

Quasi-classical description logic

Peer-reviewed author version

ZHANG, Xiaowang & LIN, Zuoquan (2012) Quasi-classical description logic. In:
JOURNAL OF MULTIPLE-VALUED LOGIC AND SOFT COMPUTING, 18 (3-4), p. 291-327.

Handle: <http://hdl.handle.net/1942/17805>

Quasi-Classical Description Logic

XIAOWANG ZHANG^{*}, ZUOQUAN LIN

Department of Information Science, Peking University, P. R. China

In this paper, we present a paraconsistent description logic based on quasi-classical logic. Compared to the four-valued description logic, quasi-classical description logic satisfies all of the three basic inference rules (i.e., *modus ponens*, *modus tollens* and *disjunctive syllogism*) so that the inference ability of quasi-classical description logic is closer to that of classical logic. Quasi-classical description logic combines three inclusions (i.e., material inclusion, internal inclusion and strong inclusion) of four-valued description logic so that quasi-classical description logic satisfies the intuitive equivalence. Moreover, we develop a terminable, sound and complete tableau algorithm for quasi-classical description logic. As an important result, the complexity of reasoning problems in quasi-classical description logic is proved to be no higher than that of reasoning problems in description logic.

Key words: ontology, description logic, quasi-classical logic, paraconsistent logic, multiple-valued Logic, inconsistency-tolerant reasoning, tableau algorithm

1 INTRODUCTION

In an open, constantly changing and collaborative environment like Semantic Web, an ontology may often contain inconsistencies due to many reasons, such as modeling errors, migration from other formalisms, merging ontologies and ontology evolution [13, 16, 10, 12, 11]. As the logical foundation of the Web Ontology Language (OWL), description logics (DLs) are unable to

^{*} email: x.zhang@pku.edu.cn

deal with inconsistency in knowledge bases. Since DLs follow the classical semantics, according to the fact *ex contradictione quodlibet*, if an ontology contains a single contradiction then the classical entailment is explosive. That is, every formula is a logical consequence of an inconsistent ontology. Thus, conclusions drawn from an inconsistent ontology are completely meaningless. To order to solve the problem, inconsistency handling in OWL and DLs has received extensive interests in the community in recent years.

There are many methods to handle inconsistencies in DLs, which can be divided into two types of approaches. One is based on the assumption that inconsistencies indicate erroneous data which are to be repaired in order to obtain a consistent ontology, e.g., by pinpointing the parts of an ontology which cause the inconsistencies and removing or weakening axioms in these parts to restore consistency [30, 27, 18, 26, 25]. The other, called *paraconsistent approach*, does not simply avoid the inconsistencies but tolerate them by applying a non-standard reasoning method to obtain meaningful answers [24, 33, 16, 20, 21, 22, 23, 11, 37, 38]. For the latter, inconsistencies are treated as a natural phenomenon in realistic data to be tolerated in reasoning. Compared with the former, the latter acknowledges and distinguishes the different epistemic statuses between “*the assertion is true*” and “*the assertion is true with conflict*”. One of paraconsistent reasoning in DLs is based on Belnap’s four-valued semantics [5]. However, the four-valued semantics is weak in a sense that it does not satisfy some classical inference rules, such as:

$$\begin{aligned} \text{modus ponens (MP)} \quad & \{C(a), C \sqsubseteq D\} \models D(a) \\ \text{modus tollens (MT)} \quad & \{\neg D(a), C \sqsubseteq D\} \models \neg C(a) \\ \text{disjunctive syllogism (DS)} \quad & \{\neg C(a), C \sqcup D\} \models D(a) \end{aligned}$$

where C, D are concepts and a an individual in DLs. A *total negation* is introduced in [20] to strengthen the capability of paraconsistent reasoning in four-valued DLs. However, since the *total negation* is not contained in the syntax of the four-valued DLs, it is difficult to define a suitable meaning for the two logical connectives “ \sqcup ” and “ \sqsubseteq ”, and the so-called *intuitive equivalence inference rule*: $\mathcal{O} \models C \sqsubseteq D$ if and only if $\mathcal{O} \models \neg C \sqcup D(a)$ for any individual a . These shortcomings are inherent limitations of four-valued logics in paraconsistent reasoning. We expect that a paraconsistent logic can satisfy as many inference rules in classical logics as possible so that its semantics could be as close to classical semantics as possible.

To find a satisfiable paraconsistent semantics, we investigate the problem of defining a suitable paraconsistent semantics for DLs based on the *quasi-classical logic (QC logic)* proposed in [6, 17] which could tolerate inconsis-

tencies by forbidding the mixing applying of both the resolution rules and the disjunction rule. This problem is challenging in that it is not straightforward to extend the QC semantics for propositional logic to DLs. Specifically, in the setting of DLs, it is difficult to define a suitable meaning for the two logical connectives “ \sqcup ” and “ \sqsubseteq ”, which are two key constructors in DLs.

We present a paraconsistent extension of description logic \mathcal{ALC} , called *quasi-classical description logic \mathcal{ALC}* (QC \mathcal{ALC} for short), which is an extension of \mathcal{ALC} with *quasi-classical semantics*. The contributions of our work are summarized as follows. Two QC semantics, called “*weak semantics*” and “*strong semantics*”, are introduced for \mathcal{ALC} . The weak semantics is a reformulation of the four-valued semantics. In contrast, the strong semantics is introduced to strengthen the capability of paraconsistent reasoning in \mathcal{ALC} . Strong semantics refines the interpretation of the disjunction of concepts in order to enhance the capability of paraconsistent reasoning. Moreover, the interpretation of a subsumption is redefined in both of the semantics so that the intuitive equivalence could hold. QC entailment (written by “ \models_Q ”) between an ontology and a formula is presented. We show that QC entailment satisfies both three basic inference rules (i.e., MP, MT and DS) and the intuitive equivalence. Therefore, QC \mathcal{ALC} is more suitable to deal with inconsistencies than four-valued description logic \mathcal{ALC} . A tableau algorithm for QC \mathcal{ALC} , called *QC tableau algorithm*, is proposed to implement paraconsistent reasoning in \mathcal{ALC} based on a notion called *complement of an axiom* which is used to reverse both the information of being true and being false under QC semantics. Furthermore, we state that our QC tableau algorithm is decidable, sound and complete in deciding whether a QC ABox is QC consistent. Finally, we show that the complexity of QC consistency checking of a QC \mathcal{ALC} ABox is PSPACE-complete.

The rest of this paper is organized as follows. In the next section, we give a short introduction of \mathcal{ALC} . In Section 3, we introduce the semantics of QC \mathcal{ALC} . In Section 4, we consider two basic reasoning tasks of QC \mathcal{ALC} , namely QC consistency problem and QC entailment problem. In Section 5, we present a QC tableau algorithm for QC \mathcal{ALC} and prove that it is decidable, sound and complete. Finally, we summarize this paper and give some future works in the concluding section.

2 PRELIMINARIES

In this section, we introduce some basic notions of DLs, a well-known family of knowledge representation formalisms. For more comprehensive back-

ground knowledge, we refer the reader to the Description Logic Handbook [1] and Chapter 3 of the Handbook of Knowledge Representation [34].

DLs are fragments of *first-order logic*. That is, they can be translated into first-order logic [7]. DLs are different from their predecessors such as semantic networks and frames in that they are equipped with a formal, logic-based semantics. In DLs, elementary descriptions are *concept names* (unary predicates) and *role names* (binary predicates). Complex descriptions are built from them inductively using concept and role constructors provided by the particular DL in consideration.

In this paper, we consider the \mathcal{ALC} which is a simple yet relatively expressive DL, where \mathcal{AL} is the abbreviation of attributive language and \mathcal{C} denotes “complement”. Let N_C and N_R be pairwise disjoint and countably infinite sets of *concept names* and *role names* respectively. Let N_I be an infinite set of *individual names*. We use the letters A and B for concept names, the letter R for role names, and the letters C and D for concepts. \top and \perp denote the *top concept* and the *bottom concept* respectively. The set of \mathcal{ALC} concepts is the smallest set such that:

- every concept name is a concept;
- if C and D are concepts, R is a role name, then the following expressions are also concepts: $\neg C$ (*full negation*), $C \sqcap D$ (*concept conjunction*), $C \sqcup D$ (*concept disjunction*), $\forall R.C$ (*value restriction on role names*) and $\exists R.C$ (*existential restriction on role names*).

For example, the concept description $Person \sqcap Female$ is an \mathcal{ALC} -concept describing those people that are female. Suppose *hasChild* is a role name, the concept description $Person \sqcap \forall hasChild.Female$ expresses those people whose children are all female. The concept $\forall hasChild.\perp \sqcap Person$ describes those people who have no children.

An interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ consists of a non-empty domain $\Delta^{\mathcal{I}}$ and a mapping $\cdot^{\mathcal{I}}$ which maps every concept to a subset of $\Delta^{\mathcal{I}}$ and every role to a subset of $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ such that the following conditions are satisfied:

$$\begin{aligned}
\top^{\mathcal{I}} &= \Delta^{\mathcal{I}} \\
\perp^{\mathcal{I}} &= \emptyset^{\mathcal{I}} \\
(\neg C)^{\mathcal{I}} &= \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}} \\
(C_1 \sqcap C_2)^{\mathcal{I}} &= C_1^{\mathcal{I}} \cap C_2^{\mathcal{I}} \\
(C_1 \sqcup C_2)^{\mathcal{I}} &= C_1^{\mathcal{I}} \cup C_2^{\mathcal{I}} \\
(\exists R.C)^{\mathcal{I}} &= \{x \mid \exists y, (x, y) \in R^{\mathcal{I}} \text{ and } y \in C^{\mathcal{I}}\} \\
(\forall R.C)^{\mathcal{I}} &= \{x \mid \forall y, (x, y) \in R^{\mathcal{I}} \text{ implies } y \in C^{\mathcal{I}}\}
\end{aligned}$$

where C, C_1, C_2 are all concepts and R a role.

A general concept inclusion axiom (GCI) or a *terminological axiom* is an inclusion statement of the form $C \sqsubseteq D$, where C and D are two (possibly complex) \mathcal{ALC} concepts (concepts for short). It is the statement about how concepts are related to each other. We use $C \equiv D$ as an abbreviation for the symmetrical pair of GCIs $C \sqsubseteq D$ and $D \sqsubseteq C$, called an *equality*. An interpretation \mathcal{I} satisfies a GCI $C \sqsubseteq D$ if and only if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$, and it satisfies a GCI $C \equiv D$ if and only if $C^{\mathcal{I}} = D^{\mathcal{I}}$. A finite set of GCIs is called a *TBox*.

An equality whose left-hand side is an atomic concept is a *definition*. That is, a definition has the form of $A \equiv C$ where A is an atomic concept and C a concept. Let A, B be atomic concepts occurring in \mathcal{T} . We say that A *directly uses* B in \mathcal{T} if B appears on the right-hand side of the definition of A , and we define *uses* to be the transitive closure of the relation *directly uses*. Then \mathcal{T} contains *cycle* if and only if there exists an atomic concept in \mathcal{T} that uses itself. Otherwise, \mathcal{T} is called *acyclic*. For instance, $\{Human \equiv Animal \sqcap \forall hasParent.Human\}$ contains cycle.

We can also formulate statements about individuals. A *concept (role) (assertion) axiom* has the form $C(a) (R(a, b))$, where C is a concept, R a role name, and a, b *individual names*. An *ABox* consists of a finite set of concept axioms and role axioms. Concept assertion axioms, role assertion axioms and GCIs are axioms. In an ABox, one describes a specific fact of an application domain in terms of concept and roles. To give a semantics to ABoxes, we need to extend interpretations to individual names. For each individual name a , \mathcal{I} maps it to an element $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$. An interpretation \mathcal{I} satisfies a concept axiom $C(a)$ if and only if $a^{\mathcal{I}} \in C^{\mathcal{I}}$. \mathcal{I} satisfies a role axiom $R(a, b)$ if and only if $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}}$. An ontology \mathcal{O} consists of a TBox and an ABox, i.e., it is a set of GCIs and assertion axioms. An interpretation \mathcal{I} is a *model* of a DL (TBox or ABox) axiom if and only if it satisfies this axiom, and it is a model of an ontology \mathcal{O} if and only if it satisfies every axiom in \mathcal{O} . A concept D *subsumes* a concept C with respect to a TBox \mathcal{T} iff each model of \mathcal{T} is a model of axiom $C \sqsubseteq D$. An ABox \mathcal{A} is *consistent* iff there exists a model of \mathcal{A} . An ABox \mathcal{A} is *consistent* with respect to a TBox \mathcal{T} iff there exists a common model of \mathcal{T} and \mathcal{A} . Given an ontology \mathcal{O} and a DL axiom ϕ , we say \mathcal{O} *entails* ϕ , denoted as $\mathcal{O} \models \phi$, if and only if every model of \mathcal{O} is a model of ϕ . A concept C is *satisfiable* with respect to a TBox \mathcal{T} if and only if there exists a model \mathcal{I} of \mathcal{T} such that $C^{\mathcal{I}} \neq \emptyset$; and *unsatisfiable* otherwise.

Two basic reasoning problems, namely, *instance checking* (checking whether an individual is an instance of a given concept) and *subsumption checking* (checking whether a concept subsumes a given concept) can be

reduced to the problem of consistency by the following lemma.

Lemma 2.1 ([1]) *Let \mathcal{O} be an ontology, C, D concepts and a an individual in \mathcal{ALC} . Then*

- (1) $\mathcal{O} \models C(a)$ if and only if $\mathcal{O} \cup \{\neg C(a)\}$ is inconsistent;
- (2) $\mathcal{O} \models C \sqsubseteq D$ if and only if $\mathcal{O} \cup \{C \sqcap \neg D(\iota)\}$ is inconsistent where ι is a new individual not occurring in \mathcal{O} .

The problem of checking consistency of an \mathcal{ALC} ABox is PSPACE-Complete [31]. The problem of checking consistency of an \mathcal{ALC} ABox with respect to an acyclic \mathcal{ALC} TBox is also PSPACE-Complete [14]. However, the problem of checking consistency of an \mathcal{ALC} ABox with respect to a general \mathcal{ALC} TBox is EXPTIME-Complete [29, 8].

The following lemma presented by Horrocks et al [15] shows that satisfiability, unsatisfiability and consistency of a concept with respect to any general TBox can be reduced to the corresponding reasoning task with respect to the empty TBox. This result is obtained by introducing a “universal” role U , that is, if y is reachable from x via a role path, then $\langle x, y \rangle \in U^{\mathcal{I}}$. Technically, the universal role U is used to add new assertion $\neg C \sqcup D(x)$, where x is an individual occurring in \mathcal{A} , into \mathcal{A} corresponding to each GCI $C \sqsubseteq D$ of a TBox so that the problem about reasoning with ABoxes and TBoxes could be reduced to the problem about reasoning with only ABoxes.

Lemma 2.2 ([15]) *Let C, D be concepts, \mathcal{A} an ABox and \mathcal{T} a TBox in \mathcal{ALC} . Define*

$$C_{\mathcal{T}} := \bigcap_{C_i \sqsubseteq D_i \in \mathcal{T}} \neg C_i \sqcup D_i.$$

Then the following properties hold:

- (1) C is satisfiable with respect to \mathcal{T} if and only if $C \sqcap C_{\mathcal{T}} \sqcap \forall U.C_{\mathcal{T}}$ is satisfiable;
- (2) D subsumes C with respect to \mathcal{T} if and only if $C \sqcap \neg D \sqcap C_{\mathcal{T}} \sqcap \forall U.C_{\mathcal{T}}$ is unsatisfiable;
- (3) \mathcal{A} is consistent with respect to \mathcal{T} if and only if $\mathcal{A} \cup \{C_{\mathcal{T}} \sqcap \forall U.C_{\mathcal{T}}(a) \mid a \in N_A(\mathcal{A})\}$ is consistent, where $N_A(\mathcal{A})$ is a set of all individuals occurring in \mathcal{A} .

Because a reasoning problem with respect to ABoxes and general TBoxes can be reduced to the same reasoning problem with respect to only ABoxes, this paper mainly considers reasoning problems with respect to ABoxes without any TBox.

3 QUASI-CLASSICAL SEMANTICS FOR DESCRIPTION LOGIC

In this section, we define *quasi-classical \mathcal{ALC}* as an extension of \mathcal{ALC} with QC semantics given in [6].

The syntax of QC \mathcal{ALC} follows that of \mathcal{ALC} with a new kind of axiom called *complement of an axiom*. Let ϕ be an axiom in \mathcal{ALC} . The complement of ϕ is denoted by $\sim \phi$. The complement of an axiom is similar to a *signed proposition* $NT\alpha$ (presented in [32]) which means that α is not true. The intuition behind complement of an axiom is to reverse both the information of being true and of being false. Note that the notion of complement of an axiom will provide a resolution-based decision procedure for QC \mathcal{ALC} .

We denote the language of QC \mathcal{ALC} as $\mathcal{L}^* = \mathcal{L} \cup \{\sim \phi \mid \phi \in \mathcal{L}\}$, where \mathcal{L} is the language of \mathcal{ALC} .

For example, let $\mathcal{A} = \{Penguin(tweety), \sim (\neg Bird(tweety)), \neg Fly(tweety), \exists HasChild.Penguin(tweety)\}$ and $\mathcal{T} = \{\sim (Bird \sqsubseteq Fly)\}$. Thus \mathcal{A} and \mathcal{T} are an ABox and a TBox of the language \mathcal{L}^* respectively. In the following, we mainly consider the language \mathcal{L}^* .

