

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:

$attribute ::= true \mid false \mid x_a = a \mid x_a \neq a$
 $\mid x_a > a \mid x_a \geq a \mid x_a < a \mid x_a \leq a$
 $\mid attribute_0 \wedge attribute_1 \mid attribute_0 \vee attribute_1$

TBexp:

$time ::= true \mid false \mid x_t \text{ BEFORE } t \mid x_t \text{ AFTER } t$
 $\mid x_t \text{ EQUAL } t \mid x_t \text{ OVERLAP } t \mid x_t \text{ DURING } t$
 $\mid time_0 \vee time_1 \mid time_0 \wedge time_1$

LBexp:

$location ::= true \mid false \mid x_l \text{ EQ } l \mid x_l \text{ OP } l \mid x_l \text{ IN } l$
 $\mid x_l \text{ NORTH } l \mid x_l \text{ SOUTH } l \mid x_l \text{ EAST } l \mid x_l \text{ WEST } l$
 $\mid x_l \text{ NORTHWEST } l \mid x_l \text{ NORTHEAST } l$
 $\mid x_l \text{ SOUTHWEST } l \mid x_l \text{ SOUTHEAST } l$
 $\mid location_0 \vee location_1 \mid location_0 \wedge location_1$

DBexp:

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

EBexp:

$e ::= agent^{time}(attribute1; attribute2; attribute3 \dots)$
 $\mid agent^{location}(attribute1; attribute2; attribute3 \dots)$
 $\mid agent^{time}_{location}(attribute1; attribute2; attribute3 \dots)$
 $\mid agent^{time}_{(location, direction)}(attribute1; attribute2; \dots)$

CEexp:

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

The operational semantics of the STeCEQL is as follows:

ABexp:

$\langle true, \sigma \rangle \rightarrow true$
 $\langle false, \sigma \rangle \rightarrow false$
 $\langle x_a = a, \sigma \rangle \rightarrow true, \text{ if } \sigma(x_a) = a$
 $\langle x_a = a, \sigma \rangle \rightarrow false, \text{ if } \sigma(x_a) \neq a$
 $\langle x_a \neq a, \sigma \rangle \rightarrow true, \text{ if } \sigma(x_a) \neq a$
 $\langle x_a \neq a, \sigma \rangle \rightarrow false, \text{ if } \sigma(x_a) = a$
 $\langle x_a > a, \sigma \rangle \rightarrow true, \text{ if } \sigma(x_a) > a$
 $\langle x_a > a, \sigma \rangle \rightarrow false, \text{ if } \sigma(x_a) \leq a$
 $\langle x_a \geq a, \sigma \rangle \rightarrow true, \text{ if } \sigma(x_a) \geq a$
 $\langle x_a \geq a, \sigma \rangle \rightarrow false, \text{ if } \sigma(x_a) < a$

$\langle x_a < a, \sigma \rangle \rightarrow true, \text{ if } \sigma(x_a) < a$
 $\langle x_a < a, \sigma \rangle \rightarrow false, \text{ if } \sigma(x_a) \geq a$
 $\langle x_a \leq a, \sigma \rangle \rightarrow true, \text{ if } \sigma(x_a) \leq a$
 $\langle x_a \leq a, \sigma \rangle \rightarrow false, \text{ if } \sigma(x_a) > a$
 $\frac{\langle attribute_0, \sigma \rangle \rightarrow b_0 \quad \langle attribute_1, \sigma \rangle \rightarrow b_1}{\langle attribute_0 \wedge attribute_1, \sigma \rangle \rightarrow b}$,

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

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

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

TBexp:

