An extending description logic for action formalism in event ontology

Wei Liu*

School of Computer Engineering and Science,
Shanghai University,
Room 226, Xingjian Building,
149 Yanchang Road, Shanghai 200072, China
Fax: +86-21-56333061
E-mail: liuw@shu.edu.cn
*Corresponding author

Wenjie Xu, Dong Wang and Xujie Zhang

School of Computer Engineering and Science,
Shanghai University,
Room 222, Xingjian Building,
149 Yanchang Road, Shanghai 200072, China
E-mail: jiex@shu.edu.cn
E-mail: ming123@shu.edu.cn
E-mail: annie4sh@shu.edu.cn

Zongtian Liu

School of Computer Engineering and Science,
Shanghai University,
Room 220, Xingjian Building,
149 Yanchang Road, Shanghai 200072, China
E-mail: ztliu@shu.edu.cn

Abstract: Event-based ontology model is a new approach for the representation of human knowledge with a higher granularity. As the basic knowledge unit, event describes the specific facts which change over time, and actions involved in event are exactly used to represent these changing processes. There exists semantic information in action as well as rich semantic relations between different actions. Therefore, in order to describe and reason about event-based knowledge effectively, this paper firstly gives a framework of event ontology, and then proposes an action formalism based on extended description logic that could describe actions with temporal information. Furthermore, some reasoning tasks about action are discussed, such as the executability and projection of action, effects of action, etc. At last, a case study of social web verifies the feasibility of this method of knowledge representation.

Keywords: description logics; DLs; action; event; event ontology; temporal information.

Reference to this paper should be made as follows: Liu, W., Xu, W., Wang, D., Zhang, X. and Liu, Z. (2014) 'An extending description logic for action formalism in event ontology', *Int. J. Computational Science and Engineering*, Vol. 9, No. 3, pp.205–214.

Biographical notes: Wei Liu received his PhD in Computer Science and Technology and did his postdoc research in information and communication engineering from Shanghai University (SHU), Shanghai, China. Currently, he is an Associate Research Fellow in the School of Computer Engineering and Science, Shanghai University. His research interests are knowledge representation, semantic ontology and agent technologies. He has published over 30 papers in various international conferences and journals in related areas.

Wenjie Xu received her BTech in Computer Science and Technology from Hubei Normal University, Huangshi Hubei, China. Currently, she is doing her MTech in Computer Science and Technology at School of Computer Engineering and Science, Shanghai University, Shanghai, China. Her research interests are knowledge representation and reasoning.

Dong Wang received his BTech in Computer Science and Technology from Nantong University, Nantong Jiangshu, China. Currently, he is doing his MTech in Computer Science and Technology at School of Computer Engineering and Science, Shanghai University, Shanghai, China. His research interests are topic detection and tracking, and complex network.

Xujie Zhang received her MTech in Computer Science and Technology from Yunnan Normal University, Kunming Yunnan, China. Currently, she is doing her PhD in Computer Science and Technology at School of Computer Engineering and Science, Shanghai University, Shanghai, China. Her research interests are data mining and ontology reasoning.

Zongtian Liu received his BTech in Aviation Material and MTech in Computer Science and Technology from Beijing University of Aeronautics and Astronautics, Beijing, China. Currently, he is a Professor and PhD Supervisor in the School of Computer Engineering and Science, Shanghai University, Shanghai, China. His research interests are artificial intelligence, software engineering, and data mining. He has published over 100 papers in various international conferences and journals in related areas.

1 Introduction

For the representation of human knowledge in semantic web, traditional ontology has obvious defects on some issues, for example, the changing processes of states, the temporal information, the non-taxonomic relations, etc. To solve these defects, many researchers in various academic fields suggest taking events instead of concepts in ontology model to describe this kind of knowledge (Zacks and Tversky, 2001; Zong-tian et al., 2009; Chang, 2003; Zhong et al., 2010). Events are specific facts which change over time, and reflect the cognitive procedure of human beings. In the research process of an event, we all focus on two aspects: the states changing in an event which usually described by action, and the temporal information of an event. As to action, certain effects may be achieved after execution of each action. And expected results may be achieved after execution of a sequence of actions. There exists semantic information in action, and rich semantic relations between different actions, such as taxonomic relation, causal relation, follow relation, accompany relation, composite relation, etc., which have provided semantic support for the inference services about actions. For temporal information, it is a temporal constraint of states changing in event which make the changing process of states more accurate. In this paper, the action formalism that we propose combines action logics with temporal information, which will contribute to the event-based knowledge processing in semantic web.

Because of description logic's (DL) advantages in semantic representation, decidability, conceptual classification representation and extensibility, most traditional ontology languages are built on the basis of DLs. At present they are used to describe many kinds of knowledge. Current research works on extending DLs for the representation of dynamic characteristic and temporal information in event-based knowledge, focus on either combining DLs with action logics (Baader et al., 2005; Zhong-zhi et al., 2004) or combining DLs with time constructors (Artale and Franconi, 1998; Artale and Franconi, 2005). These studies are separate, so it is difficult to represent the event-based knowledge accurately. In this paper, we introduce action logics and temporal information into DLs, and combine them as the main elements of event so as to achieve the goal of accurate representation of action in event. The rest of this paper is organised as follow: in Section 2, we give several new definitions about event and event ontology. Section 3 introduces an extended description logic with temporal information, in which the temporal information as a constraint of instances added to ABox. Then in Section 4, an action formalism is proposed based on this extended description logic, including the syntax and semantics of actions, some reasoning tasks and the decidability of them. In Section 5, a series of actions in the social network application, microblogging, are studied. Section 6 introduces related work, and Section 7 gives conclusions.

2 Event and event ontology

Definition 1 (event): We define *event* as a thing happens in a certain time and environment, which some actors take part in and show some action features. Event *e* can be defined as a 6-tuple formally:

$$e ::=_{def} (A, O, T, V, P, L)$$

We call elements in 6-tuple as event factors, including *action*, *objects*, *time*, *environment*, *assertions*, and *language expressions*. A detailed description of event factors can be seen in Zong-tian et al. (2009) and Liu et al. (2010). Take event *earthquake* for example, we call the earthquake happened on May 12, 2008 in China *Wenchuan earthquake*, call the earthquake happened on March 11, 2011 in Japan *Great East Japan earthquake*. There are many common characteristics in event elements of these two *earthquakes*. They belonged to natural disasters, happened in a short time, inflicted heavy casualties and property losses, etc. So, *event class* is introduced to represent a set of events with common characteristics. *Wenchuan earthquake* and *Great East Japan earthquake* are both instances of event class *earthquake*. Therefore, in our definition, event is a knowledge unit with higher granularity than concept for representation of knowledge.

