# Consistency Checking of an Ontology Concept Model Based on Description Logics

*Yu Changrui* and *Luo Yan*

Institute of System Engineering, Shanghai Jiao Tong University, 200052 Shanghai, China

**Summary**

Ontology formal model and ontology checking recently are still under hot discussion. In this paper, an ontology concept model is constructed using Description Logics. Based on model, the issue on Consistency checking of the extended ontology model is studied with the conclusion that the four kinds of term checking, including term satisfiability checking, term subsumption checking, term equivalence checking and term disjointness checking, can be reduced to the satisfiability checking, and satisfiability checking can be transformed into instantiation consistence checking.

***Key words:***

*Ontology concept model, consistency checking, description logics.*

## Introduction

Today, computers are changing from single isolated devices to entry points into a worldwide network of information exchange and business transactions. Therefore, support in the exchange of information and knowledge is becoming the key issue in computer technology. Ontology provides a shared and common understanding of a domain that can be communicated between people and across application systems [1]. Ontology plays a major role in various areas relevant to information exchange, such as knowledge engineering [2], knowledge representation [3], information modeling [4], information integration [5], semantic based retrieval [6], semantic web, knowledge management [7].

Currently, the formal ontology model suiting to model checking is still under hot discussion. Some recent researches of ontology are based on first order logic (FOL), e.g. Ontolingua [8], CycL [9], LOOM [10]. Although FOL have a more expressive power, the reasoning process are complex and most of them are even undecidable, which is not suitable for ontology model checking. There are still other ontology models written in natural language for better human's understanding, e.g. WordNet, can't support model checking because of their low formalism. Description Logics (DLs) is the name for a family of knowledge representation formalisms that represent the knowledge of an application domain by first defining the relevant concepts of the domain, and then using these concepts to specify properties of objects and individuals occurring in the domain. DL is equipped with a formal and logic-based semantics allowing inferring implicitly represented knowledge from the knowledge that is explicitly contained in the knowledge base. Although DL has a less expressive power than FOL, its inference procedures are more efficient and decidable, which is more suitable for ontology checking [11].

Refs. [6], [11] proposed a naive ontology concept model *O=<T, X, TD, XD>*, consisting of Term Set, Individual Set, Term Definition Set, and Instantiation Assertion Set, in DLs. In this paper, an extended ontology concept model *O=<T, X, TD, XD, TR>* is established in description logics through introducing Term Restriction Set to the naive model in Refs. [6], [11]. Based on the extended model, we study the issue of how to check the ontology model for preserving the consistence of ontology.

In the rest of paper, we present first, in section 2, an extended ontology concept model and its basic concepts. In section 3, Consistency checking of the extended ontology concept model is proposed and some Propositions are proofed. Finally, we finish with conclusions in section 4.

## 2. An Extended Ontology Concept Model

### 2.1 An Extended ontology concept model and its interpretation

**Definition 1**: Given a terminology description language L, an extended ontology concept model is a 5-tuples,

$$O=<T, X, TD, XD, TR> \quad (1)$$

For convenience, we call ontology below. *T* is a Term Set, *X* is an Individual Set, *TD* is a Term Definition Set, *XD* is an Instantiation Assertion Set, and *TR* is a Term Restriction Set. The elements included in *T* are also called atomic term, which is divided into two categories: atomic class term (atomic class, for short) and atomic property term (atomic property, for short).

Manuscript received December 5, 2006.
Manuscript revised December 25, 2006.

**Definition 2**: Given $O$=<$T, X, TD, XD, TR$>, an ontology interpretation is a 2-tuples,

$$I=<\Delta^I, \cdot^I> \quad (2)$$

where $\Delta^I \neq \varnothing$ is the domain of the interpretation, and $\cdot^I$ is an interpretation function, which assigns to every atomic class $C$ in $T$ a set $C^I \subseteq \Delta^I$, and to every atomic property $P$ in $T$ a binary relation $P^I \subseteq \Delta^I \times \Delta^I$, and to every individual $a$ in $X$ an element $a^I \in \Delta^I$.

