On denotational semantics of the complex event query language STeCEQL

Huiyong Li*, Yixiang Chen

Software Engineering Institute, East China Normal University, N. Zhongshan Rd. 3663, Shanghai, China

Received 12 June 2014, www.tsi.lv

Abstract

With the complex event processing technology has been widely used in processing the information of the internet of things, many scholars have proposed a lot of event query languages(EQL) for different scenarios. Early scholars generally study the operational semantics of EQL. Recently, many researchers begin to pay attention to the correctness of the operational semantics of the EQL. Some researchers have shown the correctness of the operational semantics by proven the equivalence between the denotational semantics and the operational semantics of EQL. The internet of vehicles is an important research branch of internet of things and it has a very wide range of applications. STeCEQL is a spatial and temporal constraint EQL for the internet of vehicles. In this paper, we focus on the correctness of the operational semantics of STeCEQL. We mainly establish the denotational semantics of STeCEQL. Finally, we prove the equivalence between the two semantics of STeCEQL. Therefore, the operational semantics of STeCEQL are correct.

Keywords: Complex Event Query Language, Internet of things, Mobile System, Denotational Semantics, Operational Semantics

1 Introduction

In recent years, many researchers have concerned the internet of things and they has achieved a great deal of results [1]. Internet of vehicles is an important kind of the internet of things and it has very broad applications. Unlike the other internet of things, there are a lot of non-moving agents in internet of vehicles and many fast moving agents in it. All kinds of sensors of agents in the internet of vehicles produce great amount of temporal, spatial and other data. Meanwhile, the internet of vehicles is a performance critical system, which requires real-time processing the data in the system [2, 3]. However, the database technology cannot solve the daunting task.

In order to real-time processing these data of the internet of vehicles, some researchers have introduced the complex event processing technology into it. The complex event processing technology is filtering the amounts of data flow into the events by the EQL. When there are some events occurs, the system will real-time or near real-time to make the appropriate treatment, which based on the predefined rules base. Moody has proposed an EQL SpaTec and it has been applied to monitoring the bus system of London [4, 5]. Jin has proposed an EQL CPSL and it can describe the relationship between the properties of the internet of vehicles [6]. We have proposed STeCEQL and given its syntax and the operational semantics, which can effectively describe the internet of vehicles.

The operational semantics is an important means to describe the computer language. In the early studies, the researchers only give the operational semantics of EQL. Zhu has proposed an EQL SEL and given its operational semantics [7]. Seiriö has proposed an EQL ruleCore and given its operational semantics [8]. Wu has proposed an EQL SASE and given its operational semantics [9]. Demers has proposed an EQL Cayuga and given its operational semantics [10].

In recent years, some researchers begin to concern the correctness of the EQL’s operational semantics. Michael has proposed an EQL XChange and given its operational semantics and the denotational semantics [11]. Finally, he has demonstrated the equivalence of two semantics. Darko has proposed an EQL ETALIS and demonstrated the equivalence of its two semantics [12]. The denotational semantics is more abstract than the operational semantics. The equivalence of two semantics is often used to verify the correctness of the operational semantics.

Therefore, we establish the denotational semantics of STeCEQL and proved the equivalence between the two semantics of STeCEQL in this paper. The remainder of this paper is organized as follows: Section 2 restates the syntax of STeCEQL. Section 3 defines the denotational semantics of STeCEQL. Section 4 proves the equivalence of two semantics of STeCEQL by structural inductive method. The last Section concludes this paper.

2 Syntax and operational semantics of STeCEQL

The STeCEQL can express the base events of the internet of vehicles and the complex events composed by the base

* ***Corresponding author*** e-mail: lihuiyongchina@126.com

events in a specific relationship. The syntax of the STeCEQL is as follows:

ABexp:

$attribute ::= true \mid false \mid x_a = a \mid x_a != a$

$\mid x_a > a \mid x_a >= a \mid x_a < a \mid x_a <= a$

$\mid attribute_0 \wedge attribute_1 \mid attribute_0 \vee attribute_1$

TBexp:

$time ::= true \mid false \mid x_t\ BEFORE\ \ t \mid x_t\ AFTER\ \ t$

$\mid x_t\ EQUAL\, t \mid x_t\ OVERLAP\, t \mid x_t\ DURING\, t$

$\mid time_0 \vee time_1 \mid time_0 \wedge time_1$

LBexp:

$location ::= true \mid false \mid x_l\ EQ\, l \mid x_l\ OP\, l \mid x_l\ IN\ l$

$\mid x_l\ NORTH\ l \mid x_l\ SOURTH\ l \mid x_l\ EAST\ l\ \mid x_l\ WEST\ l$

$\mid x_l\ NORTHWEST\ l \mid x_l\ NORTHEAST\ l$

$\mid x_l\ SOURTHWEST\ l \mid x_l\ SOURTHEAST\ l$

$\mid location_0 \vee location_1 \mid locaiton_0 \wedge location_1$

DBexp:

$direction ::= true \mid false \mid x_d = d \mid x_d != d$

EBexp:

$e ::= agent^{time}(attribute1; attribute2; attribute3 \cdots)$

$\mid agent_{location}(attribute1; attribute2; attribute3 \cdots)$

$\mid agent^{time}_{location}(attribute1; attribute2; attribute3 \cdots)$

$\mid agent^{time}_{(location,\, direction)}(attribute1; attribute2; \cdots)$

CEexp:

$ce ::= e1 \wedge e2 \mid e1 \vee e2$

The operational semantics of the STeCEQL is as follows:

ABexp:

$\langle true, \sigma \rangle \to true$

$\langle false, \sigma \rangle \to false$

$\langle x_a = a, \sigma \rangle \to true, \quad if\ \sigma(x_a) = a$

$\langle x_a = a, \sigma \rangle \to false, \quad if\ \sigma(x_a) \neq a$

$\langle x_a != a, \sigma \rangle \to true, \quad if\ \sigma(x_a) \neq a$

$\langle x_a != a, \sigma \rangle \to false, if\ \sigma(x_a) = a$

$\langle x_a > a, \sigma \rangle \to true, \quad if\ \sigma(x_a) > a$

$\langle x_a > a, \sigma \rangle \to false, \quad if\ \sigma(x_a) \leq a$

$\langle x_a >= a, \sigma \rangle \to true, \quad if\ \sigma(x_a) \geq a$

$\langle x_a >= a, \sigma \rangle \to false, \quad if\ \sigma(x_a) < a$

$\langle x_a < a, \sigma \rangle \to true, \quad if\ \sigma(x_a) < a$

$\langle x_a < a, \sigma \rangle \to false, \quad if\ \sigma(x_a) \geq a$

$\langle x_a <= a, \sigma \rangle \to true, \quad if\ \sigma(x_a) \leq a$

$\langle x_a <= a, \sigma \rangle \to false, \quad if\ \sigma(x_a) > a$

$$\frac{\langle attribute_0, \sigma \rangle \to b_0 \quad \langle attribute_1, \sigma \rangle \to b_1}{\langle attribute_0 \wedge attribute_1, \sigma \rangle \to b},$$

$if\ b_0 = true\ and\ b_1 = true, b = true; else\ b = false$

$$\frac{\langle attribute_0, \sigma \rangle \to b_0 \quad \langle attribute_1, \sigma \rangle \to b_1}{\langle attribute_0 \vee attribute_1, \sigma \rangle \to b},$$