Definition 2 (event class relations): Event class relations associated by the action element are mainly as follows: one taxonomic relation and four non-taxonomic relations which are *composite relation*, *causal relation*, *follow relation* and *accompany relation*. A detailed description of relations and their characteristics can be seen in Liu et al. (2010).

Definition 3 (event ontology): An event ontology is a formal, explicit specification of a shared event model that exists objectively, denoted as *EO*. The structure of event ontology can be defined as a 3-tuple:

$$EO ::= (ECs, R, Rules)$$

where *ECs* is the set of event classes, *R* indicates all relations between events. *Rules* are expressed in logic languages which can be used to describe the transformation and inference between events. Figure 1 shows a view of event ontology framework. Classes are represented by a rectangle with rounded corners. Subclass relations are represented by hollow-headed arrows. Property relations are represented by solid-headed arrows. The event factors of event are represented by curve. And a data type is represented by an oval.

There are six event factors in an *event* in Figure 1. Our research focuses on action and time of event, and omit others factor. The relations between events are described by *hasNontaxonomicRelation* or *hasTaxonomicRelation*. *Action* is classified as *AtomicAction* and *CompositeAction*. There are also taxonomic relation and non-taxonomic relations between actions. A *CompositeAction* is composed of a set of *AtomicActions* and their relations. *Time* is also classified as *TimeInstant* and *TimeInterval*. *TimeInstant* is represented by a standardised data type *dateTime*, and has a *Granularity*. *TimeInterval* consists of a start time, an end time, and a duration time. The start time and end time represent by *TimeInstant*. The relations of time are described by *hasTemporalRelation*.

3 Extended description logic with temporal information

The architecture of a knowledge representation system based on DLs can be described by a TBox with terminologies and an ABox with assertional axioms. In the real world, the terminologies in TBox remain unchanged, but the assertional axioms in ABox are changing frequently. These changes can be considered to be triggered by actions. Therefore, we propose a kind of extended description logics with temporal information for the representation of action formalism in the next section, in which the hierarchy of knowledge maintains unchanged in TBox, the temporal information is added into ABox to describe the temporal scope of instances.

3.1 Description logics

DLs are a family of formal knowledge representation formalisms that may be viewed as fragments of first-order logic (FOL) (Baader et al., 2003). The main components of DLs are *concept*, *role* and *individual*. $\mathcal{ALC}$ is the basic DLs with five constructors, include *disjunction* ($\sqcup$), *conjunction* ($\sqcap$), *negation* ($\neg$), *existential quantification* ($\exists$) and *value restriction* ($\forall$). The other expressive DLs are extended based on $\mathcal{ALC}$. Let N_C be a set of concept names, N_R be a set of role names and N_O be a set of individual names. The syntax and semantic of $\mathcal{ALC}$ is shown in Table 1, where C, $D \in N_C$ and $R \in N_R$.

An interpretation of $\mathcal{ALC}$ is of the form $\mathcal{I} = (\triangle^{\mathcal{I}}, \triangle^{\mathcal{I}})$. $\triangle^{\mathcal{I}}$ is the domain of $\mathcal{I}$ which is a non-empty set of individuals. $\triangle^{\mathcal{I}}$ is the interpretation function of $\mathcal{I}$ which maps each concept $C \in N_C$ to a subset $C^{\mathcal{I}}$ of $\triangle^{\mathcal{I}}$, each role $R \in N_R$ to a binary relation $R^{\mathcal{I}}$ of $\triangle^{\mathcal{I}} \times \triangle^{\mathcal{I}}$, and each individual $a \in N_O$ to an element $a^{\mathcal{I}} \in \triangle^{\mathcal{I}}$.

Figure 1 A view of event ontology framework

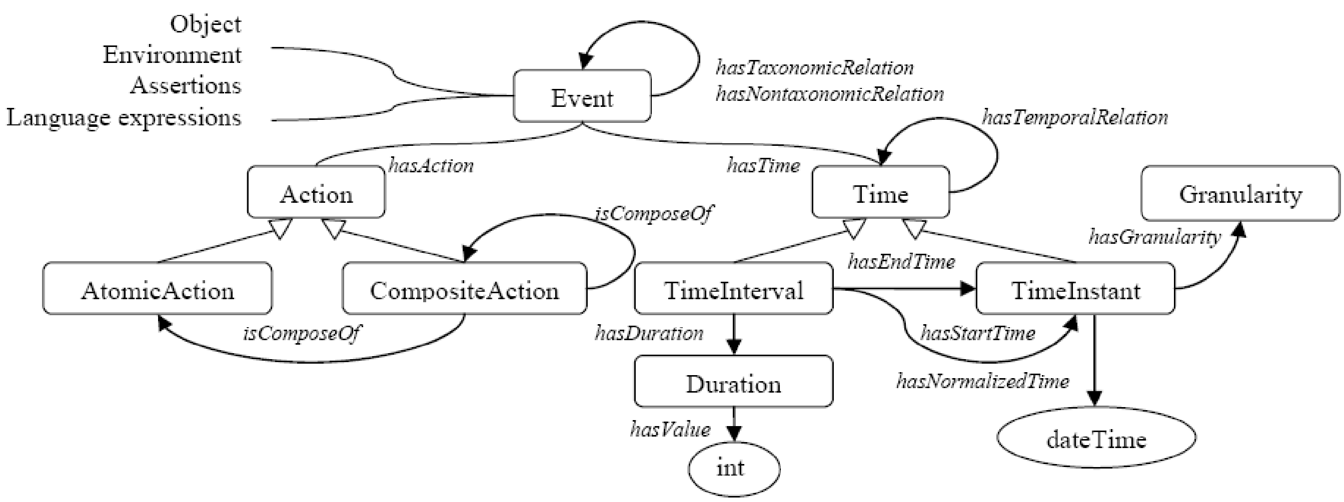

Table 1 The syntax and semantics of $\mathcal{ALC}$

Constructor	*Syntax*	*Semantics*	*Example*
Disjunction	$C \sqcup D$	$C^{\mathcal{I}} \cup D^{\mathcal{I}}$	$man \cup woman$
Conjunction	$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$	$human \cap male$
Negation	$\neg C$	$\Delta^{\mathcal{I}} \backslash C^{\mathcal{I}}$	$\neg male$
Existential quantification	$\exists R.C$	$\{x \mid \exists y(x, y) \in R^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\}\}$	$\exists hasChild.male$
Value restriction	$\forall R.C$	$\{x \mid \forall y(x, y) \in R^{\mathcal{I}} \rightarrow y \in C^{\mathcal{I}}\}\}$	$\forall hasChlid.male$

A finite set of *GCIs*(*general concept inclusions*) is called TBox $\mathcal{T}$. And a *GCI* is of the form $C \sqsubseteq D$, where $C, D \in N_C$. Write $C \equiv D$ when $C \sqsubseteq D$ and $D \sqsubseteq C$. The basic reasoning task in TBox $\mathcal{T}$ is the satisfiability of concepts which can be described that, a concept C in $\mathcal{T}$ is satisfied if there exists a model $\mathcal{I}$, $\mathcal{I} \models \mathcal{T}$ and satisfies $C^{\mathcal{I}} \neq \varnothing$. $\mathcal{I}$ is a model of $\mathcal{T}$ if it satisfies all *GCIs* in $\mathcal{T}$.