$\langle true, \sigma \rangle \rightarrow true$
 $\langle false, \sigma \rangle \rightarrow false$
 $\langle x_t \text{ BEFORE } t, \sigma \rangle \rightarrow true, \text{ if } \sigma(x_t).endn < t.start1$
 $\langle x_t \text{ BEFORE } t, \sigma \rangle \rightarrow false, \text{ if } \sigma(x_t).endn \geq t.start1$
 $\langle x_t \text{ AFTER } t, \sigma \rangle \rightarrow true, \text{ if } \sigma(x_t).start1 > t.endn$
 $\langle x_t \text{ AFTER } t, \sigma \rangle \rightarrow false, \text{ if } \sigma(x_t).start1 \leq t.endn$
 $\langle x_t \text{ EQUAL } t, \sigma \rangle \rightarrow true,$
 if $(\forall i \in N. \sigma(x_t).starti = t.starti \text{ and } \sigma(x_t).endi = t.endi)$
 $\langle x_t \text{ EQUAL } t, \sigma \rangle \rightarrow false,$
 if $(\exists i \in N. \sigma(x_t).starti \neq t.starti \text{ and } \sigma(x_t).endi \neq t.endi)$
 $\langle x_t \text{ OVERLAP } t, \sigma \rangle \rightarrow true,$
 if $(\sigma(x_t).endn \geq t.start1 \text{ and } \sigma(x_t).endn \leq t.endn)$
 or $(\sigma(x_t).start1 \geq t.start1 \text{ and } \sigma(x_t).start1 \leq t.endn)$
 $\langle x_t \text{ OVERLAP } t, \sigma \rangle \rightarrow false,$
 if $\sigma(x_t).endn < t.start1$ or $\sigma(x_t).start1 > t.endn$
 $\langle x_t \text{ DURING } t, \sigma \rangle \rightarrow true,$
 if $\sigma(x_t).start1 \geq t.start1$ and $\sigma(x_t).end1 \leq t.endn$
 $\langle x_t \text{ DURING } t, \sigma \rangle \rightarrow false,$
 if $\sigma(x_t).start1 < t.start1$ or $\sigma(x_t).end1 > t.endn$
 $\frac{\langle time, \sigma \rangle \rightarrow true}{\langle \neg time, \sigma \rangle \rightarrow false}$

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

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

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

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

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

Lexp:

$$\langle true, \sigma \rangle \rightarrow true$$

$$\langle false, \sigma \rangle \rightarrow false$$

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

if $(\forall i \in N. \sigma(x_l).row_i = l.row_i \text{ and } \sigma(x_l).column_i = l.column_i)$

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

if $(\exists i \in N. \sigma(x_l).row_i \neq l.row_i \text{ or } \sigma(x_l).column_i \neq l.column_i)$

$$\langle x_l OPl, \sigma \rangle \rightarrow true,$$

if $(\exists i, j \in N. \sigma(x_l).row_i = l.row_j \text{ and } \sigma(x_l).column_i = l.column_j)$

$$\langle x_l OPl, \sigma \rangle \rightarrow false,$$

if $(\forall i, j \in N. \sigma(x_l).row_i = l.row_j \text{ and } \forall \sigma(x_l).column_i = l.column_j)$

$$\langle x_l IN l, \sigma \rangle \rightarrow true, \text{ if } \sigma(x_l) \subset l$$

$$\langle x_l IN l, \sigma \rangle \rightarrow false, \text{ if } \sigma(x_l) \not\subset l$$

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

if $(\forall i, j \in N. \sigma(x_l).row_i < l.row_j \text{ and } \sigma(x_l).column_j = l.column_j)$

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

if $(\exists i, j \in N. \sigma(x_l).row_i \geq l.row_j \text{ or } \sigma(x_l).column_j \neq l.column_j)$

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

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

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

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

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

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

DBexp:

$$\langle true, \sigma \rangle \rightarrow true$$

$$\langle false, \sigma \rangle \rightarrow false$$

$$\langle x_d = d1, \sigma \rangle \rightarrow true, \text{ if } \sigma(x_d) = d1$$

$$\langle x_d = d1, \sigma \rangle \rightarrow false, \text{ if } \sigma(x_d) \neq d1$$

$$\langle x_d \neq d1, \sigma \rangle \rightarrow true, \text{ if } \sigma(x_d) \neq d1$$

$$\langle x_d \neq d1, \sigma \rangle \rightarrow false, \text{ if } \sigma(x_d) = d1$$

EBexp:

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

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

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

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

$$\frac{\langle t, \sigma \rangle \rightarrow b1 \quad \langle l, \sigma \rangle \rightarrow b2 \quad \langle a1, \sigma \rangle \rightarrow b3 \quad \langle a2, \sigma \rangle \rightarrow b4 \quad \dots}{\langle agent_{location}^{time}(attribute1; attribute2; \dots), \sigma \rangle \rightarrow true},$$

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

$$\frac{\langle t, \sigma \rangle \rightarrow b1 \quad \langle l, \sigma \rangle \rightarrow b2 \quad \langle a1, \sigma \rangle \rightarrow b3 \quad \langle a2, \sigma \rangle \rightarrow b4 \quad \dots}{\langle agent_{location}^{time}(attribute1; attribute2; \dots), \sigma \rangle \rightarrow false},$$