In QC \mathcal{ALC} , let A be a concept name and R a role. A and $\neg A$ are *concept literals*. A concept C is in *negation normal form* (or NNF) if negation (\neg) only occurs in front of concept names in C . A *role-involved literal* has the form $\forall R.C$ or $\exists R.C$ where C is a concept in NNF. A *literal*, denoted by a letter like L , is either a concept literal or a role-involved literal. A *clause* is the disjunction of a finite number of literals. Let $L_1 \sqcup \dots \sqcup L_n$ be a clause, then $Lit(L_1 \sqcup \dots \sqcup L_n)$ is the set of literals $\{L_1, \dots, L_n\}$ that are in the clause. A clause is an *empty clause* if it has no literals. To simplify notations, we use $\bar{\cdot}$ to denote a *complementation operation* such that $\bar{\bar{A}}$ is $\neg A$ and $\bar{\bar{(\neg A)}}$ is A .

A *QC ABox* is a finite set of concept and role assertion axioms, complements of concept assertion axioms or complements of role assertion axioms; and a *QC TBox* is a set of inclusions or complements of inclusions. A *QC ontology* consists of a QC ABox and a QC TBox.

Definition 3.1 Let $L_1 \sqcup \dots \sqcup L_n$ be a clause that includes a literal disjunct L_i . The focus of $L_1 \sqcup \dots \sqcup L_n$ by L_i , denoted by $\otimes(L_1 \sqcup \dots \sqcup L_n, L_i)$, is defined as the clause obtained by removing L_i from $Lit(L_1 \sqcup \dots \sqcup L_n)$. In the case of a clause with just one disjunct, we assume $\otimes(L, L) = \perp$.

Example 3.1 Given a clause $L_1 \sqcup L_2 \sqcup L_3$, $\otimes(L_1 \sqcup L_2 \sqcup L_3, L_2) = L_1 \sqcup L_3$.

In the following, we define a *weak interpretation* and a *strong interpretation* over domain $\Delta^{\mathcal{I}}$ by assigning to each concept C a pair $\langle +C, -C \rangle$ of

subsets of $C^{\mathcal{I}}$. Intuitively, $+C$ is the set of elements known to belong to the extension of C , while $-C$ is the set of elements known not to be contained in the extension of C . $+C$ and $-C$ are not necessarily disjoint and mutually complementary with respect to the domain. $+C$ and $-C$ are *positive extension* and *negative extension* of C respectively.

We define the *complemental set* of a set S with respect to an interpretation \mathcal{I} to be $\bar{S} = \Delta^{\mathcal{I}} \setminus S$.

In QC \mathcal{ALC} , a weak interpretation is a reformulation of a four-valued interpretation in four-valued \mathcal{ALC} (or $\mathcal{ALC4}$) presented in [20].

Definition 3.2 Let \mathcal{I} be a pair $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where $\Delta^{\mathcal{I}}$ is a domain and $\cdot^{\mathcal{I}}$ is a function assigning an element of $\Delta^{\mathcal{I}}$ to an individual, a pair $\langle +C, -C \rangle$ of subsets of $\Delta^{\mathcal{I}}$ to a concept C , and a pair $\langle +R, -R \rangle$ of subsets of $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ to a role R . In particular, $U^{\mathcal{I}} = \langle +U, -U \rangle$ where $+U = \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ and $-U = \emptyset$. \mathcal{I} is a weak interpretation in QC \mathcal{ALC} if the following conditions are satisfied:

$$\begin{aligned}
\top^{\mathcal{I}} &= \langle \Delta^{\mathcal{I}}, \emptyset \rangle \\
\perp^{\mathcal{I}} &= \langle \emptyset, \Delta^{\mathcal{I}} \rangle \\
(\neg C)^{\mathcal{I}} &= \langle -C, +C \rangle \text{ where } C^{\mathcal{I}} = \langle +C, -C \rangle \\
(C_1 \sqcap C_2)^{\mathcal{I}} &= \langle +C_1 \cap +C_2, -C_1 \cup -C_2 \rangle \\
(C_1 \sqcup C_2)^{\mathcal{I}} &= \langle +C_1 \cup +C_2, -C_1 \cap -C_2 \rangle \\
(\exists R.C)^{\mathcal{I}} &= \langle \{x \mid \exists y, (x, y) \in +R \text{ and } y \in +C\}, \\
&\quad \{x \mid \forall y, (x, y) \in +R \text{ implies } y \in -C\} \rangle \\
(\forall R.C)^{\mathcal{I}} &= \langle \{x \mid \forall y, (x, y) \in +R \text{ implies } y \in +C\}, \\
&\quad \{x \mid \exists y, (x, y) \in +R \text{ and } y \in -C\} \rangle
\end{aligned}$$

where $C_i^{\mathcal{I}} = \langle +C_i, -C_i \rangle$ for $i = 1, 2$ and $R^{\mathcal{I}} = \langle +R, -R \rangle$.

From Definition 3.2, for any weak interpretation \mathcal{I} and any concept C , the negation of a concept $(\neg C)^{\mathcal{I}}$ is a pair obtained by exchanging two parts of $C^{\mathcal{I}}$. Intuitively, $\neg C$ could be taken as a “new” concept which is symmetric to C . In the following, we show some properties of the negation of concepts based on weak interpretations.

Theorem 3.1 Let C be a concept and \mathcal{I} a weak interpretation. If $C^{\mathcal{I}} = \langle +C, -C \rangle$, then we have

- (1) $+(\neg C) = -C$;
- (2) $-(\neg C) = +C$;
- (3) $+(\neg\neg C) = +C$;
- (4) $-(\neg\neg C) = -C$.

Theorem 3.1 shows that the *law of double negation* is valid by weak interpretations.

In the following, we define the notion of weak satisfaction.

Definition 3.3 Let \models_w be a satisfiability relation between a set of QC weak interpretations and a set of axioms, called weak satisfaction. For a weak interpretation \mathcal{I} , we define \models_w as follows:

- (1) $\mathcal{I} \models_w C(a)$ if and only if $a^{\mathcal{I}} \in +C$, where $C^{\mathcal{I}} = \langle +C, -C \rangle$;
- (2) $\mathcal{I} \models_w R(a, b)$ if and only if $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in +R$, where $R^{\mathcal{I}} = \langle +R, -R \rangle$;
- (3) $\mathcal{I} \models_w C \sqsubseteq D$ if and only if $+C \subseteq +D$, where $C^{\mathcal{I}} = \langle +C, -C \rangle$ and $D^{\mathcal{I}} = \langle +D, -D \rangle$.

Here C, D are concepts, R a role and a an individual in \mathcal{L}^* .

In Definition 3.3, an inclusion is interpreted as an *internal inclusion* as defined in $\mathcal{ALC}4$. That is, a weak interpretation satisfies an inclusion $C \sqsubseteq D$ if and only if every element that is known to belong to the positive extension of C is known to belong to the positive extension of D under the weak satisfaction.

In the following, we give the interpretation of a complement of an axiom based on weak interpretations.

Definition 3.4 The interpretation of a complement of an axiom under weak interpretation is defined as follows: for any weak interpretation \mathcal{I} , concepts C, C_1, C_2 , role R and individual a in \mathcal{L}^* ,

- (1) $\mathcal{I} \models_w \sim C(a)$ if and only if $a^{\mathcal{I}} \notin +C$ where $C^{\mathcal{I}} = \langle +C, -C \rangle$.
- (2) $\mathcal{I} \models_w \sim R(a, b)$ if and only if $(a^{\mathcal{I}}, b^{\mathcal{I}}) \notin +R$ where $R^{\mathcal{I}} = \langle +R, -R \rangle$.
- (3) $\mathcal{I} \models_w \sim C \sqsubseteq D$ if and only if there exists an individual a such that $a^{\mathcal{I}} \in +C$ and $a^{\mathcal{I}} \notin +D$ where $C^{\mathcal{I}} = \langle +C, -C \rangle$ and $D^{\mathcal{I}} = \langle +D, -D \rangle$.

The following theorem provides some basic properties of the weak satisfaction.

Theorem 3.2 Let C, D, E be concepts, R a role, a, b individuals and \mathcal{I} a weak interpretation in QC \mathcal{ALC} . The following properties hold.

- (1) $\mathcal{I} \models_w \neg C(a)$ if and only if $a^{\mathcal{I}} \in -C$, where $C^{\mathcal{I}} = \langle +C, -C \rangle$.
- (2) $\mathcal{I} \models_w \neg \neg C(a)$ if and only if $\mathcal{I} \models_w C(a)$.
- (3) $\mathcal{I} \models_w C \sqcup D(a)$ if and only if $\mathcal{I} \models_w C(a)$ or $\mathcal{I} \models_w D(a)$.
- (4) $\mathcal{I} \models_w C \sqcap D(a)$ if and only if $\mathcal{I} \models_w C(a)$ and $\mathcal{I} \models_w D(a)$.
- (5) $\mathcal{I} \models_w \neg(C \sqcap D)(a)$ if and only if $\mathcal{I} \models_w \neg C \sqcup \neg D(a)$.
- (6) $\mathcal{I} \models_w \neg(C \sqcup D)(a)$ if and only if $\mathcal{I} \models_w \neg C \sqcap \neg D(a)$.
- (7) $\mathcal{I} \models_w C \sqcup (D \sqcap E)(a)$ if and only if $\mathcal{I} \models_w (C \sqcup D) \sqcap (C \sqcup E)(a)$.

- (8) $\mathcal{I} \models_w C \sqcap (D \sqcup E)(a)$ if and only if $\mathcal{I} \models_w (C \sqcap D) \sqcup (C \sqcap E)(a)$.
 (9) $\mathcal{I} \models_w \neg \exists R.C(a)$ if and only if $\mathcal{I} \models_w \forall R. \neg C(a)$.
 (10) $\mathcal{I} \models_w \neg \forall R.C(a)$ if and only if $\mathcal{I} \models_w \exists R. \neg C(a)$.

The properties (3) and (4) say that, under the weak satisfaction, an individual belongs to a disjunction (resp. conjunction) of two concepts if and only if it belongs to either (both) of the concepts. The properties (9) and (10) show that under weak satisfaction, the duality between existential restriction and value restriction holds; and the properties (2), (5) and (6), (7) and (8) respectively show that the weak satisfaction validates double the negation law, De Morgan's law and distributive law.

The following theorem shows that the intuitive equivalence between inclusion and disjunction of concepts with respect to complement of an axiom is satisfied under the weak semantics.

Theorem 3.3 *Let \mathcal{I} be a weak interpretation and C, D concepts in \mathcal{L}^* .*

$$\mathcal{I} \models_w C \sqsubseteq D \text{ iff for any individual } a, \mathcal{I} \models_w \sim C(a) \text{ or } \mathcal{I} \models_w D(a).$$

Theorem 3.3 provides for a theoretical foundation to transform the problems of subsumption checking into the problems of QC consistency checking (defined in Section 4 later).

The following theorem provides some important properties about the complement of an axiom under the weak satisfaction.

Theorem 3.4 *Given concepts C, D , individuals a, b , a role R and a weak interpretation \mathcal{I} in \mathcal{L}^* , the following properties hold.*

- (1) $\mathcal{I} \models_w \sim (C \sqcap D)(a)$ if and only if $\mathcal{I} \models_w \sim C(a)$ or $\mathcal{I} \models_w \sim D(a)$.
 (2) $\mathcal{I} \models_w \sim (C \sqcup D)(a)$ if and only if $\mathcal{I} \models_w \sim C(a)$ and $\mathcal{I} \models_w \sim D(a)$.
 (3) $\mathcal{I} \models_w \sim (\forall R.C)(a)$ if and only if there is an individual name b such that $\mathcal{I} \models_w R(a, b)$ and $\mathcal{I} \models_w \sim C(b)$.
 (4) $\mathcal{I} \models_w \sim (\exists R.C)(a)$ if and only if for any individual b if $\mathcal{I} \models_w R(a, b)$ then $\mathcal{I} \models_w \sim C(b)$.

In Theorem 3.4, the properties (1) and (2) show that the QC weak satisfaction satisfies the DeMorgan's law, when complement of an axiom is considered. The properties (3) and (4) show the complements of existential restriction assertions and value restriction assertions could be transformed into complements of assertions.

In QC \mathcal{ALC} , a weak interpretation \mathcal{I} is a *weak model* of an axiom ϕ if and only if $\mathcal{I} \models_w \phi$. \mathcal{I} is a *weak model* of a QC ontology \mathcal{O} , denoted by $\mathcal{I} \models_w \mathcal{O}$ if and only if \mathcal{I} is a weak model of every axiom in \mathcal{O} .

Note that the weak semantics does not assure that every \mathcal{ALC} ontology has weak QC models. This fact also holds for the four-valued DL defined in [33] if \top and \perp are used arbitrarily. The following example illustrates this.

Example 3.2 Consider $\mathcal{T} = \{\top \sqsubseteq \perp\}$. Since for any weak interpretation \mathcal{I} , $\top^{\mathcal{I}} = \langle \Delta^{\mathcal{I}}, \emptyset \rangle$ and $\perp^{\mathcal{I}} = \langle \emptyset, \Delta^{\mathcal{I}} \rangle$ where $\Delta^{\mathcal{I}} \neq \emptyset$ for weak interpretations, \mathcal{T} has no weak model according to Definition 3.3.

To preserve the weak satisfaction, we employ the satisfiable substitution in [21] which is presented to maintain the four-valued satisfaction in $\mathcal{ALC4}$. Given an ontology \mathcal{O} , the satisfiable form of \mathcal{O} (denoted by $SF(\mathcal{O})$) is the ontology obtained by replacing each occurrence of \top in \mathcal{O} with $NA \sqcup \neg NA$, and replacing each occurrence of \perp in \mathcal{O} with $NA \sqcap \neg NA$, where NA is a new concept name.

In Example 3.2, $\top \sqsubseteq \perp$ can be replaced by $NA \sqcup \neg NA \sqsubseteq NA \sqcap \neg NA$.

We take advantage of the substitution which preserves the weak satisfaction. That is, for any ontology \mathcal{O} , $SF(\mathcal{O})$ always has at least one weak model.

Because of the above discussion, we assume that all ontologies discussed in the rest of this paper have weak models.

Moreover, the weak satisfaction like the four-valued satisfaction does not satisfy some basic inference rules, such as MP, MT and DS. Therefore, we define a strong interpretation by redefining the interpretations of disjunction of concepts and conjunction of concepts.

Definition 3.5 Let \mathcal{I} be a pair $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ where $\Delta^{\mathcal{I}}$ is a domain and $\cdot^{\mathcal{I}}$ is a function assigning an element of $\Delta^{\mathcal{I}}$ to an individual, a pair $\langle +C, -C \rangle$ of subsets of $\Delta^{\mathcal{I}}$ to a concept C , and a pair $\langle +R, -R \rangle$ of subsets of $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ to a role R . \mathcal{I} is a strong interpretation in QC \mathcal{ALC} if the conditions in Definition 3.2, except those for conjunction of concepts and disjunction of concepts, are satisfied and the following two conditions are also satisfied:

$$\begin{aligned} \text{conjunction of concepts: } (C_1 \sqcap C_2)^{\mathcal{I}} &= \langle +C_1 \cap +C_2, (-C_1 \cup -C_2) \cap (-C_1 \cup \overline{+C_2}) \cap (\overline{+C_1} \cup -C_2) \rangle; \\ \text{disjunction of concepts: } (C_1 \sqcup C_2)^{\mathcal{I}} &= \langle (+C_1 \cup +C_2) \cap (\overline{-C_1} \cup +C_2) \cap (+C_1 \cup \overline{-C_2}), -C_1 \cap -C_2 \rangle; \end{aligned}$$

where $C^{\mathcal{I}} = \langle +C, -C \rangle$ and $C_i^{\mathcal{I}} = \langle +C_i, -C_i \rangle$ for $i = 1, 2$.

Compared with the weak interpretation, the strong interpretation of disjunction of concepts tightens the condition that an individual is known to belong to a concept. For instance, based on strong interpretations, an individual a is known to be an instance of $C_1 \sqcup C_2$ if and only if

- a is an instance of C_1 or a is an instance of C_2 ;
- if a is also an instance of $\neg C_1$ then a must be an instance of C_2 ; and
- if a is also an instance of $\neg C_2$ then a must be an instance of C_1 .

The strong interpretation of conjunction of concepts is defined by relaxing the condition that an individual is known not to be contained in the positive extension of a concept. For instance, based on strong interpretations, an individual a is known not to be contained in the positive extension of $C_1 \sqcap C_2$ if and only if

- a is an instance of $\neg C_1$ or a is an instance of $\neg C_2$;
- if a is known to be an instance of C_1 then a must be an instance of $\neg C_2$; and
- if a is an instance of C_2 then a must be an instance of $\neg C_1$.

The strong interpretation for disjunction and conjunction are introduced to ensure that the three basic inference rules (i.e., MP, MT, DS) are satisfied in QC \mathcal{ALC} .

Corresponding to Theorem 3.1, we show that the *law of double negation* is also valid based on strong interpretations.

Theorem 3.5 *Let C be a concept and \mathcal{I} a strong interpretation. If $C^{\mathcal{I}} = \langle +C, -C \rangle$, then we have*

- (1) $+(\neg C) = -C$;
- (2) $-(\neg C) = +C$;
- (3) $+(\neg\neg C) = +C$;
- (4) $-(\neg\neg C) = -C$.

Similar to the weak satisfaction, we introduce the notion of *strong satisfaction*.