An ontology model of family domain using the above model is gives as follows:

$O$=<{Person, Female, Male, Woman, Man, Mother, Father, Parent, hasChild, hasHusband, Wife, Grandmother, MotherWithoutSon, MotherWithManyChildren}, {Alice, Tom, Mary}, {Woman≡Person ⊓ Female, Man≡Person⊓Male, Mother≡Woman⊓∃hasChild.Person, Father≡Man⊓∃hasChild.Person, Parent≡Father⊔ Mother, Wife≡Woman ⊓ ∃hasHusband.Man, Grandmother≡Mother ⊓ ∃hasChild.Person, MotherWithoutSon≡ Mother ⊓ ∀hasChild.¬Man, MotherWithManyChildren≡Mother ⊓ ≥ 3,hasChild}, {Man(Tom), Woman(Alice), hasHusband(Alice, Tom), hasChild(Alice, Mary)}, {Female ∩̶ Male}>

## 2.2 Term set and term definition set

Term Set comprises a group of atomic terms. However, atomic terms can only express limited logics and simple contents, because they are just the basic element with less expressive power. So here we adopt the term constructors from Description Logic to build term formulas for the expression of more complex contents. Given $L$, we call the expression, satisfying the syntax rule below, an $L$-based term formula.

$$D, E \rightarrow C \mid \top \mid \bot \mid \neg C \mid D \sqcap E \mid \forall P.D \mid \exists P.\top \quad (3)$$

Term Set comprises a group of atomic terms. However, atomic terms can only express limited logics and simple contents, because they are just the basic element with less expressive power. So here we adopt the term constructors from Description Logic to build term formulas for the expression of more complex contents. Table 1 gives ontology interpretations to six basic term constructors.

( i ) Any atomic class $C$ is an $L$-based formula.
( ii ) The negation of an atomic class $C$, expressed as "$\neg C$", is an $L$-based formula.
( iii ) Universal Class and Empty Class, denoted as "$\top$" and "$\bot$" respectively, are also an $L$-based formula.
( iv ) The intersection of two $L$-based formulas $D$ and $E$, written as "$D \sqcap E$", is still an $L$-based formula.
( v ) Given an atomic property $P$ and an $L$-based formula $D$, the value restriction of $P$, denoted as "$\forall P.D$" is still an $L$-based formula.
( vi ) Given an atomic property $P$, the limited existential quantification of $P$, written as "$\exists P.\top$" is still an $L$-based formula.

Table 1: Basic term constructors and their ontology interpretations

| **Constructors Name** | **Term Constructors Syntax** | **Ontology Interpretation** |
|---|---|---|
| Universal Class | $\top$ | $\top^I=\Delta^I$ |
| Empty Class | $\bot$ | $\bot^I=\varnothing$ |
| Atomic Class Negation | $\neg C$ | $(\neg C)^I= \Delta^I \backslash C^I$ |
| Intersection | $D \sqcap E$ | $(D \sqcap E)^I=D^I \cap E^I$ |
| Property Value Restriction | $\forall P.D$ | $(\forall P.D)^I=\{a \in \Delta^I \mid \forall b. (a, b) \in P^I \rightarrow b \in D^I\}$ |
| Limited Existential Quantification | $\exists P.\top$ | $(\exists P.\top)^I=\{a \in \Delta^I \mid \exists b. (a, b) \in P^I\}$ |

Table 2: Extended term constructors and their ontology interpretations

| Constructors Name | Term Constructors Syntax | Ontology Interpretation |
|---|---|---|
| Union | $D \sqcup E$ | $(D \sqcup E)^I=D^I \cup E^I$ |
| Non-atomic negation | $\neg D$ | $(\neg D)^I=\Delta^I \backslash D^I$ |
| Full Existential Quantification | $\exists P.D$ | $(\exists P.C)^I=\{a \in \Delta^I \mid \exists b. (a, b) \in P^I \wedge b \in C^I\}$ |
| At-least number restriction | $\geq n,P$ | $(\geq n,P)^I=\{a \in \Delta^I \mid \lvert\{b \mid (a, b) \in P^I \}\rvert \geq n\}$ |
| At-most number restriction | $\leq n,P$ | $(\leq n,P)^I=\{a \in \Delta^I \mid \lvert\{b \mid (a, b) \in P^I \}\rvert \leq n\}$ |

Eq. (3) shows that the expressive power of term formulas strongly depends on the type of term constructors. In Eq.(3), $L$ only supports six basic constructors listed in Table 1, so L is also called basic term description language, written as $L_B$, and the corresponding term formulas constructed by $L_B$ is called $L_B$-based term formulas. To obtain more expressive languages, more complex constructors should be extended to $L_B$. Table 2 gives another set of term constructors, called extended term constructors, where $P$ is an atomic property, and $D$ and $E$ are two term formulas.

