Fuzzy Systems and Data Mining VII
A.J. Tallón-Ballesteros (Ed.)

doi:10.3233/FAIA210209

R-Calculus for Post Three-Valued Description Logic[1]

Cungen CAO [a], Lanxi HU [a,b,2] and Yuefei SUI [a,b]

[a] *Key Laboratory of Intelligent Information Processing, Institute of Computing Technology Chinese Academy of Sciences,China*

[b] *School of Computer Science and Technology University of Chinese Academy of Sciences,China*

Abstract. R-calculus is a belief revision operator satisfying AGM postulates, and belief revision in ontology engineering is ontology revision, which based logic is description logics. In Post three-valued description logic, a tableau proof system $\mathbf{T}_{\mathtt{t}}$ will be given such that $\mathbf{T}_{\mathtt{t}}$ is sound and complete for $\mathtt{t}$-satisfiability, and nonmonotonic, that is, a theory Δ is $\mathtt{t}$-satisfiable if and only if Δ is deducible in $\mathbf{T}_{\mathtt{t}}$. Based on the tableau proof system, an R-calculus $\mathbf{R}_{\mathtt{t}}$ will be given such that a configuration $\Delta|C(a)$ is reducible to $C(a),\Delta$ if and only if $C(a)$ is $\mathtt{t}$-satisfiable with Δ, if and only if reduction $\Delta|C(a) \Rightarrow C(a),\Delta$ is deducible in $\mathbf{R}_{\mathtt{t}}$.

Keywords. Post three-valued logic, Belief revision, Tableau proof system, R-calculus, Concepts

1. Introduction

Belief revision is a topic of logic, computer science and philosophy. Given a knowledge base Δ and a formula A in a logic, A is enumerated into Δ if and only if A is consistent with Δ. AGM postulates [1] are a set of basic requirements a belief revision operator should satisfy. Belief revision in ontology engineering is ontology revision, which based logic is description logics. Traditional ontology revision is based on binary-valued description logics. We consider the three-valued description logics.

In many-valued logic [2][3], it is important to give an explanation of the truth-values other than the truth $\mathtt{t}$ and the falsity $\mathtt{f}$. For example, in a three-valued logic [4], the third value $\mathtt{m}$ is interpreted as unknown or indeterminate, and the semantic definition of binary logical connectives are independent of $\mathtt{m}$. Description logics [5] are different from traditional logics, because a concept seems natural to have different counterparts. For example, in three-valued description logics, an interpretation C^I of a concept C is decomposed into three parts: $(\oslash C)^I$, consisting of these elements taking truth-value $\mathtt{t}$; $(\sim C)^I$, these taking $\mathtt{m}$, and $(\lhd C)^I$, these taking $\mathtt{f}$.

[1]Supported by the National Key Research and Development Program of China(2017YFB1002300, 2017YFC1700300), National Natural Science Foundation of China (61702234),Beijing NOVA Program(Cross-discipline, Z191100001119014).

[2]Corresponding Author: Lanxi Hu,BEIJING,CHINA; E-mail: hulanxi17b@ict.ac.cn

R-calculus [6][7] is a belief revision operator satisfying AGM postulates, and a deduction system for enumerating a formula A into a consistent theory Δ to keep the theory A', Δ consistent (denoted by $\models_{\mathtt{t}} \Delta|A \Rightarrow A', \Delta$, where $\Delta|A$ is called a configuration; $\Delta|A \Rightarrow A', \Delta$ is called a reduction, and A' is A if A is t-satisfiable with Δ and otherwise $A' = \lambda$, the empty string). A condition that there is a sound and complete R-calculus is that the based logic is decidable. Hence, there are sound and complete R-calculi for propositional logic [8], propositional modal logic [9], etc., and there is no such R-calculus for first-order logic.

Description logics are fragments of first-order logic, some of which are decidable and some are not. We consider one of many-valued description logics: Post three-valued description logics [10], where the logical language of Post logic contains a unary connective $\sim$, instead of $\neg$. For Post logic, there are sound and complete tableau proof systems, Gentzen deduction systems and deduction systems for many-placed sequents [3].