Definition 3.6 *Let \models_s be a satisfiability relation between a set of strong interpretations and a set of axioms, called strong satisfaction. For any strong interpretation \mathcal{I} , we define \models_s as follows:*

- (1) $\mathcal{I} \models_s C(a)$ if and only if $a^{\mathcal{I}} \in +C$, where $C^{\mathcal{I}} = \langle +C, -C \rangle$;
- (2) $\mathcal{I} \models_s R(a, b)$ if and only if $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in +R$, where $R^{\mathcal{I}} = \langle +R, -R \rangle$;
- (3) $\mathcal{I} \models_s C \sqsubseteq D$ if and only if $\overline{-C} \subseteq +D$, $+C \subseteq +D$ and $-D \subseteq -C$, where $C^{\mathcal{I}} = \langle +C, -C \rangle$ and $D^{\mathcal{I}} = \langle +D, -D \rangle$.

Here C, D are concepts, R a role and a an individual in QC \mathcal{ALC} .

In Definition 3.6, when defining the strong satisfaction of an inclusion based on strong interpretations, we use conditions for interpreting an inclusion by combining the interpretations of three inclusions, so-called *internal inclusion*, *material inclusion* and *strong inclusion* as defined in four-valued DL. By doing so, our strong satisfaction satisfies the intuitive equivalence that is falsified by the four-valued satisfaction.

Theorem 3.6 *Let \mathcal{I} be a strong interpretation and C, D concepts in \mathcal{L}^* .*

$\mathcal{I} \models_s C \sqsubseteq D$ if and only if $\mathcal{I} \models_s \neg C \sqcup D(a)$, for any individual a .

Theorem 3.6 provides a theoretical base of transforming the problem of reasoning with ABoxes and terminologies into the problem of reasoning with ABoxes.

The following theorem provides a slightly different view on the strong satisfaction of disjunction of concepts.

Theorem 3.7 *Let L_i ($i = 1, \dots, n$) be a literal and a an individual in \mathcal{L}^* . $\mathcal{I} \models_s L_1 \sqcup \dots \sqcup L_n(a)$ if and only if $\mathcal{I} \models_s L_1(a)$ or ... or $\mathcal{I} \models_s L_n(a)$ or $\forall i(1 \leq i \leq n), \mathcal{I} \models_s \neg L_i(a)$ implies $\mathcal{I} \models_s \otimes(L_1 \sqcup \dots \sqcup L_n, L_i)(a)$*

The following theorem provides some basic properties of the strong satisfaction. These properties are important because they will ensure the soundness and completeness of our QC tableau algorithm (presented in Section 5 later).

Theorem 3.8 *Let C, D, E be concepts, R a role, a, b individuals and \mathcal{I} a strong interpretation in \mathcal{L}^* . The following properties hold.*

- (1) $\mathcal{I} \models_s \neg C(a)$ if and only if $a^{\mathcal{I}} \in -C$, where $C^{\mathcal{I}} = \langle +C, -C \rangle$.
- (2) $\mathcal{I} \models_s \neg \neg C(a)$ if and only if $\mathcal{I} \models_s C(a)$.
- (3) $\mathcal{I} \models_s C \sqcap D(a)$ if and only if $\mathcal{I} \models_s C(a)$ and $\mathcal{I} \models_s D(a)$.
- (4) $\mathcal{I} \models_s \neg(C \sqcap D)(a)$ if and only if $\mathcal{I} \models_s \neg C \sqcup \neg D(a)$.
- (5) $\mathcal{I} \models_s \neg(C \sqcup D)(a)$ if and only if $\mathcal{I} \models_s \neg C \sqcap \neg D(a)$.
- (6) $\mathcal{I} \models_s \neg \exists R.C(a)$ if and only if $\mathcal{I} \models_s \forall R. \neg C(a)$.
- (7) $\mathcal{I} \models_s \neg \forall R.C(a)$ if and only if $\mathcal{I} \models_s \exists R. \neg C(a)$.

Property (3) says that, under the strong satisfaction, an individual belongs to a conjunction of two concepts if and only if it belongs to both concepts. Properties (6) and (7) show that under the strong satisfaction, the duality between existential restriction and value restriction holds; property (2) shows that the strong satisfaction satisfies the double complement law; and properties (4) and (5) show that the strong satisfaction satisfies the De Morgan's law.

Definition 3.7 *The interpretation of a complement of an axiom under strong interpretation is defined as follows: for any strong interpretation \mathcal{I} , concepts C, D , role R and individual a in \mathcal{L}^* ,*

- (1) $\mathcal{I} \models_s \sim C(a)$ if and only if $a^{\mathcal{I}} \notin +C$ where $C^{\mathcal{I}} = \langle +C, -C \rangle$.
- (2) $\mathcal{I} \models_s \sim R(a, b)$ if and only if $(a^{\mathcal{I}}, b^{\mathcal{I}}) \notin +R$ where $R^{\mathcal{I}} = \langle +R, -R \rangle$.
- (3) $\mathcal{I} \models_s \sim C \sqsubseteq D$ if and only if there exists an individual a such that $a^{\mathcal{I}} \in \overline{-C}$ and $a^{\mathcal{I}} \notin +D$, or $a^{\mathcal{I}} \in +C$ and $a^{\mathcal{I}} \notin +D$, or $a^{\mathcal{I}} \in -D$ and $a^{\mathcal{I}} \notin -C$, where $C^{\mathcal{I}} = \langle +C, -C \rangle$ and $D^{\mathcal{I}} = \langle +D, -D \rangle$.

The following theorem provides some important properties about the complement of an axiom under the strong satisfaction.

Theorem 3.9 *Given concepts C, D , individuals a, b and a role R in \mathcal{L}^* , the following properties hold.*

- (1) $\mathcal{I} \models_s \sim (C \sqcap D)(a)$ if and only if $\mathcal{I} \models_s \sim C(a)$ or $\mathcal{I} \models_s \sim D(a)$.
- (2) $\mathcal{I} \models_s \sim (C \sqcup D)(a)$ if and only if $\mathcal{I} \models_s \sim C(a)$ and $\mathcal{I} \models_s \sim D(a)$, or $\mathcal{I} \models_s \neg C(a)$ and $\mathcal{I} \models_s \sim D(a)$, or $\mathcal{I} \models_s \neg D(a)$ and $\mathcal{I} \models_s \sim C(a)$.
- (3) $\mathcal{I} \models_s \sim (\forall R.C)(a)$ if and only if there is an individual name b such that $\mathcal{I} \models_s R(a, b)$ and $\mathcal{I} \models_s \sim C(b)$.
- (4) $\mathcal{I} \models_s \sim (\exists R.C)(a)$ if and only if for any individual b if $\mathcal{I} \models_s R(a, b)$ then $\mathcal{I} \models_s \sim C(b)$.

In Theorem 3.9, properties (1) and (2) show that weak satisfaction satisfies double complement law and DeMorgan's law, when complement of an axiom is considered. Properties (3) and (4) show the conditions of satisfiability of existential restriction assertion axioms and complement of value restriction assertion axioms respectively.

In QC \mathcal{ALC} , a strong interpretation \mathcal{I} is a *strong model* of an axiom ϕ if and only if $\mathcal{I} \models_s \phi$. \mathcal{I} is a *strong model* of a QC ontology \mathcal{O} , denoted by $\mathcal{I} \models_s \mathcal{O}$ if and only if \mathcal{I} is a strong model of every axiom in \mathcal{O} .

The following theorem shows the relationship between weak models and strong models.

Theorem 3.10 *Let \mathcal{I} be a weak interpretation and ϕ an axiom in \mathcal{ALC} .*

$$\text{If } \mathcal{I} \models_s \phi \text{ then } \mathcal{I} \models_w \phi.$$

This theorem shows that a strong model is also a weak model. As a result, the reasoning power of strong satisfaction is no stronger than weak satisfaction. In fact, strong satisfaction is weaker than weak satisfaction. To see this, consider the following example.

Example 3.3 Let $\mathcal{A} = \{C(a), \neg C(a)\}$ be an ABox in \mathcal{L}^* . Let \mathcal{I} be a weak interpretation such that $\Delta^{\mathcal{I}} = \{a^{\mathcal{I}}\}$, $C^{\mathcal{I}} = \{\{a^{\mathcal{I}}\}, \{a^{\mathcal{I}}\}\}$ and $D^{\mathcal{I}} = \langle \emptyset, \{a^{\mathcal{I}}\} \rangle$. Clearly, $\mathcal{I} \models_w C \sqcup D(a)$ while $\mathcal{I} \not\models_s C \sqcup D(a)$ since $\mathcal{I} \not\models_s D(a)$. This example illustrates that Theorem 3.10 does not hold vice versa.

The above discussion shows that while the paraconsistent entailment \models_s satisfies some useful reasoning rules, it is too weak. It implies none of \models_s and \models_w is a suitable paraconsistent semantics for quasi-classical description logic (QCDL).

For this reason, we introduce a novel consequence relation in terms of both the weak and the strong satisfaction relations. We define a QC entailment which is of the same form as classical entailment except that we use the strong satisfaction for the assumptions and weak satisfaction for the inferences. It is well known that the less assumptions are contained in the premise of an entailment, the more conclusions can be drawn. Based on this fact, the strong satisfaction is employed to make less assumptions in the premise in order to make QC semantics stronger. On the other hand, the weak satisfaction is employed to ensure the conclusion tolerating inconsistencies.

Definition 3.8 Given a QC ontology \mathcal{O} and an axiom ϕ in QC \mathcal{ALC} , we say \mathcal{O} quasi-classically entails ϕ , denoted by $\mathcal{O} \models_Q \phi$, if and only if for every interpretation \mathcal{I} , if for any axiom ψ of \mathcal{O} , we have $\mathcal{I} \models_s \psi$, then $\mathcal{I} \models_w \phi$.

The following theorem shows that \models_Q satisfies the *resolution rule* which is a general form of MP, MT and DS.

Theorem 3.11 Let B, C, E be concepts and a an individual in \mathcal{ALC} . Then

$$\{B \sqcup C(a), \neg B \sqcup E(a)\} \models_Q C \sqcup E(a).$$

By Theorem 3.11, it is easy to check that \models_Q satisfies three basic inference rules (MP, MT and DS). For instance, let $\mathcal{A} = \{Penguin(tweety), \neg Fly(tweety), \neg Penguin \sqcup Fly(tweety), \neg Fly \sqcup Haswings(tweety)\}$ be an ABox in \mathcal{L}^* . It easily shows that $\mathcal{A} \models_Q Haswings(tweety)$.

Let's go back to Definition 3.8. According to Theorem 3.11, the strong satisfaction is employed to capture the decomposition of the set of assumptions because it satisfies the resolution rule. On the contrary, the weak satisfaction is employed to capture the composition of axioms from resolvent after applying the *disjunct rule*, i.e., $\{C(a)\} \models_w C \sqcup D(a)$. Based on the weak satisfaction and the strong satisfaction, QC entailment assures stronger ability of paraconsistent reasoning in QC \mathcal{ALC} .

The following example shows that QC entailment is nontrivial.

Example 3.4 Given a QC ontology $\mathcal{O} = \{B(a), \neg B(a)\}$ and a concept name A in QC \mathcal{ALC} . So $\{B(a), \neg B(a)\}$ is classically inconsistent. However it is not the case that $\mathcal{O} \models_Q A(a)$ holds, since there exists an interpretation \mathcal{I} such that $B^{\mathcal{I}} = \langle +B, -B \rangle$, $A^{\mathcal{I}} = \langle +A, -A \rangle$ where $+B = \{a^{\mathcal{I}}\}$, $-B = \{a^{\mathcal{I}}\}$ and $+A = -A = \emptyset$. So $\mathcal{I} \models_s B \sqcap \neg B(a)$, but $\mathcal{I} \not\models_w A(a)$ since $A(a)$ does not occur in \mathcal{O} .

Example 3.5 Now consider the universal concept $\top = A \sqcup \neg A$ where A is a concept name. Here $\not\models_Q A \sqcup \neg A(a)$ for any individual a since there exists an interpretation \mathcal{I} such that $\mathcal{I} \not\models_w A \sqcup \neg A(a)$. However, we have $\models A \sqcup \neg A(a)$ since $\models \top(a)$.

Example 3.5 shows that the *excluded middle law* fails in QC \mathcal{ALC} which does not hold in four-valued DLs. The failure of the excluded middle law still occurs in QC logic (see [6, 17]). In general, tautologies in classical \mathcal{ALC} , is not a problem in practice because software engineers use DLs for reasoning about some specification. So tautologies tell them nothing useful for their tasks. Contradictions could not be analogously entailed. Because of this, in this paper, we mainly consider ontologies with neither tautologies nor contradictions.

4 REASONING IN QUASI-CLASSICAL DESCRIPTION LOGIC

In this section, we mainly discuss the QC *consistency* problem and two basic inference problems in QC \mathcal{ALC} , namely, QC *instance checking* and QC *subsumption checking*.

Firstly, we consider the problem of QC consistency of QC \mathcal{ALC} ABoxes. In classical DLs, the problems of instance checking and subsumption checking can be reduced to the problem of satisfiability checking based on Lemma 2.1. In the following, analogously, we want to reduce the problems of QC instance checking and QC subsumption checking into the problem of consistency checking in QC \mathcal{ALC} . Thus, firstly, we introduce a kind of consistency under QC semantics, called QC *consistency*, in QC \mathcal{ALC} . In the following, we employ strong models to characterize our QC consistency for QC \mathcal{ALC} which is suggested in quasi-classical propositional logic (see [12]).

In QC \mathcal{ALC} , a concept C is QC *satisfiable* with respect to a QC TBox \mathcal{T} if there exists a strong model \mathcal{I} of \mathcal{T} such that $+C \neq \emptyset$ where $C^{\mathcal{I}} = \langle +C, -C \rangle$; and QC *unsatisfiable* with respect to \mathcal{T} otherwise. A QC ABox \mathcal{A} is QC *consistent* if there exists a strong model \mathcal{I} of \mathcal{A} , and QC *inconsistent*

otherwise. A QC ontology \mathcal{O} is *QC consistent* if there exists a strong model \mathcal{I} of its ABox and its TBox, and *QC inconsistent* otherwise.

Next, we mainly discuss the two basic inference problems (*quasi-classical instance checking* (*QC instance checking*) and *quasi-classical subsumption checking* (*QC subsumption checking*)) in QC \mathcal{ALC} .

- **QC instance checking:** an individual a is a *quasi-classical instance* (*QC instance for short*) of a concept C with respect to an ABox \mathcal{A} if $\mathcal{A} \models_Q C(a)$;
- **QC subsumption checking:** a concept C is *quasi-classically subsumed* (*QC subsumed*) by a concept D with respect to a TBox \mathcal{T} if $\mathcal{T} \models_Q C \sqsubseteq D$.

In the following theorem, we show that there exists a close relation between two basic inference problems and QC consistency problem.

Theorem 4.1 *Let \mathcal{O} be an ontology and C, D concepts in \mathcal{ALC} .*

- (1) $\mathcal{O} \models_Q C(a)$ if and only if $\mathcal{O} \cup \{\sim C(a)\}$ is QC inconsistent.
- (2) $\mathcal{O} \models_Q C \sqsubseteq D$ if and only if $\mathcal{O} \cup \{C(\iota), \sim D(\iota)\}$ is QC inconsistent where ι is a new individual not occurring in \mathcal{O} .

By Theorem 4.1, two basic inference problems can be reduced to the QC consistency problem in QC \mathcal{ALC} .

Corresponding to classical \mathcal{ALC} , in QC \mathcal{ALC} , we use $C \equiv_Q D$ as an abbreviation for the symmetrical pair of GCIs $C \sqsubseteq D$ and $D \sqsubseteq C$, we call it a *quasi-classical equality* (*QC equality for short*). A weak interpretation \mathcal{I} satisfies a QC equality $C \equiv_Q D$ if and only if $+C = +D$ where $C^{\mathcal{I}} = \langle +C, -C \rangle$ and $D^{\mathcal{I}} = \langle +D, -D \rangle$. A strong interpretation \mathcal{I} satisfies a QC equality $C \equiv_Q D$ if and only if (1) $+C = +D$ and $-C = -D$; and (2) $+C \cup -C = \Delta^{\mathcal{I}}$ and $+D \cup -D = \Delta^{\mathcal{I}}$ where $C^{\mathcal{I}} = \langle +C, -C \rangle$ and $D^{\mathcal{I}} = \langle +D, -D \rangle$. Analogously, we define *QC definitions* and *acyclic QC TBoxes*.

Note that to check two basic problems (instance checking and subsumption checking) for ontologies contains both ABoxes and TBoxes and queries in \mathcal{ALC} , the complement of concept axioms will be added into ABoxes during the process of reducing two basic problems to QC consistency problem. That is to say, TBoxes are not changed in the process of reduction. Therefore, we only need to consider a classical TBox and a QC ABox to get the following corollary which shows a similar result as Lemma 2.2 for QC semantics with respect to acyclic TBoxes.

Corollary 4.1 *Let C, D be concepts, \mathcal{A} a QC ABox and \mathcal{T} a TBox in \mathcal{ALC} . Then the following properties hold:*

- (1) *C is QC satisfiable with respect to \mathcal{T} if and only if $C \sqcap C_{\mathcal{T}} \sqcap \forall U.C_{\mathcal{T}}$ is QC satisfiable;*
- (2) *D QC subsumes C with respect to \mathcal{T} if and only if $C \sqcap \neg D \sqcap C_{\mathcal{T}} \sqcap \forall U.C_{\mathcal{T}}$ is QC unsatisfiable;*
- (3) *\mathcal{A} is QC consistent with respect to \mathcal{T} if and only if $\mathcal{A} \cup \{C_{\mathcal{T}} \sqcap \forall U.C_{\mathcal{T}}(a) \mid a \in N_A(\mathcal{A})\}$ is QC consistent, where $C_{\mathcal{T}}$ and $N_A(\mathcal{A})$ are defined in Lemma 2.2.*

By Theorem 4.1, the proof of Corollary 4.1 is similar to that of Lemma 2.2 that can be found in [29].

