

Norihiro Kamide

# EXTENDED FULL COMPUTATION-TREE LOGICS FOR PARACONSISTENT MODEL CHECKING

**Abstract.** It is known that the full computation-tree logic  $CTL^*$  is an important base logic for model checking. The bisimulation theorem for  $CTL^*$  is known to be useful for abstraction in model checking. In this paper, the bisimulation theorems for two paraconsistent four-valued extensions  $4CTL^*$  and  $4LCTL^*$  of  $CTL^*$  are shown, and a translation from  $4CTL^*$  into  $CTL^*$  is presented. By using  $4CTL^*$  and  $4LCTL^*$ , inconsistency-tolerant and spatio-temporal reasoning can be expressed as a model checking framework.

*Keywords:* bisimulation, full computation-tree logic, location operator, paraconsistent model checking.

## 1. Introduction

### 1.1. Full computation-tree logic and bisimulation

*Model checking* [10, 29, 12] is an automatic technique for verifying whether a system model  $M$ , represented as a *Kripke structure*, satisfies a specification  $\alpha$ , written as a *temporal logic* formula. Temporal logics are modal logics for describing the temporal ordering of events. Two possible views regarding the nature of time induce two types of temporal logics: linear-time temporal logics such as LTL (*linear-time temporal logic*) [28] and branching-time temporal logics such as CTL (*computation-tree logic*) [10]. It is known that



the *full computation-tree logic* CTL\* [17, 16], which is a result of integrating LTL and CTL, is an important base logic for model checking. Since concrete system models tend to be very large, the *state explosion problem* is arised, and *abstraction techniques* are needed for reducing a large concrete model to a small abstract one. As presented in [6, 12], the following *bisimulation theorem* for CTL\*, which is useful for abstraction in model checking, is well-known: If two Kripke structures  $M$  and  $M'$  are bisimulation equivalent, then for every CTL\* formula  $\alpha$ ,  $M$  satisfies  $\alpha$  if and only if  $M'$  satisfies  $\alpha$ . This theorem guarantees that we can use an efficient small abstract structure which is bisimilar to the given concrete large structure. The logic CTL\* and the bisimulation result addressed above are not sufficient for dealing with inconsistency-tolerant and spatio-temporal reasoning more appropriately. The aim of this paper is thus to extend the framework of CTL\* in order to obtain logical foundations for inconsistency-tolerant and spatio-temporal model checking.

## 1.2. Present paper's results

In this paper, two paraconsistent four-valued extensions 4CTL\* and 4LCTL\* of both CTL\* and *Nelson's paraconsistent logic N4 with strong negation* [1] are introduced, and the bisimulation theorems for 4CTL\* and 4LCTL\* are shown. A translation from 4CTL\* into CTL\* is also presented as an extension of *Rautenberg's embedding* [31] for Nelson's logics [26]. By using this translation, the existing model checking algorithms for CTL\* can be applied to 4CTL\*. The logic 4CTL\* (four-valued CTL\*) is obtained from CTL\* by adding a *strong negation operator*  $\sim$ , and the logic 4LCTL\* (locative 4CTL\*) is obtained from 4CTL\* by adding a *location operator*  $[l]$ . By using these logics, inconsistency-tolerant and spatio-temporal reasoning can be expressed as a model checking framework.

## 1.3. Strong negation and paraconsistency

A remarkable feature of 4CTL\* and 4LCTL\* is that these logics have two kinds of negations:  $\sim$  (strong negation) and  $\neg$  (usual classical negation in CTL\*). The strong negation  $\sim$ , which was first introduced in [26], has been studied by many researchers, and also applied to various non-classical logics (see e.g. [24, 21, 33, 34] and the references therein). A reason why  $\sim$  is added to various logics is that it derives the property of *paraconsistency* [27]. Roughly speaking, a consequence relation  $\models$  is called paraconsistent with

respect to a negation connective  $\sim$  if the following condition holds:  $\exists \alpha, \beta$ : formulas,  $\text{not-}[M, s \models (\alpha \wedge \sim \alpha) \rightarrow \beta]$  where  $s$  is a state of a Kripke structure  $M$ . It is known that logical systems with paraconsistency can deal with inconsistency-tolerant and uncertainty reasoning more appropriately. For example, we do not desire the situation that  $(s(x) \wedge \sim s(x)) \rightarrow d(x)$  is satisfied for any symptom  $s$  and disease  $d$  where  $\sim s(x)$  means “a person  $x$  does not have a symptom  $s$ ” and  $d(x)$  means “a person  $x$  suffers from a disease  $d$ ”, because symptoms and diseases are vague and uncertain concepts. For more information on paraconsistency, see e.g. [3, 7, 13]. A promising application of 4CTL\* and 4LCTL\* for paraconsistent reasoning may be model checking for requirements elicitation in software engineering. An application of multi-valued and paraconsistent model-checking for requirements elicitation was first studied in [14]. In requirements elicitation, different stakeholders often hold different views of how a proposed system should behave, resulting in inconsistencies between their descriptions [14].

#### 1.4. Verification and refutation

Another remarkable feature of 4CTL\* and 4LCTL\* is that these logics have two kinds of consequence relations  $\models^+$  (verification) and  $\models^-$  (refutation), which are a new device for model checking. By using these consequence relations, the notion of the ‘verification’ and the ‘refutation or falsification’ can simultaneously be represented. We believe that a model-checker based on such consequence relations can appropriately combine over- and under-approximating abstractions. In software model-checking, typical model-checkers are used for refutation as well as verification, because of their high bug-finding abilities. A software model-checker Yasm [19, 20] which is based on *Belnap’s 4-valued logic* is the first approach to combine verification and refutation based on the abstraction technique CEGAR [11]. Since Belnap’s 4-valued logic is regarded as a sublogic of Nelson’s logic N4 [1], 4CTL\* and 4LCTL\* may obtain logical justification of such combining approaches.

#### 1.5. Location operator

A peculiarity of the locative version 4LCTL\* is that it has the location operator  $[l]$  which indicates the location of propositions, e.g. a formula  $[l]\alpha$  means “proposition  $\alpha$  holds at location  $l$ .” This operator is the same setting as in [22, 23], which is regarded as a refinement of the original proposal of Kobayashi et al. [25]. In [25], such a location operator was introduced



using a structural congruence relation in formalizing a *distributed concurrent linear logic programming language*. In [22, 23], the framework of this original operator was improved in order to obtain pure logical foundations without any structural congruence relation. Assuming a space domain  $\text{Loc}$  and the operator  $[l]$  with  $l \in \text{Loc}$ , a consequence relation  $(s, l) \models^+ \alpha$  of  $4\text{LCTL}^*$  can be interpreted as “proposition  $\alpha$  holds at time (or state)  $s$  and location  $l$ ”. Various kinds of spatio-temporal situations can thus be expressed using  $4\text{LCTL}^*$ . For example, a linevenss property: ‘If we input the login-password of the host computer  $\text{Comp3}$  at one of the mobile computers  $\text{Comp1}$  and  $\text{Comp2}$ , then we will eventually be able to login  $\text{Comp3}$ ’ can directly be expressed as

$$G([\text{comp1}]password \vee [\text{comp2}]password \rightarrow F[\text{comp3}]login)$$

where the space domain  $\text{Loc}$  is  $\{\text{comp1}, \text{comp2}, \text{comp3}\}$ ,  $G$  represents ‘any-time in the future’, and  $F$  represents ‘eventually in the future’.<sup>1</sup>

### 1.6. Paraconsistency-centric approaches

It is remarked that foundations and applications of multi-valued model checking have been studied by many researchers (see e.g. [5, 8, 9, 14] and the references therein). The present paper’s result is also an example of such studies. Paraconsistency-centric approaches to multi-valued model checking were studied in [14, 15]. In [14], *multi-valued belief exploration logic framework* for merging and reasoning about inconsistent viewpoints were presented. In [15], an automated tool for paraconsistent reasoning were introduced using a *multi-valued model checking framework*. Both frameworks were based on the *multi-valued computation-tree logic*  $\chi\text{CTL}$  with the algebraic structures called *quasi-Boolean logics*. Kripke structures for these frameworks were based on a multi-valued transition relation and a multi-valued valuation (labeling) function. The multi-valued valuation function was a very general setting because it can express  $n$ -valued truth values for any natural number  $n$ . The present paper’s framework is regarded as a special case of these multi-valued frameworks, since the two valuation functions presented in this paper, which are inductively extended to  $\models^-$  and  $\models^+$ , can be transformed into a four-valued valuation function.

---

<sup>1</sup>Strictly speaking, in a  $\text{CTL}^*$  based framework,  $G$  and  $F$  must be replaced respectively by  $AG$  and  $AF$  where  $A$  is the universal path quantifier.

## 1.7. Organization of this paper

The contents of this paper are then summarized as follows. In Section 2, the logic 4CTL\* is introduced as a Kripke structure with  $\models^+$  and  $\models^-$ , and the bisimulation theorem for 4CTL\* is shown using the method presented in [12]. In Section 3, a translation from 4CTL\* into CTL\*, which is an extension of Rautenberg's embedding, is presented using an alternative (intermediate) formulation of 4CTL\* with a single consequence relation  $\models$ . In Section 4, the logic 4LCTL\* is introduced as a locative Kripke structure with a space domain Loc, and the bisimulation theorem for 4LCTL\* is shown. In Section 5, some illustrative examples such as bipolar preference modeling are presented based on 4CTL\* and 4LCTL\*. In Section 6, this paper is concluded with some remarks.

## 2. 4CTL\* and bisimulation

### 2.1. 4CTL\*

An expression ATOM means the set of atomic formulas. In the following explanation, we will follow [12]. *Formulas* of 4CTL\* are defined by combining two types of formulas: *state formulas* and *path formulas*.

DEFINITION 2.1. State formulas are obtained by the following rules.

1. If  $p \in \text{ATOM}$ , then  $p$  is a state formula.
2. If  $\alpha_1$  and  $\alpha_2$  are state formulas, then  $\sim \alpha_1$ ,  $\neg \alpha_1$ ,  $\alpha_1 \wedge \alpha_2$ ,  $\alpha_1 \vee \alpha_2$  and  $\alpha_1 \rightarrow \alpha_2$  are state formulas.
3. If  $\alpha$  is a path formula, then  $E\alpha$  and  $A\alpha$  are state formulas.