A finite set of assertional axioms is called ABox $\mathcal{A}$. ABox contains concept assertions which are of the form a:C and role assertion which are of the form (a, b): R, where a, b $\in N_O$, $C \in N_C$, and $R \in N_R$. The basic reasoning task in ABox $\mathcal{A}$ is instance checking which can be described that, an individual a is an instance of concept C if for any model $\mathcal{I}$, $\mathcal{I} \models \mathcal{T}$ and satisfies $a^{\mathcal{I}} \in C^{\mathcal{I}}$. $\mathcal{I}$ is a model of $\mathcal{A}$ if it satisfies all assertional axioms in $\mathcal{A}$.

3.2 The syntax and semantics of time

Definition 4 (time T): T is a *TimeInterval* which can be represented by an ordered pair $[t_1, t_2](t_1 \leq t_2)$. $t_1 = start(T)$ is the start time of T, $t_2 = end(T)$ is the end time of T, t_1 and t_2 are *TimeInstant*. If $t_1 = t_2$, T represents a *TimeInstant*. In fact, there are four kinds of time, $[t_1, t_2]$, $(t_1, t_2]$, $[t_1, t_2)$, and (t_1, t_2). In this paper, we take $[t_1, t_2]$ as an example.

Definition 5 (time relations): There are seven relations of T: *meets*, *before*, *overlaps*, *starts*, *ends*, *during* and *equals*, which have been researched in paper by Allen (1984), and applied widely in many fields.

The duration of time $|T|$ expresses the duration time in T. And it is composed of a duration time number and a time unit. For example, $|[14{:}00, 16{:}00]| = 2$ hours. 2 is the duration time number, *hours* is the time unit. As we know, time is with various granularities, such as *year*, *month*, *day*, *hour*, *minute*, *second* and so on. Thus, the choice of time unit and the calculation of duration number are important problems in this expression. But as usually, the temporal information in an action will not last long. So, we choose the minimum time unit which appears in the formalism as the standardised time unit. Then calculate the duration time number.

Definition 6 (semantics of time relations):

$$Meets(T_1, T_2),\ \{T_1\ Meets\ T_2 \mid t_{12} = t_{21}\}$$

$$Before(T_1, T_2),\ \{T_1\ Before\ T_2 \mid t_{12} < t_{21}\}$$

$$Overlaps(T_1, T_2),\ \{T_1\ Overlaps\ T_2 \mid t_{11} < t_{21} \wedge t_{12} < t_{22}\}$$

$$Starts(T_1, T_2),\ \{T_1\ Starts\ T_2 \mid t_{11} = t_{21} \wedge t_{12} < t_{22}\}$$

$$Ends(T_1, T_2),\ \{T_1\ Ends\ T_2 \mid t_{11} > t_{21} \wedge t_{12} = t_{22}\}$$

$$During(T_1, T_2),\ \{T_1\ During\ T_2 \mid t_{11} > t_{21} \wedge t_{12} < t_{22}\}$$

$$Equals(T_1, T_2),\ \{T_1\ Equals\ T_2 \mid t_{11} = t_{21} \wedge t_{12} = t_{22}\}$$

where $t_{11} = start(T_1)$ is the start time of T_1, $t_{12} = end(T_1)$ is the end time of T_1, and $t_{11} \leq t_{12}$. $t_{21} = start(T_2)$ is the start time of T_2, $t_{22} = end(T_2)$ is the end time of T_2, and $t_{21} \leq t_{22}$.

3.3 A temporal description logic

A temporal description logic T-$\mathcal{ALC}$ is proposed in this section to deal with temporal information in action formalism, in which TBox with terminologies is the same with $\mathcal{ALC}$, while the temporal information as a constraint of instances added to ABox. The detail definitions of ABox with temporal information and its interpretation are as follow.

Definition 7 (ABox $\mathcal{A}_T$): A finite set of assertional axioms with temporal information is called ABox $\mathcal{A}_\mathrm{T}$. Assertional axioms with temporal information contain concept assertions which are of the form $a^{[t_1,t_2]} : C$ or $a^t : C$ and role assertions which are of the form $(a,b)^{[t_1,t_2]} : R$ or $(a, b)^t : R$ where a, $b \in N_O$, $C \in N_C$, $R \in N_R$ and $t_1 \leq t_2$, t_1, t_2, t are *TimeInstant*.

In this definition, the changing processes of individual states can be expressed through the concept assertions and role assertions in $\mathcal{A}_\mathrm{T}$.

Definition 8 (interpretation $\mathcal{I}(t)$ in ABox $\mathcal{A}_T$): An interpretation of T-$\mathcal{ALC}$ in ABox $\mathcal{A}_T$ is of the form $\mathcal{I}(t) = (\Delta^{\mathcal{I}(t)}, \Delta^{\mathcal{I}(t)})$. $\Delta^{\mathcal{I}(t)}$ is the domain of $\mathcal{I}(t)$ which is a non-empty set of individuals at t and $\Delta^{\mathcal{I}(t)}$ is an interpretation function at t:

1 for each concept $C(a)$, $a \in N_O$,
- if there exists $a^t : C$, then at t, $a^{\mathcal{I}(t)} \in a^{\mathcal{I}(t)}, C^{\mathcal{I}(t)} \subseteq \Delta^{\mathcal{I}(t)}$
- if there exists $a^{[t_1,t_2]} : C$ and for every $t \in [t_1, t_2]$, then at t, $a^{\mathcal{I}(t)} \in C^{\mathcal{I}(t)}, C^{\mathcal{I}(t)} \subseteq \Delta^{\mathcal{I}(t)}$.

2 for each role $R(a, b)$, $a, b \in N_O$,
- if there exists $(a, b)^t : R$ then at t, $(a,b)^t : R \in R^{\mathcal{I}(t)}$, $R^{\mathcal{I}(t)} \subseteq \Delta^{\mathcal{I}(t)} \times \Delta^{\mathcal{I}(t)}$
- if there exists $(a,b)^{[t_1,t_2]} : R$ and for every $t \in [t_1, t_2]$, then at t, $(a,b)^{\mathcal{I}(t)} \in R^{\mathcal{I}(t)}$, $R^{\mathcal{I}(t)} \subseteq \Delta^{\mathcal{I}(t)} \times \Delta^{\mathcal{I}(t)}$.