Corollary 4.1 provides an approach to checking consistency of QC ABoxes with respect to general QC TBoxes. Because of this, the problems about reasoning with respect to QC ABoxes and general QC TBoxes can be reduced to those of reasoning with respect to QC ABoxes.

In a word, the two basic inference problems can be reduced to the problem of QC consistency checking of a QC ABox by Theorem 4.1 and Corollary 4.1 in QC \mathcal{ALC} . So we mainly discuss the QC consistency problem in the following.

5 QC TABLEAU ALGORITHM

In \mathcal{ALC} , *tableau-based algorithms* have been proposed to decide consistency of ontologies. Schmidt-Schauß and Smolka in [31] presented a tableau algorithm for checking consistency of an \mathcal{ALC} ABox. In Section 2, we have shown that the basic problems of reasoning in DLs can be reduced to the consistency problem for ABoxes by Lemma 2.1 and Lemma 2.2. The main idea of tableau-based algorithms for deciding consistency of an ABox \mathcal{A} is given as follows: the algorithm starts with the ABox \mathcal{A} (all concepts occurring in \mathcal{A} is in NNF) and applies consistency preserving expansion rules (see Figure 2.6 in the Description Logic Handbook [1]) to the ABox until no more rules to be applied. If the “complete” ABox obtained in this way does not contain an obvious contradiction (called *clash*) then \mathcal{A} is consistent, otherwise it is inconsistent. In this section, we develop a tableau algorithm called *QC tableau algorithm* for deciding QC inconsistency of \mathcal{ALC} QC ABoxes.

By Theorem 3.8 and Theorem 3.4, it suffices to put axioms $\phi \in \mathcal{L}^*$ into NNF by pushing negation inwards in polynomial time. For instance, the NNF of $\neg(\neg B_1 \sqcup \forall R.B_2)(a)$ is $B_1 \sqcap \exists R.\neg B_2(a)$ where B_1, B_2 are concept names,

\sqcap -rule:	If $C_1 \sqcap C_2(x) \in \mathcal{A}$, but $\{C_1(x), C_2(x)\} \not\subseteq \mathcal{A}$, then $\mathcal{A}' := \mathcal{A} \cup \{C_1(x), C_2(x)\}$.
\sqcup -rule:	If $C_1 \sqcup C_2(x) \in \mathcal{A}$, but $\{C_1(x), C_2(x), \neg C_1(x), \neg C_2(x)\} \cap \mathcal{A} = \emptyset$, then $\mathcal{A}' := \mathcal{A} \cup \{C_1(x)\}$, $\mathcal{A}'' := \mathcal{A} \cup \{C_2(x)\}$.
\exists -rule:	If $\exists R.C(x) \in \mathcal{A}$, but there is no individual name z such that $\{C(z), R(x, z)\} \subseteq \mathcal{A}$, then $\mathcal{A}' := \mathcal{A} \cup \{C(y), R(x, y)\}$ $\cup \{U(a, y), U(y, a) \mid a \in N_A(\mathcal{A})\}$ where y is an individual name not occurring in \mathcal{A} .
\forall -rule:	If $\{\forall R.C(x), R(x, y)\} \subseteq \mathcal{A}$, but $C(y) \notin \mathcal{A}$, then $\mathcal{A}' := \mathcal{A} \cup \{C(y)\}$.
QC-rule:	If $C_1 \sqcup C_2(x) \in \mathcal{A}$ and $\neg C_i(x) \in \mathcal{A}, i \in \{1, 2\}$, then $\mathcal{A}' := \mathcal{A} \cup \{\otimes(C_1 \sqcup C_2, C_i)(x)\}$.
$\sim \sqcap$ -rule:	If $\sim (C_1 \sqcap C_2)(x) \in \mathcal{A}$, but $\{\sim C_1(x), \sim C_2(x)\} \cap \mathcal{A} = \emptyset$, then $\mathcal{A}' := \mathcal{A} \cup \{\sim C_1(x)\}$, $\mathcal{A}'' := \mathcal{A} \cup \{\sim C_2(x)\}$.
$\sim \sqcup$ -rule:	If $\sim (C_1 \sqcup C_2)(x) \in \mathcal{A}$, but $\{\sim C_1(x), \sim C_2(x)\} \not\subseteq \mathcal{A}$, then $\mathcal{A}' := \mathcal{A} \cup \{\sim C_1(x), \sim C_2(x)\}$.
$\sim \exists$ -rule:	If $\{\sim (\exists R.C)(x), R(x, y)\} \subseteq \mathcal{A}$, but $\sim C(y) \notin \mathcal{A}$, then $\mathcal{A}' := \mathcal{A} \cup \{\sim C(y)\}$.
$\sim \forall$ -rule:	If $\sim (\forall R.C)(x) \in \mathcal{A}$, but there is no individual name z such that $\{\sim C(z), R(x, z)\} \subseteq \mathcal{A}$, then $\mathcal{A}' := \mathcal{A} \cup \{\sim C(y), R(x, y)\}$ $\cup \{U(a, y), U(y, a) \mid a \in N_A(\mathcal{A})\}$ where y is an individual name not occurring in \mathcal{A} .

TABLE 1: Quasi-Classical Expansion Rules

R a role name and a an individual. Without loss of generality, we assume that all concepts C occurring in \mathcal{A} are in NNF. Secondly, we introduce *QC expansion rules* which contain nine rules in the following definition.

Definition 5.1 *Let \mathcal{A} be a QC ABox, C_1, C_2, C concepts, R a role and x, y, z individuals, respectively. The QC expansion rules in QC \mathcal{ALC} are defined in Table 1.*

In Definition 5.1, four of the nine QC expansion rules, namely, \sqcap -rule, \sqcup -rule, \exists -rule and \forall -rule, are directly borrowed from classical expansion rules for decomposition of axioms. The QC-rule is introduced to make the *resolu-*

tion rule $\frac{C \sqcup D(a), \neg C \sqcup E(a)}{D \sqcup E(a)}$ be satisfied in the reasoning system of QC \mathcal{ALC} . Other four rules, namely, $\sim \sqcap$ -rule, $\sim \sqcup$ -rule, $\sim \exists$ -rule and $\sim \forall$ -rule are introduced to the decompose complement of axioms.

In the following, we introduce the concepts of *clash* and *complete* in QC \mathcal{ALC} . A QC ABox \mathcal{A} is called *complete* if and only if none of the QC expansion rules of Table 1 applies to it. A QC ABox \mathcal{A} contains a *clash* if for some concept name A , some role name R and some individuals a, b , $\{A(a), \sim A(a)\} \subseteq \mathcal{A}$ or $\{\neg A(a), \sim (\neg A(a))\} \subseteq \mathcal{A}$ or $\{R(a, b), \sim (R(a, b))\} \subseteq \mathcal{A}$. A QC ABox \mathcal{A} is *closed* if it contains a clash; and *clash-free* otherwise. A set of QC ABoxes \mathcal{S} is *closed* if each QC ABox of \mathcal{S} is closed; and *clash-free* otherwise.

We present our *QC tableau algorithm* for deciding the QC consistency of QC ABoxes. The QC tableau algorithm starts with a QC ABox \mathcal{A} (all concepts occurring in \mathcal{A} are in NNF). If a universal role U occurs in \mathcal{A} , then we initialize \mathcal{A} by adding all role assertions $U(a, b)$ where $a, b \in N_A(\mathcal{A})$. Then the algorithm applies consistency preserving QC expansion rules in Table 1 to the QC ABox until no more rules to be applied. If the set of QC “complete” ABoxes \mathcal{S} obtained in this way does not contain a clash then \mathcal{A} is QC consistent; otherwise, \mathcal{A} is QC inconsistent. The transformation rules that handle disjunction and at-most restrictions are *non-deterministic* in the sense that a given QC ABox is transformed into finitely many new QC ABoxes such that the original QC ABox is QC consistent if and only if *one of* the new QC ABoxes is so. For this reason, we will consider finite sets of QC ABoxes $\mathcal{S} = \{\mathcal{A}_1, \dots, \mathcal{A}_k\}$ instead of single QC ABoxes. Such a set is QC consistent iff there is some $i (1 \leq i \leq k)$ such that \mathcal{A}_i is QC consistent. A rule of Table 1 is applied to a given finite set of ABoxes \mathcal{S} as follows: it takes an element \mathcal{A} of \mathcal{S} , and replaces it by one QC ABox \mathcal{A}' , by two QC ABoxes \mathcal{A}' and \mathcal{A}'' , or by finitely many ABoxes $\mathcal{A}_{i,j}$.

As a result, we show that the QC tableau algorithm for any finite QC ABox can terminate in finite steps.

Theorem 5.1 *The QC tableau algorithm terminates.*

In the following, we show that the QC tableau algorithm for deciding the QC consistency of QC ABoxes is sound and complete. Firstly, we present a lemma as the basis of completeness.

Lemma 5.1 *Let \mathcal{A} be a QC ABox. If \mathcal{S} is a set of QC ABoxes obtained by applying a QC expansion rule in Table 1, then we have \mathcal{A} is QC consistent if and only if \mathcal{S} is QC consistent.*

From Lemma 5.1, we can show that the QC tableau algorithm for deciding the consistency of QC ABoxes is sound.

Theorem 5.2 (Soundness) *Let \mathcal{A} be a QC ABox in QC \mathcal{ALC} . If S is a set of QC ABoxes obtained by applying QC expansion rules, then we have \mathcal{A} is QC consistent if and only if S is QC consistent.*

Next, we show that the QC tableau algorithm for deciding the consistency of QC ABoxes is complete.

Theorem 5.3 (Completeness) *Any complete and clash-free QC ABox \mathcal{A} has a strong model.*

In the following examples, we use the QC tableau algorithm for QC instance checking with respect to classically inconsistent ABoxes.

Example 5.1 *Given an ABox $\mathcal{A} = \{Bird(tweety), \neg Bird \sqcup Fly(tweety), \neg Fly \sqcap Penguin(tweety)\}$ and a query $Fly(tweety)$. A sequence of new ABoxes can be obtained from $\mathcal{A} \cup \{\sim Fly(tweety)\}$ by using the tableau algorithm as follows:*

$\mathcal{A}_1 = \mathcal{A} \cup \{\sim Fly(tweety)\}$.

$\mathcal{A}_2 = \mathcal{A}_1 \cup \{Fly(tweety)\}$ for $Bird(tweety)$ and $\neg Bird \sqcup Fly(tweety)$ by using the QC-rule.

$\mathcal{A}_3 = \mathcal{A}_2 \cup \{\neg Fly(tweety), Penguin(tweety)\}$ for $\neg Fly \sqcap Penguin(tweety)$ by using the \sqcap -rule.

So $S = (\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3)$ and \mathcal{A}_3 is the last ABox of the sequence since the QC tableau algorithm terminates. It is easy to show that \mathcal{A}_2 is closed because $Fly(tweety)$ and $\sim Fly(tweety)$ are in \mathcal{A}_2 . So $\mathcal{A} \models_Q Fly(tweety)$ by Theorem 5.2 and Theorem 5.3. Though \mathcal{A} is classical inconsistent, supposed that $Wounded(tweety)$ is a new query then $Wounded(tweety)$ cannot be QC entailed by \mathcal{A} , i.e., $\mathcal{A} \not\models_Q Wounded(tweety)$, because the last ABox in the sequence which is obtained after the QC tableau algorithm is $\{Bird(tweety), \neg Bird \sqcup Fly(tweety), \neg Fly \sqcap Penguin(tweety), \sim Wounded(tweety), Fly(tweety), \neg Fly(tweety), Penguin(tweety)\}$ which is not closed.

Example 5.2 *Let $\mathcal{A} = \{\neg Boy(June), HasFriend(Mike, June), \forall HasFriend.Boy \sqcup Girl(Mike), \neg Girl(Mike)\}$ be an \mathcal{ALC} ABox and $\exists HasFriend.Girl(Mike)$ a query. A sequence S of new ABoxes can be obtained from $\mathcal{A} \cup \{\sim (\exists HasFriend.Girl)(Mike)\}$ by using the QC tableau algorithm as follows:*

$\mathcal{A}_1 = \mathcal{A} \cup \{\sim (\exists \text{HasFriend.Girl})(\text{June})\}.$
 $\mathcal{A}_2 = \mathcal{A}_1 \cup \{\text{Boy} \sqcup \text{Girl}(\text{June})\}$ for $\text{HasFriend}(\text{Mike}, \text{June})$ and
 $\forall \text{HasFriend.Boy} \sqcup \text{Girl}(\text{Mike})$ by using \forall -rule.
 $\mathcal{A}_3 = \mathcal{A}_2 \cup \{\text{Girl}(\text{June})\}$ for $\neg \text{Boy}(\text{June})$ and $\text{Boy} \sqcup \text{Girl}(\text{June})$ by using
 QC -rule.
 $\mathcal{A}_4 = \mathcal{A}_3 \cup \{\sim \text{Girl}(\text{June})\}$ for $\text{HasFriend}(\text{Mike}, \text{June})$ and
 $\sim (\exists \text{HasFriend.Girl})(\text{Mike})$ by using $\sim \exists$ -rule.

So $\mathcal{S} = (\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3, \mathcal{A}_4)$ and \mathcal{A}_4 is the last ABox of the sequence since
 the QC tableau algorithm terminates and \mathcal{A}_4 is closed because it has a clash
 $\text{Girl}(\text{June})$ and $\sim \text{Girl}(\text{June})$. So $\mathcal{A} \models_Q \exists \text{HasFriend.Girl}(\text{Mike})$ by
 Theorem 5.2 and Theorem 5.3.

Example 5.3 Given an ABox $\mathcal{A} = \{C \sqcup D(a)\}$ and a query $\phi = C(a)$.

$\mathcal{A}_1 = \{C \sqcup D(a)\} \cup \{\sim C(a)\};$
 $\mathcal{A}_{21} = \{C \sqcup D(a), \sim C(a), C(a)\}$ or $\mathcal{A}_{22} = \{C \sqcup D(a), \sim C(a), D(a)\}$ by
 using \rightarrow_{\sqcup} -rule.

We denote $\hat{\mathcal{S}} = \{\mathcal{A}_{21}, \mathcal{A}_{22}\}$. $\hat{\mathcal{S}}$ is not closed since \mathcal{A}_{22} is open. So $\mathcal{A} \not\models_Q$
 $C(a)$ by Theorem 5.2 and Theorem 5.3.

The above examples exemplifies that the QC tableau algorithm can be
 employed to implement the problem about instance checking in inconsistent
 ABoxes by Theorem 5.2 and Theorem 5.3.

Theorem 5.4 Complexity of QC consistency of QC ABoxes without a universal role U is PSPACE-Complete.

Theorem 5.5 Complexity of QC consistency of QC ABoxes with respect to a general TBox is EXPTIME-Complete.

Theorem 5.4 and Theorem 5.5 show that the complexity of checking QC
 consistency of ontologies in QC \mathcal{ALC} is no higher than that of checking consistency of ontologies in \mathcal{ALC} .

6 RELATED WORK

In this paper, following from our previous work [35, 36], we present a paraconsistent version of description logic by introducing and revising the classical QC semantics presented in [6, 17]. Compared with QC propositional logic [6] and QC first-order logic [17], the new constructor called complement of an axiom, which is helpful for implementing paraconsistent reasoning via

tableau algorithm, is introduced in the syntax of our quasi-classical DL. In this sense, we generalize classical QC logics and discuss some interesting reasoning tasks such as QC satisfiable problem and QC inconsistent problem.

There are many existing paraconsistent approaches to dealing with inconsistencies in DLs [24, 33, 16, 20, 21, 22, 23, 11, 37, 38]. In [16], inconsistency-tolerant reasoning is based on selecting maximal consistent subsets from inconsistent ontologies. Different from argumentation-based proposals in [37, 38] for inconsistency-tolerant reasoning with DL-based ontologies, our scenario is based on multi-valued semantics. Note that current paraconsistent approaches to handling inconsistency in DL-based ontologies mainly follow from multi-valued semantics such as [24, 33, 20, 21, 22, 23, 11]. The main idea of them is applying Belanp's four-valued semantics [5] to implement inconsistency-tolerant reasoning with DL-based ontologies. Compared with single four-valued interpretations to capture four-valued models in four-valued DL, two interpretations (weak interpretations and strong interpretations) are introduced to ensure the inference power of QC entailment stronger with holding paraconsistency. Weak models based on weak interpretations are similar to four-valued models while strong models based on strong interpretations are obtained by refining four-valued models. In our quasi-classical DL, three CGIs of four-valued DL, namely, material inclusion (\vdash), internal inclusion (\sqsubseteq) and strong inclusion (\rightarrow), are integrated into an inclusion. The relationship between quasi-classical DL and four-valued DL can be shown in the following theorems.