**Definition 3(Subsumption):** Given $L$ and $O$=<$T$, $X$, $TD$, $XD$, $TR$>, $D$ and $E$ are two $L$-based term formulas. We say $D$ is subsumed by $E$, if $D^I \subseteq E^I$ for any an interpretation $I$.

**Definition 4(Equivalence):** Given $L$ and $O$=<$T$, $X$, $TD$, $XD$, $TR$>, $D$ and $E$ are two $L$-based term formulas. We say $D$ and $E$ are equivalent, if $D^I = E^I$ for any an interpretation $I$.

**Definition 5(Disjointness):** Given $L$ and $O$=<$T$, $X$, $TD$, $XD$, $TR$>, $D$ and $E$ are two $L$-based term formulas. We say $D$ and $E$ are disjointness, if $D^I \cap E^I = \varnothing$ for any an interpretation $I$.

**Definition 6(Term Definition Item):** A term definition item is an equivalence relationship between two terms, written as $C \equiv D$, where $C$ is an atomic class term, called definiendum, and $D$ is a term formula, called definiens.

Given $O$=<$T$, $X$, $TD$, $XD$, $TR$>, Term Definition Set $TD$ is such a set that consists of term definition items subject to the following restrictions, written as $TD=\{C_1 \equiv D_1, C_2 \equiv D_2, \ldots, C_n \equiv D_n\}$. Where, $C_i \in T$, $D_i$ is a term formula, and every term in $D_i$ is from $T$.

( i ) for any $i, j$ ($i \neq j, 1 \leq i \leq n, 1 \leq j \leq n$), $C_i \neq C_j$ holds.
( ii ) if there exist $C_1' \equiv D_1'$, $C_2' \equiv D_2'$,…, $C_m' \equiv D_m'$ in $TD$, and $C_i'$ occurs in $D_{i\text{-}1}'$ ($1 < i \leq m$, $m \leq n$), then $C_1'$ must not occur in $D_m'$.

**Definition 7 (Model of Term Definition Item):** Given $O$=<$T$, $X$, $TD$, $XD$, $TR$>, if there exists an ontology interpretation $I$ satisfying a term definition item $A$ of $TD$, then $I$ is called a model of $A$. If $I$ is a model of all term definition item of $TD$, then we say $I$ is a model of $TD$.

**Definition 8(Defined Term & Primitive Term):** Given $O$=<$T$, $X$, $TD$, $XD$, $TR$>, atomic terms of $T$ can be divided into two sets: defined terms, which occur in the definiendum of term definition item of $TD$, written as $T_d$, and primitive terms, which occur only in the definiens, written as $T_p$.

**Definition 9(Expansion of term definition item):** Given $TD=\{C_1 \equiv D_1, C_2 \equiv D_2, \ldots, C_n \equiv D_n\}$, each term definition item in $TD$ is expanded through an iterative process by replacing each occurrence of a defined term in the definiense with the primitive terms it stands for. Since no cycle term definition is allowed in $TD$ (guaranteed by Restriction ( ii ) above), the process eventually stops and we end up with a Term Set $T'=\{C_1 \equiv D_1', C_2 \equiv D_2', \ldots, C_n \equiv D_n'\}$, where $D_i'$ contains only primitive terms and no defined terms. We say that $D_i'$ is the expansion of $D_i$ with respect to $TD$, written as $e(D_i)$, and $C_i \equiv D_i'$ is the expansion of $C_i \equiv D_i$ with respect to $TD$, and $T'$ is the expansion of $T$ with respect to $TD$, written as $e(T)$.

**Proposition1:** Given $O$=<$T$, $X$, $TD$, $XD$, $TR$>, where $TD=\{C_1 \equiv D_1, C_2 \equiv D_2, \ldots, C_n \equiv D_n\}$, and $E$ is a term formula. If $I$ is a model of $TD$, then $E^I = e(E)^I$ holds.
**Proof:** Since $I$ is a model of $TD$, we can conclude that $C_i^I = D_i^I$ holds for any term definition item $C_i \equiv D_i$ in $TD$. Then replace one of defined term $C_j$ occurring in $E$ with $D_j$ and obtain a new term formula $E'$. we have $E^I = E'^I$ since $C_i^I = D_i^I$. Moreover, $e(E)$ can be obtained through the above replacing process in finite times until all defined terms are replaced with primitive terms, so $E^I = e(E)^I$ holds.