	$\neg$
t	f
m	m
f	t

	$\sim$
t	f
m	t
f	m

For decidable description logics, a problem is to define the semantics of quantifier concept constructors. In binary ones, an element a belongs to an interpretation of concept $(\forall R.C)$ if for any element b with $(a,b) \in R^I, b \in C^I$; and an element a belongs to an interpretation of concept $\neg(\forall R.C)$ if for some element b with $(a,b) \in R^I, b \notin C^I$. Correspondingly, in Post three-valued description logic and an interpretation I, we define

- an element a belongs to the interpretation of concept $(\forall R.C)$ if for any element b with $(a,b) \in R^I, b \in C^I$;
- an element a belongs to the interpretation of concept $\sim (\forall R.C)$ if $a \notin (\forall R.C)^I$ and for any element b with $(a,b) \in (\oslash R \cup \sim R)^I, b \in (\oslash C \cup \sim C)^I$;
- an element a belongs to the interpretation of concept $\lhd(\forall R.C)$ if there is an element b such that $(a,b) \in (\oslash R \cup \sim R)^I$ and $b \in (\lhd C)^I$.

A theory (a set of statements) Δ is t-satisfiable if there is an interpretation I such that for any statement $C(a) \in \Delta, (C(a))^I \neq \mathtt{t}$. We will give a tableau proof system $\mathbf{T}_{\mathtt{t}}$ for t-satisfiability, which is sound, complete and nonmonotonic.

Based on the tableau proof system $\mathbf{T}_{\mathtt{t}}$, we construct an R-calculus $\mathbf{R}_{\mathtt{t}}$ for $\Delta|A \Rightarrow A', \Delta$. $\mathbf{R}_{\mathtt{t}}$ is shown to be sound and complete, that is,

$$\models_{\mathtt{t}} \Delta|A \Rightarrow A, \Delta \text{ iff } \Delta|A \Rightarrow A, \Delta \text{ is provable in } \mathbf{R}_{\mathtt{t}}.$$

Because $\models_{\mathtt{t}} \Delta|A \Rightarrow \Delta$ iff $\not\models_{\mathtt{t}} \Delta|A \Rightarrow A, \Delta$, we have

$$\models_{\mathtt{t}} \Delta|A \Rightarrow \Delta \text{ iff } \mathbf{R}_{\mathtt{t}} \nvdash \Delta|A \Rightarrow \Delta.$$

This paper is organized as follows: The next section defines the logical language and the semantics of Post three-valued description logic; the third section gives a tableau proof system for the description logic and shows soundness and completeness theorems; the fourth section gives an R-calculus for t-satisfiability, and the last section concludes the whole paper.

2. Post three-valued description logic

Let $\mathbf{L_3} = (\{\mathtt{t}, \mathtt{m}, \mathtt{f}\}, \oslash, \sim, \lhd, \cap, \cup)$ be an algebraical structure, where

	$\oslash$	$\sim$	$\lhd$
t	t	f	m
m	m	t	f
f	f	m	t

$\cap$	t	m	f
t	t	m	f
m	m	m	f
f	f	f	f

$\cup$	t	m	f
t	t	t	t
m	t	m	m
f	t	m	f

The logical language of Post three-valued description logic contains the following symbols:

- atomic concepts: $A_0, A_1, ...$;
- roles: $R_0, R_1, ...$;
- concept constructors: $\oslash, \sim, \lhd, \sqcap, \sqcup, \forall$.

Concepts are defined inductively as follows:

$$C ::= A | \oslash C | \sim C | \lhd C | C_1 \sqcap C_2 | C_1 \sqcup C_2 | \forall R.C,$$

where A is an atomic concept, and R is a role.

Statements are defined as follows:

$$\varphi ::= C(a) | R(a,b) | \oslash \varphi | \sim \varphi | \lhd \varphi.$$

A model M is a pair (U, I), where U is a non-empty set, and I is an interpretation such that

- ∘ for any atomic concept $A, I(A) : U \to \mathbf{L_3}$;
- ∘ for any role $R, I(R) : U^2 \to \mathbf{L_3}$.