$if\ b_0 = true\ or\ b_1 = true, b = true; else\ b = false$

TBexp:

$\langle true, \sigma \rangle \to true$

$\langle false, \sigma \rangle \to false$

$\langle x_t\ BEFORET\ t, \sigma \rangle \to true, \quad if\ \sigma(x_t).endn < t.start1$

$\langle x_t\ BEFORET\ t, \sigma \rangle \to false, \quad if\ \sigma(x_t).endn \geq t.start1$

$\langle x_t\ AFTER\ t, \sigma \rangle \to true, \quad if\ \sigma(x_t).start1 > t.endn$

$\langle x_t\ AFTER\ t, \sigma \rangle \to false, \quad if\ \sigma(x_t).start1 \leq t.endn$

$\langle x_t\ EQUAL\ t, \sigma \rangle \to true,$

$if\ (\forall i \in N.\sigma(x_t).starti = t.starti\ and\ \sigma(x_t).endi = t.endi)$

$\langle x_t\ EQUAL\ t, \sigma \rangle \to false,$

$if\ (\exists i \in N.\sigma(x_t).starti \neq t.starti\ and\ \sigma(x_t).endi \neq t.endi)$

$\langle x_t\ OVERLAP\ t, \sigma \rangle \to true,$

$if\ (\sigma(x_t).endn \geq t.start1\ and\ \sigma(x_t).endn \leq t.endn)$

$or\ (\sigma(x_t).start1 \geq t.start1\ and\ \sigma(x_t).start1 \leq t.endn)$

$\langle x_t\ OVERLAP\ t, \sigma \rangle \to false,$

$if\ \sigma(x_t).endn < t.start1\ or\ \ \sigma(x_t).start1 > t.endn$

$\langle x_t\ DURING\ t, \sigma \rangle \to true,$

$if\ \sigma(x_t).start1 \geq t.start1\ and\ \sigma(x_t).end1 \leq t.endn$

$\langle x_t\ DURING\ t, \sigma \rangle \to false,$

$if\ \sigma(x_t).start1 < t.start1\ or\ \sigma(x_t).end1 > t.endn$

$$\frac{\langle time, \sigma \rangle \to true}{\langle \neg time, \sigma \rangle \to false}$$

$$\frac{\langle time,\sigma\rangle \to false}{\langle \neg time,\sigma\rangle \to true}$$

$$\frac{\langle time_0,\sigma\rangle \to b_0 \quad \langle time_1,\sigma\rangle \to b_1}{\langle time_0 \wedge time_1,\sigma\rangle \to b},$$

$if\ b_0 = true\ and\ b_1 = true, b = true; else\ b = false$

$$\frac{\langle time_0,\sigma\rangle \to b_0 \quad \langle time_1,\sigma\rangle \to b_1}{\langle time_0 \vee time_1,\sigma\rangle \to b},$$

$if\ b_0 = true\ or\ b_1 = true, b = true; else\ b = false$

Lexp:

$\langle true,\sigma\rangle \to true$

$\langle false,\sigma\rangle \to false$

$\langle x_l\ EQ\ l,\sigma\rangle \to true,$

$if\ (\forall i \in N.\sigma(x_l).rowi = l.rowi\ and\ \sigma(x_l).columni = l.endi)$

$\langle x_l\ EQ\ l,\sigma\rangle \to false,$

$if\ (\exists i \in N.\sigma(x_l).rowi \neq l.rowi\ or\ \sigma(x_l).columni \neq l.endi)$

$\langle x_l\ OP\ l,\sigma\rangle \to true,$

$if\ (\exists i,j \in N.\sigma(x_l).rowi = l.rowj\ and\ \sigma(x_l).columni = l.columnj)$

$\langle x_l\ OP\ l,\sigma\rangle \to false,$

$if\ (\forall i,j \in N.\sigma(x_l).rowi = l.rowj\ and\ \forall\sigma(x_l).columni = l.columnj)$

$\langle x_l\ IN\ l,\sigma\rangle \to true,\quad if\ \sigma(x_l) \subset l$

$\langle x_l\ IN\ l,\sigma\rangle \to false,\quad if\ \sigma(x_l) \not\subset l$

$\langle x_l\ NORTH\ l,\sigma\rangle \to true,$

$if\ (\forall i,j \in N.\sigma(x_l).rowi < l.rowj\ and\ \sigma(x_l).columnj = l.columnj)$

$\langle x_l\ NORTH\ l,\sigma\rangle \to false,$

$if\ (\exists i,j \in N.\sigma(x_l).rowi \geq l.rowj\ or\ \sigma(x_l).columnj \neq l.columnj)$

$$\frac{\langle location,\sigma\rangle \to true}{\langle \neg location,\sigma\rangle \to false}$$

$$\frac{\langle location,\sigma\rangle \to false}{\langle \neg location,\sigma\rangle \to true}$$

$$\frac{\langle location_0,\sigma\rangle \to b_0 \quad \langle location_1,\sigma\rangle \to b_1}{\langle location_0 \wedge location_1,\sigma\rangle \to b},$$

$if\ b_0 = true\ and\ b_1 = true, b = true; else\ b = false$

$$\frac{\langle location_0,\sigma\rangle \to b_0 \quad \langle location_1,\sigma\rangle \to b_1}{\langle location_0 \vee location_1,\sigma\rangle \to b},$$

$if\ b_0 = true\ or\ b_1 = true, b = true; else\ b = false$

DBexp:

$\langle true,\sigma\rangle \to true$

$\langle false,\sigma\rangle \to false$

$\langle x_d = d1,\sigma\rangle \to true,\quad if\ \sigma(x_d) = d1$

$\langle x_d = d1,\sigma\rangle \to false,\quad if\ \sigma(x_d) \neq d1$

$\langle x_d\ != d1,\sigma\rangle \to true,\quad if\ \sigma(x_d) \neq d1$

$\langle x_d\ != d1,\sigma\rangle \to false,\quad if\ \sigma(x_d) = d1$

EBexp:

$$\frac{\langle time,\sigma\rangle \to b1 \quad \langle a1,\sigma\rangle \to b2 \quad \langle a2,\sigma\rangle \to b3 \quad \cdots}{\langle agent^{time}(attribute1; attribute2; \cdots),\sigma\rangle \to true},$$

$if\ \forall b \in (b1,b2,b3,\cdots), b = true$

$$\frac{\langle time,\sigma\rangle \to b1 \quad \langle a1,\sigma\rangle \to b2 \quad \langle a2,\sigma\rangle \to b3 \quad \cdots}{\langle agent^{time}(attribute1; attribute2; \cdots),\sigma\rangle \to false},$$

$if\ \exists b \in (b1,b2,b3,\cdots), b = false$

$$\frac{\langle t,\sigma\rangle \to b1 \quad \langle l,\sigma\rangle \to b2 \quad \langle a1,\sigma\rangle \to b3 \quad \langle a2,\sigma\rangle \to b4 \quad \cdots}{\langle agent^{time}_{location}(attribute1; attribute2; \cdots),\sigma\rangle \to true},$$