**Proposition 2:** Given $O$=<$T$, $X$, $TD$, $XD$, $TR$>, where $TD=\{C_1 \equiv D_1, C_2 \equiv D_2, \ldots, C_n \equiv D_n\}$, and $S$ is a term formula. If $I$ is a model of $e(TD)$, then there must exists a model of $TD$ $I'$, such that $S^{I'} = e(S)^I$.
**Proof:** Let $T_p=\{B_1, B_2, \ldots, B_m\}$ be the Primitive Term Set. Since $TD=\{C_1 \equiv D_1, C_2 \equiv D_2, \ldots, C_n \equiv D_n\}$, we get the Defined Term Set $T_d=\{C_1, C_2, \ldots, C_n\}$. Suppose $e(TD)=\{C_1 \equiv D_1', C_2 \equiv D_2', \ldots, C_n \equiv D_n'\}$, i.e. $D_i' = e(D_i)$. If $I$ is a model of $e(TD)$, then $C_i^I = D_i'^I$ holds for any term definition item in $e(TD)$. Then we use $I$ to build a new ontology interpretation $I'$, such that $B_i^{I'} = B_i^I$ for any primitive term $B_i$; $C_i^{I'} = D_i'^I$, for any defined term $C_i$. With the new interpretation $I'$, we have $S^{I'} = e(S)^I$ for any term formula $S$, which result in $D_i^{I'} = e(D_i)^I$. Moreover, since $C_i^{I'} = D_i'^I = e(D_i)^I$, we can conclude $C_i^{I'} = D_i^{I'}$, i.e. $I'$ is a model of $TD$.

## 2.3 Individual set and instantiation assertion set

Individual Set is a set of individuals whose names are denoted as $a$, $b$, $c$. Instantiation Assertion Set consists of a group of class instantiation assertions, property instantiation assertions and individual inequality assertions.

A class instantiation assertion, written as $C(a)$, states that individual $a$ belongs to class $C$. a property instantiation assertion, written as $P(a, b)$, states that there exists a relation $P$ between $a$ and $b$, and $b$ is called the value of $a$ about property $P$. Individual inequality assertion, written as $a \not\equiv b$, means that the two objects denoted by $a$ and $b$ are distinct.

Given an ontology interpretation $I$, if a class instantiation assertion $C(a)$ holds, then $a^I \in C^I$. If a property instantiation assertion $(a, b)$ holds, then $(a^I, b^I) \in P^I$. If an individual inequality assertion $a \not\equiv b$ holds, then $a^I \neq b^I$.

**Definition 10(Model of Instantiation Assertion):** Given $O$=<$T$, $X$, $TD$, $XD$, $TR$>, if there exists an ontology interpretation $I$ making an instantiation assertion $\alpha$ holds, then $I$ is said to be a model of $\alpha$. If $I$ is model of all the instantiation assertions in $XD$, then $I$ is called a model of $XD$.

**Definition 11 (Expansion of Instantiation Assertion):** Given $O$=<$T$, $X$, $TD$, $XD$, $TR$>. $C(a)$ is a class instantiation assertion in $XD$, and $e(C)$ is an expansion of $C$ with respect to $TD$. $e(C)(a)$ is said to be the expansion of $C(a)$ with respect to $TD$. Through transforming each class instantiation assertion into the form of expansion, we can get a new Instantiation Assertion Set $XD'$. The new set $XD'$ is called the expansion of $XD$ with respect to $D$, denoted as $e(XD)$.

### 2.4 Term restriction set

Given $O$=<$T$, $X$, $TD$, $XD$, $TR$>, the Term Restriction Set $TR$ is a set of term relationships in the form of subsumption, equivalence or disjointness, which is intended to restrict the logical relationship between terms in $T$. Suppose $D$, $E$ are two term formulas, the meaning of the three kinds of relationship are given as follows:

(ⅰ) $D\mathrm{M}E$ states that every instance of the class $D$ is also the instance of the class $E$.
(ⅱ) $D\equiv E$ states that every instance of the class $D$ is also the instance of the class $E$, and vice versa.
(ⅲ) $D \pitchfork E$ states that the two classes $D$ and $E$ have no instance in common.

$e(D)\mathrm{M}e(E)$, $e(D)\equiv e(E)$ and $e(D) \pitchfork e(E)$ are called the expansion of $D\mathrm{M}E$, $D\equiv E$ and $D \pitchfork E$ respectively.

