \mid \forall y, (x, y) \in proj^+(R^I) \Rightarrow y \in proj^-((\neg C)^I)\} >\\ =& (\exists R.\neg C)^I\end{aligned}$$

Therefore, $(\neg(\forall R.C))^I = (\exists R.\neg C)^I$.

By the same approach, we can prove that $(\neg(\exists R.C))^I = (\forall R.\neg C)^I$, $(\neg(\geq n.R))^I = (< n.R)^I$, and $(\neg(\leq n.R))^I = (> n.R)^I$. □

3.3 Expressivity of $\mathcal{SHOIN(D)}4$

We explain the expressivity of $\mathcal{SHOIN(D)}4$ by following examples.

Example 1 *Let knowledge base K be as follows*

$TBox = \exists hasPatient.Patient \sqsubset Doctor$. *(The one who has a patient must be a doctor)*
$ABox = \{Doctor(john), \neg Doctor(john), Patient(mary),$
$hasPatient(bill, mary)\}$.

Obviously, there is a contradiction in $ABox$. If it is a $\mathcal{SHOIN}(\mathcal{D})$ knowledge base, we can conclude anything from $\mathcal{K}$. But as a $\mathcal{SHOIN}(\mathcal{D})4$ knowledge base, we get positive answer to the query "is there any information indicating bill is a doctor?", since for each four-valued model of K, the following holds: $(bill, mary) \in proj^+(hasPatient^I)$, $mary \in proj^+(Patient^I), john \in proj^+(Doctor^I)$, and $john \in proj^-(Doctor^I)$. By definition 2, $bill \in proj^+((\exists hasPatient.Patient)^I)$, so $bill \in proj^+(Doctor^I)$. But we cannot get positive answer to the query " is there any information indicating bill is not a doctor?", since for the following model I of K,

$$Doctor^I =< \{john, bill\}, \{john\} >, Patient^I =< \{mary\}, \emptyset >,$$
$$hasPatient^I =< \{(bill, mary)\}, \emptyset >$$

we see that $bill \notin proj^-(Doctor^I)$.

Therefore, the $\mathcal{SHOIN}(\mathcal{D})4$ knowledge base can tolerate inconsistency without destroying useful inferences, that is it reasons para-consistently.

Example 2 *Let $\mathcal{K}$ be the following knowledge base*

$$TBox4 = \begin{cases} SurgicalTeam \sqsubset \neg ReadPatientRecordTeam \\ UrgencyTeam \sqsubset ReadPatientRecordTeam \end{cases}$$

$$ABox = \{SurgeicalTeam(john), UrgentTeam(john)\}.$$

The $\mathcal{SHOIN}(\mathcal{D})4$ knowledge base is satisfiable since it has a model as followings:

$$SurgicalTeam^I \in \{< \{john\}, \emptyset >, < \{john\}, \{john\} >\},$$
$$UrgencyTeam^I \in \{< \{john\}, \emptyset >, < \{john\}, \{john\} >\},$$
$$ReadPatientRecordTeam^I =< \{john\}, \{john\} > .$$

When queried "is there any information declining that john is allowed to read patient's record", it answers "yes" since $john \in proj^+(ReadPatientRecordTeam^I)$ for every models of $\mathcal{K}$; when queried "is there any information declining that john is not allowed to read patient's record", it answers "yes" because for every model of $\mathcal{K}$, $john \in proj^-(ReadPatientRecordTeam^I)$. However, when queried "is there any information declining that john is (not) a patient", it answers "no" since for some model of $\mathcal{K}$, $john \notin proj^{+(-)}(Patient^I)$.

In short, $\mathcal{SHOIN}(\mathcal{D})4$ gives the positive answers to both aspects of a contradiction, while remains other information not contrary. In this sense, $\mathcal{SHOIN}(\mathcal{D})4$ reflects system's information actually.

Let us consider an example including material and internal inclusion axioms:

Example 3 *We have the following knowledge: "generally speaking, the bird with a pair of swings can fly. Penguin is a kind of bird and has a pair of swings, but it cannot fly. $Tweety$ is a penguin with a pair of swings w." We can describe it by $\mathcal{SHOIN}(\mathcal{D})$ ontology ($TBox, ABox$) and $\mathcal{SHOIN}(\mathcal{D})4$ ontology ($TBox4, ABox$) as follows:*

$$TBox = \begin{cases} Bird \sqcap \exists hasWing.Wing \sqsubseteq Fly \\ Penguin \sqsubseteq Bird \\ Penguin \sqsubseteq \exists hasWing.Wing \\ Penguin \sqsubseteq \neg Fly \end{cases}$$

$$TBox4 = \begin{cases} Bird \sqcap \exists hasWing.Wing \mapsto Fly \\ Penguin \sqsubset Bird \\ Penguin \sqsubset \exists hasWing.Wing \\ Penguin \sqsubset \neg Fly \end{cases}$$

$$ABox = \{Bird(tweety), Penguin(tweety), Wing(w), hasWing(tweety, w)\}.$$

$\mathcal{K} = (TBox, ABox)$ is an unsatisfiable $\mathcal{SHOIN}(\mathcal{D})$ knowledge base from which everything follows. But $\mathcal{K}4 = (TBox4, ABox)$ is a satisfiable $\mathcal{SHOIN}(\mathcal{D})4$ knowledge base. Among its models is the following $I = (\{tweety, w\}, \cdot^I)$:

$Bird^I =< \{tweety\}, \{tweety\} >$, $Fly^I =< \emptyset, \{tweety\} >$, $Penguin^I =< \{tweety\}, \emptyset >$, $Wing^I =< \{w\}, \emptyset >$, $hasWing^I =< \{tweety\}, \{w\} >$.