Theorem 6.1 *Let C, D be concepts, a an individual, R a role and \mathcal{I} a weak interpretation in \mathcal{ALC} . We have*

- (1) $\mathcal{I} \models_w C(a)$ if and only if $\mathcal{I} \models_4 C(a)$;
- (2) $\mathcal{I} \models_w R(a, b)$ if and only if $\mathcal{I} \models_4 R(a, b)$;
- (3) $\mathcal{I} \models_w C \sqsubseteq D$ if and only if $\mathcal{I} \models_4 C \sqsubseteq D$.

Proof 1 *These properties follow directly from Definition 3.2, Definition 3.3 and the definition of four-valued interpretations in four-valued DL.*

Theorem 6.1 shows that each four-valued model of an ontology is a weak model of the ontology where inclusions are interpreted as the internal inclusions in four-valued DL.

Theorem 6.2 *Let C, D be concepts and a an individual in \mathcal{L}^* .*

- (1) *if $\mathcal{I} \models_s C(a)$ then $\mathcal{I} \models_4 C(a)$;*
- (2) *if $\mathcal{I} \models_s C \sqsubseteq D$ then $\mathcal{I} \models_4 C \propto D$, where \propto is a place-holder of \vdash, \sqsubseteq and \rightarrow and \models_4 is the four-valued entailment in four-valued DL.*

Proof 2 *This theorem follows directly from Definition 3.5, Theorem 6.1 and Theorem 3.10.*

Theorem 6.2 states that each strong model of an ontology is a four-valued model of the ontology. As a result, our QC entailment based on weak models and strong models ensures three basic inference rules, namely, MP, MT and DS, valid in QCDL while they are not so in four-valued DL.

Moreover, compared with existing reasoning algorithm for DL-based ontologies, we modify standard tableau algorithm to implement paraconsistent reasoning with taking the advantage of quasi-classical DL.

7 CONCLUSIONS

In this paper, we have introduced QC semantics into \mathcal{ALC} to handle inconsistency. QC semantics was defined by weak semantics and strong semantics. Weak semantics is a reformulation of a four-valued semantics for \mathcal{ALC} . In order to make the QC semantics satisfy the resolution rules, strong semantics was introduced by restricting disjunction of concepts in \mathcal{ALC} . Because none of weak semantics and strong semantics is a suitable paraconsistent semantics for QC DLs, we have proposed the QC semantics for DLs by combining two semantics in order to take their respective advantages. Intuitively, weak interpretations are used to retain paraconsistency while strong interpretations help in obtaining stronger reasoning power.

Compared with four-valued DL, we have redefined concept inclusion (or subsumption) in quasi-classical DL so that intuitive equivalence is valid under the QC semantics. For this purpose, concept inclusion under weak semantics is defined by internal inclusion of four-valued description logic and concept inclusion under strong semantics is defined by hybrid of three inclusions (material inclusion, internal inclusion and strong inclusion) of four-valued DL. A QC tableau algorithm for instance checking in a QC ABox has been proposed in this paper. We have proved that our algorithm is terminable, sound and complete. Moreover, we have also showed that the complexity of checking QC consistency for QC \mathcal{ALC} ABoxes is PSPACE-Complete; and the complexity of checking QC consistency for QC \mathcal{ALC} ABoxes with respect to a general TBox is EXPTIME-Complete. As a future work, based on this algorithm, we will develop a paraconsistent reasoner to implement QC reasoning tasks.

8 ACKNOWLEDGMENTS

The authors would like to thank the peer reviewers and colleagues Guilin Qi, Yue Ma, Guohui Xiao and Ulrike Sattler for their helpful and constructive comments. This work was supported by the program of the National Natural Science Foundation of China under grant number 60973003.

REFERENCES

- [1] Franz Baader, Diego Calvanese, Deborah L. McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, editors. (2003). *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press.
- [2] Franz Baader and Philipp Hanschke. (1991). A scheme for integrating concrete domains into concept languages. In *Proceedings of the 17th International Joint Conference on Artificial Intelligence(IJCAI'01)*, Sydney, Australia, August 24-30, 1991, pages 452–457. Morgan Kaufmann.
- [3] Franz Baader and Bernhard Hollunder. (1991). A terminological knowledge representation system with complete inference algorithms. In *Proceedings of the International Workshop of Processing Declarative Knowledge (PDK'91)*, Kaiserslautern, Germany, July 1-3, 1991, volume 567 of *Lecture Notes in Computer Science*, pages 67–86. Springer.
- [4] Franz Baader, Ian Horrocks, and Ulrike Sattler. (2007). Description Logics. In Frank van Harmelen, Vladimir Lifschitz, and Bruce Porter, editors, *Handbook of Knowledge Representation*. Elsevier.
- [5] N. D. Belnap. (1977). A useful four-valued logic. *Modern uses of multiple-valued logics*, pages 7–73.
- [6] Philippe Besnard and Anthony Hunter. (1995). Quasi-classical logic: Non-trivializable classical reasoning from inconsistent information. In *Proceedings of the 3rd European Conference on Symbolic and Quantitative Approaches to Reasoning and Uncertainty (EC-SQARU'95)*, Fribourg, Switzerland, July 3-5, volume 946 of *Lecture Notes in Computer Science*, pages 44–51. Springer.
- [7] Alexander Borgida. (1994). On the relationship between description logic and predicate logic. In *Proceedings of the Third International Conference on Information and Knowledge Management (CIKM'94)*, Gaithersburg, Maryland, November 29 - December 2, 1994., pages 219–225. ACM.
- [8] Francesco M. Donini and Fabio Massacci. (2000). EXPTIME tableaux for ALC. *Artificial Intelligence*, 124(1):87–138.
- [9] Michael J. Fischer and Richard E. Ladner. (1979). Propositional dynamic logic of regular programs. *Journal of Computer and System Sciences*, 18(2):194–211.
- [10] Giorgos Flouris, Zhisheng Huang, Jeff Z. Pan, Dimitris Plexousakis, and Holger Wache. (2006). Inconsistencies, negations and changes in ontologies. In *Proceedings of the 21st National Conference on Artificial Intelligence and the Eighteenth Innovative Applications of Artificial Intelligence Conference (AAAI'06)*, July 16-20, 2006, Boston, Massachusetts, USA. AAAI Press.
- [11] Sergio Alejandro Gomez, Carlos Ivan Chesnevar, and Guillermo Ricardo Simari. (2008). An argumentative approach to reasoning with inconsistent ontologies. In *Proceedings of the Knowledge Representation Ontology Workshop (KROW'08)*, Sydney, Australia, September 17, 2008., volume 90. Australian Computer Society.

- [12] John Grant and Anthony Hunter. (2006). Measuring inconsistency in knowledgebases. *Journal of Intelligence Information Systems*, 27(2):159–184.
- [13] Peter Haase, Frank van Harmelen, Zhisheng Huang, Heiner Stuckenschmidt, and York Sure. (2005). A framework for handling inconsistency in changing ontologies. In *Proceedings of the 4th International Semantic Web Conference (ISWC'05), Galway, Ireland, November 6-10, 2005*, volume 3729 of *Lecture Notes in Computer Science*, pages 353–367. Springer.
- [14] Jan Hladik and Rafael Peñaloza. (2006). PSPACE automata for description logics. In *Proceedings of the 2006 International Workshop on Description Logics (DL'06), Windermere, Lake District, UK, May 30 - June 1, 2006.*, volume 189 of *CEUR Workshop Proceedings*. CEUR-WS.org.
- [15] Ian Horrocks, Ulrike Sattler, and Stephan Tobies. (2000). Reasoning with individuals for the description logic SHIQ. In *Proceedings of the 17th International Conference on Automated Deduction (CADE'00), Pittsburgh, PA, USA, June 17-20, 2000.*, volume 1831 of *Lecture Notes in Computer Science*, pages 482–496. Springer.
- [16] Zhisheng Huang, Frank van Harmelen, and Annette ten Teije. (2005). Reasoning with inconsistent ontologies. In *Proceedings of the 19th International Joint Conference on Artificial Intelligence (IJCAI'05), Edinburgh, Scotland, UK, July 30-August 5, 2005*, pages 454–459. Professional Book Center.
- [17] Anthony Hunter. (2000). Reasoning with contradictory information using quasi-classical logic. *Journal of Logic and Computation*, 10(5):677–703.
- [18] Aditya Kalyanpur, Bijan Parsia, Matthew Horridge, and Evren Sirin. (2007). Finding all justifications of OWL DL entailments. In *Proceedings of the 6th International Semantic Web Conference/ the 2nd Asian Semantic Web Conference (ISWC'07 / ASWC'07), Busan, Korea, November 11-15, 2007.*, volume 4825 of *Lecture Notes in Computer Science*, pages 267–280. Springer.
- [19] Richard E. Ladner. (1977). The computational complexity of provability in systems of modal propositional logic. *SIAM Journal on Computing*, 6(3):467–480.
- [20] Yue Ma, Pascal Hitzler, and Zuoquan Lin. (2007). Algorithms for paraconsistent reasoning with OWL. In *Proceedings of the 4th European Semantic Web Conference (ESWC'07), Innsbruck, Austria, June 3-7, 2007*, volume 4519 of *Lecture Notes in Computer Science*, pages 399–413. Springer.
- [21] Yue Ma, Pascal Hitzler, and Zuoquan Lin. (2007). Paraconsistent resolution for four-valued description logics. In *Proceedings of the 20th International Workshop on Description Logics (DL'07), Brixen-Bressanone, near Bozen-Bolzano, Italy, 8-10 June, 2007*, volume 250 of *CEUR Workshop Proceedings*. CEUR-WS.org.
- [22] Yue Ma, Pascal Hitzler, and Zuoquan Lin. (2008). Paraconsistent reasoning for expressive and tractable description logics. In *Proceedings of the 21st International Workshop on Description Logics (DL'08), Dresden, Germany, May 13-16, 2008*, volume 353 of *CEUR Workshop Proceedings*. CEUR-WS.org.
- [23] Sergei P. Odintsov and Heinrich Wansing. (2008). Inconsistency-tolerant description logic. part ii: A tableau algorithm for $CACL^C$. *Journal of Applied Logic*, 6(3):343–360.
- [24] Peter F. Patel-Schneider. (1989). A four-valued semantics for terminological logics. *Artificial Intelligence*, 38(3):319–351.
- [25] Guilin Qi and Jianfeng Du. (2009). Model-based revision operators for terminologies in description logics. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI'09), California, USA, July 11-17, 2009.*, pages 891–897.

- [26] Guilin Qi, Peter Haase, Zhisheng Huang, Qiu Ji, Jeff Z. Pan, and Johanna Völker. (2008). A kernel revision operator for terminologies - algorithms and evaluation. In *Proceedings of the 7th International Semantic Web Conference (ISWC'08), Karlsruhe, Germany, October 26-30, 2008.*, volume 5318 of *Lecture Notes in Computer Science*, pages 419–434. Springer.
- [27] Guilin Qi, Weiru Liu, and David A. Bell. (2006). A revision-based approach to handling inconsistency in description logics. *Artificial Intelligence Review*, 26(1-2):115–128.
- [28] Guilin Qi, Jeff Z. Pan, and Qiu Ji. (2007). Extending description logics with uncertainty reasoning in possibilistic logic. In *Proceedings of the 9th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU'07), Hammamet, Tunisia, October 31 - November 2, 2007*, volume 4724 of *Lecture Notes in Computer Science*, pages 828–839. Springer.
- [29] Klaus Schild. (1991). A correspondence theory for terminological logics: Preliminary report. In *Proceedings of the 12th International Joint Conference on Artificial Intelligence (IJCAI'91), Sydney, Australia, August 24-30, 1991.*, pages 466–471. Morgan Kaufmann.
- [30] Stefan Schlobach and Ronald Cornet. (2003). Non-standard reasoning services for the debugging of description logic terminologies. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI'03), Acapulco, Mexico, August 9-15, 2003*, pages 355–362. Morgan Kaufmann.
- [31] Manfred Schmidt-Schauß and Gert Smolka. (1991). Attributive concept descriptions with complements. *Artificial Intelligence*, 48(1):1–26.
- [32] Umberto Straccia. (1997). A four-valued fuzzy propositional logic. In *Proceedings of the 15th International Joint Conference on Artificial Intelligence (IJCAI'97), Nagoya, Japan, August 23-29, 1997.*, pages 128–135. Morgan Kaufmann.
- [33] Umberto Straccia. (1997). A sequent calculus for reasoning in four-valued description logics. In *Proceedings of the 6th International Conference of Automated Reasoning with Analytic Tableaux and Related Methods, (TABLEAUX'97), Pont-à-Mousson, France, May 13-16, 1997*, volume 1227 of *Lecture Notes in Computer Science*, pages 343–357. Springer.
- [34] Frank van Harmelen, Vladimir Lifschitz, and Bruce Porter, editors. (2007). *Handbook of Knowledge Representation (Foundations of Artificial Intelligence)*. Elsevier Science.
- [35] Xiaowang Zhang and Zuoquan Lin. (2008). Paraconsistent reasoning with quasi-classical semantics in ALC. In *Proceedings of the 2nd International Conference on Web Reasoning and Rule Systems (RR'08), Karlsruhe, Germany, October 31-November 1, 2008.*, volume 5341 of *Lecture Notes in Computer Science*, pages 222–229. Springer.
- [36] Xiaowang Zhang, Guilin Qi, Yue Ma, and Zuoquan Lin. (2009). Quasi-classical semantics for expressive description logics. In *Proceedings of the 22nd International Workshop on Description Logics (DL'09), Oxford, UK, July 27-30, 2009.*, volume 477 of *CEUR Workshop Proceedings*. CEUR-WS.org.
- [37] Xiaowang Zhang, Zhihu Zhang, and Zuoquan Lin. (2009). An argumentative semantics for paraconsistent reasoning in description logic ALC. In *Proceedings of the 22nd International Workshop on Description Logics (DL'09), Oxford, United Kingdom, July 27-30, 2009.*, volume 353 of *CEUR Workshop Proceedings*. CEUR-WS.org.
- [38] Xiaowang Zhang, Zhihu Zhang, Dai Xu, and Zuoquan Lin. (2010). Argumentation-based reasoning with inconsistent knowledge bases. In *Proceedings of the 23rd Canadian Conference on Artificial Intelligence (Canadian AI'10), Ottawa, Canada, May 31 - June 2, 2010.*, volume 6085 of *Lecture Notes in Computer Science*, pages 87–99. Springer.

APPENDIX

Proof of Theorem 3.1

(1) and (2) follow directly from Definition 3.2. (3) and (4) follow directly from (1) and (2).

Proof of Theorem 3.2

In all proofs of this paper, the non-logical notation \Leftrightarrow is used to link the two logical expressions which are equivalent. We use the definitions of a weak interpretation (Definition 3.2) and the weak satisfaction (Definition 3.3) and Theorem 3.1 to prove these properties.

(1)

$$\begin{aligned} \mathcal{I} \models_w \neg C(a) &\Leftrightarrow a^{\mathcal{I}} \in +(\neg C), \text{ where } (\neg C)^{\mathcal{I}} = \langle -C, +C \rangle \\ &\Leftrightarrow a^{\mathcal{I}} \in -C \text{ where } C^{\mathcal{I}} = \langle +C, -C \rangle. \end{aligned}$$

(2)

$$\begin{aligned} \mathcal{I} \models_w \neg\neg C(a) &\Leftrightarrow a^{\mathcal{I}} \in +(\neg\neg C) \\ &\Leftrightarrow a^{\mathcal{I}} \in +C \\ &\Leftrightarrow \mathcal{I} \models_w C(a) \end{aligned}$$

(3)

$$\begin{aligned} \mathcal{I} \models_w C \sqcup D(a) &\Leftrightarrow a^{\mathcal{I}} \in +C \cup +D \\ &\Leftrightarrow a^{\mathcal{I}} \in +C \text{ or } a^{\mathcal{I}} \in +D \\ &\Leftrightarrow \mathcal{I} \models_w C(a) \text{ or } \mathcal{I} \models_w D(a), \end{aligned}$$

where $C^{\mathcal{I}} = \langle +C, -C \rangle$ and $D^{\mathcal{I}} = \langle +D, -D \rangle$.

(4)

$$\begin{aligned} \mathcal{I} \models_w C \sqcap D(a) &\Leftrightarrow a^{\mathcal{I}} \in +C \cap +D \\ &\Leftrightarrow \mathcal{I} \models_w C(a) \text{ and } \mathcal{I} \models_w D(a), \end{aligned}$$

where $C^{\mathcal{I}} = \langle +C, -C \rangle$ and $D^{\mathcal{I}} = \langle +D, -D \rangle$.

(5)

$$\begin{aligned} \mathcal{I} \models_w \neg(C \sqcap D)(a) &\Leftrightarrow a^{\mathcal{I}} \in +\neg(C \sqcap D) \\ &\Leftrightarrow a^{\mathcal{I}} \in +(\neg(C \sqcap D)) \\ &\Leftrightarrow a^{\mathcal{I}} \in +(\neg C) \cup +(\neg D) \\ &\Leftrightarrow \mathcal{I} \models_w \neg C \sqcup \neg D(a) \end{aligned}$$

where $(\neg(C \sqcap D))^{\mathcal{I}} = \langle +(\neg(C \sqcap D)), -(\neg(C \sqcap D)) \rangle$, $(\neg C)^{\mathcal{I}} = \langle +(\neg C), -(\neg C) \rangle$ and $(\neg D)^{\mathcal{I}} = \langle +(\neg D), -(\neg D) \rangle$.

(6) The proof of this property is similar to that of (5).