Path formulas are obtained by the following rules.

1. If  $\beta$  is a state formula, then  $\text{path}(\beta)$  is a path formula where  $\text{path}$  is an auxiliary function from the set of state formulas to the set of path formulas.<sup>2</sup>
2. If  $\beta_1$  and  $\beta_2$  are path formulas, then  $\sim \beta_1$ ,  $\neg \beta_1$ ,  $\beta_1 \wedge \beta_2$ ,  $\beta_1 \vee \beta_2$ ,  $\beta_1 \rightarrow \beta_2$ ,  $X\beta_1$ ,  $F\beta_1$ ,  $G\beta_1$ ,  $\beta_1 U \beta_2$  and  $\beta_1 R \beta_2$  are path formulas.

State formulas and path formulas are 4CTL\* formulas.

---

<sup>2</sup>Since the function 'path' is usually omitted [12], this rule is interpreted as follows: A state formula is a path formula.



In this definition, E (“some computation path”) and A (“all computation paths”) are called the “path quantifiers”, and X (“next”), F (“eventually”), G (“always”), U (“until”) and R (“release”) are called the “temporal operators”.

In the following, the logic 4CTL\* is defined as a Kripke structure with two consequence relations  $\models^+$  and  $\models^-$ .

**DEFINITION 2.2.** A *Kripke structure* is a structure  $\langle S, S_0, R, L^+, L^- \rangle$  such that

$S$  is the set of states,

$S_0$  is a set of initial states and  $S_0 \subseteq S$ ,

$R$  is a binary relation on  $S$  which satisfies the condition:  $\forall s \in S \exists s' \in S [(s, s') \in R]$ , and

$L^+$  and  $L^-$  are functions from  $S$  to the power set of  $AT \subseteq ATOM$ .

**DEFINITION 2.3.** A *path* in a Kripke structure is an infinite sequence of states,  $\pi = s_0, s_1, s_2, \dots$  such that  $\forall i \geq 0 [(s_i, s_{i+1}) \in R]$ . An expression  $\pi^i$  means the suffix of  $\pi$  starting at  $s_i$ .

**DEFINITION 2.4.** Let  $AT$  be a nonempty subset of  $ATOM$ . Let  $\alpha_1$  and  $\alpha_2$  be state formulas and  $\beta_1$  and  $\beta_2$  be path formulas. *Consequence relations*  $\models^+$  and  $\models^-$  on a Kripke structure  $M = \langle S, S_0, R, L^+, L^- \rangle$  are defined inductively as follows ( $\pi$  represents a path constructed from  $S$ , and  $s$  represents a state in  $S$ ):

1.  $M, s \models^+ p$  iff  $p \in L^+(s)$  for  $p \in AT$ ,
2.  $M, s \models^- p$  iff  $p \in L^-(s)$  for  $p \in AT$ ,
3.  $M, s \models^+ \sim \alpha_1$  iff  $M, s \models^- \alpha_1$ ,
4.  $M, s \models^+ \neg \alpha_1$  iff not- $[M, s \models^+ \alpha_1]$ ,
5.  $M, s \models^+ \alpha_1 \wedge \alpha_2$  iff  $M, s \models^+ \alpha_1$  and  $M, s \models^+ \alpha_2$ ,
6.  $M, s \models^+ \alpha_1 \vee \alpha_2$  iff  $M, s \models^+ \alpha_1$  or  $M, s \models^+ \alpha_2$ ,
7.  $M, s \models^+ \alpha_1 \rightarrow \alpha_2$  iff  $M, s \models^+ \alpha_1$  implies  $M, s \models^+ \alpha_2$ ,
8.  $M, s \models^+ E\beta_1$  iff there exists a path  $\pi$  from  $s$  such that  $M, \pi \models^+ \beta_1$ ,
9.  $M, s \models^+ A\beta_1$  iff for every path  $\pi$  starting from  $s$ ,  $M, \pi \models^+ \beta_1$ ,
10.  $M, \pi \models^+ \text{path}(\alpha_1)$  iff  $s$  is the first state of  $\pi$  and  $M, s \models^+ \alpha_1$ ,
11.  $M, \pi \models^+ \sim \beta_1$  iff  $M, \pi \models^- \beta_1$ ,
12.  $M, \pi \models^+ \neg \beta_1$  iff not- $[M, \pi \models^+ \beta_1]$ ,

13.  $M, \pi \models^+ \beta_1 \wedge \beta_2$  iff  $M, \pi \models^+ \beta_1$  and  $M, \pi \models^+ \beta_2$ ,
14.  $M, \pi \models^+ \beta_1 \vee \beta_2$  iff  $M, \pi \models^+ \beta_1$  or  $M, \pi \models^+ \beta_2$ ,
15.  $M, \pi \models^+ \beta_1 \rightarrow \beta_2$  iff  $M, \pi \models^+ \beta_1$  implies  $M, \pi \models^+ \beta_2$ ,
16.  $M, \pi \models^+ X\beta_1$  iff  $M, \pi^1 \models^+ \beta_1$ ,
17.  $M, \pi \models^+ F\beta_1$  iff  $\exists k \geq 0 [M, \pi^k \models^+ \beta_1]$ ,
18.  $M, \pi \models^+ G\beta_1$  iff  $\forall k \geq 0 [M, \pi^k \models^+ \beta_1]$ ,
19.  $M, \pi \models^+ \beta_1 U \beta_2$  iff  $\exists k \geq 0 [(M, \pi^k \models^+ \beta_2) \text{ and } \forall j (0 \leq j < k \text{ implies } M, \pi^j \models^+ \beta_1)]$
20.  $M, \pi \models^+ \beta_1 R \beta_2$  iff  $\forall j \geq 0 [\forall i < j \text{ not-}[M, \pi^i \models^+ \beta_1] \text{ implies } M, \pi^j \models^+ \beta_2]$
21.  $M, s \models^- \sim \alpha_1$  iff  $M, s \models^+ \alpha_1$ ,
22.  $M, s \models^- \neg \alpha_1$  iff not-[ $M, s \models^- \alpha_1$ ],
23.  $M, s \models^- \alpha_1 \wedge \alpha_2$  iff  $M, s \models^- \alpha_1$  or  $M, s \models^- \alpha_2$ ,
24.  $M, s \models^- \alpha_1 \vee \alpha_2$  iff  $M, s \models^- \alpha_1$  and  $M, s \models^- \alpha_2$ ,
25.  $M, s \models^- \alpha_1 \rightarrow \alpha_2$  iff  $M, s \models^+ \alpha_1$  and  $M, s \models^- \alpha_2$ ,
26.  $M, s \models^- E\beta_1$  iff for every path  $\pi$  starting from  $s$ ,  $M, \pi \models^- \beta_1$ ,
27.  $M, s \models^- A\beta_1$  iff there exists a path  $\pi$  from  $s$  such that  $M, \pi \models^- \beta_1$ ,
28.  $M, \pi \models^- \text{path}(\alpha_1)$  iff  $s$  is the first state of  $\pi$  and  $M, s \models^- \alpha_1$ ,
29.  $M, \pi \models^- \sim \beta_1$  iff  $M, \pi \models^+ \beta_1$ ,
30.  $M, \pi \models^- \neg \beta_1$  iff not-[ $M, \pi \models^- \beta_1$ ],
31.  $M, \pi \models^- \beta_1 \wedge \beta_2$  iff  $M, \pi \models^- \beta_1$  or  $M, \pi \models^- \beta_2$ ,
32.  $M, \pi \models^- \beta_1 \vee \beta_2$  iff  $M, \pi \models^- \beta_1$  and  $M, \pi \models^- \beta_2$ ,
33.  $M, \pi \models^- \beta_1 \rightarrow \beta_2$  iff  $M, \pi \models^+ \beta_1$  and  $M, \pi \models^- \beta_2$ ,
34.  $M, \pi \models^- X\beta_1$  iff  $M, \pi^1 \models^- \beta_1$ ,
35.  $M, \pi \models^- F\beta_1$  iff  $\forall k \geq 0 [M, \pi^k \models^- \beta_1]$ ,
36.  $M, \pi \models^- G\beta_1$  iff  $\exists k \geq 0 [M, \pi^k \models^- \beta_1]$ ,
37.  $M, \pi \models^- \beta_1 U \beta_2$  iff  $\forall j \geq 0 [\forall i < j \text{ not-}[M, \pi^i \models^- \beta_1] \text{ implies } M, \pi^j \models^- \beta_2]$ ,
38.  $M, \pi \models^- \beta_1 R \beta_2$  iff  $\exists k \geq 0 [(M, \pi^k \models^- \beta_2) \text{ and } \forall j (0 \leq j < k \text{ implies } M, \pi^j \models^- \beta_1)]$ .

The notation  $\models^*$  is used for  $\models^+$  or  $\models^-$ .  $M \models^* \alpha$  is defined by  $\forall s \in S [M, s \models^* \alpha]$ .



Since  $\models^+$  is the usual consequence relation  $\models$  of CTL\* and  $\models^-$  is the De Morgan dual of  $\models^+$ ,<sup>3</sup> the intuitive interpretations of  $\models^+$  and  $\models^-$  are “verification” and “refutation or falsification”, respectively. Roughly speaking, to introduce  $\models^-$  is thus allow to obtain the following axiom schemes w.r.t.  $\sim$ , where  $\alpha \leftrightarrow \beta$  means  $(\alpha \rightarrow \beta) \wedge (\beta \rightarrow \alpha)$ .

$$\begin{aligned}
& \sim \sim \alpha \leftrightarrow \alpha, \\
& \sim \neg \alpha \leftrightarrow \neg \sim \alpha, \\
& \sim(\alpha \wedge \beta) \leftrightarrow \sim \alpha \vee \sim \beta, \\
& \sim(\alpha \vee \beta) \leftrightarrow \sim \alpha \wedge \sim \beta, \\
& \sim(\alpha \rightarrow \beta) \leftrightarrow \alpha \wedge \sim \beta, \\
& \sim X\alpha \leftrightarrow X \sim \alpha, \\
& \sim F\alpha \leftrightarrow G \sim \alpha, \\
& \sim G\alpha \leftrightarrow F \sim \alpha, \\
& \sim E\alpha \leftrightarrow A \sim \alpha, \\
& \sim A\alpha \leftrightarrow E \sim \alpha, \\
& \sim \text{path}(\alpha) \leftrightarrow \text{path}(\sim \alpha), \\
& \sim(\alpha_1 U \alpha_2) \leftrightarrow (\sim \alpha_1) R (\sim \alpha_2), \\
& \sim(\alpha_1 R \alpha_2) \leftrightarrow (\sim \alpha_1) U (\sim \alpha_2).
\end{aligned}$$

For each state  $s$  and each formula  $\alpha$ , we can take one of the following four cases: (1)  $\alpha$  is verified at  $s$  (i.e.  $M, s \models^+ \alpha$ ), (2)  $\alpha$  is falsified at  $s$  (i.e.  $M, s \models^- \alpha$ ), (3)  $\alpha$  is both verified and falsified at  $s$ , and (4)  $\alpha$  is neither verified nor falsified at  $s$ . Thus, 4CTL\* is regarded as a four-valued logic.