Definition 9 (interpretation $\mathcal{I}(t)$ in TBox $\mathcal{T}$): If there exists a model $\mathcal{I}$ of $\mathcal{T}$ in $\mathcal{ALC}$, then $\mathcal{I}(t)(t \in [0, \mathrm{V}])$ is also a model of $\mathcal{T}$ in T-$\mathcal{ALC}$. 0 represents the start time of system, and V represents the end time of system. [0, V] represents all the valid time of system, and $\mathcal{I}(t)$ suggests that $\mathcal{I}$ is a model of $\mathcal{T}$ in T-$\mathcal{ALC}$ within all the valid time.

Obviously, the difference of interpretation $\mathcal{I}(t)$ between TBox and ABox in T-$\mathcal{ALC}$ is that the latter has the constraint of temporal scope. Due to the static structure of knowledge in TBox, we can also integrate them together if we only consider the constraint in ABox.

If there exist a knowledge base $K_{\text{T-}\mathcal{ALC}} = < \mathcal{T}, \mathcal{A}_T >$ and an interpretation $\mathcal{I}(t) = (\Delta^{\mathcal{I}(t)}, \bullet^{\mathcal{I}(t)})$ at t, then at t:

1 In TBox $\mathcal{T}$, if $C^{\mathcal{I}(t)} \subseteq D^{\mathcal{I}(t)}$ is satiable, then $\mathcal{I}(t)$ ($t \in (-\infty, +\infty)$) satisfies $C \sqsubseteq D$ ($\mathcal{I}(t) \models C \sqsubseteq D$). If $C^{\mathcal{I}(t)} \subseteq D^{\mathcal{I}(t)}$ and $D^{\mathcal{I}(t)} \subseteq C^{\mathcal{I}(t)}$ are both satiable, then $\mathcal{I}(t)$ satisfies $C \equiv D$ ($\mathcal{I}(t) \models C \equiv D$). If for every *GCIs* $\mathcal{P} \in \mathcal{T}$, $\mathcal{I}(t) \mid = \mathcal{P}$, then $\mathcal{I}(t)$ satisfies $\mathcal{T}$ ($\mathcal{I}(t) \models \mathcal{T}$).

2 In ABox $\mathcal{A}_T$, for concept assertions $a^{t'} : C$ and $a^{[t_1,t_2]} : C(t_1 < t_2)$, if C is satiable for $\mathcal{T}$, and $a^{\mathcal{I}(t)} \in C^{\mathcal{I}(t)}$ ($t \in [t_1, t_2] \cup t'$), then $\mathcal{I}(t)$ satisfies $a^t : C$ ($\mathcal{I}(t) \models a^t : C$). For role assertions $(a,b)^{t'} : R$ and $(a,b)^{[t_1,t_2]} : R(t_1 < t_2)$, if R is satiable for $\mathcal{T}$, and $(a,b)^{\mathcal{I}(t)} \in R^{\mathcal{I}(t)}$ ($t \in [t_1, t_2] \cup t'$), then $\mathcal{I}(t)$ satisfies $(a, b)^t : R$ ($\mathcal{I}(t) \models (a, b)^t : R$). If for every concept assertions and role assertions $\mathcal{Q} \in \mathcal{A}_T$, $\mathcal{I}(t) \models \mathcal{Q}$, then $\mathcal{I}(t)$ satisfies $\mathcal{A}_T$ ($\mathcal{I}(t) \models \mathcal{A}_T$).

3 If $\mathcal{I}(t)$ both satisfies $\mathcal{T}$ and $\mathcal{A}_T$ then $\mathcal{I}(t)$ is an interpretation of knowledge base $K_{\text{T-}\mathcal{ALC}}$.

It is worthwhile to note that we do not add time constructors in T-$\mathcal{ALC}$ rather than adding a temporal constraint of instances in ABox. The time constructors, which defined in Definitions 5 and 6, used to describe the time relations of actions. In other expressive DLs, constructors are added to express the complex knowledge in TBox rather than ABox. So, this kind of extension method is also appropriate for other expressive DLs. The shortcoming about this method is the increasing of space complexity which may cause the calculation complications increase. But evidence shows that it is still within an acceptable range.

4 The action formalism

In the last few years, extensive research has been dedicated to action representation and reasoning with quite significant results, such as *situation calculus* (McCarthy, 1963), *Event Calculus* (Kowalski and Sergot, 1986), *dynamic description logic* (*DDL*) (Zhong-zhi et al., 2004; Chang et al., 2007), and action formalism which was proposed by Baader et al. (2005). The main problem these action formalisms focus on is to achieve balance between the ability of expression and reasoning complexity. Temporal information is an important factor in action, and always be neglected in existing action formalisms. So, in this paper, in order to deal with temporal information, an extended DL with temporal information firstly proposed in last section. In this section, we focus on describing the syntax and semantics of actions, and reasoning tasks in the action formalism, such as the executability and projection of action, effects of actions, etc.

4.1 Definitions about action

The syntax and semantics of actions are introduced based on the extended description logic T-$\mathcal{ALC}$. An acyclic TBox $\mathcal{T}$ is used to define the background information, an ABox $\mathcal{A}_T$ is used to define the current situations.

Definition 10 (atomic action): Let $\mathcal{T}$ be an acyclic TBox, the definition of an atomic action involved in event is the form as:

$$A(x_1, \ldots, x_n, T) = (Pre, Post)$$

where

1 A is the action name

2 $x_1, \ldots, x_n$ are individual variables that denote the objects the action operate on

3 T is a predicate of time interval which denotes the execute time of action

4 *Pre* is a finite set of preconditions of DL predicates which denotes under which preconditions the action is executable

5 *Post* is a finite set of conditional expressions φ/ψ which denotes under what conditions the effects will be obtained after an action be executed. φ are conditions and ψ are effects, which are also a finite set of DL predicates. *Post* set can describe all possible effects when an action has not been executed yet.

For the conditional expression φ/ψ in the *Post* set of an action *A*, there are several explanations:

1 if *A* is executable, then ψ is true if and only if φ is true

2 if all φ/ψ are of the form $true/\psi$, then it can be written as ψ only

3 if the number of φ/ψ in *Post* set is *n*, then according to the true or false of φ, the number of possible effects of ψ is 2^n.

Therefore, except the standard reasoning tasks executability and projection of action, how to obtain effects of actions is also the main reasoning problem about action in this paper.

Definition 11 (instance of substitution): $\{a_1/x_1, \ldots, a_n/x_n\}$ is an instance of substitution about the action *A* when all the variables $\{x_1, \ldots, x_n\}$ except *T* in *A* are replaced by constants $\{a_1, \ldots, a_n\}$, both replaced in the *Pre* and *Post* set.