Given an atomic concept A and a role R, we define concepts $\oslash A, \sim A, \lhd A$ and roles $\oslash R, \sim R, \lhd R$ as follows: for any $x \in U$,

$A(x)$	$\oslash A(x)$	$\sim A(x)$	$\lhd A(x)$
t	t	f	m
m	m	t	f
f	f	m	t

$R(x,y)$	$\oslash R(x,y)$	$\sim R(x,y)$	$\lhd R(x,y)$
t	t	f	m
m	m	t	f
f	f	m	t

The interpretation C^I of a concept C is a function from U to $\mathbf{L_3}$ such that for any $x \in U$,

$$C^I(x) = \begin{cases} I(A)(x) & \text{if } C = A \\ *(C^I)(x) & \text{if } C = *C_1 \\ C_1^I(x) \cap C_2^I(x) & \text{if } C = C_1 \sqcap C_2 \\ C_1^I(x) \cup C_2^I(x) & \text{if } C = C_1 \sqcup C_2 \\ \min\{\max\{I(\sim R)(x,y), I(\lhd R)(x,y), C_1^I(y)\} : y \in U\} & \text{if } C = \forall R.C_1, \end{cases}$$

where $* \in \{\oslash, \sim, \lhd\}$.

Therefore, $C^I(x) = \mathtt{t}$[3] if

[3] In syntax, we use $\neg, \wedge, \to, \forall, \exists$ to denote the logical connectives and quantifiers; and in semantics we use $\sim, \&, \Rightarrow, \mathbf{A}, \mathbf{E}$ to denote the corresponding connectives and quantifiers.

$$\begin{cases} \mathbf{A}y \in U(I(\oslash R)(x,y) = \mathtt{t} \Rightarrow C_1^{\mathtt{I}}(\mathtt{y}) = \mathtt{t}) & \text{if } C(x) = (\oslash \forall R.C_1)(x) \\ \mathbf{A}y \in U(I(\oslash R \cup \sim R)(x,y) = \mathtt{t} \Rightarrow (\oslash C_1 \sqcup \sim C_1)^{\mathtt{I}}(\mathtt{y}) = \mathtt{t}) & \\ \&\mathbf{E}y \in U(I(\oslash R \cup \sim R)(x,y) = \mathtt{t}\&(\sim C_1)(\mathtt{y}) = \mathtt{t}) & \text{if } C(x) = (\sim \forall R.C_1)(x) \\ \mathbf{E}y \in U(I(\oslash R \cup \sim R)(x,y) = \mathtt{t}\&(\lhd C_1)(\mathtt{y}) = \mathtt{t}) & \text{if } C(x) = (\lhd \forall R.C_1)(x). \end{cases}$$

A theory Δ is $\mathtt{t}$-valid, denoted by $\models^{\mathtt{t}} \Delta$, if for any interpretation I, there is a statement $\varphi \in \Delta$ such that $(\varphi)^I = \mathtt{t}$; and Δ is $\mathtt{t}$-satisfiable, denoted by $\models_{\mathtt{t}} \Delta$, if there is an interpretation I such that for each statement $\varphi \in \Delta$, $(\varphi)^I \neq \mathtt{t}$.

Proposition 2.1. For any concept C and interpretation I, and for any $x \in U$, $C^I(x) \cup (\sim C)^I(x) \cup (\lhd C)^I(x) = \mathtt{t}$.

3. Nonmonotonic tableau proof system

Define

$$\begin{array}{l} \text{incon}(\Delta) \text{ iff } \mathbf{E}\text{p}(\text{p}, \sim \text{p}, \lhd \text{p} \in \Delta) \\ \text{con}(\Delta) \text{ iff } \neg\mathbf{E}\text{p}(\text{p}, \sim \text{p}, \lhd \text{p} \in \Delta) \end{array}$$

.

Proposition 3.1. Let Δ be a set of literals. Δ is $\mathtt{t}$-satisfiable iff con(Δ).

Nonmonotonic tableau proof system $\mathbf{T}_{\mathtt{t}}$ contains the following axioms and deduction rules: let a be a constant.

• **Axioms:**

$$\frac{\text{con}(\Delta)}{\Delta} \; (\mathtt{A}_{\mathtt{t}})$$

where Δ is a set of literals.