Assume a Kripke structure  $M = \langle S, S_0, R, L^+, L^- \rangle$  such that  $p \in L^+(s)$ ,  $p \in L^-(s)$  and  $q \notin L^+(s)$  for any distinct atomic formulas  $p$  and  $q$ . Then,  $M, s \models^+ (p \wedge \sim p) \rightarrow q$  does not hold, and hence  $\models^+$  in 4CTL\* is paraconsistent with respect to  $\sim$ .

## 2.2. Bisimulation

DEFINITION 2.5. Let  $M = \langle S, S_0, R, L^+, L^- \rangle$  and  $M' = \langle S', S'_0, R', L'^+, L'^- \rangle$  be Kripke structures with the same nonempty set  $\text{AT} \subseteq \text{ATOM}$ .

A relation  $B \subseteq S \times S'$  is a *bisimulation relation* between  $M$  and  $M'$  if and only if for all  $s$  and  $s'$ , if  $B(s, s')$  then the following conditions hold.

1.  $L^+(s) = L'^+(s')$ ,

---

<sup>3</sup>Except the cases for X, path and  $\neg$ . These cases are adapted to be the self-dual.

2.  $L^-(s) = L^-(s')$ ,
3.  $\forall s_1[R(s, s_1) \text{ implies } \exists s'_1[R'(s', s'_1) \text{ and } B(s_1, s'_1)]]$ ,
4.  $\forall s'_1[R'(s', s'_1) \text{ implies } \exists s_1[R(s, s_1) \text{ and } B(s_1, s'_1)]]$ .

The structures  $M$  and  $M'$  are *bisimulation equivalent* if there exists a bisimulation relation  $B$  such that (1)  $\forall s_0 \in S_0 \exists s'_0 \in S'_0 [B(s_0, s'_0)]$  and (2)  $\forall s'_0 \in S'_0 \exists s_0 \in S_0 [B(s_0, s'_0)]$ .

DEFINITION 2.6. Two paths  $\pi = s_0, s_1, s_2, \dots$  in a Kripke structure  $M$  and  $\pi' = s'_0, s'_1, s'_2, \dots$  in a Kripke structure  $M'$  are called *corresponding paths* if  $\forall i \geq 0 [B(s_i, s'_i)]$ .

LEMMA 2.7. Let  $s$  and  $s'$  be two states such that  $B(s, s')$ . Then for every path starting from  $s$  there is a corresponding path starting from  $s'$ , and for every path starting from  $s'$  there is a corresponding path starting from  $s$ .

PROOF. See [12]. □

LEMMA 2.8. Let  $M$  and  $M'$  be Kripke structures with the same nonempty set  $AT \subseteq ATOM$ , and  $B$  be a bisimulation relation between  $M$  and  $M'$ . Let  $\alpha$  be a 4CTL\* formula. Assume that  $B(s, s')$  and that  $\pi$  in  $M$  and  $\pi'$  in  $M'$  are corresponding paths.

If  $\alpha$  is a state formula, then

- (1)  $M, s \models^+ \alpha$  iff  $M', s' \models^+ \alpha$ ,
- (2)  $M, s \models^- \alpha$  iff  $M', s' \models^- \alpha$ ,

and if  $\alpha$  is a path formula, then

- (1)  $M, \pi \models^+ \alpha$  iff  $M', \pi' \models^+ \alpha$ ,
- (2)  $M, \pi \models^- \alpha$  iff  $M', \pi' \models^- \alpha$ .

PROOF. This lemma is proved by (simultaneous) induction on the complexity of  $\alpha$ . In the following, the Kripke structures  $M$  and  $M'$  are omitted in the expressions, since the structures are clear from the context.

Base step:  $\alpha \equiv p$  for  $p \in AT$ . By the assumption  $B(s, s')$ , we have  $L^+(s) = L^+(s')$  and  $L^-(s) = L^-(s')$ , and hence obtain (1)  $s \models^+ p$  iff  $s' \models^+ p$  and (2)  $s \models^- p$  iff  $s' \models^- p$ .

Induction step: We show only some cases.

(Case  $\alpha \equiv \alpha_1 \vee \alpha_2$  is a state formula): (1)  $s \models^+ \alpha_1 \vee \alpha_2$  iff  $s \models^+ \alpha_1$  or  $s \models^+ \alpha_2$  iff  $s' \models^+ \alpha_1$  or  $s' \models^+ \alpha_2$  (induction hypothesis) iff  $s' \models^+ \alpha_1 \vee \alpha_2$ .

(2)  $s \models^- \alpha_1 \vee \alpha_2$  iff  $s \models^- \alpha_1$  and  $s \models^- \alpha_2$  iff  $s' \models^- \alpha_1$  and  $s' \models^- \alpha_2$   
(induction hypothesis) iff  $s' \models^- \alpha_1 \vee \alpha_2$ .

(Case  $\alpha \equiv \alpha_1 \vee \alpha_2$  is a path formula): Similar to the above case.

(Case  $\alpha \equiv \sim \alpha_1$  is a state formula)<sup>4</sup>: (1)  $s \models^+ \sim \alpha_1$  iff  $s \models^- \alpha_1$  iff  $s' \models^- \alpha_1$  (induction hypothesis) iff  $s' \models^+ \sim \alpha_1$ . (2)  $s \models^- \sim \alpha_1$  iff  $s \models^+ \alpha_1$  iff  $s' \models^+ \alpha_1$  (induction hypothesis) iff  $s' \models^- \sim \alpha_1$ .

(Case  $\alpha \equiv \sim \alpha_1$  is a path formula): Similar to the above case.

(Case  $\alpha \equiv A\alpha_1$ , a state formula): We only show:  $s \models^* \alpha$  implies  $s' \models^* \alpha$ . The converse direction can be proved in a similar way, and hence omitted. We only consider the case for  $* = -$ . Suppose  $s \models^- A\alpha_1$ . Then we have that  $\exists \pi_1$ : path starting from  $s$  [ $\pi_1 \models^- \alpha_1$ ]. By Lemma 2.7, we have that (\*): there exists a corresponding path  $\pi'_1$  starting from  $s'$ , i.e.  $\pi_1$  and  $\pi'_1$  are corresponding paths. By (\*) and the induction hypothesis, we obtain (\*\*):  $\pi_1 \models^- \alpha_1$  iff  $\pi'_1 \models^- \alpha_1$ . By (\*) and (\*\*), we obtain that  $\exists \pi'_1$ : path starting from  $s'$  [ $\pi'_1 \models^- \alpha_1$ ]. Therefore  $s' \models^- A\alpha_1$ .

(Case  $\alpha \equiv \text{path}(\alpha_1)$ , a path formula): Let  $s_0$  and  $s'_0$  be the first states of  $\pi$  and  $\pi'$ , respectively.  $\pi \models^* \text{path}(\alpha_1)$  iff  $s_0 \models^* \alpha_1$  iff  $s'_0 \models^* \alpha_1$  (induction hypothesis) iff  $\pi' \models^* \text{path}(\alpha_1)$ .

(Case  $\alpha \equiv X\alpha_1$ , a path formula): We only show:  $\pi \models^* \alpha_1$  implies  $\pi' \models^* \alpha_1$ . The converse direction can be shown in a similar way. Suppose  $\pi \models^* X\alpha_1$ . Then we have  $\pi^1 \models \alpha_1$ . Since  $\pi$  and  $\pi'$  are corresponding paths by the assumption, so are  $\pi^1$  and  $\pi'^1$ . Thus we obtain  $\pi'^1 \models \alpha_1$  by the induction hypothesis, and hence  $\pi' \models^* X\alpha_1$ .

(Case  $\alpha \equiv \alpha_1 R\alpha_2$ , a path formula): We only show:  $\pi \models^* \alpha$  implies  $\pi' \models^* \alpha$ . The converse direction can be shown in a similar way. We only consider the case for  $* = -$ . Suppose  $\pi \models^- \alpha_1 R\alpha_2$ . Then we have  $\exists k \geq 0$  [ $(\pi^k \models^- \alpha_2)$  and  $\forall j (0 \leq j < k$  implies  $\pi^j \models^- \alpha_1)$ ]. Since by the assumption,  $\pi$  and  $\pi'$  are corresponding paths,  $\pi^j$  and  $\pi'^j$  are corresponding paths for any  $j$ . Thus, by the induction hypothesis, we obtain  $\pi'^k \models^- \alpha_2$  and  $\forall j (0 \leq j < k$  implies  $\pi'^j \models^- \alpha_1)$ . Therefore  $\pi' \models^- \alpha_1 R\alpha_2$ .  $\square$

The next theorem is a consequence of the preceding lemma.

**THEOREM 2.9.** *Let  $M$  and  $M'$  be Kripke structures and  $B$  be bisimulation relation between  $M$  and  $M'$ . If  $B(s, s')$ , then for every 4CTL\* formula  $\alpha$ ,*

(1)  $M, s \models^+ \alpha$  iff  $M', s' \models^+ \alpha$ ,

(2)  $M, s \models^- \alpha$  iff  $M', s' \models^- \alpha$ .

---

<sup>4</sup>In this case, we use the simultaneous induction hypothesis with respect to both  $\models^+$  and  $\models^-$ .

We then obtain the following theorem.

**THEOREM 2.10.** *Suppose that Kripke structures  $M$  and  $M'$  are bisimulation equivalent. For every 4CTL\* formula  $\alpha$ ,*

- (1)  $M \models^+ \alpha$  iff  $M' \models^+ \alpha$ ,
- (2)  $M \models^- \alpha$  iff  $M' \models^- \alpha$ .