Under this interpretation, $tweety \in proj^+(Bird^I) \cap proj^-(Bird^I)$, that is the value of $Bird^I(tweety)$ is $\top$. Similarly, $Fly^I(tweety) = f, Penguin^I(tweety) = t, Wing^I(w) = t, hasWing^I(tweety, w) = t$. We see that exceptions can be expressed by $\mathcal{SHOIN}(\mathcal{D})4$ system. We will continue to discuss the reasoning of this example in section 4.2.

Let us consider another example with number restriction constructor:

Example 4 *"The one who has at least one child is a parent. Generally speaking, parent is married. We have the fact that single Smith adopts a child Kate. " This is a possible ontology. But it can not be expressed by any classical OWL DL ontology language without contradiction. We can express it by $\mathcal{SHOIN}(\mathcal{D})4$ in a novel way:*

$$TBox = \begin{cases} \geq 1.hasChild \sqsubset Parent \\ Parent \mapsto Married \end{cases}$$

$$ABox = \{hasChild(smith, kate), \neg\, Married(smith)\}$$

This is a satisfiable $\mathcal{SHOIN}(\mathcal{D})4$ knowledge base. For example, the following is its models with domain $\{smith, kate\}$:

- M1-M4: $(\geq 1.hasChild)^I =< \{smith\}, \emptyset >, Married^I =< \{smith\}, \{smith\} >$, $hasChild^I =< \{(smith, kate)\}, \emptyset > or < \{(smith, kate)\}, \{(smith, kate)\} >$, $Parent^I =< \{smith\}, \emptyset > or < \{smith\}, \{smith\} >$;
- M5-M6: $(\geq 1.hasChild)^I =< \{smith\}, \emptyset >, Parent^I =< \{smith\}, \{smith\} >$, $hasChild^I =< \{(smith, kate)\}, \emptyset\} > or < \{(smith, kate)\}, \{(smith, kate)\} >$, $Married^I =< \emptyset, \{smith\} >$;
- M7-M8: $hasChild^I =< \{(smith, kate)\}, \{(smith, kate), (smith, smith)\} >$, $(\geq 1.hasChild)^I =< \{smith\}, \{smith\} >, Married^I =< \{smith\}, \{smith\} >$, $Parent^I =< \{smith\}, \emptyset > or < \{smith\}, \{smith\} >$;
- M9: $hasChild^I =< \{(smith, kate)\}, \{(smith, kate), (smith, smith)\} >$, $(\geq 1.hasChild)^I =< \{smith\}, \{smith\} >, Parent^I =< \{smith\}, \{smith\} >$, $Married^I =< \{smith\}, \emptyset >$;

The corresponding four-valued semantics of the above models are as shown in Table 4 (s is Smith for short and k is Kate for short).

Table 4. Four-valued Models of Example 4

	$hasChild(s,k)$	$\geq 1.hasChild(s)$	$Parent(s)$	$Married(s)$
M1-M4	$t/\top$	t	$t/\top$	$\top$
M5-M6	$t/\top$	t	$\top$	f
M7-M8	$\top$	$\top$	$t/\top$	$\top$
M9	$\top$	$\top$	$\top$	f

Since the role $hasChild$ is not reflexive (namely, no one will relate itself by this role), we declare that the semantics of $\mathcal{SHOIN}(\mathcal{D})4$ had better not refer to unreasonable interpretation like $hasChild(smith, smith)$ for nonreflexive roles. The effect of distinguishing reflex roles to DL systems is to be study.

4 Reducing $\mathcal{SHOIN}(\mathcal{D})4$ to $\mathcal{SHOIN}(\mathcal{D})$

Because we don't consider the four-valued semantics of datatype concepts, in the rest, we only denote an interpretation of $\mathcal{SHOIN}(\mathcal{D})4$ as $I = (\Delta^I, \cdot^I)$ instead of $I = (\Delta^I, \Delta^I_D, \cdot^I)$ for simpleness.

For any interpretation $I = (\Delta^I, \cdot^I)$ of $\mathcal{SHOIN}(\mathcal{D})4$ and any concept C, we can decide the semantics of C by the positive projection of C^I and $(\neg C)^I$ according to the equation $proj^+((\neg C)^I) = proj^-(C^I)$, although there's no relation between $proj^+(C^I)$ and $proj^-(C^I)$.

We introduce the following notations to characterize the relationship between four-valued and two-valued semantics.

Definition 4 *(Decomposability) The four-valued semantics of $\mathcal{SHOIN}(\mathcal{D})4$ can be decomposed into two-valued semantics of $\mathcal{SHOIN}(\mathcal{D})$, iff for any $\mathcal{SHOIN}(\mathcal{D})4$ knowledge base $\mathcal{K}$ and its concept C and object (datatype) role R, there is an two-valued $\mathcal{SHOIN}(\mathcal{D})$ knowledge base $\overline{\mathcal{K}}$ and its two concepts $\overline{C_1}$,$\overline{C_2}$ and two object (datatype) roles $\overline{R_1}$,$\overline{R_2}$ such that for any four-valued interpretation I of $\mathcal{K}$, there's a two-valued interpretation $\overline{I}$ of $\overline{\mathcal{K}}$, such that*

$$C^I = < P, N > \text{ iff } \overline{C_1}^{\overline{I}} = P, \overline{C_2}^{\overline{I}} = N.$$
$$R^I = < P_1 \times P_2, N_1 \times N_2 > \text{ iff } \overline{R_1}^{\overline{I}} = P_1 \times P_2, \overline{R_2}^{\overline{I}} = \Delta^I \times \Delta^I \setminus N_1 \times N_2.$$

where P, N, P_1, P_2, N_1 and N_2 are subsets of Δ^I.