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

$$\frac{\langle t, \sigma \rangle \rightarrow b1 \quad \langle l, \sigma \rangle \rightarrow b2 \quad \langle d, \sigma \rangle \rightarrow b3 \quad \langle a1, \sigma \rangle \rightarrow b4 \quad \langle a2, \sigma \rangle \rightarrow b5 \quad \dots}{\langle agent_{(l,d)}^{time}(attribute1; attribute2; \dots), \sigma \rangle \rightarrow true},$$

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

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

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

CEBexp:

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

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

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

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

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 $\langle \text{attribute}, \sigma \rangle \rightarrow \text{true}$ 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 $\mathbf{B}=\{\text{true}, \text{false}\}$ 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 \mathbf{B} .

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

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

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

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

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

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

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

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

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

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

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

ABexp:

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

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

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

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

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

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

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

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

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

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

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

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

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

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

$$\mathcal{A}[\![\text{attribute}_0 \wedge \text{attribute}_1]\!] = \{(\sigma, b_0 \wedge_T b_1) \mid \sigma \in \Sigma$$

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

$$\mathcal{A}[\![\text{attribute}_0 \vee \text{attribute}_1]\!] = \{(\sigma, b_0 \vee_T b_1) \mid \sigma \in \Sigma$$

TBexp:

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

$$\mathcal{T}[\![\text{time}_0 \wedge \text{time}_1]\!] = \{(\sigma, b_0 \wedge_T b_1) \mid \sigma \in \Sigma$$

$$\text{and } (\sigma, b_0) \in \mathcal{A}[\![\text{time}_0]\!] \text{ and } (\sigma, b_1) \in \mathcal{A}[\![\text{time}_1]\!]\}$$

$$\mathcal{T}[\![\text{time}_0 \vee \text{time}_1]\!] = \{(\sigma, b_0 \vee_T b_1) \mid \sigma \in \Sigma$$

$$\text{and } (\sigma, b_0) \in \mathcal{A}[\![\text{time}_0]\!] \text{ and } (\sigma, b_1) \in \mathcal{A}[\![\text{time}_1]\!]\}$$

LBexp:

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

DBexp:

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

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

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

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

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

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

EBexp:

$\mathcal{E}[\![agent^{time}(attribute_1; attribute_2; attribute_3; \dots)]\!]$

$= \{(\sigma, b_1 \wedge_T b_2 \wedge_T b_3 \wedge_T \dots) \mid \sigma \in \Sigma$

and $(\sigma, b_1) \in \mathcal{T}[\![t]\!]\}$ and $(\sigma, b_2) \in \mathcal{A}[\![a_1]\!]\}$

and $(\sigma, b_3) \in \mathcal{A}[\![a_2]\!]\}$ and $(\sigma, b_4) \in \mathcal{A}[\![a_3]\!]\}$

$\mathcal{E}[\![agent_{location}(attribute_1; attribute_2; attribute_3; \dots)]\!]$

$= \{(\sigma, b_1 \wedge_T b_2 \wedge_T b_3 \wedge_T \dots) \mid \sigma \in \Sigma$

and $(\sigma, b_1) \in \mathcal{L}[\![location]\!]\}$ and $(\sigma, b_2) \in \mathcal{A}[\![a_1]\!]\}$

and $(\sigma, b_3) \in \mathcal{A}[\![a_2]\!]\}$ and $(\sigma, b_4) \in \mathcal{A}[\![a_3]\!]\}$

$\mathcal{E}[\![agent_{location}^{time}(attribute_1; attribute_2; attribute_3; \dots)]\!]$

$= \{(\sigma, b_1 \wedge_T b_2 \wedge_T b_3 \wedge_T \dots) \mid \sigma \in \Sigma$

and $(\sigma, b_1) \in \mathcal{T}[\![t]\!]\}$ and $(\sigma, b_2) \in \mathcal{L}[\![location]\!]\}$

and $(\sigma, b_3) \in \mathcal{A}[\![a_1]\!]\}$ and $(\sigma, b_4) \in \mathcal{A}[\![a_2]\!]\}$ and $(\sigma, b_5) \in \mathcal{A}[\![a_3]\!]\}$

$\mathcal{E}[\![agent^{time}_{(location, direction)}(attribute_1; attribute_2; \dots)]\!]$