• **Deduction rules for modalities:**

$$\frac{f(*_1, *_2)C_1(a), \Delta}{*_1 *_2 C_1(a), \Delta} (*_1 *_2)$$

where $*_1, *_2 \in \{\lambda, \sim, \lhd\}$ and $f(*_1, *_2)$ is defined as follows:

$f(*_1,*_2)$	$\oslash$	$\sim$	$\lhd$
$\oslash$	$\oslash$	$\sim$	$\lhd$
$\sim$	$\sim$	$\lhd$	$\oslash$
$\lhd$	$\lhd$	$\oslash$	$\sim$

• **Deduction rules for logical connectives:**

$$\frac{\left\{\begin{array}{l}\oslash C_1(a),\Delta\\ \oslash C_2(a),\Delta\end{array}\right.}{\oslash(C_1\sqcap C_2)(a),\Delta}(\oslash\sqcap)\quad \frac{\left[\begin{array}{l}\oslash C_1(a),\Delta\\ \oslash C_2(a),\Delta\end{array}\right.}{\oslash(C_1\sqcup C_2)(a),\Delta}(\oslash\sqcup)\quad \frac{\left[\begin{array}{l}\sim R(a,d),\Delta\\ \lhd R(a,d),\Delta\\ C_1(d),\Delta\end{array}\right.}{\oslash(\forall R.C_1)(a),\Delta}(\oslash\forall)$$

$$\frac{\left[\begin{array}{l}\left\{\begin{array}{l}\sim C_1(a),\Delta\\ \sim C_2(a),\Delta\end{array}\right.\\ \left\{\begin{array}{l}\oslash C_1(a),\Delta\\ \sim C_2(a),\Delta\end{array}\right.\\ \left\{\begin{array}{l}\sim C_1(a),\Delta\\ \oslash C_2(a),\Delta\end{array}\right.\end{array}\right.}{\sim(C_1\sqcap C_2)(a),\Delta}(\sim\sqcap)\quad \frac{\left[\begin{array}{l}\left\{\begin{array}{l}\sim C_1(a),\Delta\\ \sim C_2(a),\Delta\end{array}\right.\\ \left\{\begin{array}{l}\lhd C_1(a),\Delta\\ \sim C_2(a),\Delta\end{array}\right.\\ \left\{\begin{array}{l}\sim C_1(a),\Delta\\ \lhd C_2(a),\Delta\end{array}\right.\end{array}\right.}{\sim(C_1\sqcup C_2)(a),\Delta}(\sim\sqcup)\quad \frac{\left\{\begin{array}{l}\left[\begin{array}{l}\lhd R(a,d),\Delta\\ \oslash C_1(d),\Delta\\ \sim C_1(d),\Delta\end{array}\right.\\ \left\{\begin{array}{l}\lhd R(a,c),\Delta\\ \sim C_1(c),\Delta\end{array}\right.\end{array}\right.}{\sim(\forall R.C_1)(a),\Delta}(\sim\forall)$$

$$\frac{\left[\begin{array}{l}\lhd C_1(a),\Delta\\ \lhd C_2(a),\Delta\end{array}\right.}{\lhd(C_1\sqcap C_2)(a),\Delta}(\lhd\sqcap)\quad \frac{\left\{\begin{array}{l}\lhd C_1(a),\Delta\\ \lhd C_2(a),\Delta\end{array}\right.}{\lhd(C_1\sqcup C_2)(a),\Delta}(\lhd\sqcup)\quad \frac{\left\{\begin{array}{l}\left[\begin{array}{l}\oslash R(a,c),\Delta\\ \sim R(a,c),\Delta\end{array}\right.\\ \lhd C_1(c),\Delta\end{array}\right.}{\lhd(\forall R.C_1)(a),\Delta}(\lhd\forall)$$

where d is a constant, c is a new constant, and $\dfrac{\left\{\begin{array}{l}\delta_1\\ \delta_2\end{array}\right.}{\delta}$ means that δ_1 implies δ and δ_2 implies δ; and $\dfrac{\left[\begin{array}{l}\delta_1\\ \delta_2\end{array}\right.}{\delta}$ means that δ_1 and δ_2 imply δ.