### 3. Translation from 4CTL\* into CTL\*

In order to obtain a translation from 4CTL\* into CTL\*, an alternative formulation of 4CTL\* is defined as a Kripke structure with a single consequence relation  $\models$ .

**DEFINITION 3.11.** Let AT1 and AT2 be nonempty subsets of ATOM and  $AT2^\sim = \{\sim p \mid p \in AT2\}$ . A *single-consequence Kripke structure* is a structure  $\langle S, S_0, R, L \rangle$  such that  $S, S_0$  and  $R$  are the same as that in Definition 2.2, and

$L$  is a function from  $S$  to the power set of  $AT1 \cup AT2^\sim$ .

A path in a single-consequence Kripke structure is defined in a similar way as in Definition 2.3.

**DEFINITION 3.12.** Let  $\alpha_1$  and  $\alpha_2$  be state formulas and  $\beta_1$  and  $\beta_2$  be path formulas. A consequence relation  $\models$  on a single-consequence Kripke structure  $M = \langle S, S_0, R, L \rangle$  is defined inductively as follows ( $\pi$  represents a path constructed from  $S$ , and  $s$  represents a state in  $S$ ):

1.  $M, s \models p$  iff  $p \in L(s)$  for  $p \in AT1$ ,
2.  $M, s \models \neg\alpha_1$  iff not- $[M, s \models \alpha_1]$ ,
3.  $M, s \models \alpha_1 \wedge \alpha_2$  iff  $M, s \models \alpha_1$  and  $M, s \models \alpha_2$ ,
4.  $M, s \models \alpha_1 \vee \alpha_2$  iff  $M, s \models \alpha_1$  or  $M, s \models \alpha_2$ ,
5.  $M, s \models \alpha_1 \rightarrow \alpha_2$  iff  $M, s \models \alpha_1$  implies  $M, s \models \alpha_2$ ,
6.  $M, s \models E\beta_1$  iff there exists a path  $\pi$  from  $s$  such that  $M, \pi \models \beta_1$ ,
7.  $M, s \models A\beta_1$  iff for every path  $\pi$  starting from  $s$ ,  $M, \pi \models \beta_1$ ,
8.  $M, \pi \models \text{path}(\alpha_1)$  iff  $s$  is the first state of  $\pi$  and  $M, s \models \alpha_1$ ,
9.  $M, \pi \models \neg\beta_1$  iff not- $[M, \pi \models \beta_1]$ ,

10.  $M, \pi \models \beta_1 \wedge \beta_2$  iff  $M, \pi \models \beta_1$  and  $M, \pi \models \beta_2$ ,
11.  $M, \pi \models \beta_1 \vee \beta_2$  iff  $M, \pi \models \beta_1$  or  $M, \pi \models \beta_2$ ,
12.  $M, \pi \models \beta_1 \rightarrow \beta_2$  iff  $M, \pi \models \beta_1$  implies  $M, \pi \models \beta_2$ ,
13.  $M, \pi \models \mathbf{X}\beta_1$  iff  $M, \pi^1 \models \beta_1$ ,
14.  $M, \pi \models \mathbf{F}\beta_1$  iff  $\exists k \geq 0 [M, \pi^k \models \beta_1]$ ,
15.  $M, \pi \models \mathbf{G}\beta_1$  iff  $\forall k \geq 0 [M, \pi^k \models \beta_1]$ ,
16.  $M, \pi \models \beta_1 \mathbf{U}\beta_2$  iff  $\exists k \geq 0 [(M, \pi^k \models \beta_2) \text{ and } \forall j (0 \leq j < k \text{ implies } M, \pi^j \models \beta_1)]$ ,
17.  $M, \pi \models \beta_1 \mathbf{R}\beta_2$  iff  $\forall j \geq 0 [\forall i < j \text{ not-}[M, \pi^i \models \beta_1] \text{ implies } M, \pi^j \models \beta_2]$ ,
18.  $M, s \models \sim p$  iff  $\sim p \in L(s)$  for  $\sim p \in \text{AT2}^\sim$ ,
19.  $M, s \models \sim \sim \alpha_1$  iff  $M, s \models \alpha_1$ ,
20.  $M, s \models \sim \neg \alpha_1$  iff not-[ $M, s \models \sim \alpha_1$ ],
21.  $M, s \models \sim(\alpha_1 \wedge \alpha_2)$  iff  $M, s \models \sim \alpha_1$  or  $M, s \models \sim \alpha_2$ ,
22.  $M, s \models \sim(\alpha_1 \vee \alpha_2)$  iff  $M, s \models \sim \alpha_1$  and  $M, s \models \sim \alpha_2$ ,
23.  $M, s \models \sim(\alpha_1 \rightarrow \alpha_2)$  iff  $M, s \models \alpha_1$  and  $M, s \models \sim \alpha_2$ ,
24.  $M, s \models \sim \mathbf{E}\beta_1$  iff for every path  $\pi$  starting from  $s$ ,  $M, \pi \models \sim \beta_1$ ,
25.  $M, s \models \sim \mathbf{A}\beta_1$  iff there exists a path  $\pi$  from  $s$  such that  $M, \pi \models \sim \beta_1$ ,
26.  $M, \pi \models \sim \text{path}(\alpha_1)$  iff  $s$  is the first state of  $\pi$  and  $M, s \models \sim \alpha_1$ ,
27.  $M, \pi \models \sim \sim \beta_1$  iff  $M, \pi \models \beta_1$ ,
28.  $M, \pi \models \sim \neg \beta_1$  iff not-[ $M, \pi \models \sim \beta_1$ ],
29.  $M, \pi \models \sim(\beta_1 \wedge \beta_2)$  iff  $M, \pi \models \sim \beta_1$  or  $M, \pi \models \sim \beta_2$ ,
30.  $M, \pi \models \sim(\beta_1 \vee \beta_2)$  iff  $M, \pi \models \sim \beta_1$  and  $M, \pi \models \sim \beta_2$ ,
31.  $M, \pi \models \sim(\beta_1 \rightarrow \beta_2)$  iff  $M, \pi \models \beta_1$  and  $M, \pi \models \sim \beta_2$ ,
32.  $M, \pi \models \sim \mathbf{X}\beta_1$  iff  $M, \pi^1 \models \sim \beta_1$ ,
33.  $M, \pi \models \sim \mathbf{F}\beta_1$  iff  $\forall k \geq 0 [M, \pi^k \models \sim \beta_1]$ ,
34.  $M, \pi \models \sim \mathbf{G}\beta_1$  iff  $\exists k \geq 0 [M, \pi^k \models \sim \beta_1]$ ,
35.  $M, \pi \models \sim(\beta_1 \mathbf{U}\beta_2)$  iff  $\forall j \geq 0 [\forall i < j \text{ not-}[M, \pi^i \models \sim \beta_1] \text{ implies } M, \pi^j \models \sim \beta_2]$ ,
36.  $M, \pi \models \sim(\beta_1 \mathbf{R}\beta_2)$  iff  $\exists k \geq 0 [(M, \pi^k \models \sim \beta_2) \text{ and } \forall j (0 \leq j < k \text{ implies } M, \pi^j \models \sim \beta_1)]$ .

An equivalence between the single-consequence Kripke structure with  $\models$  and the Kripke structure with  $\models^+$  and  $\models^-$  is shown below.

**THEOREM 3.13.** *Let  $AT$ ,  $AT1$  and  $AT2$  be nonempty subset of  $ATOM$  and  $AT2^\sim := \{\sim p \mid p \in AT2\}$ . Suppose that  $M$  is a Kripke structure  $\langle S, S_0, R, L^+, L^- \rangle$  with  $L^+, L^-: S \rightarrow 2^{AT}$ ,  $N$  is a single-consequence Kripke structure  $\langle S, S_0, R, L \rangle$  with  $L: S \rightarrow 2^{AT1} \cup AT2^\sim$ , and moreover satisfying that for any  $s \in S$ ,  $[p \in L(s) \text{ iff } p \in L^+(s)]$  and  $[\sim p \in L(s) \text{ iff } p \in L^-(s)]$ . Let  $s$  be a state in  $S$  or a path in  $M$  and  $N$ . For any 4CTL\* formula  $\alpha$ ,*

- (1)  $M, s \models^+ \alpha \text{ iff } N, s \models \alpha$ ,
- (2)  $M, s \models^- \alpha \text{ iff } N, s \models \sim \alpha$ .

**PROOF.** This theorem is proved by (simultaneous) induction on the complexity of  $\alpha$ .

Base step:

(Case  $\alpha \equiv p$  for  $p \in ATOM$ ): (1):  $M, s \models^+ p$  iff  $p \in L^+(s)$  iff  $p \in L(s)$  iff  $N, s \models p$ . (2):  $M, s \models^- p$  iff  $p \in L^-(s)$  iff  $\sim p \in L(s)$  iff  $N, s \models \sim p$ .

(Case  $\alpha \equiv \sim p$  for  $p \in ATOM$ ): (1):  $M, s \models^+ \sim p$  iff  $M, s \models^- p$  iff  $p \in L^-(s)$  iff  $\sim p \in L(s)$  iff  $N, s \models \sim p$ . (2):  $M, s \models^- \sim p$  iff  $M, s \models^+ p$  iff  $p \in L^+(s)$  iff  $p \in L(s)$  iff  $N, s \models p$  iff  $N, s \models \sim \sim p$ .

Induction step: We show only some cases.

(Case  $\alpha \equiv \sim \beta$ ): (1):  $M, s \models^+ \sim \beta$  iff  $M, s \models^- \beta$  iff  $N, s \models \sim \beta$  (induction hypothesis). (2):  $M, s \models^- \sim \beta$  iff  $M, s \models^+ \beta$  iff  $N, s \models \beta$  (induction hypothesis) iff  $N, s \models \sim \sim \beta$ .

(Case  $\alpha \equiv \alpha_1 \wedge \alpha_2$ ): (1):  $M, s \models^+ \alpha_1 \wedge \alpha_2$  iff  $M, s \models^+ \alpha_1$  and  $M, s \models^+ \alpha_2$  iff  $N, s \models \alpha_1$  and  $N, s \models \alpha_2$  (induction hypothesis) iff  $N, s \models \alpha_1 \wedge \alpha_2$ . (2):  $M, s \models^- \alpha_1 \wedge \alpha_2$  iff  $M, s \models^- \alpha_1$  or  $M, s \models^- \alpha_2$  iff  $N, s \models \sim \alpha_1$  or  $N, s \models \sim \alpha_2$  (induction hypothesis) iff  $N, s \models \sim(\alpha_1 \wedge \alpha_2)$ .