(7)

$$\begin{aligned}
\mathcal{I} \models_w C \sqcup (D \sqcap E)(a) &\Leftrightarrow a^{\mathcal{I}} \in +C \cup +(D \sqcap E) \\
&\Leftrightarrow a^{\mathcal{I}} \in +C \cup (+D \sqcap +E) \\
&\Leftrightarrow a^{\mathcal{I}} \in (+C \sqcap +D) \cup (+C \sqcap +E) \\
&\Leftrightarrow \mathcal{I} \models_w (C \sqcup D) \sqcap (C \sqcup E)(a)
\end{aligned}$$

where $(D \sqcap E)^{\mathcal{I}} = \langle +(D \sqcap E), -(D \sqcap E) \rangle$, $C^{\mathcal{I}} = \langle +C, -C \rangle$, $D^{\mathcal{I}} = \langle +D, -D \rangle$ and $E^{\mathcal{I}} = \langle +E, -E \rangle$.

(8) The proof of this property is similar to that of (7).

(9) and (10) follow directly from Definition 3.2.

Proof of Theorem 3.3

$$\begin{aligned}
\mathcal{I} \models_w C \sqsubseteq D &\Leftrightarrow \text{for any individual } a, \text{ if } a^{\mathcal{I}} \in +C \text{ then } a^{\mathcal{I}} \in +D \\
&\Leftrightarrow a^{\mathcal{I}} \notin +C \text{ or } a^{\mathcal{I}} \in +D \\
&\Leftrightarrow \mathcal{I} \models_w \sim C(a) \text{ or } \mathcal{I} \models_w D(a)
\end{aligned}$$

where $C^{\mathcal{I}} = \langle +C, -C \rangle$ and $D^{\mathcal{I}} = \langle +D, -D \rangle$.

Proof of Theorem 3.4

We apply Definition 3.3, Definition 3.4 and Theorem 3.2 to prove this theorem.

(1)

$$\begin{aligned}
\mathcal{I} \models_w \sim (C \sqcap D)(a) &\Leftrightarrow a^{\mathcal{I}} \notin +(C \sqcap D) \\
&\Leftrightarrow a^{\mathcal{I}} \notin +C \text{ or } a^{\mathcal{I}} \notin D \\
&\Leftrightarrow \mathcal{I} \models_w \sim C(a) \text{ or } \mathcal{I} \models_s \sim D(a).
\end{aligned}$$

(2)

$$\begin{aligned}
\mathcal{I} \models_w \sim (C \sqcup D)(a) &\Leftrightarrow a^{\mathcal{I}} \notin +(C \sqcup D) \\
&\Leftrightarrow a^{\mathcal{I}} \notin +C \text{ and } a^{\mathcal{I}} \notin D \\
&\Leftrightarrow \mathcal{I} \models_w \sim C(a) \text{ and } \mathcal{I} \models_w \sim D(a).
\end{aligned}$$

(3)

$$\begin{aligned}
\mathcal{I} \models_w \sim (\forall R.C)(a) &\Leftrightarrow a^{\mathcal{I}} \notin +(\forall R.C) \\
&\Leftrightarrow \text{there is an individual name } b \text{ such that} \\
&\quad (a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}} \text{ and } a^{\mathcal{I}} \notin +C \\
&\Leftrightarrow \mathcal{I} \models_w R(a, b) \text{ and } \mathcal{I} \models_w \sim C(b).
\end{aligned}$$

(4)

$$\begin{aligned}
\mathcal{I} \models_w \sim (\exists R.C)(a) &\Leftrightarrow a^{\mathcal{I}} \notin +(\exists R.C) \\
&\Leftrightarrow (a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}} \Rightarrow b^{\mathcal{I}} \notin +C \\
&\Leftrightarrow \mathcal{I} \models_w \sim C(b).
\end{aligned}$$

Here $S^{\mathcal{I}} = \langle +S, -S \rangle$ and S is a place-holder of $C, D, R, C \sqcap D, C \sqcup D, \forall R.C$ and $\exists R.C$.

Proof of Theorem 3.5

(1) and (2) follow directly from Definition 3.5. (3) and (4) follow directly from (1) and (2) respectively.

Proof of Theorem 3.6

$$\begin{aligned} & \mathcal{I} \models_s \neg C \sqcup D(a) \text{ for any individual } a \in N_I \\ \Leftrightarrow & \text{ for any individual } a \in N_I, (a^{\mathcal{I}} \in +(\neg C) \text{ or } a^{\mathcal{I}} \in +D) \text{ and} \\ & (a^{\mathcal{I}} \in +C \text{ implies } a^{\mathcal{I}} \in +D) \text{ and } (a^{\mathcal{I}} \in -D \text{ implies } a^{\mathcal{I}} \in +(\neg C)); \\ \Leftrightarrow & \overline{-C} \subseteq +D, +C \subseteq +D \text{ and } -D \subseteq -C \end{aligned}$$

where $C^{\mathcal{I}} = \langle +C, -C \rangle$ and $C^{\mathcal{I}} = \langle +D, -D \rangle$.

Proof of Theorem 3.7

We prove this property by induction over n .

Base step: $n = 1$, it easy to check that the theorem holds. Suppose $n = 2$, i.e., $\mathcal{I} \models_s L_1 \sqcup L_2(a)$. By Definition 3.5, $(L_1 \sqcup L_2)^{\mathcal{I}} = \langle (+L_1 \cup +L_2) \cap (\overline{-L_1} \cup +L_2), -L_1 \cap -L_2 \rangle$, where $L_i^{\mathcal{I}} = \langle +L_i, -L_i \rangle$ for $i = 1, 2$.

$$\begin{aligned} & \mathcal{I} \models_s L_1 \sqcup L_2(a) \\ \Leftrightarrow & a^{\mathcal{I}} \in (+L_1 \cup +L_2) \cap (\overline{-L_1} \cup +L_2) \cap (+L_1 \cup \overline{-L_2}) \\ \Leftrightarrow & a^{\mathcal{I}} \in (+L_1 \cup +L_2), a^{\mathcal{I}} \in (\overline{-L_1} \cup +L_2) \text{ and } a^{\mathcal{I}} \in (+L_1 \cup \overline{-L_2}) \\ \Leftrightarrow & a^{\mathcal{I}} \in +L_1 \text{ or } a^{\mathcal{I}} \in +L_2, \text{ if } a^{\mathcal{I}} \in -L_1 \text{ then } a^{\mathcal{I}} \in +L_2 \text{ and} \\ & \text{if } a^{\mathcal{I}} \in -L_2 \text{ then } a^{\mathcal{I}} \in +L_1 \\ \Leftrightarrow & \mathcal{I} \models_s L_1(a) \text{ or } \mathcal{I} \models_s L_2(a) \text{ and } \mathcal{I} \models_s \neg L_1(a) \text{ implies } \mathcal{I} \models_s L_2(a) \\ & \text{and } \mathcal{I} \models_s \neg L_2(a) \text{ implies } \mathcal{I} \models_s L_1(a). \end{aligned}$$

Induction step: suppose that $n > 2$, $\mathcal{I} \models_s L_1 \sqcup \dots \sqcup L_n(a) \Leftrightarrow \mathcal{I} \models_s L_1(a) \text{ or } \dots \text{ or } \mathcal{I} \models_s L_n(a)$ and $\forall i (1 \leq i \leq n), \mathcal{I} \models_s \neg L_i(a)$ implies $\mathcal{I} \models_s \otimes(L_1 \sqcup \dots \sqcup L_n, L_i)(a)$. Now we consider $n + 1$, i.e.,

$$\begin{aligned} & \mathcal{I} \models_s L_1 \sqcup \dots \sqcup L_{n+1}(a) \\ \Leftrightarrow & \mathcal{I} \models_s L_1 \sqcup \dots \sqcup L_n(a) \text{ or } \mathcal{I} \models_s L_{n+1}(a) \text{ and} \\ & \mathcal{I} \models_s \neg L_{n+1}(a) \text{ implies } \mathcal{I} \models_s L_1 \sqcup \dots \sqcup L_n(a) \\ \Leftrightarrow & \mathcal{I} \models_s L_1(a) \text{ or } \dots \text{ or } \mathcal{I} \models_s L_{n+1}(a) \text{ and} \\ & \forall i (1 \leq i \leq n + 1), \mathcal{I} \models_s \neg L_i(a) \\ & \text{implies } \mathcal{I} \models_s \otimes(L_1 \sqcup \dots \sqcup L_{n+1}, L_i)(a) \end{aligned}$$

Proof of Theorem 3.8

We apply Definition 3.5, Definition 3.6 and Theorem 3.5 to prove this theorem.

(1)

$$\begin{aligned}\mathcal{I} \models_s \neg C(a) &\Leftrightarrow a^{\mathcal{I}} \in -(\neg C) \text{ where } (\neg C)^{\mathcal{I}} = \langle -C, +C \rangle; \\ &\Leftrightarrow a^{\mathcal{I}} \in -C, \text{ where } C^{\mathcal{I}} = \langle +C, -C \rangle.\end{aligned}$$

(2)

$$\begin{aligned}\mathcal{I} \models_s \neg\neg C(a) &\Leftrightarrow a^{\mathcal{I}} \in +(\neg\neg C) \\ &\Leftrightarrow a^{\mathcal{I}} \in +C \\ &\Leftrightarrow \mathcal{I} \models_s C(a).\end{aligned}$$

(3)

$$\begin{aligned}\mathcal{I} \models_s C \sqcap D(a) &\Leftrightarrow a^{\mathcal{I}} \in (+C \sqcap +D) \\ &\Leftrightarrow a^{\mathcal{I}} \in +C \text{ and } a^{\mathcal{I}} \in +D \\ &\Leftrightarrow \mathcal{I} \models_s C(a) \text{ and } \mathcal{I} \models_s D(a).\end{aligned}$$

(4)

$$\begin{aligned}\mathcal{I} \models_s \neg(C \sqcap D)(a) & \\ \Leftrightarrow a^{\mathcal{I}} \in -(C \sqcap D) & \\ \Leftrightarrow a^{\mathcal{I}} \in (-C \sqcap -D) \cap (-C \cup \overline{+D}) \cap (\overline{+C} \cup -D) & \\ \Leftrightarrow a^{\mathcal{I}} \in (+(\neg C) \sqcap +(\neg D)) \cap (+(\neg C) \cup \overline{-(\neg D)}) \cap (\overline{-(\neg C)} \cup +(\neg D)) & \\ \Leftrightarrow a^{\mathcal{I}} \in +(\neg C \sqcup \neg D) & \\ \Leftrightarrow \mathcal{I} \models_s \neg C \sqcup \neg D(a). &\end{aligned}$$

(5) The proof of this property is similar to that of (4).

(6) and (7) follow directly from Definition 3.5 and Definition 3.6.

Proof of Theorem 3.9

We apply Definition 3.5, Definition 3.7, Theorem 3.7 and Theorem 3.8 to prove this theorem.

(1)

$$\begin{aligned}\mathcal{I} \models_s \sim (C \sqcap D)(a) &\Leftrightarrow a^{\mathcal{I}} \notin +(C \sqcap D) \\ &\Leftrightarrow a^{\mathcal{I}} \notin +C \text{ or } a^{\mathcal{I}} \notin +D \\ &\Leftrightarrow \mathcal{I} \models_s \sim C(a) \text{ or } \mathcal{I} \models_s \sim D(a).\end{aligned}$$

(2)

$$\begin{aligned}\mathcal{I} \models_s \sim (C \sqcup D)(a) &\Leftrightarrow a^{\mathcal{I}} \notin +(C \sqcup D) \\ &\Leftrightarrow a^{\mathcal{I}} \notin +C \text{ and } a^{\mathcal{I}} \notin +D, \text{ or} \\ &\quad a^{\mathcal{I}} \in -C \text{ and } a^{\mathcal{I}} \notin +D, \text{ or} \\ &\quad a^{\mathcal{I}} \in D \text{ and } a^{\mathcal{I}} \notin +C \\ &\Leftrightarrow \mathcal{I} \models_s \sim C(a) \text{ and } \mathcal{I} \models_s \sim D(a), \text{ or} \\ &\quad \mathcal{I} \models_s \neg C(a) \text{ and } \mathcal{I} \models_s \sim D(a), \text{ or} \\ &\quad \mathcal{I} \models_s \neg D(a) \text{ and } \mathcal{I} \models_s \sim C(a).\end{aligned}$$

(3)

$$\begin{aligned}
\mathcal{I} \models_s \sim (\forall R.C)(a) &\Leftrightarrow a^{\mathcal{I}} \notin +(\forall R.C) \\
&\Leftrightarrow \text{there is an individual name } b \text{ such that} \\
&\quad (a^{\mathcal{I}}, b^{\mathcal{I}}) \in +R \text{ and } a^{\mathcal{I}} \notin +C \\
&\Leftrightarrow \mathcal{I} \models_s R(a, b) \text{ and } \mathcal{I} \models_s \sim C(b).
\end{aligned}$$

(4)

$$\begin{aligned}
\mathcal{I} \models_s \sim (\exists R.C)(a) &\Leftrightarrow a^{\mathcal{I}} \notin +(\exists R.C) \\
&\Leftrightarrow (a^{\mathcal{I}}, b^{\mathcal{I}}) \in +R \Rightarrow a^{\mathcal{I}} \notin +C \\
&\Leftrightarrow \mathcal{I} \models_s \sim C(b).
\end{aligned}$$

Here $S^{\mathcal{I}} = \langle +S, -S \rangle$ and S is a place-holder of $C, D, R, C \sqcap D, C \sqcup D, \forall R.C$ and $\exists R.C$.

Proof of Theorem 3.10

An axiom ϕ can only have three forms, namely, $R(a, b)$, $C \sqsubseteq D$ and $C(a)$ where C, D are concepts, R is a role name and a, b are individuals.

- (1) When ϕ is $R(a, b)$ or $C \sqsubseteq D$, it is easy to prove the theorem by Definition 3.3 and Definition 3.6.
- (2) When ϕ is $C(a)$, there are two cases, namely, C is a concept name and a complex concept. This theorem clearly holds when C is an atomic concept. In the following, we mainly discuss the case that C is a complex concept by induction over the number n of connectives and quantifiers in C .

Base step: when $n = 1$, C is in one of the following five forms, namely, $\neg A(a)$, $D \sqcap E$, $D \sqcup E$, $\forall R.D$ and $\exists R.D$ where A is an atomic concept, D, E are concepts, R is a role name and a, b are individuals.

- (a) When ϕ has one of the following forms, namely, $\neg A(a)$, $\forall R.D(a)$ and $\exists R.D(a)$, the strong interpretation of ϕ is equivalent to the weak interpretation of ϕ by Definition 3.2 and Definition 3.5. Therefore, this theorem clearly holds.
- (b) Suppose $\phi = C \sqcap D(a)$. If $\mathcal{I} \models_s C \sqcap D(a)$ then $a^{\mathcal{I}} \in +C \cap +D$ by Definition 3.5 where $C^{\mathcal{I}} = \langle +C, -C \rangle$ and $D^{\mathcal{I}} = \langle +D, -D \rangle$. Therefore, $\mathcal{I} \models_w C \sqcap D(a)$ by Definition 3.2.
- (c) Suppose $\phi = C \sqcup D(a)$. If $\mathcal{I} \models_s C \sqcup D(a)$ then $a^{\mathcal{I}} \in (+C \cup +D) \cap (\neg C \cup +D) \cap (+C \cup \neg D)$ by Definition 3.5. So $a^{\mathcal{I}} \in +C \cap +D$ where $C^{\mathcal{I}} = \langle +C, -C \rangle$ and $D^{\mathcal{I}} = \langle +D, -D \rangle$. Therefore, $\mathcal{I} \models_w C \sqcup D(a)$ by Definition 3.2.

Induction step: assume when the number of connectives and quantifiers in C is n , the theorem holds. We reduce axioms with the number $n + 1$ of connectives and quantifiers into axioms with the number n of connectives or quantifiers by equivalently eliminating one connective or quantifier. For instance, suppose $\phi = C \sqcap (D \sqcup E)(a)$ where C, D, E are concepts and a is an individual. If $\mathcal{I} \models_s C \sqcap (D \sqcup E)(a)$ then $a^{\mathcal{I}} \in +C$ and $a^{\mathcal{I}} \in +(D \sqcup E)$ where $C^{\mathcal{I}} = \langle +C, -C \rangle$ and $(D \sqcup E)^{\mathcal{I}} = \langle +(D \sqcup E), -(D \sqcup E) \rangle$, that is, $\mathcal{I} \models_s C(a)$ and $\mathcal{I} \models_s D \sqcup E(a)$. Thus, $\mathcal{I} \models_w C(a)$ and $\mathcal{I} \models_w D \sqcup E(a)$. Then $\mathcal{I} \models_w C \sqcap (D \sqcup E)(a)$.

In a word, we have proved that if $\mathcal{I} \models_s C(a)$ then $\mathcal{I} \models_w C(a)$ by Theorem 3.2 and Theorem 3.8.

Proof of Theorem 3.11

Let $\mathcal{A} = \{B \sqcup C(a), \neg B \sqcup E(a)\}$. We assume that \mathcal{I} is a strong interpretation of $\{B \sqcup C(a), \neg B \sqcup E(a)\}$, i.e., $\mathcal{I} \models_s B \sqcup C(a)$ and $\mathcal{I} \models_s \neg B \sqcup E(a)$. Thus, $a^{\mathcal{I}} \in (+B \cup +C) \cap (\overline{-B} \cup +C) \cap (+B \cup \overline{-C})$ and $a^{\mathcal{I}} \in (-B \cup +E) \cap (\overline{+B} \cup +E) \cap (-B \cup \overline{-E})$ by Definition 3.5. Then $a^{\mathcal{I}} \in (+B \cup +C) \cap (-B \cup +E)$. Observe that

- (1) If $a^{\mathcal{I}} \in +B$ then $a^{\mathcal{I}} \in +E$.
- (2) If $a^{\mathcal{I}} \notin +B$ then $a^{\mathcal{I}} \in +C$. Therefore, $a^{\mathcal{I}} \in +C$ or $a^{\mathcal{I}} \in +E$.