**Definition 12(Model of Term Restriction):** Given $O$=<$T$, $X$, $TD$, $XD$, $TR$>, if there exists an ontology interpretation $I$ satisfying a term relation $R$ in $TR$, then we say $I$ is a model of $R$. If $I$ is the model of all the term relation in $TR$, then $I$ is called a model of $TR$.

**Definition 13(Expansion of Term Restriction Set):** Given $O$=<$T$, $X$, $TD$, $XD$, $TR$>. For convenience, we assume $TR=\{D_1\mathrm{M}E_1, D_2\equiv E_2, D_3 \pitchfork E_3\}$. If each term relation in $TR$ has been transformed into the expansion form, a new Term Restriction Set $TR'=\{e(D_1)\mathrm{M}e(E_1), e(D_2)\equiv e(E_2), e(D_3) \pitchfork e(E_3)\}$ is obtained. $TR'$ is said to be the expansion of $TR$, written as $e(TR)$.

**Definition 14(Model of Expansion of Term Restriction Set):** Given $O$=<$T$, $X$, $TD$, $XD$, $TR$>, if there exists an ontology interpretation $I$ satisfying all the term interpretations in $e(TR)$, then we call $I$ a model of $e(TR)$.

**Proposition 3:** Given $O$=<$T$, $X$, $TD$, $XD$, $TR$>, if $TD$ and $TR$ have a model $I$ in common, then $I$ also is a model of $e(TR)$.

**Proof:** Let $TR$ be $\{D_1\mathrm{M}E_1, D_2\equiv E_2, D_3 \pitchfork E_3\}$ for the sake of simplicity, then we have $e(TR)=\{e(D_1)\mathrm{M}e(E_1), e(D_2)\equiv e(E_2), e(D_3) \pitchfork e(E_3)\}$. If $TD$ and $TR$ have a common model $I$, then $D_1^I\subseteq E_1^I$, $D_2^I=E_2^I$ and $D_3^I\cap E_3^I=\varnothing$ hold since $I$ is a model of $TR$. Moreover, $I$ is also a model of $TD$, so we can conclude that $C^I=e(C)^I$ holds for any term formula $C$ according to Proposition 1. With the above conclusion, we can further obtain that $e(D_1)^I\subseteq e(E_1)^I$, $e(D_2)^I=e(E_2)^I$, $e(D_3)^I\cap e(E_3)^I=\varnothing$. So $I$ is a model of $e(TR)$.

**Proposition 4:** Given $O$=<$T$, $X$, $TD$, $XD$, $TR$>, where $TD=\{C_1\equiv D_1, C_2\equiv D_2,\ldots, C_n\equiv D_n\}$. $T_p=\{B_1, B_2,\ldots, B_m\}$ is the set of primitive terms in $TD$, and $S$ is a term formula. If $I$ is a model of $e(TR)$, then there must exist a common model $I'$ of both $TD$ and $TR$, such that $S^{I'}=e(S)^I$.
**Proof:** Since $TD=\{C_1\equiv D_1, C_2\equiv D_2,\ldots, C_n\equiv D_n\}$, the set of defined terms in $TD$ is $T_d=\{C_1, C_2,\ldots, C_n\}$. Suppose that $e(TD)=\{C_1\equiv D_1', C_2\equiv D_2',\ldots, C_n\equiv D_n'\}=\{C_1\equiv e(D_1), C_2\equiv e(D_2),\ldots, C_n\equiv e(D_n)\}$ and $TR=\{D_1\mathrm{M}E_1, D_2\equiv E_2, D_3 \pitchfork E_3\}$, we can get $e(TR)=\{e(D_1)\mathrm{M}e(E_1), e(D_2)\equiv e(E_2), e(D_3) \pitchfork e(E_3)\}$. If $I$ is a model of $e(TR)$, then $e(D_1)^I\subseteq e(E_1)^I$, $e(D_2)^I=e(E_2)^I$, $e(D_3)^I \cap e(E_3)^I=\varnothing$. Then we use $I$ to construct a new ontology interpretation $I'$, such that $B_i^{I'}=B_i^I$, for any primitive term $B_i$; $C_i^{I'}=D_i'^{I}$, for any defined term $C_i$; $a^{I'}=a^I$, for any individual $a$. With the new constructed interpretation $I'$, we can conclude $S^{I'}=e(S)^I$ for any term formula $S$, which result in $D_i^{I'}=e(D_i)^I$. Moreover, since $C_i^{I'}=D_i'^{I}=e(D_i)^I$, we can conclude $C_i^{I'}=D_i^{I'}$, i.e. $I'$ is a model of $TD$. Furthermore according to the above conclusion: $e(D_1)^I\subseteq e(E_1)^I$, $e(D_2)^I=e(E_2)^I$ and $e(D_3)^I \cap e(E_3)^I=\varnothing$, we can obtain that $D_1^{I'}\subseteq E_1^{I'}$, $D_2^{I'}=E_2^{I'}$ and $D_3^{I'} \cap E_3^{I'}=\varnothing$, i.e. $I'$ is a model of $TR$.