The decomposability of $\mathcal{SHOIN}(\mathcal{D})4$ means that the four-valued semantics of concept C and role R can be divided into the two-valued semantics of two $\mathcal{SHOIN}(\mathcal{D})$ concepts $\overline{C_1}, \overline{C_2}$ and roles $\overline{R_1}, \overline{R_2}$. Arieli [15, 16] provides some techniques to reduce some models of four-valued logic to classical two-valued semantics. Yue [17] proposes a formula transformation technique to distinguish material implication and internal implication of four-valued logic. We will further study transformation technique to decompose the four-valued semantics of $\mathcal{SHOIN}(\mathcal{D})4$ in the next section. Furthermore, we will see that the decomposability of $\mathcal{SHOIN}(\mathcal{D})4$ enables the inference of $\mathcal{SHOIN}(\mathcal{D})4$ to be reduced to that of $\mathcal{SHOIN}(\mathcal{D})$.

4.1 Concept, Role and Axiom Transformations

Let $\mathcal{L}$ be a $\mathcal{SHOIN}(\mathcal{D})4$ language, $\mathcal{L} = \{C, R, a \mid C$ is a concept name, R is a role name, a is an individual$\}$. $\mathcal{A}(\mathcal{L})$ is set of atomic concepts of $\mathcal{L}$. $\overline{\mathcal{L}} = \{\overline{C}, \overline{\neg C}, R^+, R^=, \overline{a} \mid C, R, a \in \mathcal{L}, \overline{C}, \overline{\neg C}$ are the concept transformations of C and $\neg C$ respectively, $R^+, R^=$ are two role transformations of role R. $\overline{a}$ is the renamed name of individual a in $I\}$.

Concept transformation and role transformation of a $\mathcal{SHOIN}(\mathcal{D})4$ concept C and a role R are defined as follows:

Definition 5 *For any given concept $C \in \mathcal{L}$, $\overline{C} \in \overline{\mathcal{L}}$ is the concept transformation of C, such that*

(1) If $C = A, A \in \mathcal{A}(\mathcal{L})$, then $\overline{C} = A^+$;
(2) If $C = \neg A, A \in \mathcal{A}(\mathcal{L})$, then $\overline{C} = A^-$;
(3) If $C = \top$, then $\overline{C} = \top$;
(4) If $C = \bot$, then $\overline{C} = \bot$;
(5) If $C = E \sqcap D$, then $\overline{C} = \overline{E} \sqcap \overline{D}$;
(6) If $C = E \sqcup D$, then $\overline{C} = \overline{E} \sqcup \overline{D}$;
(7) If $C = \exists R.D$ where R is an object role or a datatype role, then $\overline{C} = \exists R^+.\overline{D}$;
(8) If $C = \forall R.D$ where R is an object role or a datatype role, then $\overline{C} = \forall R^+.\overline{D}$;
(9) If $C = \geq n.R$ where R is an object role or a datatype role, then $\overline{C} = \geq n.R^+$;
(10) If $C = \leq n.R$ where R is an object role or a datatype role, then $\overline{C} = \leq n.R^=$;
(11) If $C = \neg\neg D$, then $\overline{C} = \overline{D}$;
(12) If $C = \neg(E \sqcap D)$, then $\overline{C} = \overline{\neg E} \sqcup \overline{\neg D}$;
(13) If $C = \neg(E \sqcup D)$, then $\overline{C} = \overline{\neg E} \sqcap \overline{\neg D}$;
(14) If $C = \neg(\exists R.D)$ where R is an object role, then $\overline{C} = \forall R^+.\overline{\neg D}$;
(15) If $C = \neg(\forall R.D)$ where R is an object role, then $\overline{C} = \exists R^+.\overline{\neg D}$;
(16) If $C = \neg(\geq n.R)$ where R is an object role or a datatype role, then $\overline{C} = \leq (n-1).R^=$;
(17) If $C = \neg(\leq n.R)$ where R is an object role or a datatype role, then $\overline{C} = \geq (n+1).R^+$;
(18) If $C = \{o_1, ...\}$ where o_i is an individual, then $\overline{C} = \{\overline{o}_1, ...\}$;
(19) $(R^-)^+ = (R^+)^-, (R^-)^= = (R^=)^-$

Based on the concept and role transformations, we give the axiom transformations as follows:

Definition 6 *The transformation of axioms of $\mathcal{SHOIN}(\mathcal{D})4$ are defined as follows:*

(1) $\overline{C_1 \mapsto C_2} = \neg\overline{\neg C_1} \sqsubseteq \overline{C_2}$;
$\overline{C_1 \sqsubset C_2} = \overline{C_1} \sqsubseteq \overline{C_2}$;
$\overline{C_1 \to C_2} = \{\overline{C_1} \sqsubseteq \overline{C_2}, \overline{\neg C_2} \sqsubseteq \overline{\neg C_1}\}$.
where, $C_i (i = 1, 2)$ *is a concept.*

(2) $\overline{R_1 \mapsto R_2} = R_1^{=} \sqsubseteq R_2^{+}$;
$\overline{R_1 \sqsubset R_2} = R_1^{+} \sqsubseteq R_2^{+}$;
$\overline{R_1 \to R_2} = \{R_1^{+} \sqsubseteq R_2^{+}, R_1^{=} \sqsubseteq R_2^{=}\}$.
where, $R_i (i = 1, 2)$ *is an object role or a datatype role.*