Definition 3.2. A theory Δ is provable in $\mathbf{T}_{\mathtt{t}}$, denoted by $\vdash_{\mathtt{t}} \Delta$, if there is a sequence $\{\Delta_1,...,\Delta_n\}$ of theories such that $\Delta_n = \Delta$, and for each $1 \leq i \leq n, \Delta_i$ is either an axiom or deduced from the previous theories by one of the deduction rules in $\mathbf{T}_{\mathtt{t}}$.

Theorem 3.3. For any theory Δ, $\models_{\mathtt{t}} \Delta$ iff $\vdash_{\mathtt{t}} \Delta$.

Because $\models^{\mathtt{t}} \Delta$ if and only if $\not\models_{\mathtt{t}} \Delta$, we have the following

Corollary 3.4. For any theory Δ, $\models^{\mathtt{t}} \Delta$ iff $\nvdash_{\mathtt{t}} \Delta$.

4. R-calculus

Intuitively, a statement $\oslash(C_1\sqcap C_2)(a)$ is enumerable into Δ to preserve the $\mathtt{t}$-satisfiability of Δ, if either $\oslash C_1(a)$ or $\oslash C_2(a)$ is enumerable into Δ; and $\oslash(C_1\sqcup C_2)(a)$ is enumerable into Δ if $\oslash C_1(a)$ is enumerable into Δ and $\oslash C_2(a)$ is enumerable into $\Delta\cup\{\oslash C_1(a)\}$.

Statement $\sim(C_1\sqcap C_2)(a)$ is enumerable into Δ, if (1) either $\sim C_1(a)$ or $\sim C_2(a)$ is enumerable into Δ; (2) either $\oslash C_1(a)$ or $\sim C_2(a)$ is enumerable into $\Delta\cup\{(\sim C_1\sqcap\sim C_2)(a)\}$, and (3) either $\sim C_1(a)$ or $\oslash C_2(a)$ is enumerable into $\Delta\cup\{(\sim C_1\sqcap\sim C_2)(a),(\oslash C_1\sqcap\sim C_2)(a)\}$; and statement $\sim(C_1\sqcup C_2)(a)$ is enumerable into Δ, if (4) either $\sim C_1(a)$ or $\sim C_2(a)$ is enumerable into Δ; (5) either $\lhd C_1(a)$ or $\sim C_2(a)$ is enumerable into $\Delta\cup\{(\sim C_1\sqcap\sim C_2)(a)\}$, (6) either $\sim C_1(a)$ or $\lhd C_2(a)$ is enumerable into $\Delta\cup\{(\sim C_1\sqcap\sim C_2)(a),(\lhd C_1\sqcap\sim C_2)(a)\}$.

Statement $\lhd(C_1\sqcap C_2)(a)$ is enumerable into Δ if $\lhd C_1(a)$ is enumerable into Δ, and $\lhd C_2(a)$ is enumerable into $\Delta\cup\{\lhd C_1(a)\}$; and $\lhd(C_1\sqcup C_2)(a)$ is enumerable into Δ if either $\lhd C_1(a)$ or $\lhd C_2(a)$ is enumerable into Δ.

A statement $\oslash(C_1\sqcap C_2)(a)$ is not enumerable into Δ, if $\oslash C_1(a)$ and $\oslash C_2(a)$ are not enumerable into Δ; and $\oslash(C_1\sqcup C_2)(a)$ is not enumerable into Δ if either $\oslash C_1(a)$ is not enumerable into Δ, or $\oslash C_2(a)$ is not enumerable into $\Delta\cup\{\oslash C_1(a)\}$.