$if\ \forall b \in (b1,b2,b3,b4\cdots), b = true$

$$\frac{\langle t,\sigma\rangle \to b1 \quad \langle l,\sigma\rangle \to b2 \quad \langle a1,\sigma\rangle \to b3 \quad \langle a2,\sigma\rangle \to b4 \quad \cdots}{\langle agent^{time}_{location}(attribute1; attribute2; \cdots),\sigma\rangle \to false},$$

$if\ \exists b \in (b1,b2,b3,b4\cdots), b = false$

$$\frac{\langle t,\sigma\rangle \to b1 \quad \langle l,\sigma\rangle \to b2 \quad \langle d,\sigma\rangle \to b3 \quad \langle a1,\sigma\rangle \to b4 \quad \langle a2,\sigma\rangle \to b5 \quad \cdots}{\langle agent^{t}_{(l,d)}(attribute1; attribute2; \cdots),\sigma\rangle \to true},$$

$if\ \forall b \in (b1,b2,b3,b4,b5\cdots), b = true$

$$\frac{\langle t,\sigma\rangle \to b1 \quad \langle l,\sigma\rangle \to b2 \quad \langle d,\sigma\rangle \to b3 \quad \langle a1,\sigma\rangle \to b4 \quad \langle a2,\sigma\rangle \to b5 \quad \cdots}{\langle agent^{time}_{(location,direction)}(attribute1; attribute2; \cdots),\sigma\rangle \to false},$$

$if\ \exists b \in (b1,b2,b3,b4,b5\cdots), b = false$

CEBexp:

$$\frac{\langle e1,\sigma\rangle \to b1 \quad \langle e2,\sigma\rangle \to b2}{\langle e1 \wedge e2,\sigma\rangle \to true},\quad if\ \forall b \in (b1,b2), b \equiv true$$

$$\frac{\langle e1,\sigma\rangle \to b1 \quad \langle e2,\sigma\rangle \to b2}{\langle e1 \wedge e2,\sigma\rangle \to false},\quad if\ \exists b \in (b1,b2), b \equiv false$$

$$\frac{\langle e1,\sigma\rangle \to b1 \quad \langle e2,\sigma\rangle \to b2}{\langle e1 \vee e2,\sigma\rangle \to true},\quad if\ \exists s \in (s1,s2), b \equiv true$$

$$\frac{\langle e1,\sigma\rangle \to b1 \quad \langle e2,\sigma\rangle \to b2}{\langle e1 \vee e2,\sigma\rangle \to false},\quad if\ \forall b \in (b1,b2), b \equiv false$$

3 Denotational semantics of STeCEQL

Let the states set Σ is composed by the function σ that from the storage set to different attribute values set. And then, $\sigma(X)$ is the value of the storage unit X under the state σ. The ordered pair $<attribute,\sigma>\to true$ means that the value of the expression **attribute** is true under the state σ. The value of the complex event expressions is Boolean. Let Boolean set is **B**={true, false} and the element of the set express by **b**. Therefore, in the STeCEQL, the denotational functions of the all kinds of Boolean expressions are the mappings from the states set Σ to the Boolean set **B**.

Numeric Boolean expressions **attribute∈ABexp**, denotational function $\mathcal{A}[\![attribute]\!]:\Sigma\to B$.

Temporal Boolean expressions **time∈TBexp**, denotational function $\mathcal{T}[\![time]\!]:\Sigma\to B$.

Spatial Boolean expressions **location∈LBexp**, denotational function $\mathcal{L}[\![location]\!]:\Sigma\to B$.

Directional Boolean expressions **direction∈DBexp**, denotational function $\mathcal{D}[\![direction]\!]:\Sigma\to B$.

Event Boolean expressions **e∈EBexp**, denotational function $\mathcal{E}[\![e]\!]:\Sigma\to B$.

We define the denotational semantic function by the structural induction as below:

$\mathcal{A}:\text{ABexp}\to(\Sigma\to B)$

$\mathcal{T}:\text{TBexp}\to(\Sigma\to B)$

$\mathcal{L}:\text{LBexp}\to(\Sigma\to B)$

$\mathcal{D}:\text{DBexp}\to(\Sigma\to B)$

$\mathcal{E}:\text{EBexp or CEBexp}\to(\Sigma\to B)$

ABexp:

$\mathcal{A}[\![true]\!]=\{(\sigma,true)\mid\sigma\in\Sigma\}$

$\mathcal{A}[\![false]\!]=\{(\sigma,false)\mid\sigma\in\Sigma\}$

$\mathcal{A}[\![x_a=a]\!]=\{(\sigma,true)\mid\sigma\in\Sigma\ and\ \sigma(x_a)=a\}$

$\cup\{(\sigma,false)\mid\sigma\in\Sigma\ and\ \sigma(x_a)\neq a\}$

$\mathcal{A}[\![x_a\,!=a]\!]=\{(\sigma,true)\mid\sigma\in\Sigma\ and\ \sigma(x_a)\neq a\}$

$\cup\{(\sigma,false)\mid\sigma\in\Sigma\ and\ \sigma(x_a)=a\}$

$\mathcal{A}[\![x_a>a]\!]=\{(\sigma,true)\mid\sigma\in\Sigma\ and\ \sigma(x_a)>a\}$

$\cup\{(\sigma,false)\mid\sigma\in\Sigma\ and\ \sigma(x_a)\leq a\}$

$\mathcal{A}[\![x_a>=a]\!]=\{(\sigma,true)\mid\sigma\in\Sigma\ and\ \sigma(x_a)\geq a\}$

$\cup\{(\sigma,false)\mid\sigma\in\Sigma\ and\ \sigma(x_a)<a\}$

$\mathcal{A}[\![x_a<a]\!]=\{(\sigma,true)\mid\sigma\in\Sigma\ and\ \sigma(x_a)>a\}$

$\cup\{(\sigma,false)\mid\sigma\in\Sigma\ and\ \sigma(x_a)\leq a\}$

$\mathcal{A}[\![x_a<=a]\!]=\{(\sigma,true)\mid\sigma\in\Sigma\ and\ \sigma(x_a)\geq a\}$

$\cup\{(\sigma,false)\mid\sigma\in\Sigma\ and\ \sigma(x_a)<a\}$

$\mathcal{A}[\![attribute_0\wedge attribute_1]\!]=\{(\sigma,b_0\wedge_T b_1)\mid\sigma\in\Sigma$

$and\ (\sigma,b_0)\in\mathcal{A}[\![a_0]\!]\ and\ (\sigma,b_1)\in\mathcal{A}[\![a_1]\!]\}$

$\mathcal{A}[\![attribute_0\vee attribute_1]\!]=\{(\sigma,b_0\vee_T b_1)\mid\sigma\in\Sigma$

TBexp:

$\mathcal{T}[\![true]\!]=\{(\sigma,true)\mid\sigma\in\Sigma\}$

$\mathcal{T}[\![false]\!]=\{(\sigma,false)\mid\sigma\in\Sigma\}$

$\mathcal{T}[\![x_t\ BEFORE\ t]\!]=\{(\sigma,true)\mid\sigma\in\Sigma\ and\ \sigma(x_t).n<t.1\}$