For action $A(x_1, \ldots, x_n, T) = (Pre, Post)$ defined in Definition 10, we call it abstract action. Let $\{a_1/x_1, \ldots, a_n/x_n\}$ is an instance of substitution, and *A* is replaced by $\{a_1/x_1, \ldots, a_n/x_n\}$, there are three kinds of action:

1 *action not be executed with certain effects*, in which *A* is not executed, and all the conditional expressions φ/ψ in *Post* set are the form of ψ only

2 *action not be executed with uncertain effects*, in which *A* is not executed, and there exist at least one conditional expressions φ/ψ in *Post* set which is not the form of ψ

3 *action be executed*, in which *A* is executed, the *Post* set have been processed, and only one kind of effects left.

T is the time of action, and it only can be replaced by a specific time in situation 3. Actions in situation 1 and 3 express the certain effects in their *Post* set which call certain actions, while actions in 2 express the uncertain effects in their *Post* set which call uncertain actions.

Let $\mathcal{T}$ be an acyclic TBox, a composite action *A* for $\mathcal{T}$ is a finite sequence of atomic actions $A_1, \ldots, A_k$ for $\mathcal{T}$. Then each action $A_1, \ldots, A_k$ has *composite relation* with *A*, and there may be also other relations among $A_1, \ldots,$ and A_k, such as *causal relation*, *follow relation*, *accompany relation* we defined in Definition 2.

Definition 12 [principle of inertia (PoI)]: In our action formalism, *PoI* means that predicates in ABox change if and only if they are trigger by actions, otherwise they remain unchanged, and can be merged with original concepts. A formal explanation about *PoI* is that: let $\mathcal{A}_T$ be an ABox at *t*, $\mathcal{I}(t)$ be an interpretation, $C(a)$ and R(a, b) are concept assertion and role assertion in $\mathcal{A}_T$, some actions be executed from time *t* to *t'*. If $C(a)$ and $R(a, b)$ not be changed by triggers of these actions, $C(a)$ and $R(a, b)$ are still true at *t'*:

1 For the form of $C(a)^t$ and $R(a, b)^t$: $\mathcal{A}_T$ need to be changed into $\mathcal{A}_T' = \{\mathcal{A}_T \cup C(a)^{[t,t']} \cup R(\mathrm{a,b})^{[t,t']}\} \setminus \{C(a)^t \cup R(\mathrm{a,b})^t\}$. and $\mathcal{I}(t)$ also need to be changed into $\mathcal{I}'(t') = \{\mathcal{I}(t) \cup C(a)^{t'} \cup R(\mathrm{a,b})^{t'}\} \setminus \{C(a)^t \cup R(\mathrm{a,b})^t\}$.

2 For the form of $C(a)^{[t_1,t]}$ and $R(a,b)^{[t_2,t]} (t_1 < t,\ t_2 < t)$: $\mathcal{A}_T$ need to be changed into

$$\mathcal{A}_T' = \{\mathcal{A}_T \cup C(a)^{[t_1,t']} \cup R(a,b)^{[t_2,t']}\} \setminus \{C(a)^{[t_1,t]} \cup R(a,b)^{[t_2,t]}\}.$$

Definition 13 (semantics of atomic action): Let $\mathcal{T}$ be an acyclic TBox, $\mathcal{A}_T$ be an ABox at *t*, $\mathcal{I}(t)$ be an interpretation, and $\mathcal{I}(t) \models \mathcal{T}, \mathcal{A}_T$. Let *A* be an atomic action which has not been executed at *t*, and *A* is executable for $\mathcal{T}$ and $\mathcal{A}_T$. If *A* be executed, then it changes some states and produces some new effects. That is to say, $\mathcal{A}_T$ will be changed to another $\mathcal{A}_T'$, $\mathcal{I}(t)$ will be changed to another $\mathcal{I}'(t')$, and $\mathcal{I}'(t') \models \mathcal{T}$. $[t, t'](t'>t)$ is the execute time of *A*. The semantics can be written as $\mathcal{I}(t) \Rightarrow_A^{\mathcal{T},\mathcal{A}_T} \mathcal{I}'(t')$ and $\mathcal{A}_T \Rightarrow_A^{\mathcal{T}} \mathcal{A}_T'$.

In the above definition, the semantics of action can be described as the changing process of an interpretation $\mathcal{I}(t)$ transform into another interpretation $\mathcal{I}'(t')$, while the ABox $\mathcal{A}_T$ also changes into $\mathcal{A}_T'$ over time. For each conditional expression φ/ψ in the *Post* set of *A*, the new interpretation $\mathcal{I}'(t')$ and new ABox $\mathcal{A}_T'$ after *A* executed can be calculated as following, where $C, D \in N_C$, $R,S \in N_R$, $a, b \in N_O$:

1 For each primitive concept $C(a)$ or $\neg C(a)$ in ψ, if set $(Pre \cup \varphi)$ is satisfied and consistent at *t*, then *A* is executable. If *A* be executed between *t* and *t'*, then

$$\mathcal{I}(t') = \left\{\mathcal{I}(t) \cup C(a)^{t'}\right\} \setminus \left\{\neg C(a)^t \cup D(a)^t\right\} \text{ or}$$
$$\mathcal{I}(t') = \left\{\mathcal{I}(t) \cup \neg C(a)^{t'}\right\} \setminus \left\{C(a)^t \cup D(a)^t\right\}$$

$$\mathcal{A}_{T'} = \left\{\mathcal{A}_T \cup \varphi \cup C(a)^{t'}\right\} \setminus \left\{D(a)^t\right\} \text{ or}$$
$$\mathcal{A}_{T'} = \left\{\mathcal{A}_T \cup \varphi \cup \neg C(a)^{t'}\right\} \setminus \left\{D(a)^t\right\}$$

where *D* is the conflicting concept with *C* if there exists inconsistency of $\{\mathcal{I}(t) \cup C(a)^t\}$ or $\{\mathcal{A}_T \cup Pre \cup \varphi \cup C(a)^t\}$, otherwise $D(a)^t = \varnothing$. After that, update $\mathcal{I}(t')$ into $\mathcal{I}'(t')$, update $\mathcal{A}_{T'}$ except $\neg C(a)^t$ or $C(a)^t$ into $\mathcal{A}_T'$ according to *PoI*.