$= \{(\sigma, b_1 \wedge_T b_2 \wedge_T b_3 \wedge_T \dots) \mid \sigma \in \Sigma$

and $(\sigma, b_1) \in \mathcal{T}[\![t]\!]\}$ and $(\sigma, b_2) \in \mathcal{L}[\![l]\!]\}$ and $(\sigma, b_3) \in \mathcal{D}[\![d]\!]\}$

and $(\sigma, b_4) \in \mathcal{A}[\![a_1]\!]\}$ and $(\sigma, b_5) \in \mathcal{A}[\![a_2]\!]\}$ and $(\sigma, b_6) \in \mathcal{A}[\![a_3]\!]\}$

CEBexp:

$\mathcal{E}[\![e_1 \wedge e_2]\!] = \{(\sigma, b_1 \wedge_T b_2) \mid \sigma \in \Sigma$

and $(\sigma, b_1) \in \mathcal{E}[\![e_1]\!]\}$ and $(\sigma, b_2) \in \mathcal{E}[\![e_2]\!]\}$

$\mathcal{E}[\![e_1 \vee e_2]\!] = \{(\sigma, b_1 \vee_T b_2) \mid \sigma \in \Sigma$

and $(\sigma, b_1) \in \mathcal{E}[\![e_1]\!]\}$ and $(\sigma, b_2) \in \mathcal{E}[\![e_2]\!]\}$

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 $attribute \in \mathbf{ABexp}$, we have $\mathcal{A}[\![attribute]\!] = \{(\sigma, b) \mid \langle attribute, \sigma \rangle \rightarrow b\}$.

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

$P(attribute) \Leftrightarrow_{def}$

$\mathcal{A}[\![attribute]\!] = \{(\sigma, b) \mid \langle attribute, \sigma \rangle \rightarrow b\}$

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 $b \equiv true$ and $\langle true, \sigma \rangle \rightarrow true$.

Conversely, if $\langle true, \sigma \rangle \rightarrow true$, then the only possible derive is $b \equiv true$, thus $(\sigma, b) \in \mathcal{A}[\![true]\!]$.

The case: $attribute \equiv (x_a = a)$, x_a is the storage unit.

By the definition:

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

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

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 $\langle x_a = a, \sigma \rangle \rightarrow true$.

Conversely, suppose $\langle x_a = a, \sigma \rangle \rightarrow true$, then there must be a derivation as below:

$$\frac{\sigma(x_a) = a}{\langle x_a = a, \sigma \rangle \rightarrow true}$$

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

Hence, $(\sigma, true) \in \mathcal{A}[\![true]\!] \Leftrightarrow \langle x_a = a, \sigma \rangle \rightarrow true$

Similarly,

$(\sigma, false) \in \mathcal{A}[\![true]\!] \Leftrightarrow \langle x_a = a, \sigma \rangle \rightarrow false$.

Thus, we can get:

$\mathcal{A}[\![x_a = a]\!] \Leftrightarrow \{(\sigma, b) \mid \langle x_a = a, \sigma \rangle \rightarrow b\}$

The case: $attribute \equiv (attribute_0 \wedge attribute_1)$, let $attribute_0$ and $attribute_1$ are \mathbf{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

$\langle \text{attribute}_0, \sigma \rangle \rightarrow b_0$ and $\langle \text{attribute}_1, \sigma \rangle \rightarrow b_1$
Hence, we can derive
 $\langle \text{attribute}_0 \wedge \text{attribute}_1, \sigma \rangle \rightarrow b, b = b_0 \wedge_T b_1$.
Conversely, every derivation of
 $\langle \text{attribute}_0 \wedge \text{attribute}_1, \sigma \rangle \rightarrow b$ must have the follows:

$$\frac{\begin{array}{c} \vdots \\ \langle \text{attribute}_0, \sigma \rangle \rightarrow b_0 \end{array} \quad \frac{\begin{array}{c} \vdots \\ \langle \text{attribute}_1, \sigma \rangle \rightarrow b_1 \end{array}}{\langle \text{attribute}_1, \sigma \rangle \rightarrow b_1}}{\langle \text{attribute}_0 \wedge \text{attribute}_1, \sigma \rangle \rightarrow b}$$

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

Because the $P(\text{attribute}_0)$ and $P(\text{attribute}_1)$ are true,
 $(\sigma, b_0) \in \mathcal{A}[\text{attribute}_0]$ and $(\sigma, b_1) \in \mathcal{A}[\text{attribute}_1]$.