$\cup\{(\sigma,false)\mid\sigma\in\Sigma\ and\ \sigma(x_t).endn\geq t.start1\}$

$\mathcal{T}[\![x_t\ AFTER\ t]\!]=\{(\sigma,true)\mid\sigma\in\Sigma\ and\ \sigma(x_t).1>t.n\}$

$\cup\{(\sigma,false)\mid\sigma\in\Sigma\ and\ \sigma(x_t).start1\leq t.endn\}$

$\mathcal{T}[\![x_t\ EQUAL\ t]\!]=\{(\sigma,true)\mid\sigma\in\Sigma$

$and\ (\forall i\in N.\sigma(x_t).si=t.si\ \ and\ \ \sigma(x_t).ei=t.ei)\}$

$\cup\{(\sigma,false)\mid\sigma\in\Sigma$

$and\ (\exists i\in N.\sigma(x_t).si\neq t.si\ and\ \sigma(x_t).ei\neq t.ei)\}$

$\mathcal{T}[\![x_t\ OVERLAP\ t]\!]=\{(\sigma,true)\mid\sigma\in\Sigma$

$and\ (\sigma(x_t).endn\geq t.start1\ and\ \sigma(x_t).endn\leq t.start1)$

$or\ (\sigma(x_t).start1\geq t.start1\ and\ \sigma(x_t).start1\leq t.endn))\}$

$\cup\{(\sigma,false)\mid\sigma\in\Sigma\ and\ \sigma(x_t).n<t.1\ or\ \sigma(x_t).1>t.n\}$

$\mathcal{T}[\![x_t\ DURING\ t]\!]=\{(\sigma,true)\mid\sigma\in\Sigma$

$and\ \sigma(x_t).s1\geq t.s1\ and\ \sigma(x_t).e1\geq t.en\}$

$\cup\{(\sigma,false)\mid\sigma\in\Sigma$

$and\ \sigma(x_t).start1<t.start1\ and\ \sigma(x_t).end1>t.endn\}$

$\mathcal{T}[\![time_0\wedge time_1]\!]=\{(\sigma,b_0\wedge_T b_1)\mid\sigma\in\Sigma$

$and\ (\sigma,b_0)\in\mathcal{A}[\![time_0]\!]\ and\ (\sigma,b_1)\in\mathcal{A}[\![time_1]\!]\}$

$\mathcal{T}[\![time_0\vee time_1]\!]=\{(\sigma,b_0\vee_T b_1)\mid\sigma\in\Sigma$

$and\ (\sigma,b_0)\in\mathcal{A}[\![time_0]\!]\ and\ (\sigma,b_1)\in\mathcal{A}[\![time_1]\!]\}$

LBexp:

$\mathcal{L}[\![true]\!]=\{(\sigma,true)\mid\sigma\in\Sigma\}$

$\mathcal{L}[\![false]\!]=\{(\sigma,false)\mid\sigma\in\Sigma\}$

$\mathcal{L}[\![x_l\ EQ\ l]\!]=\{(\sigma,true)\mid\sigma\in\Sigma$

$and\ \forall i\in N.\sigma(x_l).rowi=l.rowi\ and\ \sigma(x_l).columni=l.columni\}$

$\cup\{(\sigma,false)\mid\sigma\in\Sigma$

$and\ \exists i\in N.\sigma(x_l).rowi\neq l.rowi\ or\ \sigma(x_l).columni\neq l.columni\}$

$\mathcal{L}[\![x_l\ OP\ l]\!]=\{(\sigma,true)\mid\sigma\in\Sigma$

$and\ \exists i,j\in N.\sigma(x_l).rowi=l.rowj\ and\ \sigma(x_l).columni=l.columnj\}$

$\cup\{(\sigma,false)\mid\sigma\in\Sigma$

$and\ \forall i,j\in N.\sigma(x_l).rowi\neq l.rowj\ and\ \sigma(x_l).columni\neq l.columnj\}$

$\mathcal{L}[\![x_l\ IN\ l]\!]=\{(\sigma,true)\mid\sigma\in\Sigma\ and\ \sigma(x_l)\subset l\}$

$\cup\{(\sigma,false)\mid\sigma\in\Sigma\ and\ \sigma(x_l)\not\subset l\}$

$\mathcal{L}[\![x_l\ NORTH\ l]\!]=\{(\sigma,true)\mid\sigma\in\Sigma$

$and\ \forall i,j \in N.\sigma(x_l).ri < l.rj\ and\ \sigma(x_l).cj = l.cj\}$

$\cup\{(\sigma, false) \mid \sigma \in \Sigma\ and\ \exists i,j \in N.\sigma(x_l).ri \geq l.rj\}$

$\mathcal{L}[\![location_0 \wedge location_1]\!] = \{(\sigma, b_0 \wedge_T b_1) \mid \sigma \in \Sigma$

$and\ (\sigma,b_0) \in \mathcal{A}[\![location_0]\!]\ and\ (\sigma,b_1) \in \mathcal{A}[\![location_1]\!]\}$

$\mathcal{L}[\![location_0 \vee location_1]\!] = \{(\sigma, b_0 \vee_T b_1) \mid \sigma \in \Sigma$

$and\ (\sigma,b_0) \in \mathcal{A}[\![location_0]\!]\ and\ (\sigma,b_1) \in \mathcal{A}[\![locaiton_1]\!]\}$

DBexp:

$\mathcal{D}[\![true]\!] = \{(\sigma,true) \mid \sigma \in \Sigma\}$

$\mathcal{D}[\![false]\!] = \{(\sigma,false) \mid \sigma \in \Sigma\}$

$\mathcal{D}[\![x_d = d]\!] = \{(\sigma,true) \mid \sigma \in \Sigma\ and\ \sigma(x_d) = d\}$

$\cup\{(\sigma, false) \mid \sigma \in \Sigma\ and\ \sigma(x_d) \neq d\}$

$\mathcal{D}[\![x_d\ != d]\!] = \{(\sigma,true) \mid \sigma \in \Sigma\ and\ \sigma(x_d) \neq d\}$

$\cup\{(\sigma, false) \mid \sigma \in \Sigma\ and\ \sigma(x_d) = d\}$

EBexp:

$\mathcal{E}[\![agent^{time}(attribute1; attribute2; attribute3; \cdots)]\!]$

$= \{(\sigma, b1 \wedge_T b2 \wedge_T b3 \wedge_T \cdots) \mid \sigma \in \Sigma$

$and\ (\sigma,b1) \in \mathcal{T}[\![t]\!]\ and\ (\sigma,b2) \in \mathcal{A}[\![a1]\!]$

$and\ (\sigma,b3) \in \mathcal{A}[\![a2]\!] and\ (\sigma,b4) \in \mathcal{A}[\![a3]\!]\}$

$\mathcal{E}[\![agent_{location}(attribute1; attribute2; attribute3; \cdots)]\!]$

$= \{(\sigma, b1 \wedge_T b2 \wedge_T b3 \wedge_T \cdots) \mid \sigma \in \Sigma$

$and\ (\sigma,b1) \in \mathcal{L}[\![location]\!]\ and\ (\sigma,b2) \in \mathcal{A}[\![a1]\!]$

$and\ (\sigma,b3) \in \mathcal{A}[\![a2]\!] and\ (\sigma,b4) \in \mathcal{A}[\![a3]\!]\}$

$\mathcal{E}[\![agent^{time}_{locaiton}(attribute1; attribute2; attribute3; \cdots)]\!]$