(Case  $\alpha \equiv A\beta$ ): (1):  $M, s \models^+ A\beta$  iff  $\forall \pi$ : path starting from  $s$  [ $M, \pi \models^+ \beta$ ] iff  $\forall \pi$ : path starting from  $s$  [ $N, \pi \models \beta$ ] (induction hypothesis) iff  $N, s \models A\beta$ . (2):  $M, s \models^- A\beta$  iff  $\exists \pi$ : path starting from  $s$  [ $M, \pi \models^- \beta$ ] iff  $\exists \pi$ : path starting from  $s$  [ $N, \pi \models \sim \beta$ ] (induction hypothesis) iff  $N, s \models \sim A\beta$ .

(Case  $\alpha \equiv \beta_1 U \beta_2$  and  $s$  is a path  $\pi$ ): (1):  $M, \pi \models^+ \beta_1 U \beta_2$  iff  $\exists k \geq 0$  [ $(M, \pi^k \models^+ \beta_2)$  and  $\forall j$  ( $0 \leq j < k$  implies  $M, \pi^j \models^+ \beta_1$ )] iff  $\exists k \geq 0$  [ $(N, \pi^k \models \beta_2)$  and  $\forall j$  ( $0 \leq j < k$  implies  $N, \pi^j \models \beta_1$ )] (induction hypothesis) iff  $N, \pi \models \beta_1 U \beta_2$ . (2):  $M, \pi \models^- \beta_1 U \beta_2$  iff  $\forall j \geq 0$  [ $\forall i < j$  not- $[M, \pi^i \models^- \beta_1]$  implies  $M, \pi^j \models^- \beta_2$ ] iff  $\forall j \geq 0$  [ $\forall i < j$  not- $[N, \pi^i \models \sim \beta_1]$  implies  $N, \pi^j \models \sim \beta_2$ ] (induction hypothesis) iff  $N, \pi \models \sim(\beta_1 U \beta_2)$ .  $\square$



Next, the logic CTL\* is defined.

**DEFINITION 3.14.** Let AT be a nonempty subset of ATOM. A Kripke structure for CTL\* is a single-consequence Kripke structure  $\langle S, S_0, R, L \rangle$ , where  $L$  is a function from  $S$  to the power set of AT. The consequence relation  $\models$  on a Kripke structure for CTL\* is obtained from Definition 3.12 by deleting the conditions 18–36.

In order to distinguish the consequence relations, expressions  $\models_{4\text{CTL}^*}$  and  $\models_{\text{CTL}^*}$  are used for 4CTL\* and CTL\*, respectively.

In the following, a translation function from 4CTL\* into CTL\* is introduced. This is an extension of the translation function originally presented by Rautenberg [31].

**DEFINITION 3.15.** Let AT be a nonempty subset of ATOM and  $\text{AT}' := \{p' \mid p \in \text{AT}\}$ . The language  $\mathcal{L}^\sim$  (the set of formulas) of 4CTL\* is defined by using AT,  $\sim$ ,  $\neg$ ,  $\rightarrow$ ,  $\wedge$ ,  $\vee$ , E, A, path, X, F, G, U and R. The language  $\mathcal{L}$  of CTL\* is obtained from  $\mathcal{L}^\sim$  by adding  $\text{AT}'$  and by deleting  $\sim$ .

A function  $f$  from  $\mathcal{L}^\sim$  to  $\mathcal{L}$  is defined as follows.

$$\begin{aligned}
 f(p) &:= p \text{ and } f(\sim p) := p' \in \text{AT}', \text{ for any } p \in \text{AT}, \\
 f(\alpha \circ \beta) &:= f(\alpha) \circ f(\beta), \text{ where } \circ \in \{\wedge, \vee, \rightarrow, \text{U}, \text{R}\}, \\
 f(\sim \sim \alpha) &:= f(\alpha), \\
 f(\sim(\circ \alpha)) &:= \circ f(\sim \alpha), \text{ where } \circ \in \{\neg, \text{X}\}, \\
 f(\sim(\alpha \wedge \beta)) &:= f(\sim \alpha) \vee f(\sim \beta), \\
 f(\sim(\alpha \vee \beta)) &:= f(\sim \alpha) \wedge f(\sim \beta), \\
 f(\sim(\alpha \rightarrow \beta)) &:= f(\alpha) \wedge f(\sim \beta), \\
 f(\sim(\text{E}\alpha)) &:= \text{A}f(\sim \alpha), \\
 f(\sim(\text{A}\alpha)) &:= \text{E}f(\sim \alpha), \\
 f(\sim(\text{F}\alpha)) &:= \text{G}f(\sim \alpha), \\
 f(\sim(\text{G}\alpha)) &:= \text{F}f(\sim \alpha), \\
 f(\sim(\alpha \text{U}\beta)) &:= f(\sim \alpha) \text{R}f(\sim \beta), \\
 f(\sim(\alpha \text{R}\beta)) &:= f(\sim \alpha) \text{U}f(\sim \beta).
 \end{aligned}$$

**THEOREM 3.16.** Let AT be a nonempty subset of ATOM,  $\text{AT}^\sim := \{\sim p \mid p \in \text{AT}\}$  and  $\text{AT}' := \{p' \mid p \in \text{AT}\}$ . Let  $\alpha$  be a 4CTL\* formula in  $\mathcal{L}^\sim$ , and  $f$  be a function defined in Definition 3.15. Suppose that  $M$  is a single-consequence Kripke structure  $\langle S, S_0, R, L \rangle$  for 4CTL\* with  $L: S \rightarrow 2^{\text{AT}} \cup \text{AT}^\sim$ , and that

$N$  is a Kripke structure  $\langle S, S_0, R, L' \rangle$  for CTL\* with  $L': S \rightarrow 2^{\text{AT} \cup \text{AT}'}$  such that  $L'$  is obtained from  $L$  by replacing  $\text{AT}^\sim$  by  $\text{AT}'$  satisfying that  $[p \in L(s)$  iff  $p \in L'(s)]$  and  $[\sim p \in L(s)$  iff  $p' \in L'(s)]$ . Let  $s$  be a state in  $S$  or a path in  $M$  and  $N$ .

$$M, s \models_{4\text{CTL}^*} \alpha \quad \text{iff} \quad N, s \models_{\text{CTL}^*} f(\alpha).$$

PROOF. This theorem is proved by induction on the complexity of  $\alpha$ .

Base step:

(Case  $\alpha \equiv p \in \text{AT}$ ):  $M, s \models_{4\text{CTL}^*} p$  iff  $p \in L(s)$  iff  $p \in L'(s)$  iff  $N, s \models_{\text{CTL}^*} p$  iff  $N, s \models_{\text{CTL}^*} f(p)$ .

(Case  $\alpha \equiv \sim p \in \text{AT}^\sim$ ):  $M, s \models_{4\text{CTL}^*} \sim p$  iff  $\sim p \in L(s)$  iff  $p' \in L'(s)$  iff  $N, s \models_{\text{CTL}^*} p' \in \text{AT}'$  iff  $N, s \models_{\text{CTL}^*} f(\sim p)$ .

Induction step: We show only some cases.

(Case  $\alpha \equiv \sim \sim \alpha_1$ ):  $M, s \models_{4\text{CTL}^*} \sim \sim \alpha_1$  iff  $M, s \models_{4\text{CTL}^*} \alpha_1$  iff  $N, s \models_{\text{CTL}^*} f(\alpha_1)$  (induction hypothesis) iff  $N, s \models_{\text{CTL}^*} f(\sim \sim \alpha_1)$ .

(Case  $\alpha \equiv \sim(\alpha_1 \wedge \alpha_2)$ ):  $M, s \models_{4\text{CTL}^*} \sim(\alpha_1 \wedge \alpha_2)$  iff  $M, s \models_{4\text{CTL}^*} \sim \alpha_1$  or  $M, s \models_{4\text{CTL}^*} \sim \alpha_2$  iff  $N, s \models_{\text{CTL}^*} f(\sim \alpha_1)$  or  $N, s \models_{\text{CTL}^*} f(\sim \alpha_2)$  (induction hypothesis) iff  $N, s \models_{\text{CTL}^*} f(\sim \alpha_1) \vee f(\sim \alpha_2)$  iff  $N, s \models_{\text{CTL}^*} f(\sim(\alpha_1 \wedge \alpha_2))$ .

(Case  $\alpha \equiv \sim \text{E}\beta$ ):  $M, s \models_{4\text{CTL}^*} \sim \text{E}\beta$  iff  $\forall \pi$ : path starting from  $s$   $[M, \pi \models_{4\text{CTL}^*} \sim \beta]$  iff  $\forall \pi$ : path starting from  $s$   $[N, \pi \models_{\text{CTL}^*} f(\sim \beta)]$  (induction hypothesis) iff  $N, s \models_{\text{CTL}^*} \text{A}f(\sim \beta)$  iff  $N, s \models_{\text{CTL}^*} f(\sim \text{E}\beta)$ .

(Case  $\alpha \equiv \sim(\beta_1 \text{R}\beta_2)$  and  $s$  is a path  $\pi$ ):  $M, \pi \models_{4\text{CTL}^*} \sim(\beta_1 \text{R}\beta_2)$  iff  $\exists k \geq 0$   $[(M, \pi^k \models_{4\text{CTL}^*} \sim \beta_2)$  and  $\forall j$  ( $0 \leq j < k$  implies  $M, \pi^j \models_{4\text{CTL}^*} \sim \beta_1$ )] iff  $\exists k \geq 0$   $[(N, \pi^k \models_{\text{CTL}^*} f(\sim \beta_2))$  and  $\forall j$  ( $0 \leq j < k$  implies  $N, \pi^j \models_{\text{CTL}^*} f(\sim \beta_1)$ )] (induction hypothesis) iff  $N, \pi \models_{\text{CTL}^*} f(\sim \beta_1) \text{U} f(\sim \beta_2)$  iff  $N, \pi \models_{\text{CTL}^*} f(\sim(\beta_1 \text{R}\beta_2))$ .  $\square$

By using Theorem 3.16, the existing model checking algorithms for CTL\* can be used for 4CTL\*: Assuming  $\text{AT}' = \{p' \mid p \in \text{AT} \subseteq \text{ATOM}\}$ , the translation function  $f$  interpretes formulas with  $\sim$ .