2 For each primitive role R(a, b) or $\neg R$(a, b) in ψ, if set ($Pre \cup \varphi$) is satisfied and consistent at t, A is executable. If A be executed between t and t', then

$$\mathcal{I}(t') = \left\{\mathcal{I}(t) \cup R(a,b)^{t'}\right\} \backslash \left\{\neg R(a,b)^t \cup S(a,b)^t\right\} \text{ or}$$
$$\mathcal{I}(t') = \left\{\mathcal{I}(t) \cup \neg R(a,b)^{t'}\right\} \backslash \left\{R(a,b)^t \cup S(a,b)^t\right\}$$

$$\mathcal{A}_{T'} = \left\{\mathcal{A}_T \cup \varphi \cup R(a,b)^{t'}\right\} \backslash \left\{S(a,b)^t\right\} \text{ or}$$
$$\mathcal{A}_{T'} = \left\{\mathcal{A}_T \cup \varphi \cup \neg R(a,b)^{t'}\right\} \backslash \left\{S(a,b)^t\right\}$$

where S is the conflicting role with R if there exists inconsistency of $\{\mathcal{I}(t) \cup R(\text{a, b})^{t}\}$ or $\{\mathcal{A}_T \cup Pre \cup \varphi \cup R(\text{a, b})^t\}$, otherwise $R(\text{a, b})^t = \varnothing$. After that, update $\mathcal{I}(t')$ into $\mathcal{I}'(t')$, update $\mathcal{A}_{T'}$ except $\neg R(\text{a, b})^t$ or $R(\text{a, b})^t$ into $\mathcal{A}_T'$ according to *PoI*.

3 For the other concept $C(a)$ and R(a, b) in ψ, C can be spread out into a set of primitive concepts and roles according to the acyclicity of $\mathcal{T}$. So $\mathcal{I}'(t')$ and $\mathcal{A}_T'$ can also be obtained through those primitive concepts and roles.

So, we can conclude that $\mathcal{I}'(t')$ describes the states at t', but $\mathcal{A}_T'$ describes the states and states changing in [0, t'].

4.2 *Reasoning tasks about actions and their decidability*

Reasoning is the process of discovering new knowledge which can not be obtained directly from the current knowledge base. Before trying to carry out the action, we want to know whether it is indeed executable before executing the action, i.e., whether the *Pre* set are satisfied at current time, which we call executability. If the action is executable, we want to know whether achieves the desired effects, i.e., whether an assertion that we want to make true really holds after executing the action, which we call projection (Baader et al., 2005).

Definition 14 (consistency of atomic action): Let $\mathcal{T}$ be an acyclic TBox, A be an atomic action for $\mathcal{T}$, $\mathcal{I}(t)$ be an interpretation and $\mathcal{A}_T$ be an ABox at t before A be executed, $\mathcal{I}(t) \models \mathcal{T}, \mathcal{A}_T$:

1 If the *Pre* set of A is consistency for $\mathcal{I}(t)$ and $\mathcal{A}_T$, then A is consistency for *Pre*.

2 If there exists conditional expressions $\varphi_1 / \psi \in Post$, $\varphi_2 / \neg\psi \in Post$ in the *Post* set of A, where φ_1 and φ_2 are satisfied for $\mathcal{I}(t)$ and $\mathcal{A}_T$ at t, then the effects may be inconsistent after A be executed. So A is inconsistent in the *Post* set.

3 If A is consistency in *Pre* set and not inconsistent in *Post* set, then A is consistency for $\mathcal{T}$, $\mathcal{I}(t)$ and $\mathcal{A}_T$.

Definition 15 (executability and projection of atomic actions): Let $\mathcal{T}$ be an acyclic TBox, A be an atomic action for $\mathcal{T}$, $\mathcal{I}(t)$ be an interpretation and $\mathcal{A}_T$ be an ABox at t before A be executed:

1 Executability: if and only if all the concepts and roles assertions in *Pre* set of A are satisfied at t before A be executed, $\mathcal{I}(t) \models Pre$.

2 Projection: the assertion ψ is a consequence of executing A, if and only if for all models $\mathcal{I}(t)$ of $\mathcal{A}_T$ and $\mathcal{T}$, all the new interpretation $\mathcal{I}'(t')$ with $\mathcal{I}(t) \Rightarrow_A^{\mathcal{T}, \mathcal{A}_T}$ $\mathcal{I}'(t')$, we have $\mathcal{I}'(t') \models \psi$.

Lemma 1: Executability of atomic action can be reduced to a standard reasoning Problem the satisfiability problem in T-ALC.

Proof:

1 according to the definition of executability of atomic actions, it can be described as the satisfiability problem of description logic with temporal information, $\mathcal{I}(t) \models Pre$

2 the *Pre* set of action is a set of concept assertions and role assertions before the action be executed

3 the way of temporal information integrated into ALC will not cause the additional complexity of reasoning.

Therefore, the executability of atomic actions is still decidable, and can be translated into the satisfiability problem of concepts and roles in the *Pre* set.

Lemma 2: Executability and projection of atomic action can be reduced to each other in polynomial time.

The effects of actions can be reasoned through the semantics of action in Definition 13, in which we give all the steps of processing the conditional expression φ/ψ in the *Post* set. For example, there exist two actions which are not being executed:

$$A_1(a,\ b,\ c, T_1) = (\{C(a), D(b), \neg C(c)\},\ \{R(a,b),\ S(b,\ c)\})$$

$$A_2(a,\ b,\ c, T_2) = (\{C(a), R(a,b), \neg C(c)\},\ \{D(b)/S(b,c), \neg D(b)/\neg R(a,\ b)\})$$

where $C, D \in N_C$, $R, S \in N_R$, $a, b, c \in N_O$, and T_1, T_2 are *TimeInterval*. As we can see, A_1 is the *action not be executed with certain effects*, and A_2 is the *action not be executed with uncertain effects*. Let $\mathcal{I}(t_1)$ be an interpretation at t (*Meets*(t, T_1)), and suppose A_1 is executable at t, $\mathcal{I}(t) \models \{C(a)^t, D(b)^t, \neg C(c)^t\}$. If A_1 be executed, then for each conditional expression in the *Post* set, we can obtain that

$\{R(a, b)^{t_1}, S(b, c)^{t_1}\}$ ($Meets(T_1, t1)$) is the effects of A_1. Suppose A_2 is also executable at t, $\mathcal{I}(t)|=\{C(a)^t, R(a, b)^t, \neg C(c)^t\}$. If A_2 be executed, then for each conditional expression in the *Post* set, we can obtain two kinds effects of $A_2 : \{D(b)^{t_2}, S(a, b)^{t_2}\}$ ($Meets(T_2, t_2)$) is the effects of A_2 if $\{D(b) \cup Pre_2\}$ is satisfied at t, or $\{D(b)^{t_2}, \neg R(a, b)^{t_2}\}$ is the effects of A_2 if $\{\neg D(b) \cup Pre_2\}$ is satisfied at t. The new interpretation and ABox can not be obtained in this situation because the incompleteness of knowledge base. A complete case described with this action formalism is to be presented in the next section.