$= \{(\sigma, b1 \wedge_T b2 \wedge_T b3 \wedge_T \cdots) \mid \sigma \in \Sigma$

$and\ (\sigma,b1) \in \mathcal{T}[\![t]\!]\ and\ (\sigma,b2) \in \mathcal{L}[\![location]\!]$

$and\ (\sigma,b3) \in \mathcal{A}[\![a1]\!]\ and\ (\sigma,b4) \in \mathcal{A}[\![a2]\!] and\ (\sigma,b5) \in \mathcal{A}[\![a3]\!]\}$

$\mathcal{E}[\![agent^{time}_{(locaiton,direction)}(attribute1; attribute2; \cdots)]\!]$

$= \{(\sigma, b1 \wedge_T b2 \wedge_T b3 \wedge_T \cdots) \mid \sigma \in \Sigma$

$and\ (\sigma,b1) \in \mathcal{T}[\![t]\!]\ and\ (\sigma,b2) \in \mathcal{L}[\![l]\!]\ and\ (\sigma,b3) \in \mathcal{D}[\![d]\!]$

$and\ (\sigma,b4) \in \mathcal{A}[\![a1]\!]\ and\ (\sigma,b5) \in \mathcal{A}[\![a2]\!] and\ (\sigma,b6) \in \mathcal{A}[\![a3]\!]\}$

CEBexp:

$\mathcal{E}[\![e1 \wedge e2]\!] = \{(\sigma, b1 \wedge_T b2) \mid \sigma \in \Sigma$

$and\ (\sigma,b1) \in \mathcal{E}[\![e1]\!]\ and\ (\sigma,b2) \in \mathcal{E}[\![e2]\!]\}$

$\mathcal{E}[\![e1 \vee e2]\!] = \{(\sigma, b1 \vee_T b2) \mid \sigma \in \Sigma$

$and\ (\sigma,b1) \in \mathcal{E}[\![e1]\!]\ and\ (\sigma,b2) \in \mathcal{E}[\![e2]\!]\}$

4 Equivalence between operational semantics and denotational semantics

The operational semantics of STeCEQL describes the behavioural characteristics of each step. The denotational semantics is more abstract than the operational semantics. The denotational semantics describes the relationships between the state sets. To illustrate the correctness of the operation semantics, we prove the equivalence between the operational semantics and the denotational semantics of STeCEQL.

Theorem 1: For every expression **attribute∈ABexp**, we have $\mathcal{A}[\![attribute]\!] = \{(\sigma,b) \mid <attribute, \sigma> \to b\}$.

Proof. We prove the theorem by structural induction. We have that

$P(attribute) \Leftrightarrow_{def}$

$\mathcal{A}[\![attribute]\!] = \{(\sigma,b) \mid <attribute, \sigma> \to b\}$

The case: attribute≡true.

Let $(\sigma,b) \in \mathcal{A}[\![true]\!] \Leftrightarrow \sigma \in \Sigma\ and\ b \equiv true$.

Obviously, if $(\sigma,b) \in \mathcal{A}[\![true]\!]$, then b≡true and $<true, \sigma> \to true$.

Conversely, if $<true, \sigma> \to true$, then the only possible derive is b≡true, thus $(\sigma,b) \in \mathcal{A}[\![true]\!]$.

The case: attribute≡(x_a=a), x_a is the storage unit.

By the definition:

$\mathcal{A}[\![x_a = a]\!] = \{(\sigma,true) \mid \sigma \in \Sigma\ and\ \sigma(x_a) = a\}$

$\cup\{(\sigma, false) \mid \sigma \in \Sigma\ and\ \sigma(x_a) \neq a\}$.

Then $(\sigma,true) \in \mathcal{A}[\![x_a = a]\!] \Leftrightarrow \sigma \in \Sigma\ and\ \sigma(x_a) = a$.

If $(\sigma,true) \in \mathcal{A}[\![x_a = a]\!]$, then $\sigma(x_a) = a$.

By the operational semantics of the expression, we get $<x_a{=}a, \sigma> \to true$.

Conversely, suppose $<x_a{=}a, \sigma> \to true$, then there must be a derivation as below:

$$\frac{\sigma(x_a) = a}{<x_a{=}a, \sigma> \to true}$$

Thus, $(\sigma,true) \in \mathcal{A}[\![true]\!]$.

Hence, $(\sigma,true) \in \mathcal{A}[\![true]\!] \Leftrightarrow <x_a{=}a, \sigma> \to true$

Similarly,

$(\sigma, false) \in \mathcal{A}[\![true]\!] \Leftrightarrow <x_a{=}a, \sigma> \to false$.

Thus, we can get: $\mathcal{A}[\![x_a{=}a]\!] \Leftrightarrow \{(\sigma,b) \mid <x_a{=}a, \sigma> \to b\}$

The case: attribute≡($attribute_0$∧$attribute_1$), let **$attribute_0$** and **$attribute_1$** are **ABexp**.

Suppose P(**$attribute_0$**) and P(**$attribute_1$**) are true.

By the definition:

$(\sigma,b) \in \mathcal{A}[\![attribute_0 \wedge attribute_1]\!]$

$\Leftrightarrow \sigma \in \Sigma\ and\ \exists b_0, b_1 . b = b_0 \wedge_T b_1$

$and\ (\sigma,b_0) \in \mathcal{A}[\![attribute_0]\!]\ and\ (\sigma,b_1) \in \mathcal{A}[\![attribute_1]\!]$.

Thus, suppose $(\sigma,b) \in \mathcal{A}[\![attribute_0 \wedge attribute_1]\!]$, then $\exists b_0, b_1\ \ (\sigma,b_0) \in \mathcal{A}[\![a_0]\!]\ and\ (\sigma,b_1) \in \mathcal{A}[\![a_1]\!]$

By the suppose, the P(**$attribute_0$**) and P(**$attribute_1$**) are true, then

$< attribute_0, \sigma > \to b_0$ *and* $< attribute_1, \sigma > \to b_1$

Hence, we can derive $< attribute_0 \wedge attribute_1, \sigma > \to b,\ \ b = b_0 \wedge_T b_1$.

Conversely, every derivation of $< attribute_0 \wedge attribute_1, \sigma > \to b$ must have the follows:

$$\dfrac{\dfrac{\vdots}{< attribute_0, \sigma > \to b_0} \quad \dfrac{\vdots}{< attribute_1, \sigma > \to b_1}}{< attribute_0 \wedge attribute_1, \sigma > \to b}$$

For a **b₀** and **b₁**,we can derive $b = b_0 \wedge_T b_1$.

Because the P(**attribute₀**) and P(**attribute₁**) are true, $(\sigma, b_0) \in \mathcal{A}[\![attribute_0]\!]$ *and* $(\sigma, b_1) \in \mathcal{A}[\![attribute_1]\!]$.

Hence, $(\sigma, b) \in \mathcal{A}[\![attribute]\!]$.

The proofs of other cases are completely analogous.

We finish the proof of this theorem.