(3) $\overline{Trans(R)} = \{Trans(R^{+})\}$
where, R *is an object role.*

(4) $\overline{a : C} = \overline{a} : \overline{C}$, $\overline{a = b} = \overline{a} = \overline{b}$, $\overline{a \neq b} = \overline{a} \neq \overline{b}$
where, a, b *are individuals,* C *is a concept.*

Definition 7 *(Classical Induced KB) We say the* classical induced KB *of any given* $\mathcal{SHOIN(D)}4$ *knowledge base* $\mathcal{K}$*, written* $\overline{\mathcal{K}}$*, if all axioms in* $\overline{\mathcal{K}}$ *are exactly the transformations of axioms in* $\mathcal{K}$*.*

Obviously, concept, role and axiom transformations can be finished in polynomial time.

4.2 $\mathcal{SHOIN(D)}4$ Reasoning

In this section we first show the decomposability of $\mathcal{SHOIN(D)}4$, and then prove that the standard reasoning problems of $\mathcal{SHOIN(D)}4$ can be reduced to those of classical $\mathcal{SHOIN(D)}$.

Definition 8 *(Classical Induced Interpretation) Let* $I = (\Delta^I, \cdot^I)$ *be an interpretation of* $\mathcal{SHOIN(D)}4$*, and* $\overline{\mathcal{K}}$ *be the classical induced KB of* $\mathcal{K}$*.* I'*s classical induced interpretation* $\overline{I} = (\Delta^{\overline{I}}, \cdot^{\overline{I}})$ *is defined as follows:*

- I *and* $\overline{I}$ *have the same domain, i.e.* $\Delta^{\overline{I}} = \Delta^I$;
- I *and* $\overline{I}$ *interpret instance names in the same way, i.e.* $\overline{a}^{\overline{I}} = a^I$;
- *For any atomic concept* A*, if* $A^I = < P, Q >$*, then* $(A^{+})^{\overline{I}} = P, (A^{-})^{\overline{I}} = Q$;
- *For any object or datatype role* R*, if* $R^I = < P_1 \times P_2, N_1 \times N_2 >$*, then* $(R^{+})^{\overline{I}} = P_1 \times P_2$*, and* $(R^{=})^{\overline{I}} = \Delta^{\overline{I}} \times \Delta^{\overline{I}} \setminus N_1 \times N_2$.

The semantics of complex concepts are obtained in the standard way.

Definition 9 *(Four-valued Induced Interpretation) Let* $\overline{I}$ *be the interpretation of an* $\mathcal{SHOIN(D)}$ *knowledge base* $\mathcal{K}$*,* $\overline{I}$*'s four-valued induced interpretation* $I = (\Delta^I, \cdot^I)$ *is defined as follows*

- I *and* $\overline{I}$ *have the same domain, i.e.* $\Delta^{\overline{I}} = \Delta^I$;
- I *and* $\overline{I}$ *interpret instance names in the same way, i.e.* $\overline{a}^{\overline{I}} = a^I$;
- *For any primitive concept* A*, if* $(A^{+})^{\overline{I}} = P, (A^{-})^{\overline{I}} = Q$*, then* $A^I = < P, Q >$;
- *For any object and datatype role* R*, if* $(R^{+})^{\overline{I}} = P_1 \times P_2$*, and* $(R^{=})^{\overline{I}} = Q_1 \times Q_2$*, then* $R^I = < P_1 \times P_2, \Delta^I \times \Delta^I \setminus Q_1 \times Q_2 >$.

The semantics of complex concepts are obtained according to definition 2.

From definitions 8 and 9, the classical induced KB of a $\mathcal{SHOIN}(\mathcal{D})4$ knowledge base $\mathcal{K}$ is two-valued theory, whose constructors are those of $\mathcal{SHOIN}(\mathcal{D})$. Therefore, we can change a $\mathcal{SHOIN}(\mathcal{D})4$ knowledge base into a $\mathcal{SHOIN}(\mathcal{D})$ one by transformation technique.

Lemma 5 *The semantics of $\mathcal{SHOIN}(\mathcal{D})4$ can be decomposed to two-valued semantics of $\mathcal{SHOIN}(\mathcal{D})$.*

Proof. Let $\mathcal{K}$ be a $\mathcal{SHOIN}(\mathcal{D})4$ knowledge base and C be a concept. For any interpretation I, we prove by structure induction that $C^I =< P, N >$ iff $\overline{C}^{\overline{I}} = P, \overline{\neg C}^{\overline{I}} = N$, where $\overline{I}$ is the Classical Induced Interpretation of I.

Case: C is an atomic concept A is easy by definition 9, 8.

Case: $C = \neg D$. $\overline{C} = \overline{\neg D}, \overline{\neg C} = \overline{D}$,

- Suppose $C^I =< P, N >$. Then $D^I =< N, P >$. By induction assumption, we know $\overline{D}^{\overline{I}} = N, \overline{\neg D}^{\overline{I}} = P$. That is $\overline{\neg C}^{\overline{I}} = N, \overline{C}^{\overline{I}} = P$.
- Whereas, suppose $\overline{C}^{\overline{I}} = P, \overline{\neg C}^{\overline{I}} = N$. Then $\overline{D}^{\overline{I}} = N, \overline{\neg D}^{\overline{I}} = P$. By induction assumption, we know $D^I =< N, P >$. Through the semantics of negation, we know $C^I =< P, N >$.