## 4. 4LCTL\* and bisimulation

### 4.1. 4LCTL\*

DEFINITION 4.17. Let  $\text{Loc}$  be a finite nonempty set of *locations*, and  $l \in \text{Loc}$ . State and path formulas of 4LCTL\* are obtained from Definition 2.1 by adding the following rules. (1) if  $\alpha$  is a state formula, then so is  $[l]\alpha$ . (2) if  $\beta$  is a path formula, then so is  $[l]\beta$ . State and path formulas of 4LCTL\* are 4LCTL\* formulas.



DEFINITION 4.18. A *locative Kripke structure* is a structure  $\langle \text{Loc}, S, S_0, R, L^+, L^- \rangle$  such that  $S, S_0, R$  are the same as those in Definition 2.2,  $L^+, L^-$  are functions from  $S \times \text{Loc}$  to the power set of the nonempty AT ( $\subseteq \text{ATOM}$ ), and  $\text{Loc}$  is a finite nonempty set of locations.

A path in a locative Kripke structure can be defined similarly in Definition 2.3, which is independent of  $\text{Loc}$ .

DEFINITION 4.19. Let AT be a nonempty subset of ATOM. Let  $\alpha_1$  and  $\alpha_2$  be state formulas of 4LCTL\* and,  $\beta_1$  and  $\beta_2$  be path formulas of 4LCTL\*. *Consequence relations*  $\models^+$  and  $\models^-$  on a locative Kripke structure  $M = \langle \text{Loc}, S, S_0, R, L^+, L^- \rangle$  is defined inductively as follows ( $\pi$  represents a path constructed from  $S$ ,  $s$  represents a state in  $S$ , and  $l \in \text{Loc}$ ):

1.  $M, (s, l) \models^+ p$  iff  $p \in L^+(s, l)$  for  $p \in \text{AT}$  and  $l \in \text{Loc}$ ,
2.  $M, (s, l) \models^- p$  iff  $p \in L^-(s, l)$  for  $p \in \text{AT}$  and  $l \in \text{Loc}$ ,
3.  $M, (s, l) \models [k]\alpha_1$  iff  $M, (s, k) \models \alpha_1$ ,
4.  $M, (s, l) \models^+ \sim \alpha_1$  iff  $M, (s, l) \models^- \alpha_1$ ,
5.  $M, (s, l) \models^+ \neg \alpha_1$  iff  $\text{not-}[M, (s, l) \models^+ \alpha_1]$ ,
6.  $M, (s, l) \models^+ \alpha_1 \wedge \alpha_2$  iff  $M, (s, l) \models^+ \alpha_1$  and  $M, (s, l) \models^+ \alpha_2$ ,
7.  $M, (s, l) \models^+ \alpha_1 \vee \alpha_2$  iff  $M, (s, l) \models^+ \alpha_1$  or  $M, (s, l) \models^+ \alpha_2$ ,
8.  $M, (s, l) \models^+ \alpha_1 \rightarrow \alpha_2$  iff  $M, (s, l) \models^+ \alpha_1$  implies  $M, (s, l) \models^+ \alpha_2$ ,
9.  $M, (s, l) \models^+ E\beta_1$  iff there exists a path  $\pi$  from  $s$  such that  $M, (\pi, l) \models^+ \beta_1$ ,
10.  $M, (s, l) \models^+ A\beta_1$  iff for every path  $\pi$  starting from  $s$ ,  $M, (\pi, l) \models^+ \beta_1$ ,
11.  $M, (\pi, l) \models^+ \text{path}(\alpha_1)$  iff  $s$  is the first state of  $\pi$  and  $M, (s, l) \models^+ \alpha_1$ ,
12.  $M, (\pi, l) \models [k]\beta_1$  iff  $M, (\pi, k) \models \beta_1$ ,
13.  $M, (\pi, l) \models^+ \sim \beta_1$  iff  $M, (\pi, l) \models^- \beta_1$ ,
14.  $M, (\pi, l) \models^+ \neg \beta_1$  iff  $\text{not-}[M, (\pi, l) \models^+ \beta_1]$ ,
15.  $M, (\pi, l) \models^+ \beta_1 \wedge \beta_2$  iff  $M, (\pi, l) \models^+ \beta_1$  and  $M, (\pi, l) \models^+ \beta_2$ ,
16.  $M, (\pi, l) \models^+ \beta_1 \vee \beta_2$  iff  $M, (\pi, l) \models^+ \beta_1$  or  $M, (\pi, l) \models^+ \beta_2$ ,
17.  $M, (\pi, l) \models^+ \beta_1 \rightarrow \beta_2$  iff  $M, (\pi, l) \models^+ \beta_1$  implies  $M, (\pi, l) \models^+ \beta_2$ ,
18.  $M, (\pi, l) \models^+ X\beta_1$  iff  $M, (\pi^1, l) \models^+ \beta_1$ ,
19.  $M, (\pi, l) \models^+ F\beta_1$  iff  $\exists k \geq 0 [M, (\pi^k, l) \models^+ \beta_1]$ ,

20.  $M, (\pi, l) \models^+ G\beta_1$  iff  $\forall k \geq 0 [M, (\pi^k, l) \models^+ \beta_1]$ ,
21.  $M, (\pi, l) \models^+ \beta_1 U \beta_2$  iff  $\exists k \geq 0 [[M, (\pi^k, l) \models^+ \beta_2]$  and  $\forall j [0 \leq j < k$  implies  $M, (\pi^j, l) \models^+ \beta_1]]$
22.  $M, (\pi, l) \models^+ \beta_1 R \beta_2$  iff  $\forall j \geq 0 [\forall i < j \text{ not-}[M, (\pi^i, l) \models^+ \beta_1]$  implies  $M, (\pi^j, l) \models^+ \beta_2]$
23.  $M, (s, l) \models^- \sim \alpha_1$  iff  $M, (s, l) \models^+ \alpha_1$ ,
24.  $M, (s, l) \models^- \neg \alpha_1$  iff  $\text{not-}[M, (s, l) \models^- \alpha_1]$ ,
25.  $M, (s, l) \models^- \alpha_1 \wedge \alpha_2$  iff  $M, (s, l) \models^- \alpha_1$  or  $M, (s, l) \models^- \alpha_2$ ,
26.  $M, (s, l) \models^- \alpha_1 \vee \alpha_2$  iff  $M, (s, l) \models^- \alpha_1$  and  $M, (s, l) \models^- \alpha_2$ ,
27.  $M, (s, l) \models^- \alpha_1 \rightarrow \alpha_2$  iff  $M, (s, l) \models^+ \alpha_1$  and  $M, (s, l) \models^- \alpha_2$ ,
28.  $M, (s, l) \models^- E\beta_1$  iff for every path  $\pi$  starting from  $s$ ,  $M, (\pi, l) \models^- \beta_1$ ,
29.  $M, (s, l) \models^- A\beta_1$  iff there exists a path  $\pi$  from  $s$  such that  $M, (\pi, l) \models^- \beta_1$ ,
30.  $M, (\pi, l) \models^- \text{path}(\alpha_1)$  iff  $s$  is the first state of  $\pi$  and  $M, (s, l) \models^- \alpha_1$ ,
31.  $M, (\pi, l) \models^- \sim \beta_1$  iff  $M, (\pi, l) \models^+ \beta_1$ ,
32.  $M, (\pi, l) \models^- \neg \beta_1$  iff  $\text{not-}[M, (\pi, l) \models^- \beta_1]$ ,
33.  $M, (\pi, l) \models^- \beta_1 \wedge \beta_2$  iff  $M, (\pi, l) \models^- \beta_1$  or  $M, (\pi, l) \models^- \beta_2$ ,
34.  $M, (\pi, l) \models^- \beta_1 \vee \beta_2$  iff  $M, (\pi, l) \models^- \beta_1$  and  $M, (\pi, l) \models^- \beta_2$ ,
35.  $M, (\pi, l) \models^- \beta_1 \rightarrow \beta_2$  iff  $M, (\pi, l) \models^+ \beta_1$  and  $M, (\pi, l) \models^- \beta_2$ ,
36.  $M, (\pi, l) \models^- X\beta_1$  iff  $M, (\pi^1, l) \models^- \beta_1$ ,
37.  $M, (\pi, l) \models^- F\beta_1$  iff  $\forall k \geq 0 [M, (\pi^k, l) \models^- \beta_1]$ ,
38.  $M, (\pi, l) \models^- G\beta_1$  iff  $\exists k \geq 0 [M, (\pi^k, l) \models^- \beta_1]$ ,
39.  $M, (\pi, l) \models^- \beta_1 U \beta_2$  iff  $\forall j \geq 0 [\forall i < j \text{ not-}[M, (\pi^i, l) \models^- \beta_1]$  implies  $M, (\pi^j, l) \models^- \beta_2]$ ,
40.  $M, (\pi, l) \models^- \beta_1 R \beta_2$  iff  $\exists k \geq 0 [[M, (\pi^k, l) \models^- \beta_2]$  and  $\forall j [0 \leq j < k$  implies  $M, (\pi^j, l) \models^- \beta_1]]$ .

$M \models^* \alpha$  is defined by  $\forall s \in S \forall l \in \text{Loc} [M, (s, l) \models^* \alpha]$ .

An expression  $M \models^* \alpha$  can intuitively be interpreted as “If a proposition  $\alpha$  can be verified (or refuted) at any time in the future for all spaces in a world  $M$ , then the proposition is the eternal truth (or falsehood) in the world” [22]. The proposed setting of the location operator  $[l]$  represents the

discrete space interpretation in which a location is regarded as a point and is independent of other locations.