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

The proofs of other cases are completely analogous.
We finish the proof of this theorem.

Theorem 2: For every expression $\text{time} \in \mathbf{TBexp}$, we have

$$\mathcal{T}[\text{time}] = \{(\sigma, b) \mid \langle \text{time}, \sigma \rangle \rightarrow b\}$$

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

$$P(\text{time}) \Leftrightarrow_{\text{def}} \mathcal{T}[\text{time}] = \{(\sigma, b) \mid \langle \text{time}, \sigma \rangle \rightarrow b\}$$

The case: $\text{time} \equiv \text{true}$

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

Obviously, if $(\sigma, b) \in \mathcal{T}[\text{true}]$ then
 $b \equiv \text{true}$ and $\langle \text{true}, \sigma \rangle \rightarrow \text{true}$.

Conversely, if $\langle \text{true}, \sigma \rangle \rightarrow \text{true}$, then the only possible derive is $b \equiv \text{true}$, thus $(\sigma, b) \in \mathcal{T}[\text{true}]$.

The case: $\text{time} \equiv (x_t \text{ BEFORE } t)$, x_t is the storage unit.

By the definition:

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

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

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

Then

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

$$\Leftrightarrow \sigma \in \Sigma \text{ and } \sigma(x_t).endn < t.start1.$$

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

By the operational semantics of the expression, we get
 $\langle x_t \text{ BEFORE } t, \sigma \rangle \rightarrow \text{true}$.

Conversely, suppose $\langle x_t \text{ BEFORE } t, \sigma \rangle \rightarrow \text{true}$,
then there must be a derivation as below:

$$\frac{\sigma(x_t).endn < t.start1}{\langle x_t \text{ BEFORE } t, \sigma \rangle \rightarrow \text{true}}$$

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

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

$$\Leftrightarrow \langle x_t \text{ BEFORE } t, \sigma \rangle \rightarrow \text{true}.$$

Similarly,

$$(\sigma, \text{false}) \in \mathcal{T}[x_t \text{ BEFORE } t] \Leftrightarrow \langle x_t = a, \sigma \rangle \rightarrow \text{false}.$$

Thus we can get:

$$\mathcal{T}[x_t \text{ BEFORE } t] \Leftrightarrow \{(\sigma, b) \mid \langle x_t \text{ BEFORE } t, \sigma \rangle \rightarrow b\}$$

The case: $\text{time} \equiv (\text{time}_0 \wedge \text{time}_1)$, let time_0 and time_1 are \mathbf{TBexp} .

Suppose $P(\text{time}_0)$ and $P(\text{time}_1)$ are true.

By the definition:

$$(\sigma, b) \in \mathcal{T}[\text{time}_0 \wedge \text{time}_1] \Leftrightarrow \sigma \in \Sigma$$

$$\text{and } \exists b_0, b_1. b = b_0 \wedge_T b_1$$

$$\text{and } (\sigma, b_0) \in \mathcal{T}[\text{time}_0] \text{ and } (\sigma, b_1) \in \mathcal{T}[\text{time}_1]$$

Thus, suppose $(\sigma, b) \in \mathcal{T}[\text{time}_0 \wedge \text{time}_1]$, then

$$(\sigma, b_0) \in \mathcal{T}[\text{time}_0] \text{ and } (\sigma, b_1) \in \mathcal{T}[\text{time}_1].$$

By the suppose, the $P(\text{time}_0)$ and $P(\text{time}_1)$ are true,
then

$$\langle \text{time}_0, \sigma \rangle \rightarrow b_0 \text{ and } \langle \text{time}_1, \sigma \rangle \rightarrow b_1$$

Hence, we can

derive $\langle \text{time}_0 \wedge \text{time}_1, \sigma \rangle \rightarrow b, b = b_0 \wedge_T b_1$.

Conversely, every derivation of
 $\langle \text{time}_0 \wedge \text{time}_1, \sigma \rangle \rightarrow b$ must have the follows:

$$\frac{\begin{array}{c} \vdots \\ \langle \text{time}_0, \sigma \rangle \rightarrow b_0 \end{array} \quad \frac{\begin{array}{c} \vdots \\ \langle \text{time}_1, \sigma \rangle \rightarrow b_1 \end{array}}{\langle \text{time}_1, \sigma \rangle \rightarrow b_1}}{\langle \text{time}_0 \wedge \text{time}_1, \sigma \rangle \rightarrow b}$$

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

Because the $P(\text{time}_0)$ and $P(\text{time}_1)$ are true,
 $(\sigma, b_0) \in \mathcal{T}[\text{time}_0]$ and $(\sigma, b_1) \in \mathcal{T}[\text{time}_1]$.