Statement $\sim(C_1\sqcap C_2)(a)$ is not enumerable into Δ if either (1) $\sim C_1(a)$ and $\sim C_2(a)$ are not enumerable into Δ, or (2) $\oslash C_1(a)$ and $\sim C_2(a)$ are not enumerable into $\Delta\cup\{(\sim$

$C_1 \sqcap \sim C_2)(a)\}$, or (3) $\sim C_1(a)$ and $\oslash C_2(a)$ are not enumerable into $\Delta \cup \{(\sim C_1 \sqcap \sim C_2)(a), (\oslash C_1 \sqcap \sim C_2)(a)\}$; and statement $\sim (C_1 \sqcup C_2)(a)$ is not enumerable into Δ, if either (4) $\sim C_1(a)$ and $\sim C_2(a)$ are not enumerable into Δ, or (5) $\triangleleft C_1(a)$ and $\sim C_2(a)$ are not enumerable into $\Delta \cup \{(\sim C_1 \sqcap \sim C_2)(a)\}$, or (6) $\sim C_1(a)$ and $\triangleleft C_2(a)$ are not enumerable into $\Delta \cup \{(\sim C_1 \sqcap \sim C_2)(a), (\triangleleft C_1 \sqcap \sim C_2)(a)\}$.

Statement $\triangleleft(C_1 \sqcap C_2)(a)$ is not enumerable into Δ if either $\triangleleft C_1(a)$ is not enumerable into Δ or $\triangleleft C_2(a)$ is not enumerable into $\Delta \cup \{\triangleleft C_1(a)\}$; and $\triangleleft(C_1 \sqcup C_2)(a)$ is not enumerable into Δ if $\triangleleft C_1(a)$ and $\triangleleft C_2(a)$ is not enumerable into Δ.

Given a theory Δ and a statement φ, we use Δ to revise φ and obtain φ', Δ, denoted by

$$\Delta|\varphi \Rightarrow \varphi', \Delta,$$

if

$$\varphi' = \begin{cases} \varphi \text{ if } \Delta \text{ is } \mathtt{t}\text{-satisfiable with } \varphi \\ \lambda \text{ otherwise.} \end{cases}$$

R-calculus $\mathbf{R}_\mathtt{t}$ consists of the following axioms and deduction rules:

• **Axioms:**

$$\frac{\vdash_\mathtt{t} \Delta \Rightarrow \vdash_\mathtt{t} l, \Delta}{\Delta|l \Rightarrow l, \Delta} (\mathscr{A}_\mathtt{t})$$

• **Deduction rules for modalities:**

$$\frac{\Delta|f(*_1, *_2)C_1(a) \Rightarrow f(*_1, *_2)C_1(a), \Delta}{\Delta| *_1 *_2 C_1(a) \Rightarrow *_1 *_2 C_1(a), \Delta} (*_1 *_2)$$

• **Deduction rules for logical connectives:**

$$\frac{\begin{cases} \Delta|\oslash C_1(a) \Rightarrow \oslash C_1(a), \Delta \\ \Delta|\oslash C_2(a) \Rightarrow \oslash C_2(a), \Delta \end{cases}}{\Delta|\oslash(C_1 \sqcap C_2)(a) \Rightarrow \oslash(C_1 \sqcap C_2)(a), \Delta} (\oslash\sqcap)$$

$$\frac{\begin{bmatrix} \Delta|\oslash C_1(a) \Rightarrow \oslash C_1(a), \Delta \\ \Delta, \oslash C_1(a)|\oslash C_2(a) \Rightarrow \oslash C_1(a), \oslash C_2(a), \Delta \end{bmatrix}}{\Delta|\oslash(C_1 \sqcup C_2)(a) \Rightarrow \oslash(C_1 \sqcup C_2)(a), \Delta} (\oslash\sqcup)$$

$$\frac{\begin{cases} \Delta|\sim R(a,c) \Rightarrow \sim R(a,c), \Delta \\ \Delta|\triangleleft R(a,c) \Rightarrow \triangleleft R(a,c), \Delta \\ \Delta|\oslash C_1(c) \Rightarrow \oslash C_1(c), \Delta \end{cases}}{\Delta|\oslash(\forall R.C_1)(a) \Rightarrow \oslash(\forall R.C_1)(a), \Delta} (\oslash\forall)$$