Case: $C = D \sqcup E$. $\overline{C} = \overline{D} \sqcup \overline{E}$, and $\overline{\neg C} = \overline{\neg D} \sqcap \overline{\neg E}$,

- Suppose $C^I =< P, N >$,$D^I =< P_1, N_1 >, E^I =< P_2, N_2 >$. Then $P_1 \cup P_2 = P, N_1 \cap N_2 = N$. By induction hypothesis, we know $\overline{D}^{\overline{I}} = P_1, \overline{\neg D}^{\overline{I}} = N_1, \overline{E}^{\overline{I}} = P_2$, and $\overline{\neg E}^{\overline{I}} = N_2$. Therefore $\overline{C}^{\overline{I}} = \overline{D}^{\overline{I}} \cup \overline{E}^{\overline{I}} = P_1 \cup P_2 = P$, and $\overline{\neg C}^{\overline{I}} = \overline{\neg D}^{\overline{I}} \cap \overline{\neg E}^{\overline{I}} = N_1 \cap N_2 = N$.
- Whereas, suppose $\overline{C}^{\overline{I}} = P, \overline{\neg C}^{\overline{I}} = N, \overline{D}^{\overline{I}} = P', \overline{\neg D}^{\overline{I}} = N'$, and $\overline{E}^{\overline{I}} = P'', \overline{\neg E}^{\overline{I}} = N''$. By the definition of semantics, $P = P' \cup P'', N = N' \cap N''$. By induction hypothesis, $D^I =< P', N' >, E^I =< P'', N'' >$. Therefore, $C^I =< P' \cup P'', N' \cap N'' >=< P, N >$ by definition of semantics of $\mathcal{SHOIN}(\mathcal{D})4$.

Case: $C = D \sqcap E$. the proposition holds likewise.

Case: $C = \forall R.D$. $\overline{C} = \forall R.\overline{D}$ and $\overline{\neg C} = \exists R.\overline{\neg D}$,

- Suppose $C^I =< P, N >$, $D^I =< P_1, N_1 >$. By semantics definition, we know $P = \{x \mid \forall y, R(x, y) \Rightarrow y \in proj^+(D^I)\}$, $N = \{x \mid \exists y, R(x, y) \wedge y \in proj^-(D^I)\}$. By induction hypothesis, $\overline{D}^{\overline{I}} = P_1$ and $\overline{\neg D}^{\overline{I}} = N_1$. Therefore, $N_1 = proj^-(D^I)$. (Note that $P_1 = proj^+(D^I)$)

$$\begin{aligned}
\overline{C}^{\overline{I}} &= (\forall R.\overline{D})^{\overline{I}} = \{x \mid \forall y, R(x, y) \Rightarrow y \in (\overline{D})^{\overline{I}}\} \\
&= \{x \mid \forall y, R(x, y) \Rightarrow y \in P_1\} = P, \\
\overline{\neg C}^{\overline{I}} &= (\exists R.\overline{\neg D})^{\overline{I}} = \{x \mid \exists y, R(x, y) \wedge y \in (\overline{\neg D})^{\overline{I}}\} \\
&= \{x \mid \exists y, R(x, y) \wedge y \in N_1\} = N.
\end{aligned}$$

– Whereas, suppose $\overline{C}^{\overline{I}} = P, \overline{\neg C}^{\overline{I}} = N, \overline{D}^{\overline{I}} = P', \overline{\neg D}^{\overline{I}} = N'$. By the definition of semantics,

$$P = \overline{C}^{\overline{I}} = (\forall R.\overline{D})^{\overline{I}} = \{x \mid \forall y, R(x,y) \Rightarrow y \in P'\},$$
$$N = \overline{\neg C}^{\overline{I}} = (\exists R.\overline{\neg D})^{\overline{I}} = \{x \mid \exists y, R(x,y) \wedge y \in N'\}.$$

By induction hypothesis, $D^I =< P', N' >$. Furthermore, by the semantics of $\mathcal{SHOIN(D)}4$, we know

$$C^I =< \{x \mid \forall y, R(x,y) \Rightarrow y \in P'\}, \{x \mid \exists y, R(x,y) \wedge y \in N'\} >=< P, N >$$

Case: $C = \exists R.D$. the lemma holds likewise.
Case: $C =\geq n.R$. $\overline{C} =\geq n.R^+$,$\overline{\neg C} =\leq (n-1).R^=$:

– Suppose $C^I =< P, N >, R^I =< P_1 \times P_2, N_1 \times N_2 >$. By definition 2,

$$\begin{aligned} P &= \{x \mid \sharp(y.(x,y) \in proj^+(R^I)) \geq n\} = \{x \mid \sharp(y.(x,y) \in P_1 \times P_2) \geq n\} \\ &= \{x \mid \sharp(y.(x,y) \in (R^+)^{\overline{I}}) \geq n\} = (\geq n.R^+)^{\overline{I}} = \overline{C}^{\overline{I}}, \\ N &= \{x \mid \sharp(y.(x,y) \notin proj^-(R^I)) < n\} \\ &= \{x \mid \sharp(y.(x,y) \in \Delta^I \times \Delta^I \setminus N_1 \times N_2) < n\} \\ &= \{x \mid \sharp(y.(x,y) \in (R^=)^{\overline{I}}) < n\} = (\leq (n-1).R^=)^{\overline{I}} = \overline{\neg C}^{\overline{I}}, \end{aligned}$$

Note that $(R^+)^{\overline{I}} = P_1 \times P_2, (R^=)^{\overline{I}} = \Delta^I \times \Delta^I \setminus N_1 \times N_2$ by definition 8.