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

The proofs of other cases are completely analogous.

We finish the proof of this theorem.

Theorem 3: For every expression $\text{location} \in \mathbf{LBexp}$, we have $\mathcal{L}[\text{location}] = \{(\sigma, b) \mid \langle \text{location}, \sigma \rangle \rightarrow b\}$

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

$$P(\text{location}) \Leftrightarrow_{\text{def}}$$

$$\mathcal{L}[\text{location}] = \{(\sigma, b) \mid \langle \text{location}, \sigma \rangle \rightarrow b\}$$

The case: $\text{location} \equiv \text{true}$.

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

Obviously, if $(\sigma, b) \in \mathcal{L}[\text{true}]$, then
 $b \equiv \text{true}$ and $\langle \text{true}, \sigma \rangle \rightarrow \text{true}$.

Conversely, if $\langle \text{true}, \sigma \rangle \rightarrow \text{true}$, then the only possible derive is $b \equiv \text{true}$, thus $(\sigma, b) \in \mathcal{L}[\text{true}]$.

The case: $\text{location} \equiv (x_t \text{ EQ } l)$, x_t is the storage unit.

By the definition:

$$\mathcal{L}[\llbracket x_l EQl \rrbracket] = \{(\sigma, true) \mid \sigma \in \Sigma\}$$

and $\forall i \in N. \sigma(x_l).row_i = l.row_i$ and $\sigma(x_l).column_i = l.end_i$

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

and $(\exists i \in N. \sigma(x_l).ri \neq l.ri$ or $\sigma(x_l).ci \neq l.ci)$

$$\text{Then } (\sigma, true) \in \mathcal{L}[\llbracket x_l EQl \rrbracket] \Leftrightarrow$$

$\sigma \in \Sigma$ and $\forall i \in N. \sigma(x_l).row_i = l.row_i$ and $\sigma(x_l).column_i = l.end_i$.

If $(\sigma, true) \in \mathcal{L}[\llbracket x_l EQl \rrbracket]$, then

$\forall i \in N. \sigma(x_l).row_i = l.row_i$ and $\sigma(x_l).column_i = l.end_i$.

By the operational semantics of the expression, we get $\langle x_l EQl, \sigma \rangle \rightarrow true$.

Conversely, suppose $\langle x_l EQl, \sigma \rangle \rightarrow true$, then there must be a derivation as below:

$$\frac{\forall i \in N. \sigma(x_l).row_i = l.row_i \text{ and } \sigma(x_l).column_i = l.end_i}{\langle x_l EQl, \sigma \rangle \rightarrow true}$$

Thus, $(\sigma, true) \in \mathcal{L}[\llbracket x_l EQl \rrbracket]$.

Hence, $(\sigma, true) \in \mathcal{L}[\llbracket x_l EQl \rrbracket] \Leftrightarrow \langle x_l EQl, \sigma \rangle \rightarrow true$.

Similarly,

$(\sigma, false) \in \mathcal{L}[\llbracket x_l EQl \rrbracket] \Leftrightarrow \langle x_l EQl, \sigma \rangle \rightarrow false$.

Thus we can get:

$$\mathcal{L}[\llbracket x_l EQl \rrbracket] \Leftrightarrow \{(\sigma, b) \mid \langle x_l EQl, \sigma \rangle \rightarrow b\}$$

The case: $location \equiv (location_0 \wedge location_1)$, let $location_0$ and $location_1$ are LBexp.

Suppose $P(location_0)$ and $P(location_1)$ are true.

By the definition:

$$(\sigma, b) \in \mathcal{L}[\llbracket location_0 \wedge location_1 \rrbracket] \Leftrightarrow \sigma \in \Sigma$$

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

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

By suppose, the $P(location_0)$ and $P(location_1)$ are true, then

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

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

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

$$\frac{\begin{array}{c} \vdots \\ \langle location_0, \sigma \rangle \rightarrow b_0 \end{array} \quad \frac{\begin{array}{c} \vdots \\ \langle location_1, \sigma \rangle \rightarrow b_1 \end{array}}{\langle location_1, \sigma \rangle \rightarrow b_1}}{\langle location_0 \wedge location_1, \sigma \rangle \rightarrow b}$$

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

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

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

The proofs of other cases are completely analogous.