Theorem 2: For every expression time**∈TBexp**, we have

$\mathcal{T}[\![time]\!] = \{(\sigma, b) \mid < time, \sigma > \to b\}$

Proof. We prove the theorem by structural induction. We have that

$P(time) \Leftrightarrow_{def} \mathcal{T}[\![time]\!] = \{(\sigma, b) \mid < time, \sigma > \to b\}$

The case: time≡true

Let $(\sigma, b) \in \mathcal{T}[\![true]\!] \Leftrightarrow \sigma \in \Sigma$ *and* $b \equiv true$.

Obviously, if $(\sigma, b) \in \mathcal{T}[\![true]\!]$ then $b \equiv true$而且 $< true, \sigma > \to true$.

Conversely, if $< true, \sigma > \to true$, then the only possible derive is b≡true, thus $(\sigma, b) \in \mathcal{T}[\![true]\!]$.

The case: time≡(x_t BEFORE t), x_t is the storage unit.

By the definition:

$\mathcal{T}[\![x_t\ BEFORE\ t]\!] = \{(\sigma, true)$

$\mid \sigma \in \Sigma$ *and* $\sigma(x_t).endn < t.start1\}$

$\cup \{(\sigma, false) \mid \sigma \in \Sigma$ *and* $\sigma(x_t).endn \geq t.start1\}$

Then

$(\sigma, true) \in \mathcal{T}[\![x_t\ BEFORE\ t]\!]$

$\Leftrightarrow \sigma \in \Sigma$ *and* $\sigma(x_t).endn < t.start1$.

If $(\sigma, true) \in \mathcal{T}[\![x_t\ BEFORE\ t]\!]$, then $\sigma(x_t).endn < t.start1$.

By the operational semantics of the expression, we get $< x_t\ BEFORE\ t, \sigma > \to true$.

Conversely, suppose $< x_t\ BEFORE\ t, \sigma > \to true$, then there must be a derivation as below:

$$\dfrac{\sigma(x_t).endn < t.start1}{< x_t\ BEFORE\ t, \sigma > \to true}$$

Thus, $(\sigma, true) \in \mathcal{T}[\![x_t\ BEFORE\ t]\!]$.

Hence, $(\sigma, true) \in \mathcal{T}[\![x_t\ BEFORE\ t]\!]$ $\Leftrightarrow < x_t\ BEFORE\ t, \sigma > \to true$.

Similarly,

$(\sigma, false) \in \mathcal{T}[\![x_t\ BEFORE\ t]\!] \Leftrightarrow < x_a = a, \sigma > \to false$.

Thus we can get:

$\mathcal{T}[\![x_t\ BEFORE\ t]\!] \Leftrightarrow \{(\sigma, b) \mid < x_t\ BEFORE\ t, \sigma > \to b\}$

The case: time≡(time₀∧time₁), let **time₀** and **time₁** are **TBexp**.

Suppose P(**time₀**) and P(**time₁**) are true.

By the definition:

$(\sigma, b) \in \mathcal{T}[\![time_0 \wedge time_1]\!] \Leftrightarrow \sigma \in \Sigma$

and $\exists b_0, b_1 . b = b_0 \wedge_T b_1$

and $(\sigma, b_0) \in \mathcal{T}[\![time_0]\!]$ *and* $(\sigma, b_1) \in \mathcal{T}[\![time_1]\!]$

Thus, suppose $(\sigma, b) \in \mathcal{T}[\![time_0 \wedge time_1]\!]$, then $(\sigma, b_0) \in \mathcal{T}[\![time_0]\!]$ *and* $(\sigma, b_1) \in \mathcal{T}[\![time_1]\!]$.

By the suppose, the P(**time₀**) and P(**time₁**) are true, then

$< time_0, \sigma > \to b_0$ *and* $< time_1, \sigma > \to b_1$

Hence, we can derive $< time_0 \wedge time_1, \sigma > \to b,\ \ b = b_0 \wedge_T b_1$.

Conversely, every derivation of $< time_0 \wedge time_1, \sigma > \to b$ must have the follows:

$$\dfrac{\dfrac{\vdots}{< time_0, \sigma > \to b_0} \quad \dfrac{\vdots}{< time_1, \sigma > \to b_1}}{< time_0 \wedge time_1, \sigma > \to b}$$

For a b_0 and b_1, we can derive $b = b_0 \wedge_T b_1$

Because the P(**time₀**) and P(**time₁**) are true, $(\sigma, b_0) \in \mathcal{T}[\![time_0]\!]$ *and* $(\sigma, b_1) \in \mathcal{T}[\![time_1]\!]$.

Hence, $(\sigma, b) \in \mathcal{T}[\![time]\!]$.

The proofs of other cases are completely analogous.

We finish the proof of this theorem.

Theorem 3: For every expression **location∈LBexp**, we have $\mathcal{L}[\![location]\!] = \{(\sigma, b) \mid < location, \sigma > \to b\}$

Proof. We prove the theorem by structural induction. We have that

$P(location) \Leftrightarrow_{def}$

$\mathcal{L}[\![location]\!] = \{(\sigma, b) \mid < location, \sigma > \to b\}$

The case: location≡true.

Let $(\sigma, b) \in \mathcal{L}[\![true]\!] \Leftrightarrow \sigma \in \Sigma$ *and* $b \equiv true$

Obviously, if $(\sigma, b) \in \mathcal{L}[\![true]\!]$, then $b \equiv true$ *and* $< true, \sigma > \to true$.

Conversely, if $< true, \sigma > \to true$, then the only possible derive is b≡true, thus $(\sigma, b) \in \mathcal{L}[\![true]\!]$.

The case: location≡(x_l EQ l), x_l is the storage unit.

By the definition:

$$\mathcal{L}[\![x_l\ EQ\, l]\!] = \{(\sigma, true) \mid \sigma \in \Sigma$$
$$and\ \forall i \in N.\sigma(x_l).rowi = l.rowi\ and\ \sigma(x_l).columni = l.endi\}$$
$$\cup\{(\sigma, false) \mid \sigma \in \Sigma$$
$$and\ (\exists i \in N.\sigma(x_l).ri \neq l.ri\ or\ \sigma(x_l).ci \neq l.ci)\}$$

Then $(\sigma, true) \in \mathcal{L}[\![x_l\ EQ\, l]\!] \Leftrightarrow$ $\sigma \in \Sigma\ and\ \forall i \in N.\sigma(x_l).rowi = l.rowi\ and\ \sigma(x_l).columni = l.endi$.

If $(\sigma, true) \in \mathcal{L}[\![x_l\ EQ\, l]\!]$, then $\forall i \in N.\sigma(x_l).rowi = l.rowi\ and\ \sigma(x_l).columni = l.endi$.

By the operational semantics of the expression, we get $< x_l\ EQ\, l, \sigma > \rightarrow true$.

Conversely, suppose $< x_l\ EQ\, l, \sigma > \rightarrow true$, then there must be a derivation as below:

$$\frac{\forall i \in N.\sigma(x_l).rowi = l.rowi\ and\ \sigma(x_l).columni = l.endi}{< x_l\ EQ\, l, \sigma > \rightarrow true}$$

Thus, $(\sigma, true) \in \mathcal{L}[\![x_l\ EQ\, l]\!]$.

Hence, $(\sigma, true) \in \mathcal{L}[\![x_l\ EQ\, l]\!] \Leftrightarrow < x_l\ EQ\, l, \sigma > \rightarrow true$.