– Whereas, Suppose $(\geq n.R^+)^{\overline{I}} = P, (\leq (n-1).R^=)^{\overline{I}} = N, (R^+)^{\overline{I}} = P_1 \times P_2, (R^=)^{\overline{I}} = \Delta_1 \times \Delta_2 \setminus N_1 \times N_2$. Then $P = \{x \mid \sharp(y.(x,y) \in P_1 \times P_2) \geq n\}, N = \{x \mid \sharp(y.(x,y) \notin N_1 \times N_2) < n\}$. By definition 9, $R^I =< P_1 \times P_2, N_1 \times N_2 >$. By definition 2,

$$\begin{aligned} C^I &= < \{x \mid \sharp(y.(x,y) \in P_1 \times P_2) \geq n\}, \{x \mid \sharp(y.(x,y) \notin N_1 \times N_2) < n\} > \\ &= < P, N > . \end{aligned}$$

Case: $C =\leq n.R$. the lemma can be proven in the same way.

In all, let $C_1 = \overline{C}, C_2 = \overline{\neg C}$, we see that for any concept C, $C^I =< P, N >$ iff $C_1^{\overline{I}} = P$ and $C_2^{\overline{I}} = N$.

For any role R, from definition 9 and 8, we can see that $R^I =< P_1 \times P_2, N_1 \times N_2 >$ iff $\overline{R_1}^{\overline{I}} = P_1 \times P_2, \overline{R_2}^{\overline{I}} = \Delta^I \times \Delta^I \setminus N_1 \times N_2$. □

Theorem 6 *The interpretation $I = (\Delta^I, \cdot^I)$ is a model of knowledge base $\mathcal{K}$ iff there is a model of $\overline{\mathcal{K}}$, say $\overline{I} = (\Delta^{\overline{I}}, \cdot^{\overline{I}})$, which is the classical induced knowledge base of $\mathcal{K}$.*

Proof. (Necessity) For any interpretation I of $\mathcal{K}$, let the interpretation $\overline{I}$ be I's classical induced interpretation. According to the relationship between $\overline{\mathcal{K}}$ and $\mathcal{K}$, for any $\overline{\mathcal{K}}$'s inclusion of the form $\overline{\neg\neg C} \sqsubseteq \overline{D} \in \overline{\mathcal{K}}, C \mapsto D \in \mathcal{K}$. Suppose $C^I =< P_1, N_1 >$, $D^I =< P_2, N_2 >$. By lemma 5, $\overline{\neg C}^{\overline{I}} = N_1, \overline{D}^{\overline{I}} = P_2$. Therefore, $(\neg\overline{\neg C})^{\overline{I}} = \Delta^{\overline{I}} \setminus N_1 = \Delta^I \setminus N_1$. I satisfies $C \mapsto D$. So $\Delta^I \setminus N_1 \subseteq P_2$. Therefore, $(\neg\overline{\neg C})^{\overline{I}} \subseteq \overline{D}^{\overline{I}}$. That is, $\overline{I}$ satisfies $\neg\overline{\neg C} \sqsubseteq \overline{D}$.

For any $\overline{\mathcal{K}}$'s inclusion of the form $\overline{C} \sqsubseteq \overline{D} \in \overline{\mathcal{K}}$, $\overline{\neg C} \sqsubseteq \overline{\neg D} \notin \overline{\mathcal{K}}$, $C \sqsubset D \in \mathcal{K}$. Suppose $C^I =< P_1, N_1 >$, $D^I =< P_2, N_2 >$. By lemma 5, $\overline{C}^{\overline{I}} = P_1, \overline{D}^{\overline{I}} = P_2$. I satisfies $C \sqsubset D$. Therefore, $P_1 = proj^+(C^I) \subseteq proj^+(D^I) = P_2$, that is $\overline{I}$ satisfies $\overline{C} \sqsubseteq \overline{D}$.

For any $\overline{K}$'s inclusion pair of the form $\{\overline{C} \sqsubseteq \overline{D}, \overline{\neg D} \sqsubseteq \overline{\neg C}\} \subseteq \overline{\mathcal{K}}$, $C \rightarrow D \in \mathcal{K}$. Suppose $C^I =< P_1, N_1 >, D^I =< P_2, N_2 >$. By lemma 5, $\overline{C}^{\overline{I}} = P_1, \overline{D}^{\overline{I}} = P_2, \overline{\neg C}^{\overline{I}} = N_1, \overline{\neg D}^{\overline{I}} = N_2$. I satisfies $C \rightarrow D$. Therefore, $P_1 = proj^+(C^I) \subseteq proj^+(D^I) = P_2, N_2 = proj^-(D^I) \subseteq proj^-(C^I) = N_1$, that is $\overline{I}$ satisfies $\{\overline{C} \sqsubseteq \overline{D}, \overline{\neg D} \sqsubseteq \overline{\neg C}\}$.

For any assertion of the form $\overline{a}{:}\overline{C}$. $a{:}C$ belongs to the $\mathcal{K}$. Suppose $C^I =< P, N >$, $a^I = \delta_0 \in \Delta^I$, then $(\overline{C})^{\overline{I}} = P, \overline{a}^{\overline{I}} = \delta_0$. Because I satisfies $a{:}C$, $\delta_0 \in P$, that is $\overline{I}$ satisfies $\overline{a}{:}\overline{C}$. Finally, since $a^I = (\overline{a})^{\overline{I}}$, $b^I = (\overline{b})^{\overline{I}}$, $(\overline{a})^{\overline{I}} = (\overline{b})^{\overline{I}}$ iff $a^I = b^I$, $(\overline{a})^{\overline{I}} \neq (\overline{b})^{\overline{I}}$ iff $a^I \neq b^I$.