We finish the proof of this theorem.

Theorem 4: For every expression $direction \in \mathbf{LBexp}$, we have $\mathcal{D}[\llbracket direction \rrbracket] = \{(\sigma, b) \mid \langle direction, \sigma \rangle \rightarrow b\}$

Proof. We prove the theorem by structural induction.

We have that $P(direction) \Leftrightarrow_{def}$

$$\mathcal{D}[\llbracket direction \rrbracket] = \{(\sigma, b) \mid \langle direction, \sigma \rangle \rightarrow b\}$$

The case: $direction \equiv true$.

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

Obviously, if $(\sigma, b) \in \mathcal{D}[\llbracket true \rrbracket]$, then $b \equiv true$ and $\langle true, \sigma \rangle \rightarrow true$.

Conversely, if $\langle true, \sigma \rangle \rightarrow true$, then the only possible derive is $b \equiv true$, thus $(\sigma, b) \in \mathcal{D}[\llbracket true \rrbracket]$.

The case: $direction \equiv (x_d = d)$, x_d is the storage unit.

By the definition:

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

Then

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

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

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

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

$$\frac{\sigma(x_d) = d}{\langle x_d = d, \sigma \rangle \rightarrow true}$$

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

Hence, $(\sigma, true) \in \mathcal{D}[\llbracket true \rrbracket] \Leftrightarrow \langle x_d = d, \sigma \rangle \rightarrow true$.

Similarly,

$(\sigma, false) \in \mathcal{D}[\llbracket true \rrbracket] \Leftrightarrow \langle x_d = d, \sigma \rangle \rightarrow false$.

Thus, we can get:

$$\mathcal{D}[\llbracket x_d = d \rrbracket] \Leftrightarrow \{(\sigma, b) \mid \langle x_d = d, \sigma \rangle \rightarrow b\}$$

The proofs of other cases are completely analogous.

We finish the proof of this theorem.

Theorem 5: For every expression $e \in \mathbf{EBexp}$, we have $\mathcal{E}[\llbracket e \rrbracket] = \{(\sigma, b) \mid \langle e, \sigma \rangle \rightarrow b\}$

Proof. We prove the theorem by structural induction.

We have that $P(e) \Leftrightarrow_{def} \mathcal{E}[\llbracket e \rrbracket] = \{(\sigma, b) \mid \langle e, \sigma \rangle \rightarrow b\}$.

The

case:

$e \equiv agent^{time}(attribute_1; attribute_2; attribute_3)$, let **time** is **EBexp**, **attributes** are **ABexp**,

Suppose $P(\mathbf{time})$ and $P(\mathbf{attribute})$ s are true.

By the definition:

$$(\sigma, b) \in \mathcal{E}[\llbracket agent^{time}(attribute_1; attribute_2; attribute_3) \rrbracket]$$

$$\Leftrightarrow \sigma \in \Sigma \text{ and } \exists b_1, b_2, b_3, b_4. b = b_1 \wedge_T b_2 \wedge_T b_3 \wedge_T b_4$$

and $(\sigma, b_1) \in \mathcal{T}[\text{time}]$ and $(\sigma, b_2) \in \mathcal{A}[\text{attribute}_1]$
 and $(\sigma, b_3) \in \mathcal{A}[\text{attribute}_2]$ and $(\sigma, b_4) \in \mathcal{A}[\text{attribute}_3]$
 Thus, suppose
 $(\sigma, b) \in \mathcal{E}[\text{agent}^{\text{time}}(\text{attribute}_1; \text{attribute}_2; \text{attribute}_3)]$,
 then $\exists b_1, b_2, b_3, b_4, (\sigma, b_1) \in \mathcal{T}[\text{time}]$ and $(\sigma, b_2) \in \mathcal{A}[\text{a1}]$
 and $(\sigma, b_3) \in \mathcal{A}[\text{a2}]$ and $(\sigma, b_4) \in \mathcal{A}[\text{a3}]$.

By the suppose P(**time**) and P(**attribute**)s are true,
 then $\langle \text{time}, \sigma \rangle \rightarrow b_1$ and $\langle \text{attribute}_1, \sigma \rangle \rightarrow b_2$
 and $\langle \text{attribute}_2, \sigma \rangle \rightarrow b_3$ and $\langle \text{attribute}_3, \sigma \rangle \rightarrow b_4$.