To introduce  $[l]$  is to allow the following axiom schemes with respect to  $[l]$ : for any  $l, l_i, l_j \in \text{Loc}$ ,

$$\begin{aligned} [l_i][l_j]\alpha &\leftrightarrow [l_i]\alpha, \\ [l](\alpha \# \beta) &\leftrightarrow ([l]\alpha) \# ([l]\beta), \quad \# \in \{\wedge, \vee, \rightarrow, \text{U}, \text{R}\}, \\ [l](\# \alpha) &\leftrightarrow \#([l]\alpha), \quad \# \in \{\sim, \neg, \text{E}, \text{A}, \text{X}, \text{F}, \text{G}\}, \end{aligned}$$

where an expression  $\alpha \leftrightarrow \beta$  means  $(\alpha \rightarrow \beta) \wedge (\beta \rightarrow \alpha)$ . The first axiom scheme  $[l_i][l_j]\alpha \leftrightarrow [l_i]\alpha$  intuitively means that each location  $l$  is the absolute address of locations, i.e. the location  $l$  refers to the same location anywhere [25]. The second and third axiom schemes intuitively mean that the truth is time- and space-independent, i.e. “space” is almost independent of “time”. As mentioned in [23, 22], the following inference rule called *space induction rule* is also true:

$$\frac{(\forall l \in \text{Loc})([l]\alpha)}{\alpha}.$$

In this rule, for example, if  $[l_1]\alpha$ ,  $[l_2]\alpha$  and  $[l_3]\alpha$  with  $\text{Loc} = \{l_1, l_2, l_3\}$  hold, then  $\alpha$  holds. It is also remarked that if  $\text{Loc} = \{l\}$ , then  $[l]$  derives the modal logic S4-like axiom schemes:

$$\begin{aligned} [l](\alpha \rightarrow \beta) &\rightarrow ([l]\alpha \rightarrow [l]\beta), \\ [l]\alpha &\rightarrow [l][l]\alpha, \\ [l]\alpha &\rightarrow \alpha, \end{aligned}$$

and hence  $[l]$  is more expressive (or stronger) than the S4-type modal operator. Since the case that  $\text{Loc}$  is empty corresponds to the 4CTL\* case, 4LCTL\* is regarded as a natural generalization or extension of both CTL\* and 4CTL\*.

## 4.2. Bisimulation

**DEFINITION 4.20.** Let  $M = \langle \text{Loc}, S, S_0, R, L^+, L^- \rangle$  and  $M' = \langle \text{Loc}, S', S'_0, R', L'^+, L'^- \rangle$  be locative Kripke structures with the common (nonempty) sets  $\text{AT} (\subseteq \text{ATOM})$  and  $\text{Loc}$ . The definition of bisimulation w.r.t.  $M$  and  $M'$  is almost the same as that in Definition 2.5, since the space domain  $\text{Loc}$  is independent of this definition.<sup>5</sup> The notion of the ‘corresponding paths’ in locative Kripke structures is the same as that in Definition 2.6.

<sup>5</sup>Of course, the conditions  $L^+(s) = L'^+(s')$  and  $L^-(s) = L'^-(s')$  in Definition 2.5 must be replaced by  $L^+(s, l) = L'^+(s', l)$  and  $L^-(s, l) = L'^-(s', l)$ , respectively.

The lemma concerning the notion of the ‘corresponding paths’ also holds for the same setting as Lemma 2.7.

LEMMA 4.21. *Let  $M$  and  $M'$  be locative Kripke structures with the common (nonempty) sets  $AT$  ( $\subseteq ATOM$ ) and  $Loc$ , and  $B$  be a bisimulation relation between  $M$  and  $M'$ . Let  $\alpha$  be a 4LCTL\* formula and  $l \in Loc$ . Assume that  $B(s, s')$  and that  $\pi$  in  $M$  and  $\pi'$  in  $M'$  are corresponding paths.*

*If  $\alpha$  is a state formula, then*

- (1)  $M, (s, l) \models^+ \alpha$  iff  $M', (s', l) \models^+ \alpha$ ,
- (2)  $M, (s, l) \models^- \alpha$  iff  $M', (s', l) \models^- \alpha$ ,

*and if  $\alpha$  is a path formula, then*

- (1)  $M, (\pi, l) \models^+ \alpha$  iff  $M', (\pi', l) \models^+ \alpha$ ,
- (2)  $M, (\pi, l) \models^- \alpha$  iff  $M', (\pi', l) \models^- \alpha$ .

PROOF. This lemma is proved by induction on the complexity of  $\alpha$ . The proof is almost the same as that in Lemma 2.8. In the following, the locative Kripke structures  $M$  and  $M'$  are omitted in the expressions. The base step is obvious. We only consider the following cases for the induction step.

(Case  $\alpha \equiv [k]\alpha_1$  is a state formula or a path formula):  $(s, l) \models^* [k]\alpha_1$  iff  $(s, k) \models^* \alpha_1$  iff  $(s', k) \models^* \alpha_1$  (induction hypothesis) iff  $(s', l) \models^* [k]\alpha_1$ .

(Case  $\alpha \equiv \alpha_1 U \alpha_2$ , a path formula): We only show:  $(\pi, l) \models^* \alpha$  implies  $(\pi', l) \models^* \alpha$ . The converse direction can be shown in a similar way. We only consider the case for  $* = +$ . Suppose  $(\pi, l) \models^+ \alpha_1 U \alpha_2$ . Then we have  $\exists k \geq 0$   $[(\pi^k, l) \models^+ \alpha_2]$  and  $\forall j [0 \leq j < k$  implies  $(\pi^j, l) \models^+ \alpha_1]$ . Since by the assumption,  $\pi$  and  $\pi'$  are corresponding paths,  $\pi^j$  and  $\pi'^j$  are corresponding paths for any  $j$ . Thus, by the induction hypothesis, we obtain  $(\pi'^k, l) \models^+ \alpha_2$  and  $\forall j [0 \leq j < k$  implies  $(\pi'^j, l) \models^+ \alpha_1]$ . Therefore  $(\pi', l) \models^+ \alpha_1 U \alpha_2$ .  $\square$

THEOREM 4.22. *Let  $M$  and  $M'$  be locative Kripke structures and  $B$  be a bisimulation relation between  $M$  and  $M'$ . If  $B(s, s')$ , then for every 4LCTL\* formula  $\alpha$  and any  $l \in Loc$ ,*

- (1)  $M, (s, l) \models^+ \alpha$  iff  $M', (s', l) \models^+ \alpha$ ,
- (2)  $M, (s, l) \models^- \alpha$  iff  $M', (s', l) \models^- \alpha$ .

THEOREM 4.23. *Suppose that locative Kripke structures  $M$  and  $M'$  are bisimulation equivalent. For every 4LCTL\* formula  $\alpha$ ,*

- (1)  $M \models^+ \alpha$  iff  $M' \models^+ \alpha$ ,
- (2)  $M \models^- \alpha$  iff  $M' \models^- \alpha$ .



## 5. Illustrative examples

### 5.1. Bipolar preferences

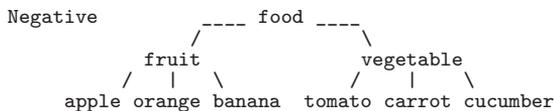
It is known that *preference modeling* is a basic activity for any type of decision aiding process. Classical logic is used as a description language in traditional preference modeling. On the other hand, the classical logic is not always suitable to formalize real-life situations, since it is unable to handle inconsistent and uncertain information. A issue in preference modeling is to represent preference statements of the types ‘I prefer  $\alpha$  to  $\beta$ ’, which is called a *positive preference*, and ‘I disprefer  $\alpha$  to  $\beta$ ’, which is called a *negative preference*.<sup>6</sup> It is remarked that the negative preference is not always the converse of the positive preference, since the notion ‘preferences’ in our real life is uncertain and inconsistent. Both positive and negative preferences, which are also called *bipolar preferences*, have recently been modelled based on some non-classical logics (see e.g. [2, 4] and the references therein).

In the framework based on 4CTL\*, the bipolar preference statements ‘I prefer  $\alpha$  to  $\beta$ ’ and ‘I disprefer  $\alpha$  to  $\beta$ ’ can respectively be expressed in a Kripke structure as single-step transitions  $s_i \rightarrow s_{i+1}$  with  $\beta \in L^+(s_i)$  and  $\alpha \in L^+(s_{i+1})$ , and  $t_j \rightarrow t_{j+1}$  with  $\beta \in L^-(t_j)$  and  $\alpha \in L^-(t_{j+1})$ . A generalization of positive (or negative) preference expressions:  $s_1 \rightarrow s_2 \rightarrow s_3 \rightarrow \dots \rightarrow s_n$  where  $L^+(s_i) = \alpha_i$  (or  $L^-(s_i) = \alpha_i$ ) means an ascending order  $\alpha_1 \leq \alpha_2 \leq \alpha_3 \leq \dots \leq \alpha_n$  of positive (or negative) preferences.

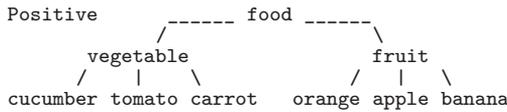
In the framework based on the located version 4LCTL\*, information on locations can be added to the preference descriptions. For example, the statement ‘Japanese people disprefer apple to orange in general’ can be expressed as  $s_i \rightarrow s_{i+1}$  with *orange*  $\in L^-(s_i, \text{Japan})$  and *apple*  $\in L^-(s_{i+1}, \text{Japan})$ , where  $\text{Loc} = \{\text{Japan}, \text{China}, \dots\}$ .

### 5.2. Taxonomic trees

An example for taxonomic hierarchies with bipolar preferences for foods is addressed based on the following two taxonomic trees.



<sup>6</sup>It is explained in [4] that a positive preference is a statement of the type ‘I like  $\alpha$ , and I like  $\beta$  even more than  $\alpha$ ’, and a negative preference is a statement of the type ‘I don’t like  $\alpha$ , and I really don’t like  $\beta$ ’.