Thus, $\mathcal{I} \models_w C \sqcup E(a)$ by Definition 3.2. So if $\mathcal{I} \models_s B \sqcup C(a)$ and $\mathcal{I} \models_s \neg B \sqcup E(a)$ then $\mathcal{I} \models_w C \sqcup E(a)$. Hence, $\mathcal{A} \models_Q C \sqcup E(a)$.

Proof of Theorem 4.1

- (1) (\Leftarrow) Suppose that $\mathcal{O} \not\models_Q C(a)$, by Definition 3.8 there exists an interpretation \mathcal{I} such that $\mathcal{I} \models_s \mathcal{O}$ but $\mathcal{I} \not\models_w C(a)$, i.e., $\mathcal{I} \not\models_s C(a)$ by Theorem 3.10. Since $\mathcal{I} \models_s \mathcal{O}$, $\mathcal{I} \models_s \mathcal{O} \cup \{\sim C(a)\}$, that is, \mathcal{I} is a strong model of $\mathcal{O} \cup \{\sim C(a)\}$ which contradicts the premise that $\mathcal{O} \cup \{\sim C(a)\}$ is QC inconsistent by the definition of QC inconsistency.

(\Rightarrow) Suppose that $\mathcal{O} \cup \{\sim C(a)\}$ is QC consistent. Thus, there exists an interpretation \mathcal{J} such that $\mathcal{J} \models_s \mathcal{O} \cup \{\sim C(a)\}$ by the definition of QC inconsistency. Then $\mathcal{J} \models_s \mathcal{O}$ and $\mathcal{J} \models_s \sim C(a)$. That is, $a^{\mathcal{J}} \notin +C$ where $C^{\mathcal{J}} = \langle +C, -C \rangle$. Then $\mathcal{J} \not\models_w C(a)$, i.e., $\mathcal{J} \models_w \sim C(a)$ by Definition 3.4. Therefore, $\mathcal{O} \not\models_Q C(a)$ which contradicts the premise that $\mathcal{O} \models_Q C(a)$.

- (2) (\Leftarrow) Suppose that $\mathcal{O} \not\models_Q C \sqsubseteq D$, by Definition 3.8 there exists an interpretation \mathcal{I} such that $\mathcal{I} \models_s \mathcal{O}$ but $\mathcal{I} \not\models_w C \sqsubseteq D$, i.e., $\mathcal{I} \not\models_s$

$C \sqsubseteq D$ by Theorem 3.10. Thus, there exists an individual a such that if $\mathcal{I} \models_s C(a)$ then $\mathcal{I} \models_s \sim D(a)$ by Definition 3.7 and Theorem 3.3. Then $\mathcal{I} \models_s \{C(a), \sim D(a)\}$. Since $\mathcal{I} \models_s \mathcal{O}$, $\mathcal{I} \models_s \mathcal{O} \cup \{C(a), \sim D(a)\}$, that is, \mathcal{I} is a QC strong model of $\mathcal{O} \cup \{C(a), \sim D(a)\}$ which contradicts the premise that for any individual ι , $\mathcal{O} \cup \{C(\iota), \sim D(\iota)\}$ is QC inconsistent by the definition of QC inconsistency.

(\Rightarrow) Suppose that there exists an individual b such that $\mathcal{O} \cup \{C(b), \sim D(b)\}$ is QC consistent. Thus, there exists an interpretation \mathcal{J} such that $\mathcal{J} \models_s \mathcal{O} \cup \{C(b), \sim D(b)\}$ by the definition of QC inconsistency. Then $\mathcal{J} \models_s \mathcal{O}$, $\mathcal{J} \models_s C(b)$ and $\mathcal{J} \models_s \sim D(b)$. That is, $b^{\mathcal{J}} \in +C$ and $b^{\mathcal{J}} \notin +D$ where $C^{\mathcal{J}} = \langle +C, -C \rangle$ and $D^{\mathcal{J}} = \langle +D, -D \rangle$. Then $\mathcal{J} \not\models_w C \sqsubseteq D$, i.e., $\mathcal{J} \models_w \sim C \sqsubseteq D$ by Definition 3.4. Therefore, $\mathcal{O} \not\models_Q C \sqsubseteq D$ which contradicts the premise that $\mathcal{O} \models_Q C \sqsubseteq D$.

Proof of Theorem 5.1

We only prove that there cannot be an infinite sequence of rule application, i.e., $\mathcal{A} \rightarrow \mathcal{S}_1 \rightarrow \mathcal{S}_2 \rightarrow \dots$ where \mathcal{S}_i ($i = 0, 1, \dots$) is a set of QC ABoxes which is obtained from \mathcal{A} by using QC expansion rules after the i th step. Given a finite ABox \mathcal{A} built from N_C , N_R and N_I whose cardinalities are all finite, we can obtain two facts:

- (1) all concept axioms occurring in a QC ABox in one of the sets \mathcal{S}_i are of the form $C(x)$ where $C \in N_C$; and
- (2) if a QC ABox in \mathcal{S}_i contains the role assertion axioms $R(x, y)$, then the maximal role depth of concepts occurring in concept assertion axioms for y is strictly smaller than the maximal role depth of concepts occurring in concept assertion axioms for x . Assume to the contrary that there is an infinite sequence $\mathcal{A} \rightarrow \mathcal{S}_1 \rightarrow \mathcal{S}_2 \rightarrow \dots$. We shall map each QC ABox \mathcal{A}_j in \mathcal{S}_i to a set of elements $\Psi(\mathcal{A}_j)$ of a set Q which is equipped with a well-founded strict partial ordering \gg defined by Baader and Hanschke in [2] as follows: $\Psi(\mathcal{A}) \gg \Psi(\mathcal{A}')$ if \mathcal{A}' is obtained from \mathcal{A} by applying a QC transform rule at a time. Since the ordering is well-founded, i.e., has no infinitely decreasing chains, we get a contradiction with the ordering based on the above two facts.

Proof of Lemma 5.1

We consider nine QC expansion rules to prove this property. We assume that \mathcal{S} is obtained by applying QC expansion rule as follows.

- (1) \sqcap -rule, that is, \mathcal{A} contains $C_1 \sqcap C_2(a)$ and but not both $C_1(a)$ and $C_2(a)$. Thus the only member of \mathcal{S} is $\mathcal{A} \cup \{C_1(a), C_2(a)\}$. Since the strong model of $\{C_1 \sqcap C_2(a)\}$ is the same as the strong model of $\{C_1(a), C_2(a)\}$ by Theorem 3.8 (3). Therefore, this property is satis-

fied.

- (2) \sqcup -rule, that is, \mathcal{A} contains $C_1 \sqcup C_2(a)$ and but neither $C_i(a)$ nor $\neg C_i(a)$ for $i = 1, 2$. Thus two members of \mathcal{S} are $\mathcal{A} \cup \{C_1(a)\}$ and $\mathcal{A} \cup \{C_2(a)\}$. Since the strong model of $\{C_1 \sqcup C_2(a)\}$ is the same as the strong model of $\{C_1(a)\}$ or $\{C_2(a)\}$ by Theorem 3.7. Therefore, this property is satisfied.
- (3) \exists -rule, that is, \mathcal{A} contains $\exists R.C(a)$, but there is no individual name c such that $C(c)$ and $R(a, c)$ are in \mathcal{A} . Thus the only member of \mathcal{S} is $\mathcal{A} \cup \{C(b), R(a, b)\} \{U(x, b), U(b, x) \mid x \in N_A(\mathcal{A})\}$ where b is an individual name not occurring in \mathcal{A} . Since the strong model of $\{\exists R.C(a)\}$ is the same as the strong model of $\{C_1(b), R(a, b)\} \{U(x, b), U(b, x) \mid x \in N_A(\mathcal{A})\}$ by Definition 3.5. Therefore, this property is satisfied.
- (4) \forall -rule, that is, \mathcal{A} contains $\forall R.C(a)$ and $R(a, b)$, but it does not contain $C(b)$. Thus the only member of \mathcal{S} is $\mathcal{A} \cup \{C(b)\}$. Since the strong model of $\{\forall R.C(a), R(a, b)\}$ is the same as the strong model of $\{C(b)\}$ by Definition 3.5. Therefore, this property is satisfied.
- (5) QC-rule, that is, \mathcal{A} contains $C_1 \sqcup C_2(a)$ and $\neg C_i(a), i \in \{1, 2\}$. Thus the only member of \mathcal{S} is $\mathcal{A} \cup \{\otimes(C_1 \sqcup C_2, C_i)(a)\}$. Since the strong model of $\{C_1 \sqcup C_2(a)\}$ is the same as the strong model of $\{\otimes(C_1 \sqcup C_2, C_i)(a)\}$ by Theorem 3.7. Therefore, this property is satisfied.
- (6) \sim_{\sqcap} -rule, that is, \mathcal{A} contains $\sim (C_1 \sqcap C_2)(a)$, but neither $\sim C_1(a)$ nor $\sim C_2(a)$. Thus the two members of \mathcal{S} are $\mathcal{A} \cup \{\sim C_1(a)\}$ or $\mathcal{A} \cup \{\sim C_2(a)\}$. Since the strong model of $\{\sim (C_1 \sqcap C_2)(a)\}$ is the same as the strong model of $\{\sim C_1(a)\}$ or $\{\sim C_2(a)\}$ by Theorem 3.9 (1). Therefore, this property is satisfied.
- (7) \sim_{\sqcup} -rule, that is, \mathcal{A} contains $\sim (C_1 \sqcup C_2)(a)$, but not both $\sim C_1(a)$ and $\sim C_2(a)$. Thus the only member of \mathcal{S} is $\mathcal{A} \cup \{\sim C_1(x), \sim C_2(x)\}$. Since the strong model of $\{\sim (C_1 \sqcup C_2)(a)\}$ is the same as the strong model of $\{\sim C_1(a), \sim C_2(a)\}$ by Theorem 3.9 (2). Therefore, this property is satisfied.
- (8) \sim_{\exists} -rule, that is, \mathcal{A} contains $\sim (\exists R.C)(a)$ and $R(a, b)$, but it does not contain $\sim C(b)$. Thus the only member of \mathcal{S} is $\mathcal{A} \cup \{\sim C(y)\}$. Since the strong model of $\{\sim (\exists R.C)(a), R(a, b)\}$ is the same of $\{\sim C(b)\}$ by Theorem 3.9 (4). Therefore, this property is satisfied.

- (9) \sim_{\forall} -rule, that is, \mathcal{A} contains $\sim (\forall R.C)(a)$, but there is no individual name c such that $\sim C(c)$ and $R(a, c)$ are in \mathcal{A} . Thus the only member of \mathcal{S} is $\mathcal{A} \cup \{\sim C(b), R(a, b)\} \cup \{U(x, b), U(b, x) \mid x \in N_A(\mathcal{A})\}$ where b is an individual name not occurring in \mathcal{A} . Since the strong model of $\{\sim (\forall R.C)(a)\} \cup \{U(x, b), U(b, x) \mid x \in N_A(\mathcal{A})\}$ is the same of $\{\sim C(b), R(a, b)\}$ by Theorem 3.9 (3). Therefore, this property is satisfied.
- (10) \forall -rule for the universal role U , that is, \mathcal{A} contains $\forall U.C(a)$, for any individual $y \in N_A(\mathcal{A})$, \mathcal{A} contains $U(a, y)$, but it does not contain $C(y)$. Thus the only member of \mathcal{S} is $\mathcal{A} \cup \{C(y)\}$. Since the strong model of $\{\forall U.C(a), R(a, y)\}$ is the same as the strong model of $\{C(y)\}$ by Definition 3.5. Therefore, this property is satisfied. We analogously prove the $\sim \exists$ -rule for the universal role U .

In short, this property holds for any QC expansion rule in Table 1.

Proof of Theorem 5.2

It easily shows that this property holds because QC inconsistency can be maintained by applying any QC expansion rule at a time based on Lemma 5.1.

Proof of Theorem 5.3

Let $\mathcal{I}_{\mathcal{A}}$ be a *canonical strong interpretation* induced by \mathcal{A} such that:

- (1) the domain $\Delta^{\mathcal{I}_{\mathcal{A}}}$ of $\mathcal{I}_{\mathcal{A}}$ consists of all the individual names occurring in \mathcal{A} ;
- (2) for all concept names A we define $A^{\mathcal{I}_{\mathcal{A}}} = \langle +A, -A \rangle$;
- (3) for all role names R we define $R^{\mathcal{I}_{\mathcal{A}}} = \langle +R, -R \rangle$

where

$$\left\{ \begin{array}{l} +A = \{a^{\mathcal{I}_{\mathcal{A}}} \mid A(a) \in \mathcal{A}\} \setminus \{b^{\mathcal{I}_{\mathcal{A}}} \mid \sim A(b) \in \mathcal{A}\}; \\ -A = \{a^{\mathcal{I}_{\mathcal{A}}} \mid \neg A(a) \in \mathcal{A}\} \setminus \{b^{\mathcal{I}_{\mathcal{A}}} \mid \sim (\neg A(b)) \in \mathcal{A}\}; \\ +R = \{(a^{\mathcal{I}_{\mathcal{A}}}, b^{\mathcal{I}_{\mathcal{A}}}) \mid R(a, b) \in \mathcal{A}\} \setminus \{(c^{\mathcal{I}_{\mathcal{A}}}, d^{\mathcal{I}_{\mathcal{A}}}) \mid \sim R(a, b) \in \mathcal{A}\}; \\ -R = (\Delta^{\mathcal{I}_{\mathcal{A}}} \times \Delta^{\mathcal{I}_{\mathcal{A}}}) \setminus +R. \\ +U = \{(a^{\mathcal{I}_{\mathcal{A}}}, b^{\mathcal{I}_{\mathcal{A}}}) \mid a, b \in N_A(\mathcal{A})\} \\ -U = \emptyset \end{array} \right.$$

Therefore, we need to show that $\mathcal{I}_{\mathcal{A}}$ is a strong model of \mathcal{A} . Firstly, by definition, $\mathcal{I}_{\mathcal{A}}$ satisfies all the role assertions in \mathcal{A} . Next, we will show that for any concept assertion $\phi \in \mathcal{A}$, it satisfies ϕ as well by induction on the

structure of concept descriptions.

(Basic step): $\phi = A(a)$ where A is a concept name and a is an individual in \mathcal{ALC} . Since \mathcal{A} is clash-free, i.e., \mathcal{A} does not contain any clash, and $A(a) \in \mathcal{A}$, $\sim A(a) \notin \mathcal{A}$. Thus $a^{\mathcal{I}_{\mathcal{A}}} \in +A$ by the definition of $\mathcal{I}_{\mathcal{A}}$. Then $\mathcal{I}_{\mathcal{A}} \models_s A(a)$ by Definition 3.6.

(Induction step): every expressive concept assertion ϕ has six basic forms as follows: $\neg C(a)$, $\sim C(a)$, $C \sqcap D(a)$, $C \sqcup D(a)$, $\forall R.C(a)$ and $\exists R.C(a)$ where C, D are concepts, R is a role name and a is an individual in \mathcal{ALC} . We inductively assume that $\mathcal{I}_{\mathcal{A}}$ satisfies all concept assertions of \mathcal{A} the number of whose connectives is less than that of ϕ and $C^{\mathcal{I}_{\mathcal{A}}} = \langle +C, -C \rangle$, $D^{\mathcal{I}_{\mathcal{A}}} = \langle +D, -D \rangle$ and $R^{\mathcal{I}_{\mathcal{A}}} = \langle +R, -R \rangle$. We will consider six cases in the following proof.