For any $\overline{\mathcal{K}}$'s role inclusion $R_1^= \sqsubseteq R_2^+, R_1 \mapsto R_2 \in \mathcal{K}$. Assume $R_1^I =< P_1^1 \times P_2^1, N_1^1 \times N_2^1 >$, $R_2^I =< P_1^2 \times P_2^2, N_1^2 \times N_2^2 >$. By definition 8, $(R_2^+)^{\overline{I}} = P_1^2 \times P_2^2$, $R_1^= = \Delta^I \times \Delta^I \setminus N_1^1 \times N_2^1$. I satisfies $R_1 \mapsto R_2$, so $(R_1^)\overline{I} == \Delta^I \times \Delta^I \setminus N_1^1 \times N_2^1 \subseteq P_1^2 \times P_2^2 = (R_2^+)^{\overline{I}}$. That is. $\overline{I}$ satisfies $R_1^= \sqsubseteq R_2^+$.

For any $\overline{\mathcal{K}}$'s role inclusion $R_1^+ \sqsubseteq R_2^+, R_1 \sqsubset R_2 \in \mathcal{K}$. Suppose $R_1^I =< P_1^1 \times P_2^1, N_1^1 \times N_2^1 >$, $R_2^I =< P_1^2 \times P_2^2, N_1^2 \times N_2^2 >$. By definition 8, $(R_2^+)^{\overline{I}} = P_1^2 \times P_2^2$, $R_1^+ = P_1^1 \times P_2^1$. I satisfies $R_1 \sqsubset R_2$, so $(R_1^+)\overline{I} = P_1^1 \times P_2^1 \subseteq P_1^2 \times P_2^2 = (R_2^+)^{\overline{I}}$. That is. $\overline{I}$ satisfies $R_1^+ \sqsubseteq R_2^+$.

For any $Tran(R^+) \in \overline{I}$, $Trans(R) \in \mathcal{K}$. Suppose $R^I =< P_1^1 \times P_2^1, N_1^1 \times N_2^1 >$. I satisfies $Trans(R)$, then $R^I = (R^I)^+$, that is $P_1 \times P_2 = (P_1 \times P_2)^+$. By definition 8, $(R^+)^{\overline{I}} = ((R^+)^{\overline{I}})^+$. That is, $Trans(R^+)$.

(sufficiency) For any interpretation $\overline{I} = (\Delta^{\overline{I}}, \cdot^{\overline{I}})$ of $\overline{K}$, let I be the four-valued semantics of $\overline{I}$. By the similar approach, we can prove that the proposition holds. □

For $\mathcal{SHOIN}(\mathcal{D})$, the inclusion axioms can be reduced to unsatisfiability of concepts. The following corollary shows similar results of inclusion axioms and concept satisfiability of $\mathcal{SHOIN}(\mathcal{D})4$.

Corollary 7 *For a $\mathcal{SHOIN}(\mathcal{D})4$ ontology $\overline{\mathcal{K}}$, the material inclusion axiom $C \mapsto D$ holds in $\mathcal{K}$ iff $\neg\overline{\neg C} \sqcap \neg\overline{D}$ is unsatisfiable in $\overline{\mathcal{K}}$; The internal inclusion axiom $C \sqsubset D$ holds in K iff $\overline{C} \sqcap \neg\overline{D}$ is unsatisfiable in $\overline{\mathcal{K}}$; The strong inclusion axiom $C \rightarrow D$ holds in $\mathcal{K}$ iff $\overline{C} \sqcap \neg\overline{D}, \overline{\neg D} \sqcap \neg\overline{\neg C}$ is unsatisfiable in $\overline{\mathcal{K}}$.*

We explain by the following example that $\mathcal{SHOIN}(\mathcal{D})4$ can express inconsistency in a knowledge base, meanwhile inference is done by calling existing reasoning techniques which is based on two-valued semantics.

Example 5 *(Example 3 contd.) By transformations, we obtain the following classical induced KB, $\overline{\mathcal{K}}$, of the example 3:*

$$\overline{TBox} = \begin{cases} \neg Bird^- \sqcap \neg\forall.\overline{hasWing}.Wing^- \sqsubseteq Fly^+ \\ Penguin^+ \sqsubseteq Bird^+ \\ Penguin^+ \sqsubseteq \exists\overline{hasWing}.Wing^+ \\ Penguin^+ \sqsubseteq Fly^- \end{cases}$$

$\overline{ABox} = \{Peguin^+(\overline{tweety}), Bird^+(\overline{tweety}), Wing^+(\overline{w}), hasWing^+(\overline{tweety}, \overline{w})\}$.

By classical tableaux algorithm, $Fly^-(\overline{tweety})$ holds, that is, tweety cannot fly. But $Fly^+(\overline{tweety})$ does not holds, which means that $\overline{\mathcal{K}}$ is not trivial. So is the original $\mathcal{SHOIN}(\mathcal{D})4$ knowledge base by theorem 6.

5 Related Work

Patel-Schneider [10] has proposed four-valued semantics for a terminological logic to equip it with a tractable inclusion relation, while we use the similar method but with some extensions to equip ontology language OWL DL with the ability to represent and reason with contradictions. The direct effect of inclusion form on the validity of algorithm makes the inclusion get only very easy cases in [10]. The algorithm used in [10] for computing inclusion is tractable, while it cannot treat concept union, full negation and exist quantification. Therefore, Patel-Schneider restricts the language without these constructors in [10]. However, the description logic $\mathcal{SHOIN}(\mathcal{D})4$ studied in this paper includes all of these constructors. Moreover, we proves that the standard inferences of $\mathcal{SHOIN}(\mathcal{D})4$ can be converted to those of $\mathcal{SHOIN}(\mathcal{D})$. Thus, both the complexity and decidability of $\mathcal{SHOIN}(\mathcal{D})4$ are the same as those of $\mathcal{SHOIN}(\mathcal{D})$.