Similarly,

$(\sigma, false) \in \mathcal{L}[\![x_l\ EQ\, l]\!] \Leftrightarrow < x_a = a, \sigma > \rightarrow false$.

Thus we can get:

$\mathcal{L}[\![x_l\ EQ\, l]\!] \Leftrightarrow \{(\sigma, b) \mid < x_l\ EQ\, l, \sigma > \rightarrow b\}$

The case: locaiton≡($location_0$∧$location_1$), let $location_0$ and $location_1$ are LBexp.

Suppose P(**location₀**) and P(**location₁**) are true.

By the definition:

$$(\sigma, b) \in \mathcal{L}[\![location_0 \wedge location_1]\!] \Leftrightarrow \sigma \in \Sigma$$
$$and\ \exists b_0, b_1.b = b_0 \wedge_T b_1$$
$$and\ (\sigma, b_0) \in \mathcal{L}[\![location_0]\!]\ and\ (\sigma, b_1) \in \mathcal{L}[\![location_1]\!]$$

By suppose, the P(**location₀**) and P(**location₁**) are true, then

$(\sigma, b_0) \in \mathcal{L}[\![location_0]\!]$ and $(\sigma, b_1) \in \mathcal{L}[\![location_1]\!]$

Hence, we can derive $< location_0 \wedge location_1, \sigma > \rightarrow b$, $b = b_0 \wedge_T b_1$.

Conversely, every derivation of $< location_0 \wedge location_1, \sigma > \rightarrow b$ must have the follows:

$$\frac{\dfrac{\vdots}{< location_0, \sigma > \rightarrow b_0} \quad \dfrac{\vdots}{< location_1, \sigma > \rightarrow b_1}}{< location_0 \wedge location_1, \sigma > \rightarrow b}$$

For a b_0 and b_1, we can derive $b = b_0 \wedge_T b_1$.

Because the P(**location₀**) and P(**location₁**) are true, $(\sigma, b_0) \in \mathcal{L}[\![location_0]\!]$ and $(\sigma, b_1) \in \mathcal{L}[\![location_1]\!]$.

Hence, $(\sigma, b) \in \mathcal{L}[\![location]\!]$.

The proofs of other cases are completely analogous.

We finish the proof of this theorem.

Theorem 4: For every expression **direction∈LBexp**, we have $\mathcal{D}[\![direction]\!] = \{(\sigma, b) \mid < direction, \sigma > \rightarrow b\}$

Proof. We prove the theorem by structural induction. We have that $P(direction) \Leftrightarrow_{def}$ $\mathcal{D}[\![direction]\!] = \{(\sigma, b) \mid < direction, \sigma > \rightarrow b\}$

The case: direction≡true.

Let $(\sigma, b) \in \mathcal{D}[\![true]\!] \Leftrightarrow \sigma \in \sum\ and\ b \equiv true$.

Obviously, if $(\sigma, b) \in \mathcal{D}[\![true]\!]$, then $b \equiv true$ and $< true, \sigma > \rightarrow true$.

Conversely, if $< true, \sigma > \rightarrow true$, then the only possible derive is b≡true, thus $(\sigma, b) \in \mathcal{D}[\![true]\!]$.

The case: direction≡(x_d=d), x_d is the storage unit.

By the definition:

$$\mathcal{D}[\![x_d = d]\!] = \{(\sigma, true) \mid \sigma \in \Sigma\ and\ \sigma(x_d) = d\}$$
$$\cup\{(\sigma, false) \mid \sigma \in \Sigma\ and\ \sigma(x_d) \neq d\}$$

Then

$(\sigma, true) \in \mathcal{D}[\![x_d = d]\!] \Leftrightarrow \sigma \in \Sigma\ and\ \sigma(x_d) = d$.

If $(\sigma, true) \in \mathcal{D}[\![x_d = d]\!]$, then $\sigma(x_d) = d$.

By the operational semantics of the expression, we get $< x_d = d, \sigma > \rightarrow true$.

Conversely, suppose $< x_d = d, \sigma > \rightarrow true$, then there must be a derivation as below:

$$\frac{\sigma(x_d) = d}{< x_d = d, \sigma > \rightarrow true}$$

Thus, $(\sigma, true) \in \mathcal{D}[\![true]\!]$.

Hence, $(\sigma, true) \in \mathcal{D}[\![true]\!] \Leftrightarrow < x_d = d, \sigma > \rightarrow true$.

Similarly,

$(\sigma, false) \in \mathcal{D}[\![true]\!] \Leftrightarrow < x_d = d, \sigma > \rightarrow false$.

Thus, we can get: $\mathcal{D}[\![x_d = d]\!] \Leftrightarrow \{(\sigma, b) \mid < x_d = d, \sigma > \rightarrow b\}$

The proofs of other cases are completely analogous.

We finish the proof of this theorem.

Theorem 5: For every expression **e∈EBexp**, we have $\mathcal{E}[\![e]\!] = \{(\sigma, b) \mid < e, \sigma > \rightarrow b\}$

Proof. We prove the theorem by structural induction. We have that $P(e) \Leftrightarrow_{def} \mathcal{E}[\![e]\!] = \{(\sigma, b) \mid < e, \sigma > \rightarrow b\}$.

The case: e≡ $agent^{time}(attribute_1; attribute_2; attribute_3)$, let **time** is **EBexp**, **attributes** are **ABexp**,

Suppose P(**time**) and P(**attribute**)s are true.

By the definition:

$$(\sigma, b) \in \mathcal{E}[\![agent^{time}(attribute_1; attribute_2; attribute_3)]\!]$$
$$\Leftrightarrow \sigma \in \sum\ and\ \exists b_1, b_2, b_3, b_4.b = b_1 \wedge_T b_2 \wedge_T b_3 \wedge_T b_4$$

$and\ (\sigma, b_1) \in \mathcal{T}[\![time]\!]\ and\ (\sigma, b_2) \in \mathcal{A}[\![attribute_1]\!]$

$and\ (\sigma, b_3) \in \mathcal{A}[\![attribute_2]\!]\ and\ (\sigma, b_4) \in \mathcal{A}[\![attribute_3]\!]$

Thus, suppose $(\sigma, b) \in \mathcal{E}[\![agent^{time}(attribute1; attribute2; attribute3)]\!]$, then $\exists b_1, b_2, b_3, b_4, (\sigma, b_1) \in \mathcal{T}[\![time]\!]$ and $(\sigma, b_2) \in \mathcal{A}[\![a1]\!]$ and $(\sigma, b_3) \in \mathcal{A}[\![a2]\!]$and $(\sigma, b_4) \in \mathcal{A}[\![a3]\!]$.

By the suppose P(**time**) and P(**attribute**)s are true, then $< time, \sigma > \to b_1$ and $< attribute_1, \sigma > \to b_2$ $and < attribute_2, \sigma > \to b_3 and < attribute_3, \sigma > \to b_4$.

Hence, we can derive $< agent^{time}(attribute1; attribute2; attribute3), \sigma > \to b,$ $b = b_1 \wedge_T b_2 \wedge_T b_3 \wedge_T b_4$.