## 3. Consistency Checking of Ontology Concept model

### 3.1 Term checking

(1) Term satisfiability checking
Given $O$=<$T$, $X$, $TD$, $XD$, $TR$> and a term formula $D$, if there exists an ontology interpretation $I$, such that $D^I\neq\varnothing$, then $D$ is said to be satisfiable and unsatisfiable otherwise. If there exists a model of $TR$, such that $D^I\neq\varnothing$, then $D$ is satisfiable with respect to $TR$. If there exists a common model of both $TR$ and $TD$, such that $D^I\neq\varnothing$, then $D$ is said to be satisfiable with respect to $TR$ and $TD$, or else unsatisfiable with respect to $TR$ and $TD$.

(2) Term subsumption checking

Given $O$=<$T$, $X$, $TD$, $XD$, $TR$> and two term formulas $D$ and $E$, if $D^I \subseteq E^I$ holds for all the ontology interpretation $I$, then $D$ is said to be subsumed by $E$, or $E$ subsumes $D$, denoted as $\models D \mathrm{M} E$. If for all the models $I$ of $TR$, $D^I \subseteq E^I$ holds, then we say $TR$ entails that $D$ is subsumed by $E$, denoted as $TR \models D \mathrm{M} E$. If $D^I \subseteq E^I$ holds for all the common models $I$ of both $TR$ and $TD$, then we say $TR$ and $TD$ jointly entails that $D$ is subsumed by $E$, denoted as $(TR+TD) \models D \mathrm{M} E$.

(3) Term equivalence checking
Given $O$=<$T$, $X$, $TD$, $XD$, $TR$> and two term formulas $D$ and $E$, if $D^I=E^I$ holds for all the ontology interpretation $I$, then we say $D$ and $E$ are equivalent, written as $\models D \equiv E$. If $D^I=E^I$ holds for all the models $I$ of $TR$, then we say $TR$ entails that $D$ and $E$ are equivalent, written as $TR \models D \equiv E$. If for all the common models $I$ for both $TR$ and $TD$, $D^I=E^I$ holds, then we say $TR$ and $TD$ jointly entails that $D$ and $E$ are equivalent, written as $(TR+TD) \models D \equiv E$.

(4) Term disjointness checking
Given $O$=<$T$, $X$, $TD$, $XD$, $TR$> and two term formulas $D$ and $E$, if $D^I \cap E^I=\varnothing$ holds for all the ontology interpretation $I$, then we call $D$ and $E$ are disjoint, written as $\models D \pitchfork E$. If in all models $I$ of $TR$, $D^I \cap E^I=\varnothing$ holds, then we call $TR$ entails that $D$ and $E$ are disjoint, written as $TR \models D \pitchfork E$. If $D^I \cap E^I=\varnothing$ holds in all the common models $I$ for both $TR$ and $TD$, then we call $TR$ and $TD$ jointly entails that $D$ and $E$ are disjoint, written as $(TR+TD) \models D \pitchfork E$.

**Proposition 5(Reduction to Subsumption):** Given $O$=<$T$, $X$, $TD$, $XD$, $TR$>, for any two term formulas $D$, $E$:
( i ) $D$ is unsatisfiable with respect to $TR$ and $TD$, iff $(TR+TD) \models D \mathrm{M} \bot$;
( ii ) $(TR+TD) \models D \equiv E$, iff $(TR+TD) \models D \mathrm{M} E$ and $(TR+TD) \models E \mathrm{M} D$;
( iii ) $(TR+TD) \models D \pitchfork E$, iff $(TR+TD) \models (D \sqcap E) \mathrm{M} \bot$.