and

$$\frac{\left[\begin{array}{l}\left\{\begin{array}{l}\Delta|\sim C_1(a)\Rightarrow\sim C_1(a),\Delta\\ \Delta|\sim C_2(a)\Rightarrow\sim C_2(a),\Delta\end{array}\right.\\ \left\{\begin{array}{l}\Delta,X|\oslash C_1(a)\Rightarrow\oslash C_1(a),X,\Delta\\ \Delta,X|\sim C_2(a)\Rightarrow\sim C_2(a),X,\Delta\end{array}\right.\\ \left\{\begin{array}{l}\Delta,X,Y|\sim C_1(a)\Rightarrow\sim C_1(a),X,Y,\Delta\\ \Delta,X,Y|\oslash C_2(a)\Rightarrow\oslash C_2(a),X,Y,\Delta\end{array}\right.\end{array}\right.}{\Delta|\sim(C_1\sqcap C_2)(a)\Rightarrow\sim(C_1\sqcap C_2)(a),\Delta}(\sim\sqcap)$$

$$\frac{\left[\begin{array}{l}\left\{\begin{array}{l}\Delta|\sim C_1(a)\Rightarrow\sim C_1(a),\Delta\\ \Delta|\sim C_2(a)\Rightarrow\sim C_2(a),\Delta\end{array}\right.\\ \left\{\begin{array}{l}\Delta,X|\lhd C_1(a)\Rightarrow\lhd C_1(a),X,\Delta\\ \Delta,X|\sim C_2(a)\Rightarrow\sim C_2(a),X,\Delta\end{array}\right.\\ \left\{\begin{array}{l}\Delta,X,Z|\sim C_1(a)\Rightarrow\sim C_1(a),X,Z,\Delta\\ \Delta,X,Z|\lhd C_2(a)\Rightarrow\lhd C_2(a),X,Z,\Delta\end{array}\right.\end{array}\right.}{\Delta|\sim(C_1\sqcup C_2)\Rightarrow\sim(C_1\sqcup C_2)(a),\Delta}(\sim\sqcup)$$

$$\frac{\begin{array}{l}\left\{\begin{array}{l}\left[\begin{array}{l}\Delta|\lhd R(a,c)\Rightarrow\lhd R(a,c),\Delta\\ \Delta,\lhd R(a,c)|\oslash C_1(c)\Rightarrow\lhd R(a,c),\oslash C_1(c)\Delta\end{array}\right.\\ \Delta,\lhd R(a,c),\oslash C_1(c)|\sim C_1(c)\Rightarrow\sim C_1(c),\lhd R(a,c),\oslash C_1(c)\Delta\end{array}\right.\\ \left\{\begin{array}{l}\left[\begin{array}{l}\Delta|\oslash R(a,d)\Rightarrow\oslash R(a,d),\Delta\\ \Delta,\oslash R(a,d)|\sim R(a,d)\Rightarrow\sim R(a,d),\oslash R(a,d),\Delta\end{array}\right.\\ \Delta|\sim C_1(d)\Rightarrow\sim C_1(d),\Delta\end{array}\right.\end{array}}{\Delta|\sim(\forall R.C_1)(a)\Rightarrow\sim(\forall R.C_1)(a),\Delta}(\sim\forall)$$

where $X=(\sim C_1\sqcap\sim C_2)(a), Y=(\oslash C_1\sqcap\sim C_2)(a), Z=(\lhd C_1\sqcap\sim C_2)(a)$, and

$$\frac{\left[\begin{array}{l}\Delta|\lhd C_1(a)\Rightarrow\lhd C_1(a),\Delta\\ \Delta,\lhd C_1(a)|\lhd C_2(a)\Rightarrow\lhd C_2(a),\lhd C_1(a),\Delta\end{array}\right.}{\Delta|\lhd(C_1\sqcap C_2)(a)\Rightarrow\lhd(C_1\sqcap C_2)(a),\Delta}(\lhd\sqcap)$$

$$\frac{\left\{\begin{array}{l}\Delta|\lhd C_1(a)\Rightarrow\lhd C_1(a),\Delta\\ \Delta|\lhd C_2(a)\Rightarrow\lhd C_2(a),\Delta\end{array}\right.}{\Delta|\lhd(C_1\sqcup C_2)(a)\Rightarrow\lhd(C_1\sqcup C_2)(a),\Delta}(\lhd\sqcup)$$

$$\frac{\left\{\begin{array}{l}\left[\begin{array}{l}\Delta|\oslash R(a,d)\Rightarrow\oslash R(a,d),\Delta\\ \Delta,\oslash R(a,d)|\sim R(a,d)\Rightarrow\sim R(a,d),\oslash R(a,d),\Delta\end{array}\right.\\ \Delta|\lhd C_1(d)\Rightarrow\lhd C_1(d),\Delta\end{array}\right.}{\Delta|\lhd(\forall R.C_1)(a)\Rightarrow\lhd(\forall R.C_1)(a),\Delta}(\lhd\forall)$$

where d is a constant and c is a new constant.