Conversely, every derivation of $< agent^{time}(attribute1; attribute2; attribute3), \sigma > \to b$ must have the follows:

$$\dfrac{\dfrac{\vdots}{< t, \sigma > \to b_1} \quad \dfrac{\vdots}{< a_1, \sigma > \to b_2} \quad \dfrac{\vdots}{< a_2, \sigma > \to b_3}}{< agent_t(a_1, a_2, a_3), \sigma > \to b}$$

For a **time** and **attributes**, we can derive $b = b_1 \wedge_T b_2 \wedge_T b_3 \wedge_T b_4$.

Because the P(**time**) and P(**attribute**) are true, $(\sigma, b_1) \in \mathcal{T}[\![time]\!]\ and\ (\sigma, b_2) \in \mathcal{A}[\![attribute_1]\!]$ $and\ (\sigma, b_3) \in \mathcal{A}[\![attribute_2]\!] and\ (\sigma, b_4) \in \mathcal{A}[\![attribute_3]\!]$.

Hence,

$(\sigma, b) \in \mathcal{E}[\![agent^{time}(attribute1; attribute2; attribute3)]\!]$.

The proofs of other cases are completely analogous.

We finish the proof of this theorem.

Theorem 6: For every expression **ce∈CEBexp**, we have $\mathcal{E}[\![ce]\!] = \{(\sigma, b) \mid < ce, \sigma > \to b\}$

Proof. We prove the theorem by structural induction. We have that $P(e) \Leftrightarrow_{def} \mathcal{E}[\![ce]\!] = \{(\sigma, b) \mid < ce, \sigma > \to b\}$.

The case: **ce≡(e₁∧e₂)**, let $\mathbf{e_1}$ and $\mathbf{e_2}$ are **EBexp**, Suppose P($\mathbf{e_1}$) and P($\mathbf{e_2}$) are true.

By the definition: $(\sigma, b) \in \mathcal{E}[\![e_1 \wedge e_2]\!] \Leftrightarrow \sigma \in \Sigma$ $and\ \exists b_0, b_1 . b = b_0 \wedge_T b_1\ and\ (\sigma, b_0) \in \mathcal{E}[\![e_1]\!]\ and\ (\sigma, b_1) \in \mathcal{E}[\![e_2]\!]$.

Thus, suppose $(\sigma, b) \in \mathcal{E}[\![e_1 \wedge e_2]\!]$, then $\exists b_0, b_1, (\sigma, b_0) \in \mathcal{E}[\![e_1]\!]$ and $(\sigma, b_1) \in \mathcal{E}[\![e_2]\!]$.

By the suppose P($\mathbf{e_1}$) and P($\mathbf{e_2}$) are true, then $< e_1, \sigma > \to b_0$ and $< e_2, \sigma > \to b_1$.

Hence, we can derive $< e_1 \wedge e_2, \sigma > \to b, b = b_0 \wedge_T b_1$.

Conversely, every derivation of $< e_1 \wedge e_2, \sigma > \to b$ must have the follows:

$$\dfrac{\dfrac{\vdots}{< e_1, \sigma > \to b_0} \quad \dfrac{\vdots}{< e_2, \sigma > \to b_1}}{< e_1 \wedge e_2, \sigma > \to b}$$

For a $\mathbf{e_1}$ and $\mathbf{e_2}$, we can derive $b = b_0 \wedge_T b_1$.

Because the P($\mathbf{e_1}$) and P($\mathbf{e_2}$) are true, $(\sigma, b_0) \in \mathcal{E}[\![e_1]\!]\ and\ (\sigma, b_1) \in \mathcal{E}[\![e_2]\!]$

Hence, $(\sigma, b) \in \mathcal{E}[\![ce]\!]$.

The proofs of other cases are completely analogous.

We finish the proof of this theorem.

Up to date, we have finished the proof of equivalence between the operational semantics and the denotational semantics of STeCEQL.

5 Conclusion and outlook

In this paper, focusing on the correctness of the operational semantics of the EQL STeCEQL, we give the denotational semantics of it and prove the equivalence of two semantics of STeCEQL by structural inductive method. From the view of formal semantics of computer language, the equivalence of the operational semantics and the denotational semantics show the correctness of its operational semantics.

Since the internet of vehicles is a typical real-time distributed mobile networked system, we will study the processing algorithm of the STeCEQL in next steps

Acknowledgments

This work is supported by the National Basic Research Program of China (No. 2011CB302802), the National Natural Science Foundation of China (No.61370100 and No.61021004) and Shanghai Knowledge Service Platform Project (No. ZF1213).

References

[1] Atzori L, Iera A, Morabito G 2010 The internet of things: A survey *Computer networks* **54**(15) 2787-805

[2] Chen Y 2012 Stec: A location-triggered specification language for real-time systems *Object/Component/Service-Oriented Real-Time Distributed Computing Workshops (ISORCW), 15th IEEE International Symposium on, IEEE* pp 1-6

[3] Wu H, Chen Y, Zhang M 2013 On Denotational Semantics of Spatial-Temporal Consistency Language-SteC *Theoretical Aspects of Software Engineering (TASE), 2013 International Symposium on, IEEE* 113-20

[4] Schwiderski-Grosche S, Moody K 2009 The SpaTeC composite event language for spatio-temporal reasoning in mobile systems *Proceedings of the Third ACM International Conference on Distributed Event-Based Systems, ACM* p 11

[5] Moody K, Bacon J, Evans D 2010 *Implementing a practical spatio-temporal composite event language, From active data management to event-based systems and more* Springer Berlin Heidelberg, pp 108-23

[6] Jin B, Zhuo W, Hu J 2013 Specifying and detecting spatio-temporal events in the internet of things *Decision Support Systems* **55**(1) 256-69

[7] Zhu D, Sethi A S 2001 SEL, a new event pattern specification language for event correlation *Computer Communications and Networks, Proceedings. Tenth International Conference on IEEE* p 586-9

[8] Seiriö M, Berndtsson M 2005 *Design and implementation of an ECA rule markup language, Rules and rule markup languages for the semantic web* Springer Berlin Heidelberg, p 98-112

[9] Wu E, Diao Y, Rizvi S 2006 High-performance complex event processing over streams *Proceedings of the 2006 ACM SIGMOD international conference on Management of data. ACM* p 407-18

[10] Demers A J, Gehrke J, Panda B 2007 Cayuga: A General Purpose Event Monitoring System *CIDR* p 412-22

[11] Eckert M 2008 *Complex event processing with XChange EQ: language design, formal semantics, and incremental evaluation for querying events* LMU München: Faculty of Mathematics, München

[12] Anicic D, Rudolph S, Fodor P 2012 Stream reasoning and complex event processing in etalis *Semantic Web* **3**(4) 397-407

Authors

Huiyong Li, born on February 2, 1980, Taiyuan, China

Current position, grades: PhD student of software engineering institute, East China Normal University.
University studies: M.Sc. in Computer Sciences (2011) from Taiyuan University of Science and Technology.
Scientific interests: different aspects of Internet of Things and Real-time Distributed Systems.

Yixiang Chen, born on March 12, 1961, Xuzhou, China

Current position, grades: Chair of the MoE Key Research Center for Software/Hardware Co-design Engineering, East China Normal University.
University studies: PhD in Mathematics (1995) from Sichuan University.
Scientific interests: different aspects of Real-time System, Formal Models, Formal Semantics of Programming, Foundation of Computations and Trustworthy Network.