The material inclusion proposed in this paper is a method to deal with knowledge with exceptions, which is based on four-valued logic. Salem et al. [4] adopt possibilistic logic and lexicographical inference combining MSP algorithm [18] which can make a stratification among the knowledge with exceptions according to the principle that the ones with higher stratification will be preferential to others which conflict with the former and are of lower stratifications. Huang [11] introduces syntax relevance to selection function whereby some consistent sub-theory(sub-theories) can be selected to be reasoned with. Both approaches mentioned above infer on a consistent sub-theory, while the approach in this paper acknowledges contradictions and allow them to join in reasoning instead of ignoring them. Note that conclusions deduced in this way may contain contradiction also. However, as we have seen, the inconsistencies are localized without destroying useful conclusions.

In this paper, four-valued ontology can be changed into a two-valued one so that we can make full use of existing inference systems instead of studying new mechanism. Formula transformations have been used to compute complex model [15–17]. However, those works are all based on propositional language. The extension of transformation techniques to ontology language is a core work of this paper as well.

6 Conclusion

We defined four-valued semantics for concepts, object (datatype) roles, and axioms, thus forming an inconsistent tolerance description logic $\mathcal{SHOIN}(\mathcal{D})4$. Mature reasoning mechanisms of classical description logic remain useful for $\mathcal{SHOIN}(\mathcal{D})4$ because of concept and axiom transformations. Since $\mathcal{SHOIN}(\mathcal{D})$ is the underlying logic system of OWL DL, $\mathcal{SHOIN}(\mathcal{D})4$ provides us with an approach to infer with inconsistent OWL DL ontology by classical inference mechanism.

The underlying idea through this paper is considering contradiction as static or hidden — that is, it is based on paraconsistent logic. Another method to treat contradiction is nonmonotonic logic which views inconsistency as dynamic and modifiable. We will further compare and combine these two methods to find out excellent methods for semantic web to reason with inconsistencies in the future work.

References

1. F. Patel-Schneider Peter and Horrocks Ian. Owl web ontology language semantics and abstract syntax. *W3C Recommendation*, 10 February, 2004.
2. Baader Franz, Horrocks Ian, and Ulrike Sattler. Description logic as ontology languages for the semantic web. *In Proceedings of the Workshop on Methods for Modalities 2001*, 2001.
3. Horrocks Ian and F. Patel-Schneider Peter. Reducing owl entailment to description logic satisfiability. *J. Web Sem.*, 1(4):345–357, 2004.
4. Salem BENFERHAT, Rania EL BAIDA, and Frédéric CUPPENS. A stratification-based approach for handling conflicts in access control. *Symposium on Access Control Models and Technologies (SACMAT'03)*, June 2-3, 2003.
5. Zuoquan Lin and Wei Li. Para-consistent logic(I)— the study of traditional para-consistent logic. *Computer Science*, 21(5), 1994.
6. Zuoquan Lin and Wei Li. Para-consistent logic(II)— the study of new para-consistent logic. *Computer Science*, 21(6), 1994.
7. Zuoquan Lin and Wei Li. Para-consistent logic(III)— the logic base of para-consistent logic. *Computer Science*, 22(1), 1995.
8. N.D.Belnap. A useful four-valued logic. *Modern uses of multiple-valued logic*, pages 7–73, 1977.
9. N.D.Belnap. How a computer should think. *Comtemporary Aspects of Philosophy: Proceedings of the Oxford International Symposium*, pages 30–56, 1977.
10. Peter F.Patel-Schneider. A four-valued semantics for terminological logics. *Artificial Intelligence*, 38:319–351, 1989.
11. Zhisheng Huang, Frank van Harmelen, and Annette ten Teije. Reasoning with inconsistent ontologies. In Leslie Pack Kaelbling and Alessandro Saffiotti, editors, *IJCAI*, pages 454–459. Professional Book Center, 2005.
12. Melvin Fitting. Bilattices are nice things. *proceedings of Conference on Self-Reference*, 2002.
13. Ofer Arieli and Arnon Avron. The value of the four values. *Artificial Intelligence*, 102:97–141, 1998.
14. Ofer Arieli and Avron A. Reasoning with logical bilattices. *J.Logic,Language and Information*, 5(1):25–63, 1996.
15. Ofer Arieli and Marc Denecker. Modeling paraconsistent reasoning by classical logic. In Thomas Eiter and Klaus-Dieter Schewe, editors, *FoIKS*, volume 2284 of *Lecture Notes in Computer Science*, pages 1–14. Springer, 2002.

16. Ofer Arieli. Paraconsistent preferential reasoning by signed quantified boolean formulae. In Ramon López de Mántaras and Lorenza Saitta, editors, *ECAI*, pages 773–777. IOS Press, 2004.
17. Anbu Yue, Yue Ma, and Zuoquan Lin. Four-valued semantics for default logic. In Luc Lamontagne and Mario Marchand, editors, *Canadian Conference on AI*, volume 4013 of *Lecture Notes in Computer Science*, pages 195–205. Springer, 2006.
18. Salem Benferhat, Didier Dubois, and Henri Prade. Nonmonotonic reasoning, conditional objects and possibility theory. *Artif. Intell.*, 92(1-2):259–276, 1997.