Definition 4.1. A reduction $\Delta|C(a)\Rightarrow C(a),\Delta$ is provable in $\mathbf{R}_{\mathtt{t}}$, denoted by $\vdash_{\mathtt{t}}\Delta|C(a)\Rightarrow C(a),\Delta$, if there is a sequence $\{\delta_1,...,\delta_n\}$ of reductions such that $\delta_n=\Delta|C(a)\Rightarrow C(a),\Delta$, and for each $1\leq i\leq n,\delta_i$ is either an axiom or deduced from the previous theories by one of the deduction rules in $\mathbf{R}_{\mathtt{t}}$.

Theorem 4.2. For any theory Δ and statement $C(a)$, $\models_{\mathtt{t}}\Delta|C(a)\Rightarrow C(a),\Delta$ iff $\vdash_{\mathtt{t}}\Delta|C(a)\Rightarrow C(a),\Delta$.

Because $\models_{\mathtt{t}}\Delta|C(a)\Rightarrow\Delta$ if and only if $\not\models_{\mathtt{t}}\Delta|C(a)\Rightarrow C(a),\Delta$, we have the following

Corollary 4.3. For any theory Δ and statement $C(a)$, $\models_{\mathtt{t}}\Delta|C(a)\Rightarrow\Delta$ iff $\nvdash_{\mathtt{t}}\Delta|C(a)\Rightarrow C(a),\Delta$.

5. Conclusions

This paper gave an R-calculus $\mathbf{R}_{\mathtt{t}}$ for t-satisfiability in Post three-valued description logic, which is sound and complete. Similarly there are R-calculi $\mathbf{R}_{\mathtt{m}}$ and $\mathbf{R}_{\mathtt{f}}$ for m-satisfiability and f-satisfiability, respectively, and there are transformations between $\mathbf{R}_{\mathtt{t}},\mathbf{R}_{\mathtt{m}}$ and $\mathbf{R}_{\mathtt{f}}$, just as transformations between $\mathbf{T}_{\mathtt{t}},\mathbf{T}_{\mathtt{m}}$ and $\mathbf{T}_{\mathtt{f}}$.

References

[1] Alchourrón CE, Gärdenfors P, Makinson D. On the logic of theory change: partial meet contraction and revision functions. Journal of Symbolic Logic.1985;50:510-30.

[2] Urquhart A. Basic many-valued logic. In: D. Gabbay, F. Guenthner, editors. Handbook of Philosophical Logic 2 (2nd edition). Dordrecht: Kluwer.2001:p.249-95.

[3] Zach R. Proof theory of finite-valued logics, Technical Report TUW-NDARRAY185.2-Z.1-93. Wien: Institut Für Computersprachen, Technische Universität Wien.1993.

[4] Avron A. Natural 3-valued logics: Characterization and proof theory. Journal of Symbolic Logic.1991;56:276-94.

[5] Baader F, Calvanese D, McGuinness DL, Nardi D, Patel-Schneider PF. The Description Logic Handbook: Theory, Implementation, Applications. Cambridge:Cambridge University Press.2003.

[6] Rademaker A. A Proof Theory for Description Logics, SpringerBriefs in Computer Science. London:Springer.2012.

[7] Li W. R-calculus: an inference system for belief revision. The Computer Journal.2007;50:378-90.

[8] Li W, Sui Y. The Sound and Complete R-Calculi With Respect To Pseudo-Revision and Pre-Revision. International Journal of Intelligence Science. 2013;3:110-17.

[9] Fermé E, Hansson SO. AGM 25 years, twenty-five years of research in belief change. Journal of Philosophical Logic. 2011;40:295-331.

[10] Post EL. Determination of all closed systems of truth tables. Bulletin American Mathematical Society. 1920;26:437.