Hence, we can derive
 $\langle \text{agent}^{\text{time}}(\text{attribute}_1; \text{attribute}_2; \text{attribute}_3), \sigma \rangle \rightarrow b$,
 $b = b_1 \wedge_T b_2 \wedge_T b_3 \wedge_T b_4$.

Conversely, every derivation of
 $\langle \text{agent}^{\text{time}}(\text{attribute}_1; \text{attribute}_2; \text{attribute}_3), \sigma \rangle \rightarrow b$
 must have the follows:

$$\frac{\begin{array}{c} \vdots \\ \langle t, \sigma \rangle \rightarrow b_1 \end{array} \quad \begin{array}{c} \vdots \\ \langle a_1, \sigma \rangle \rightarrow b_2 \end{array} \quad \begin{array}{c} \vdots \\ \langle a_2, \sigma \rangle \rightarrow b_3 \end{array}}{\langle \text{agent}_t(a_1, a_2, a_3), \sigma \rangle \rightarrow b}$$

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,
 $(\sigma, b_1) \in \mathcal{T}[\text{time}]$ and $(\sigma, b_2) \in \mathcal{A}[\text{attribute}_1]$
 and $(\sigma, b_3) \in \mathcal{A}[\text{attribute}_2]$ and $(\sigma, b_4) \in \mathcal{A}[\text{attribute}_3]$.

Hence,
 $(\sigma, b) \in \mathcal{E}[\text{agent}^{\text{time}}(\text{attribute}_1; \text{attribute}_2; \text{attribute}_3)]$.

The proofs of other cases are completely analogous.

We finish the proof of this theorem.

Theorem 6: For every expression $\mathbf{ce} \in \mathbf{CEBexp}$, we
 have $\mathcal{E}[\mathbf{ce}] = \{(\sigma, b) \mid \langle \mathbf{ce}, \sigma \rangle \rightarrow b\}$

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

$$P(e) \Leftrightarrow_{\text{def}} \mathcal{E}[\mathbf{ce}] = \{(\sigma, b) \mid \langle \mathbf{ce}, \sigma \rangle \rightarrow b\}.$$

The case: $\mathbf{ce} \equiv (\mathbf{e}_1 \wedge \mathbf{e}_2)$, 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: $(\sigma, b) \in \mathcal{E}[\mathbf{e}_1 \wedge \mathbf{e}_2] \Leftrightarrow \sigma \in \Sigma$
 and $\exists b_0, b_1. b = b_0 \wedge_T b_1$ and $(\sigma, b_0) \in \mathcal{E}[\mathbf{e}_1]$ and $(\sigma, b_1) \in \mathcal{E}[\mathbf{e}_2]$.

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*

Thus, suppose $(\sigma, b) \in \mathcal{E}[\mathbf{e}_1 \wedge \mathbf{e}_2]$, then
 $\exists b_0, b_1, (\sigma, b_0) \in \mathcal{E}[\mathbf{e}_1]$ and $(\sigma, b_1) \in \mathcal{E}[\mathbf{e}_2]$.

By the suppose P(\mathbf{e}_1) and P(\mathbf{e}_2) are true, then
 $\langle \mathbf{e}_1, \sigma \rangle \rightarrow b_0$ and $\langle \mathbf{e}_2, \sigma \rangle \rightarrow b_1$.

Hence, we can derive
 $\langle \mathbf{e}_1 \wedge \mathbf{e}_2, \sigma \rangle \rightarrow b, b = b_0 \wedge_T b_1$.

Conversely, every derivation of $\langle \mathbf{e}_1 \wedge \mathbf{e}_2, \sigma \rangle \rightarrow b$
 must have the follows:

$$\frac{\begin{array}{c} \vdots \\ \langle \mathbf{e}_1, \sigma \rangle \rightarrow b_0 \end{array} \quad \begin{array}{c} \vdots \\ \langle \mathbf{e}_2, \sigma \rangle \rightarrow b_1 \end{array}}{\langle \mathbf{e}_1 \wedge \mathbf{e}_2, \sigma \rangle \rightarrow b}$$

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,
 $(\sigma, b_0) \in \mathcal{E}[\mathbf{e}_1]$ and $(\sigma, b_1) \in \mathcal{E}[\mathbf{e}_2]$

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

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).

*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.