**Proposition 6 (Reduction to Unsatisfiability):** Given $O$=<$T$, $X$, $TD$, $XD$, $TR$>, for any two term formulas $D$, $E$:
( i ) $(TR+TD) \models D \mathrm{M} E$, iff $D \sqcap \neg E$ is unsatisfiable with respect to $TR$ and $TD$;
( ii ) $(TR+TD) \models D \equiv E$, iff both $D \sqcap \neg E$ and $\neg D \sqcap E$ are unsatisfiable with respect to $TR$ and $TD$;
( iii ) $(TR+TD) \models D \pitchfork E$, iff $D \sqcap E$ is unsatisfiable with respect to $TR$ and $TD$.

Proposition 5 and 6 can be proved easily, so the proofs are omitted here. The facts stated in the two propositions imply that all the four kinds of term checking can be reduced to the (un)satisfiability or subsumption.

**Proposition 7:** Given $O$=<$T$, $X$, $TD$, $XD$, $TR$>. $D$ and $E$ are two term formulas. $e(D)$ is the expansion of $D$ with respect to $TD$, and $e(E)$ is the expansion of $E$ with respect to $TD$. We have:
( i ) $D$ is satisfiable with respect to $TR$ and $TD$, iff $e(D)$ is satisfiable with respect to $e(TR)$;
( ii ) $(TR+TD) \models D \mathrm{M} E$, iff $e(TR) \models e(D) \mathrm{M} e(E)$;
( iii ) $(TR+TD) \models D \equiv E$, iff $e(TR) \models e(D) \equiv e(E)$;
( iv ) $(TR+TD) \models D \pitchfork E$, iff $e(TR) \models e(D) \pitchfork e(E)$.
**Proof (sufficient condition):** If $e(D)$ is satisfiable with respect to $e(TR)$, then there must exist such a model $I$ of $e(TR)$ that satisfying $e(D)^I \neq \varnothing$. According to Proposition 4, if $I$ is a model of $e(TR)$, then there must exist a common model $I'$ of both $TD$ and $TR$, such that $D^{I'}=e(D)^I$. So we can get $D^{I'} \neq \varnothing$ from $e(D)^I \neq \varnothing$ and $D^{I'}=e(D)^I$. That is $D$ is satisfiable with respect to $TR$ and $TD$.
**Proof (necessary condition):** If $D$ is satisfiable with respect to $TR$ and $TD$, then there must exist a common model $I$ of both $TR$ and $TD$, such that $D^I \neq \varnothing$. Since $I$ is a model of $TD$, $D^I=e(D)^I$ holds based on Proposition 1. Therefore we have $e(D)^I \neq \varnothing$. Moreover, according to Proposition 4, we can conclude that $I$ also is a model of $e(TR)$. So $e(D)$ is satisfiable with respect to $e(TR)$.

The remaining three proofs can be conducted similarly, so they are omitted here due to the limit of the space. Proposition 7 states that problems of term checking can be solved through checking the expansion of the terms.

## 3.2 Instantiation checking

**Definition 15 (Consistence of Instantiation Assertion):** Given $O$=<$T$, $X$, $TD$, $XD$, $TR$> and an instantiation assertion $\alpha$. If there exists such an ontology interpretation $I$ that is a model of $\alpha$, then we say $\alpha$ is consistent, and inconsistent otherwise. If $I$ is not only a model of $\alpha$, but also a common model of both $TD$ and $TR$, then we say $\alpha$ is consistent with respect to $TD$ and $TR$. If $I$ is a model of $XD$, then we say $XD$ is consistent. If $I$ is not only a model of $XD$, but also a common model of both $TD$ and $TR$, then we say $XD$ is consistent with respect to $TD$ and $TR$.

**Proposition 8:** Given $O$=<$T$, $X$, $TD$, $XD$, $TR$> and a class instantiation assertion $C(a)$, we have:
( i ) $C(a)$ is consistent with respect to $TD$ and $TR$, iff $e(C)(a)$ is consistent with respect to $e(TR)$.
( ii ) $XD$ is consistent with respect to $TD$ and $TR$, iff $e(XD)$ is consistent with respect to $e(TR)$.
**Proof (Sufficient Condition):** If $e(C)(a)$ is consistent with respect to $e(TR)$, then there exists a model $I$ of $e(TR)$ satisfying $a^I \in e(C)^I$. Since $I$ is a model of $e(TR)$, there must exits a common model $I'$ of both $TD$ and $TR$ such that $a^{I'} \in a^I$ and $C^{I'}=e(C)^I$ according to Proposition 4 So we can obtain $a^{I'} \in C^{I'}$, i.e. $C(a)$ is consistent with respect to $TD$ and $TR$.