5 A case study in social network

Microblogging is a new form of communication in which users can broadcast and share information about their activities, opinion and status in short posts distributed by instant messages, mobile phones, e-mail or the web. Some microblogging services offer features such as privacy settings, which allow users to control who can read their microblogs and construct their own small communities. Currently, most notable microblogging services are Twitter, Tumblr, Plurk, Jaiku and so on. In China, the biggest microblogging community is Weibo. Microblogging services have also emerged as an important source of real-time news updates for recent crisis situations, such as the Mumbai terror attacks or Iran protests. The short nature of updates allows users to post news items quickly, reaching its audience in seconds. For example, such a short message as "North Korea on Friday launched a long-range rocket that appears to have disintegrated soon after blastoff and fallen into the ocean". attracted more than 10,000 users to comment and forward it. Also, this message is followed by several subsequent related messages, such as "North Korea has admitted the satellite launch failures", "The United Nations condemned North Korea to launch satellite", etc. Normally, an original news or activity message posted by user involves one or several events, and there are semantic relationships between the users' actions in microblogging community. So, it is possible analyse the usage behaviour of microblogging services by describing and reasoning about the actions with formalism we proposed.

As a case study, we try to describe and reasoning about the users' actions in Weibo community, as well, to describe and reasoning about the actions involved in the events or activities posted by users. At first, give some predicates and construct a knowledge base about users in Weibo, then discuss user's actions in Weibo, at last, focus on the development process of events posted by users with time.

Firstly, each *user* in Weibo needs some predicates to describe. The necessary predicates are *username*, *password* and *email*. While the defaults are *gender*, *age*, *location*, *school*, etc. Moreover, there are some relations between users such as *isFriendOf*, *atSameAge, atSameLocation*, *atSameSchool*, etc. Through these concepts and roles, at $t = 0$, a knowledge base $K = <\mathcal{T}, \mathcal{A}_T>$ about users can be constructed.

$\mathcal{T}$ = {*username, password, email, age, location, school, hasUsername*(*user, username*), *hasPassword*(*user, password*), *hasEmail*(*user, email*), *hasGender*(*user, username*), *hasAge*(*user, age*), *hasLocation*(*user, location*), *hasSchool*(*user, school*), *atSameAge*(*user, user*), *atSameLocation*(*user, user*), *atSameSchool*(*user, user*), *isFriendOf*(*user, user*),

$gender = female \sqcup male,$

$user = (hasUsername.username \sqcap hasPassword.password \sqcap hasEmail.email) \sqcup hasGender.gender \sqcup hasAge.age \sqcup hasLocation.location \sqcup hasSchool.school,$

beLogIn, beLogOut, event, bePosted(*event, user*), *comment, hasComment*(*event, comment*), *commentFrom*(*comment, user*)}

$\mathcal{A}_T = \{\varnothing\}$

In this knowledge base, some information about users is defined.

Secondly, seven actions can be extracted in this case, which are

$\mathbb{A}$ = {

register(*username, password, email, T*),

logIn(*user, username, password, T*),

post(*user, event, T*),

delete(*user, event, T*),

forwarding(*user, user, event, T*),

comment(*user, user, event, T*),

logOut(*user, T*)}.

For each action, the detail description is as follow:

$register(u, x, y, z, T) = (\{username(x), password(y), email(z)\}, \{user(u), age(x_1)/hasAge(u, x_1), location(x_2)/hasLocation(u, x_2), school(x_3)/hasSchool(u, x_3), hasUsername(u,x), hasPassword(u,y), hasEmail(u,z)\})$

$logIn(u, x, y, T) = (\{user(u), hasUsername(u,x), hasPassword(u,y)\}, \{beLogIn(u)^{t2}\})$

$post(u, e, T) = (\{user(u), beLogIn(u)^{t1}, event(e)\}, \{bePosted(e,u)^{t2}\})$

$delete(u, e, T) = (\{user(u), beLogIn(u)^{t1}, event(e), bePosted(e,u)^{t1}\}, \{\neg bePosted(e,u)^{t2}\})$

forwarding(*u1, u2, e, T*) =({*user*(*u1*), *user*(*u2*), *beLogIn*$(u2)^{t1}$, *isFriendOf*$(u2,u1)^{t1}$, *event*(*e*), *bePosted*$(e,u1)^{t1}$},{*bePosted*$(e,u2)^{t2}$})

comment(*u1, u2, e, T*) =({*user*(*u1*), *user*(*u2*), *beLogIn*$(u2)^{t1}$, *event*(*e*), *bePosted*$(e,u1)^{t1}$}, {*comment*(*c*)/*hasComment*$(e, c)^{t2}$, *commentFrom*(*c, u2*)})

logOut(*u, T*) =({*user*(*u*), *beLogIn*$(u)^{t1}$}, {*beLogOut*$(u)^{t2}$})

Knowledge base $K = \langle \mathcal{T}, \mathcal{A}_T \rangle$ and action set $\mathbb{A}$ defined above can construct an action formalism which we propose in this paper. Some inference services about action, such as the executability and projection of action can be obtained through checking the satisfiability of predicates in knowledge base. For example, action *post* can not be executed after action *login*, while action *login* can not be executed after action *register*. Thus, the inference services which we need can be obtained through executing a sequence of actions in $\mathbb{A}$.

Thirdly, in most cases, an original news or activity posted by user can be regarded as an event. So, it is necessary to describe and reason about actions involved in 'events' posted by users, especially in some events with semantic relations.

For example, *Lee*, *Tom*, *Lily* and *Jane* are classmates in the same school. Miss *Wang* is their teacher. Miss *Wang*(*user*(*Wang*)) and the students' parent(*user*(*Lee*), *user*(*Tom*), *user*(*Lily*), *user*(*Jane*), *atSameSchool*(*Lee*, *Tom*), *atSameSchool*(*Tom*, *Lily*), *atSameSchool*(*Lily*, *Jane*)), each of them has a account in Weibo, and follows each other. One day, *Lee* and *Tom* had a fight in school. Miss *Wang* posted this event on her Weibo. Then, this event aroused bitter controversy over Weibo among Miss *Wang* and parents. The posted events with time are as follows:

> *Wang:* @Lee and @Tom just had a fight in School. @Lily, @Jane.
>
> *Wang:* I have criticized them @Lee, @Tom, and let them apologize to each other for the fight. @Lily, @Jane.
>
> *Lee:* Thanks to teacher @Wang. I also criticized Lee after school, and ordered him to go face the wall to ponder upon his faults. @Tom, @Lily, @Jane.
>
> *Tom:* Thanks to teacher @Wang. I also criticized Tom, and let him call @Lee apologize for his fault. @Lily, @Jane.
>
> *Wang:* @Lee and @Tom get along with each other very well in school today. @Lily, @Jane.
>
> *Lily:* Hope they are all good friends, and get along with each other. @Wang, @Lee, @Tom, @Jane.