- Case 1: $\phi = \neg C(a)$. Since \mathcal{A} is clash-free, and $\neg C(a) \in \mathcal{A}$ then $\sim (\neg C(a)) \notin \mathcal{A}$. Thus $a^{\mathcal{I}_{\mathcal{A}}} \in -C$ by the definition of $\mathcal{I}_{\mathcal{A}}$. Then $\mathcal{I}_{\mathcal{A}} \models_s \neg C(a)$ by Definition 3.6.
- Case 2: $\phi = \sim C(a)$. Since \mathcal{A} is clash-free, and $\sim C(a) \in \mathcal{A}$ then $C(a) \notin \mathcal{A}$. Thus $a^{\mathcal{I}_{\mathcal{A}}} \in \Delta^{\mathcal{I}_{\mathcal{A}}} \setminus +C$ by the definition of $\mathcal{I}_{\mathcal{A}}$. Then $\mathcal{I}_{\mathcal{A}} \models_s \sim C(a)$ by Definition 3.6.
- Case 3: $\phi = C \sqcap D(a)$. Since \mathcal{A} is clash-free, and $C \sqcap D(a) \in \mathcal{A}$, $\sim (C \sqcap D(a)) \notin \mathcal{A}$. Thus $\{C(a), D(a)\} \subseteq \mathcal{A}$ because \mathcal{A} is complete. Based on the inductive hypothesis, $\mathcal{I}_{\mathcal{A}} \models_s C(a)$ and $\mathcal{I}_{\mathcal{A}} \models_s D(a)$, i.e., $a^{\mathcal{I}_{\mathcal{A}}} \in +C$ and $a^{\mathcal{I}_{\mathcal{A}}} \in +D$. By Definition 3.5, $a^{\mathcal{I}_{\mathcal{A}}} \in +C \cap +D$. By Theorem 3.9 Item 1, $a^{\mathcal{I}_{\mathcal{A}}} \notin \Delta^{\mathcal{I}_{\mathcal{A}}} \setminus +C$ or $a^{\mathcal{I}_{\mathcal{A}}} \notin \Delta^{\mathcal{I}_{\mathcal{A}}} \setminus +D$. Thus $a^{\mathcal{I}_{\mathcal{A}}} \in +(C \sqcap D)$ by the definition of $\mathcal{I}_{\mathcal{A}}$. Then $\mathcal{I}_{\mathcal{A}} \models_s C \sqcap D(a)$ by Definition 3.6.
- Case 4: $\phi = C \sqcup D(a)$. Since \mathcal{A} is clash-free, and $C \sqcup D(a) \in \mathcal{A}$, $\sim (C \sqcup D(a)) \notin \mathcal{A}$. We consider two cases:
 - (a) if $\{\neg C(a), \neg D(a)\} \cap \mathcal{A} = \emptyset$, then $C(a) \in \mathcal{A}$ or $D(a) \in \mathcal{A}$ because \mathcal{A} is complete. Based on the inductive hypothesis, $\mathcal{I}_{\mathcal{A}} \models_s C(a)$ or $\mathcal{I}_{\mathcal{A}} \models_s D(a)$, i.e., $a^{\mathcal{I}_{\mathcal{A}}} \in +C$ or $a^{\mathcal{I}_{\mathcal{A}}} \in +D$. Thus $a^{\mathcal{I}_{\mathcal{A}}} \in +C \cup +D$ by Definition 3.5. By Theorem 3.9 Item 2, $a^{\mathcal{I}_{\mathcal{A}}} \notin \Delta^{\mathcal{I}_{\mathcal{A}}} \setminus +C$ or $a^{\mathcal{I}_{\mathcal{A}}} \notin \Delta^{\mathcal{I}_{\mathcal{A}}} \setminus +D$. Thus $a^{\mathcal{I}_{\mathcal{A}}} \in +(C \sqcup D)$ by the definition of $\mathcal{I}_{\mathcal{A}}$. Then $\mathcal{I}_{\mathcal{A}} \models_s C \sqcup D(a)$ by Definition 3.6.
 - (b) if $\{\neg C(a), \neg D(a)\} \cap \mathcal{A} \neq \emptyset$ and we assume $\neg C(a) \in \mathcal{A}$ without loss of generality, then $D(a) \in \mathcal{A}$ because \mathcal{A} is complete. Based on the inductive hypothesis, $\mathcal{I}_{\mathcal{A}} \models_s D(a)$, $a^{\mathcal{I}_{\mathcal{A}}} \in +D$. Then

$a^{\mathcal{I}_A} \in (+C \cup +D) \cap ((\Delta^{\mathcal{I}_A} \setminus -C) \cup +D)$ by Definition 3.5. By Theorem 3.9 Item 2, $a^{\mathcal{I}_A} \notin \Delta^{\mathcal{I}_A} \setminus +C$ or $a^{\mathcal{I}_A} \notin \Delta^{\mathcal{I}_A} \setminus +D$; and $a^{\mathcal{I}_A} \notin -C$ or $a^{\mathcal{I}_A} \notin \Delta^{\mathcal{I}_A} \setminus +D$. Thus $a^{\mathcal{I}_A} \in +(C \sqcup D)$ by the definition of \mathcal{I}_A . Then $\mathcal{I}_A \models_s C \sqcup D(a)$ by Definition 3.6.

- Case 5: $\phi = \forall R.C(a)$. Since \mathcal{A} is clash-free, and $\forall R.C(a) \in \mathcal{A}$ then $\sim \forall R.C(a) \notin \mathcal{A}$. If there exists a role assertion $R(a, b) \in \mathcal{A}$, then $C(b) \in \mathcal{A}$ because \mathcal{A} is complete. Based on the inductive hypothesis, $\mathcal{I}_A \models_s C(b)$, i.e., $b^{\mathcal{I}_A} \in +C$. By Definition 3.5, $a^{\mathcal{I}_A} \in \{x \mid \forall y, (x, y) \in +R \text{ implies } y \in +C\}$. By Theorem 3.9 Item 3, there is not any individual b such that $(a^{\mathcal{I}_A}, b^{\mathcal{I}_A}) \in +R$ and $b^{\mathcal{I}_A} \in \Delta^{\mathcal{I}_A} \setminus +C$. Thus $a^{\mathcal{I}_A} \in +\forall R.C$ by the definition of \mathcal{I}_A . Then $\mathcal{I}_A \models_s \forall R.C(a)$ by Definition 3.6.
- Case 6: $\phi = \exists R.C(a)$. Since \mathcal{A} is clash-free, if $\exists R.C(a) \in \mathcal{A}$ then $\sim \exists R.C(a) \notin \mathcal{A}$. Thus there exists an individual t occurring in \mathcal{A} such that $C(t) \in \mathcal{A}$ because \mathcal{A} is complete. Based on the inductive hypothesis, $\mathcal{I}_A \models_s C(t)$, i.e., $t^{\mathcal{I}_A} \in +C$. By Definition 3.5, $a^{\mathcal{I}_A} \in \{x \mid \exists y, (x, y) \in +R \text{ and } y \in +C\}$. By Theorem 3.9 Item 4, there is an individual b such that $(a^{\mathcal{I}_A}, b^{\mathcal{I}_A}) \in +R$ and $b^{\mathcal{I}_A} \notin \Delta^{\mathcal{I}_A} \setminus +C$. Thus $a^{\mathcal{I}_A} \in +\exists R.C$ by the definition of \mathcal{I}_A . Then $\mathcal{I}_A \models_s \exists R.C(a)$ by Definition 3.6.

In short, \mathcal{I}_A is a strong model of \mathcal{A} . By induction, we could show that it satisfies all concept assertions of \mathcal{A} as well.

Proof of Theorem 5.4

Firstly, we need to show that the problem of checking QC consistency of QC ABoxes is PSPACE by employing the analogical proof of the problem that decides the consistency of \mathcal{ALC} ABoxes in [4]. Let \mathcal{A} be a QC ABox. We denote $|\mathcal{A}|$ as the size of \mathcal{A} . Intuitively $|\mathcal{A}|$ is the length required to write \mathcal{A} down, where we assume that the length required to write atomic concept, the negation of concept name and atomic role is “1”. Formally, we define the size of QC ABoxes as follows:

$$\begin{aligned}
|\mathcal{A}| &= \sum_{C(a) \in \mathcal{A}} (|C| + 1) + \sum_{R(a,b) \in \mathcal{A}} 3 \\
|A| = |\neg A| &= 1 \text{ for a concept name } A (\text{including } \top, \perp) \\
|\sim D| &= |D| + 1 \\
|D_1 \sqcap D_2| = |D_1 \sqcup D_2| &= |D_1| + |D_2| + 1 \\
|\exists R.D| = |\forall R.D| &= |D| + 2
\end{aligned}$$

We observe that the QC tableau algorithm builds a completion forest in a monotonic way; that is, all QC expansion rules in Table 1 either add new

concept assertions and role assertions to a QC ABox or new QC ABoxes to the set of QC ABoxes \mathcal{S} , but never remove anything. We call an individual occurring in \mathcal{A} an *old individual*. Other individuals not occurring in \mathcal{A} before are generated by the \exists -rule or $\sim \forall$ -rule, and we call them *new individuals*; we call the other rules *augmenting rules*, because they only augment existing individuals. Note that no new role assertion $U(a, y)$, where $a \in N_A(\mathcal{A})$ and y is a new individual, will be added since without the universal role U . In contrast to role assertions on between old individuals, role assertions between new individuals are of a particular shape: each new individual is found in a QC ABox of \mathcal{S} with an old individual in \mathcal{A} . Other new assertions over old individuals are added by the expansion rules, and these expansion rules only add subconcept assertions over old individuals or role assertions over old individuals where subconcepts and roles are in \mathcal{A} . However, new concept assertion over new individuals or role assertion over new individuals being added by the expansion rules is limited by the number of \exists or $\sim \forall$ occurring in \mathcal{A} and a new ABox added by the expansion rules is limited by the number of \sqsubset and $\sim \sqsubset$ occurring in \mathcal{A} . Since there are at most $|\mathcal{A}|$ such subconcepts, each QC ABox can be stored in space polynomial in $|\mathcal{A}|$. Moreover, for each concept D in each QC ABox \mathcal{A}_i , any QC ABox \mathcal{A}_j which is in front of \mathcal{A}_i in the queue of QC ABoxes contains a larger concept of D . Hence the maximum size of concepts in QC ABoxes strictly decreases along the queue of QC ABoxes by applying the QC expansion rules step by step, and thus the length of the queue of QC ABoxes is bounded by $\max\{C \mid C(a) \in \mathcal{A}\}$.

Initially, we start \mathcal{A} . Then we apply the QC expansion rules in Table 1 exhaustively. Finally, we note that the QC expansion rules can be applied in an arbitrary order, that is, the correctness proof for the QC tableau algorithm does not rely on a specific application order. Therefore, we can use the following order: firstly, all augmenting rules are exhaustively applied assertions over old individuals and a set of QC ABoxes is obtained.

Secondly, we apply the \exists -rule on the form $\exists R.C(a)$ and the $\sim \forall$ -rule on the form $\sim \forall R.C(a)$ to a set of QC ABoxes and a new concept assertion over a new individual and a new role assertion over a new individual are added by applying the \exists -rule and the $\sim \forall$ -rule in a time. Then we apply the \forall -rule or $\sim \exists$ -rule exhaustively to assertions over new individuals. Then we recursively apply the same procedure to new assertions over both old individuals and new individuals, i.e., exhaustively apply the augmenting rules, and then deal with the existential restrictions one at a time. As usual, the QC algorithm stops if a clash occurs; otherwise, when all existential restrictions of a new assertion over a new individual t or value restrictions of complement of a

concept assertion over a new individual t have been dealt with, we can delete all assertions over t , i.e., delete all assertions with the form $C(t)$ and reuse the space. Thus, we can investigate the whole of a QC ABox which generates from \mathcal{A} with only keeping a single block in memory at any time. This block is of length linear in $|\mathcal{A}|$, and can thus be stored with a QC ABox in size polynomial in $|\mathcal{A}|$. We can continue the investigation of all QC ABoxes of \mathcal{S} in the same manner, the QC tableau algorithm only requires space polynomial in $|\mathcal{A}|$. Therefore, the problem of checking QC consistency of QC ABoxes is PSPACE.

In the following, we need to show that there exists a problem with the complexity in PSPACE-Complete which is no more complex than the problem of checking QC consistency of QC ABoxes. Manfred Schmidt-Schauß et al [31] had proved that the complexity of checking consistency of DL \mathcal{ALC} ABoxes is PSPACE-Complete. We need only to prove that the problem of checking consistency of DL \mathcal{ALC} ABoxes can be reduced to the problem of instance checking in QC \mathcal{ALC} . Now we consider the problem of deciding the satisfiability of a concept C with respect to an empty ABox. Let C_{new} be a new concept name not occurring in C . C is unsatisfiable if and only if $\{C \sqcup C_{new}(a)\} \models C_{new}(a)$. If $\{C \sqcup C_{new}(a)\} \models C_{new}(a)$, then we have a classical DL tableau proof. Let \mathcal{A} be $\{\{C \sqcup C_{new}(a), \neg C_{new}(a)\}\}$. If \mathcal{A} is classical inconsistent, then $\{C \sqcup C_{new}(a)\} \models C_{new}(a)$. That is to say, the satisfiability of a concept C with respect to an empty ABox is reduced to checking the consistency of \mathcal{A} . If \neg is transformed into \sim in \mathcal{A} , i.e., $\{C \sqcup C_{new}(a), \sim C_{new}(a)\}$ (denoted by \mathcal{A}_Q), then the problem of checking the consistency of \mathcal{A} can be reduced to the problem of checking the QC consistency of \mathcal{A}_Q . Now, we only need to show that there exists a QC DL tableau proof. Firstly, we use the \sqcup -rule in Table 1 for \mathcal{A}_Q and we obtain two ABoxes $\mathcal{A}' = \{C(a), \sim C_{new}(a)\}$ and $\mathcal{A}'' = \{C_{new}(a), \sim C_{new}(a)\}$. We mainly consider \mathcal{A}' because \mathcal{A}'' contains a clash. Five expansion rules (namely, \sqcup -rule, \sqcap -rule, \forall -rule, \exists -rule and QC-rule) in Table 1 are enough to decide whether $\{C(a), \sim C_{new}(a)\}$ is closed or not because $C(a)$ is a classical axiom and C_{new} is a new concept name different from C . Moreover, such five rules can be captured by standard rules in \mathcal{ALC} (see Figure 2.6 [1]). (Note that (1) the \sqcup -rule and the QC-rule in Table 1 can be captured by the standard \sqcup -rule in \mathcal{ALC} ; and (2) the \sqcap -rule, the \forall -rule and the \exists -rule directly inherit the corresponding standard rules in \mathcal{ALC}). Thus, we employ the QC tableau algorithm to decide whether $\{C \sqcup C_{new}(a), \sim C_{new}(a)\}$ is QC consistent by applying standard tableau algorithm. It is easy to see that in such case, the classical DL tableau proof is transformed into a QC DL

tableau proof. That is, following from the symbol transformation, the problem of whether $\{C \sqcup C_{new}(a)\} \models C_{new}(a)$ can be reduced to the problem of whether $\{C \sqcup C_{new}(a), \sim C_{new}(a)\}$ is QC consistent. By Theorem 5.2 and Theorem 5.3, it is equivalent to $C \sqcup C_{new}(a) \models_Q C_{new}(a)$. Now, we have reduced the problem of deciding whether C is unsatisfiable to deciding whether $C \sqcup C_{new}(a) \models_Q C_{new}(a)$. The reduction is in polynomial time.

So the complexity of checking QC consistency of QC ABoxes is PSPACE-Complete.

Proof of Theorem 5.5

Firstly, we need to show that the problem of checking QC consistency of a QC \mathcal{ALC} ABox with respect to a general TBox is in EXPTIME by employing the analogical proof of the complexity of the problem deciding the consistency of an \mathcal{ALC} ABox with respect to a general TBox in [29, 8]

Two reasoning problems, namely, instance checking and subsumption checking in \mathcal{ALC} , is PSPACE-Complete if ABoxes involve only a single atomic role (see [19, 8]). However, two reasoning problems in \mathcal{ALC} is EXPTIME if involving a universal role U which is taken as a transitive-reflexive closure of roles (see [9, 8, 15]). By Lemma 2.1 and Theorem 4.1, two reasoning problems could be equivalently reduced into the problem of deciding consistency of ABox with a universal role U . Lemma 2.2 states that the problem about deciding consistency of an \mathcal{ALC} ABox with respect to a general TBox is equivalent to the problem of deciding consistency of an \mathcal{ALC} ABox with a universal role U . That is to say, the problem of deciding consistency of an \mathcal{ALC} ABox with respect to a general TBox is also in EXPTIME. Based on the proof of Theorem 5.4, we analogously show that the problem of consistency of an \mathcal{ALC} ABox with a universal role U can be reduced to the problem of instance checking in a QC \mathcal{ALC} ABox with a universal role U . In addition, Corollary 4.1 tells that the problem about deciding QC consistency of a QC \mathcal{ALC} ABox with respect to a general TBox is equivalent to the problem about deciding QC consistency of a QC \mathcal{ALC} ABox with a universal role U . Thus the problem about deciding QC consistency of a QC \mathcal{ALC} ABox with respect to a general TBox is also EXPTIME.

In the following, we need to show that there exists a problem with the complexity EXPTIME-Complete which is no more complex than the problem of checking QC consistency of a QC ABox with respect to a general TBox. In [29, 8], we know that the complexity of checking consistency of an \mathcal{ALC} ABox with respect to a general TBox is EXPTIME-Complete. We only prove that the problem of checking consistency of an ABox with respect to a general TBox can be reduced to the problem of checking QC consistency of a QC

ABox with respect to a general TBox. Let $\mathcal{O} = (\mathcal{T}, \mathcal{A})$ be a new consistent \mathcal{ALC} ontology and $C(a)$ an \mathcal{ALC} concept assertion where \mathcal{T} is a general \mathcal{ALC} TBox, \mathcal{A} an ABox and $C \neq \top$. By Lemma 2.1, $\mathcal{O} \models C(a)$ if and only if $(\mathcal{T}, \mathcal{A} \cup \{\neg C(a)\})$ is inconsistent. Moreover, by Theorem 4.1, $\mathcal{O} \models C(a)$ if and only if $(\mathcal{T}, \mathcal{A} \cup \{\sim C(a)\})$ is QC inconsistent. In addition, $\mathcal{O} \models_Q C(a)$ if and only if $\mathcal{O} \models C(a)$ since \mathcal{O} is consistent and $C \neq \top$. Then $(\mathcal{T}, \mathcal{A} \cup \{\neg C(a)\})$ is inconsistent if and only if $(\mathcal{T}, \mathcal{A} \cup \{\sim C(a)\})$ is QC inconsistent. Therefore, we could show that the problem about checking consistency of an ABox with respect to a general TBox can be reduced to the problem about checking QC consistency of a QC ABox with respect to a general TBox.

So the complexity of checking QC consistency of a QC \mathcal{ALC} ABox with respect to a general TBox is EXPTIME-Complete.