**Proof (Necessary Condition):** If $C(a)$ is consistent with respect to $TD$ and $TR$, then there exists a common model $I$ of both $TD$ and $TR$ such that $a^I \in C^I$. Since $I$ is a model of $TD$, we have $C^I = e(C)^I$ according to Proposition 1, which imply that $a^I \in e(C)^I$. Moreover Proposition 3 states that $I$ also is a model of $e(TR)$. So we can conclude that $e(C)(a)$ is consistent with respect to $e(TR)$.

**Proposition 9:** Given an ontology $O$=<$T$, $X$, $TD$, $XD$, $TR$> and a term formula $C$, $C$ is satisfiable with respect to $TD$ and $TR$, iff $C(a)$ is consistent with respect to $TD$ and $TR$, where $a$ is an arbitrarily chosen individual name.

Proposition 8 states that the consistence of an instantiation assertion is equivalent to the consistence of the expansion of the instantiation assertion. Proposition 9 states that the consistence of instantiation assertion can be determined through checking the satisfiability of the term.

## 4. Conclusions

In this paper, an extended ontology model is constructed in DLs extending the naive model built in Refs. [6],[11],[12]. Based on the extended model, issues of consistency checking of the extended ontology concept model, including term checking and instantiation checking, are studied with the conclusion that: the four kinds of term checking, including term satisfiability checking, term subsumption checking, term equivalence checking and term disjointness checking, can be reduced to the satisfiability checking, and satisfiability checking can be transformed into instantiation checking. The problem of instantiation consistence can be decided by Tableau Algorithm (see Ref. [12] for details).

### Acknowledgments

This paper is supported by the Natural Science Fund of China (**serial No.70501022**)

## References

[1] Guarino N. Formal ontology and information systems. Proceedings of FOIS'98, Trento, Italy [C]. Amsterdam: IOS Press.1998. 3-15.

[2] Gaines, B. Editorial: using explicit ontologies in knowledge-based system development. International Journal of Human-Computer Systems, 1997,46: 181-190

[3] Guarino N. Formal ontology, conceptual analysis and knowledge representation. International Journal of Human and Computer Studies, 1995,43(5/6): 625-640.

[4] Ashenhurst R L. Ontological aspects of information modeling. Minds and Machines, 1996, 6: 287-394.

[5] Mena E, Kashyap V, Illarramendi A, et al. Domain specific ontologies for semantic information brokering on the global information infrastructure. Guarino N. (ed.) Formal Ontology in Information Systems. Amsterdam: IOS Press, 1998

[6] Wang Hongwei, Wu Jiachun, Jiang Fu. Study on formal ontology model and its application to semantic retrieval. Li Xiaoming (ed.) Advances of Search Engine and Web Mining in China. Beijing: Higher Education Press, 2003. 205-213.

[7] Deng Kai, Wu Jiachun, Wang Hongwei. Construct knowledge library using ontology. Information Science, 2003,(1):106-108.

[8] Gruber T R. Ontolingua: a translation approach to portable ontology specifications. Knowledge Acquisition, 1993,5(2): 199–220.

[9] Lenat D, Guha R. Building large knowledge based system: representation and inference in the CYC project. London: Addison-Wesley Press, 1990.

[10] MacGregor R. Inside the LOOM classifier. SIGART Bulletin, 1991,2(3):88-92.

[11] Wang Hongwei, Wu Jiachun, Jiang Fu. Study on naive ontology model based on Description Logics. System Engineering, 2003, 21(3):101-106.

[12] Baader F, Sattler U. Tableau algorithm for description logics. Dyckhoff R. (ed.) Proceeding of the International Conference on Automated Reasoning with Tableaux and Related Methods (Tableaux 2000), Vol.1847 of Lecture Notes in AI. Scotland, St Andrews: Springer-Verlag. 1-18.

**Yu Changrui** received the B.S. in Power Engineering from Northeast University of Electric Power in China in 1996. He received M.S. degree and PhD degree in Management Science and Engineering from University of Shanghai for Science and Technology in 2000 and 2003, respectively. Now he is a postdoctor in Institute of System Engineering, Shanghai Jiao Tong University. His research interests include conceptual modeling, semantic web, decision support system and so on.

**Luo Yan** received her PhD degree in Management Science and Engineering from University of Shanghai for Science and Technology in 2005, respectively. Now she is a postdoctor in Institute of System Engineering, Shanghai Jiao Tong University. Her research interests include conceptual modeling, management information system , decision support system and so on.