Around this original event *fight*, some subsequent events happened. Each event has one or more actions. We can extract several actions through these events, such as *fight*, *criticize*, *apologize*, *facingTheWall*, *call*, *getAlong*, etc. There are also several relations between them. For example, (*fight*, *criticize*), (*fight*, *apologize*), (*fight*, *facingTheWall*) are *causal relations*. (*criticize*, *facingTheWall*), (*apologize*, *getAlong*) are *follow* relations. Thus, a knowledge base with higher granularity can be constructed using these events and relations to support further reasoning service.

This is the case study in social web about the action formalism which we propose under the framework of event ontology. In this case, we defined some static knowledge in a knowledge base, extracted some actions for the application, and realised the development process of events with time.

6 Related work

The representation and reasoning about action is an important research topic for dynamic knowledge representation. Many formal methods are gradually emerging, but they all focus on one aspect, either too expressive to decidability, or weak expressive, or ignore temporal information. For example, *situation calculus* which first proposed by McCarthy (1963) can be seen as a sorted FOL framework. It provides a methodology to axiomatics the effects of actions, and defines its semantics using second-order axioms which may lead to undecidability (Reiter, 1991; De Giacomo et al., 2000). *Event calculus* (Kowalski and Sergot, 1986) is another logical mechanism for reasoning about time and events, which infers what is true when given what happens when and what actions do. Later, Shanahan (1999, 2000) studied three typical problems about action reasoning, the frame problem, the qualification problem and the ramification problem. However, it can not describe the procedure of action execution. Badder et al. (2005) presented a first proposal for integration DLs and action formalisms into a decidable hybrid formalism. The basic reasoning tasks executability and projection of action are mutually polynomially reducible. Zhong-zhi et al. (2004) (Chang et al., 2007) also proposed a DDL which can be used to describe and reason about both static and dynamic knowledge. It provides the syntax, semantics and reasoning about action. After, Chang et al. (2010) also proposed another kind of extended dynamic description logic EDDL(X) based on DDL.

The methods above describe and reason about action from various perspectives. But they are all not effective enough to express dynamic knowledge accurate. For example, the last two methods above are both based on DLs which inherit many advantages of DLs, but they ignore the temporal information of action which impacts the description of the changing processes of action. Therefore, on the basis of analysing the problems of existing action formalisms, and making use of the expressive power, decidable reasoning, extendibility of DLs, the research of this paper focuses on the accurate representation of action based on the framework of event ontology.

7 Conclusions

In order to research human knowledge in semantic web, this paper takes event as the unit of human knowledge, which is more close to the cognitive procedure of human beings. Under this background, we first introduce a set of definitions about event, event class relations and event ontology. Then we define the syntax and semantics of event time based on the Allen's research. Further we integrate description logic with time for describing event actions. After that, an action formalism based on the extended description logic is proposed according to the framework of event ontology, which includes the representation of actions, reasoning about actions and their decidability. At last, a case study of social web shows the effectiveness of this method of knowledge representation. Our further study work is to research the relations between atomic actions which are an important issue for the research of composite action.

Acknowledgements

This paper is supported by the Internal Natural Science Foundation of China, No. 60975033 and No. 61074135, Shanghai Natural Science Foundation of China, No. 12ZR1410900.

References

Allen, J.F. (1984) 'Towards a general theory of action and time', *Artificial Intelligence*, Vol. 23, No. 2, pp.123–154.

Artale, A. and Franconi, E. (1998) 'A temporal description logic for reasoning about actions and plans', *Journal of Artificial Intelligence Research*, Vol. 9, No. 1, pp.463–506.

Artale, A. and Franconi, E. (2005), 'Temporal description logics', *Foundations of Artificial Intelligence*, Vol. 1, pp.375–388.

Baader, F. et al. (2003) *The Description Logic Handbook: Theory, Implementation, and Applications*, Cambridge University Press, Cambridge.

Baader, F. et al. (2005) 'Integrating description logics and action formalisms: first results', *Proc. of the 12th Nat. Conf. on Artificial Intelligence (AAAI '05)*, pp.572–577.

Chang, J.h. (2003) 'Event structure and argument linking in Chinese', *Language and Linguistics*, Vol. 4, No. 2, pp.317–351.

Chang, L. et al. (2007) 'Dynamic description logic: embracing actions into description logic', *Proc. of the 20th International Workshop on Description Logics (DL '07)*, pp.243–250.

Chang, L. et al. (2010) 'Family of extended dynamic description logics', *Journal of Software*, Vol. 21, No. 1, pp.1–13.

De Giacomo, G. et al. (2000) 'ConGolog, a concurrent programming language based on the situation calculus', *Artificial Intelligence*, Vol. 121, Nos. 1–2, pp.109–169.

Kowalski, R. and Sergot, M. (1986) 'A logic-based calculus of events', *New Generation Computing*, Vol. 4, No. 1, pp.67–95.

Liu, W. et al. (2010) 'Extending OWL for modeling event-oriented ontology', *Proc. 4th International Conference on Complex, Intelligent and Software Intensive Systems (CISIS –2010)*, pp.581–586.

McCarthy, J. (1963) *Situations, Actions, and Causal Laws*, Tech. report, Repr. Semantic Information Processing, Minsky, M. (Ed.), Dept. of Computer Science, Stanford Univ. Calif., MIT Press, Cambridge, pp.410–417.

Reiter, R. (1991) 'The frame problem in the situation calculus: a simple solution (sometimes) and a completeness result for goal regression', *Artificial Intelligence and Mathematical Theory of Computation: Papers in Honor of John McCarthy*, pp.359–380, Academic Press Professional, Inc., San Diego, CA.

Shanahan, M. (1999) 'The ramification problem in the event calculus', *International Joint Conference on Artificial Intelligence*, pp.140–146.

Shanahan, M. (2000) 'An abductive event calculus planner', *The Journal of Logic Programming*, Vol. 44, Nos. 1–3, pp.207–240.

Zacks, J.M. and Tversky, B. (2001) 'Event structure in perception and conception', *Psychological Bulletin*, Vol. 127, pp.3–4.

Zhong, Z. et al. (2010) 'Event ontology and its evaluation', *Journal of Information and Computational Science*, Vol. 7, No. 1, pp.95–101.

Zhong-zhi, S. et al. (2004) 'A logic foundation for the semantic web', *Science in China Ser. E Information Sciences*, Vol. 34, No. 10, pp.1123–1138.

Zong-tian, L. et al. (2009) 'Research on event-oriented ontology model', *Computer Science*, Vol. 36, No. 11, pp.189–192.