In these expressions, the negative preference orders are presented as

$$\begin{aligned}
 &fruit \leq vegetable, \\
 &apple \leq orange \leq banana \leq tomato \leq carrot \leq cucumber,
 \end{aligned}$$

and the positive preference orders are presented as

$$\begin{aligned}
 &vegetable \leq fruit, \\
 &cucumber \leq tomato \leq carrot \leq orange \leq apple \leq banana.
 \end{aligned}$$

In order to represent the taxonomic trees using a Kripke structure, the following set of atomic formulas are assumed.

$$\begin{aligned}
 AT = \{ &food, fruit, vegetable, apple, orange, banana, \\
 &cucumber, tomato, carrot \}.
 \end{aligned}$$

A Kripke structure  $M = \langle S, S_0, R, L^+, L^- \rangle$  for the taxonomic trees can be defined as

$$\begin{aligned}
 S &= \{s_0, s_1, s_2, s_3, s_4, s_5\}, & S_0 &= \{s_0\}, \\
 R &= \{(s_0, s_1), (s_1, s_2), (s_2, s_3), (s_3, s_4), (s_4, s_5), (s_5, s_5)\}, \\
 L^-(s_0) &= \{food, fruit, apple\}, \\
 L^-(s_1) &= \{food, fruit, orange\}, \\
 L^-(s_2) &= \{food, fruit, banana\}, \\
 L^-(s_3) &= \{food, vegetable, tomato\}, \\
 L^-(s_4) &= \{food, vegetable, carrot\}, \\
 L^-(s_5) &= \{food, vegetable, cucumber\}, \\
 L^+(s_0) &= \{food, vegetable, cucumber\}, \\
 L^+(s_1) &= \{food, vegetable, tomato\}, \\
 L^+(s_2) &= \{food, vegetable, carrot\}, \\
 L^+(s_3) &= \{food, fruit, orange\}, \\
 L^+(s_4) &= \{food, fruit, apple\}, \\
 L^+(s_5) &= \{food, fruit, banana\}.
 \end{aligned}$$



By using this structure, for example, the statement ‘I prefer banana to apple’ is expressed as  $s_4 \rightarrow s_5$  with  $apple \in L(s_4)$  and  $banana \in L(s_5)$ , and the statement ‘I disprefer vegetable to fruit’ is expressed as  $s_2 \rightarrow s_3$  with  $fruit \in L^-(s_2)$  and  $vegetable \in L^-(s_3)$ . This Kripke structure is regarded as a database representing the taxonomic trees for food preferences. Then we can check some properties for this database. For example, consider the following statements.

- (1) There is ‘orange’ in this database, i.e. there is a state  $s_i$  such that  $orange \in L^+(s_i)$ .
- (2) ‘Orange’ is ‘fruite’, i.e. there is a state  $s_i$  such that  $\{orange, fruite\} \subseteq L^+(s_i)$ .
- (3) The root of the underlying taxonomic trees is ‘food’, i.e. the proposition  $food$  holds for all states.

These statements are respectively expressed as  $M \models^+ AForange$ ,  $M \models^+ AF(orange \wedge fruite)$ , and  $M \models^+ AGfood$ .

It is remarked that by using the framework based on a locative Kripke structure for 4LCTL\*, location information can be added to the expression defined above.

### 5.3. Biomedical ontologies

It is known that biomedical ontology constructions are important issues of life science, and various ontologies such as the Open Biomedical Ontologies (OBO) and the Gene Ontologies (GO) have been proposed by many researchers (see e.g. [18, 32] and the references therein). *Biomedical ontologies* are a kind of knowledge representation models with hierarchies of biomedical vocabularies, called *controlled vocabularies*, and are usually represented as trees or directed acyclic graphs with some relations labelled by is-a (sub-type relation), part-of (inclusion relation), located-in (spatial relation) and preceded-by (temporal relation) [32].

Tree and directed acyclic graph structures can appropriately be described by a (locative) Kripke structure, and the reasoning about time, space and uncertainty can naturally be represented by the virtue of 4CTL\* and 4LCTL\*. A data base using such ontologies can thus be represented using a model checking framework based on the logics. A verification (or search) for a biological process called *pathway* can be performed by finding a path in a (locative) Kripke structure.

## 6. Concluding remarks

In this paper, the bisimulation theorems for  $4CTL^*$  and  $4LCTL^*$  were shown, and the translation from  $4CTL^*$  into  $CTL^*$  was presented using the single-consequence Kripke structure for  $4CTL^*$ . In addition, some illustrative examples including bipolar preference modeling and taxonomic tree representation were presented based on  $4CTL^*$  and  $4LCTL^*$ . It was thus shown that  $4CTL^*$  and  $4LCTL^*$  are useful as base logics for paraconsistent model checking: The bisimulation results guarantee to be able to apply abstraction, and the translation result allows us to use the existing  $CTL^*$  model checking algorithms.

It is pointed out that although the idea of using two kinds of consequence relations  $\models^+$  and  $\models^-$  in Kripke structures is originally due to Routley [30], the idea of using a single consequence relation  $\models$  with respect to  $\sim$  is a new idea introduced in this paper, and the natural translation technique can thus be realized using the single-consequence Kripke structure. The single-consequence Kripke structure-based bisimulation theorem can also be shown for  $4CTL^*$ .

It is remarked that in a Kripke structure  $\langle S, S_0, R, L^+, L^- \rangle$  for  $4CTL^*$ , to adopt the condition  $\forall s \in S [L^+(s) \cap L^-(s) = \emptyset]$  is equivalent to the fact that the underlying logic becomes a three-valued logic which is called here  $3CTL^*$ . Although  $3CTL^*$  is not paraconsistent, the corresponding bisimulation and translation results can be obtained. These results for  $3CTL^*$  can also be adapted for the locative version  $3LCTL^*$ . In addition, the bisimulation result with Loc can be adapted for the  $\sim$ -free part  $LCTL^*$  of  $3LCTL^*$  (or  $4LCTL^*$ ), i.e.  $CTL^*$  with the location operator.

## References

- [1] A. Almkudad and D. Nelson, “Constructible falsity and inexact predicates”, *Journal of Symbolic Logic* 49 (1984), 231–233.
- [2] S. Benferhat, D. Dubois, S. Kaci, and H. Prade, “Bipolar possibility theory in preference modeling: Representation, fusion and optimal solutions”, *Information Fusion* 7(1) (2006), 135–150.
- [3] J.-Y. Béziau, “The future of paraconsistent logic”, *Logical Studies* 2 (1999). Online available.



- [4] S. Bistarelli, M.S. Pini, F. Rossi, and K.B. Venable, “Positive and negative preferences”, Proceeding of the 7th International Workshop on Preferences and soft constraints (Soft 2005) held in conjunction with CP2005, Sitges, Spain, 2005.
- [5] G. Bruns and P. Godefroid, “Model checking with multi-valued logics”, Proceedings of ICALP2004, *Lecture Notes in Computer Science* 3142 (2004), pp. 281–293.
- [6] M.C. Browne, E.M. Clarke, and O. Grumberg, “Characterizing finite Kripke structures in propositional temporal logic”, *Theoretical Computer Science* 59 (1988), pp. 115–131.
- [7] W.A. Carnielli, M.E. Coniglio, and J. Marcos, “Logics of formal inconsistency”, in D. Gabbay and F. Guentner (eds.), *Handbook of Philosophical Logic*, vol. 14, pp. 1–124, Kluwer Academic Publishers, 2004.
- [8] M. Chechik, B. Devereux, S. Easterbrook, and A. Gurfinkel, “Multi-valued symbolic model-checking”, *ACM Transactions on Software Engineering and Methodology* 12(4) (2004), 1–38.
- [9] M. Chechik, and W. MacCaull, “CTL model-checking over logics with non-classical negations”, Proceedings of 33rd IEEE International Conference on Multi-valued logics (ISMVL’03), pp. 293–300, 2003.
- [10] E.M. Clarke and E.A. Emerson, “Design and synthesis of synchronization skeletons using branching time temporal logic”, Proceedings of the Workshop on logic of Programs, *Lecture Notes in Computer Science* 131 (1981), 52–71.
- [11] E.M. Clarke, O. Grumberg, S. Jha, Y. Lu, and H. Veith, “Counterexample-guided abstraction refinement for symbolic model checking”, *Journal of the ACM* 50(5) (2003), 752–794.
- [12] E.M. Clarke, O. Grumberg, and D.A. Peled, *Model checking*, The MIT Press, 1999.
- [13] C.V. Damásio and L.M. Pereira, “A survey of paraconsistent semantics for logic programs”, in D. Gabbay and P. Smets (eds.), *Handbook of Defeasible Reasoning and Uncertainty Management Systems*, vol. 2, pp. 421–320, Kluwer Academic Publishers, 1998.
- [14] S. Easterbrook, and M. Chechik, “A framework for multi-valued reasoning over inconsistent viewpoints”, Proceedings of the 23rd International Conference on Software Engineering (ICSE 2001), pp. 411–420.
- [15] S. Easterbrook, and M. Chechik, “Automated paraconsistent reasoning via model checking”, Proceedings of IJCAI Inconsistency Workshop, 8 pages, 2001.
- [16] E.A. Emerson and J.Y. Halpern, “‘Sometimes’ and ‘not never’ revisited: on branching versus linear time temporal logic”, *Journal of the ACM* 33(1) (1986), 151–178.

- [17] E.A. Emerson and P. Sistla, “Deciding full branching time logic”, *Information and Control* 61 (1984), 175–201.
- [18] Ken I. Fukuda, “Biomedical ontologies and their applications” (in Japanese), *Journal of the Japanese Society for Artificial Intelligence* 22(1) (2007), 70–76.
- [19] A. Gurfinkel, O. Wei, and M. Chechik, “Yasm: a software model-checker for verification and refutation”, *Proceedings of Computer Aided Verification (CAV’06)*, pp. 170–174, 2006.
- [20] A. Gurfinkel, and M. Chechik, “Why wast a perfectly good abstraction?”, *Proceedings of TACAS’06, Lecture Notes in Computer Science* 3920 (2006), 212–226.
- [21] N. Kamide, “Gentzen-type methods for bilattice negation”, *Studia Logica* 80 (2005), 265–289.
- [22] N. Kamide, “A spatial modal logic with a location interpretation”, *Mathematical Logic Quarterly* 51(4) (2005), 331–341.
- [23] N. Kamide, “Linear and affine logics with temporal, spatial and epistemic operators”, *Theoretical Computer Science* 353(1–3) (2006), 165–207.
- [24] N. Kamide, “Foundations of paraconsistent resolution”, *Fundamenta Informaticæ* 71(4) (2006), 419–441.
- [25] N. Kobayashi, T. Shimizu, and A. Yonezawa, “Distributed concurrent linear logic programming”, *Theoretical Computer Science* 227 (1999), 185–220.
- [26] D. Nelson, “Constructible falsity”, *Journal of Symbolic Logic* 14 (1949), 16–26.
- [27] G. Priest, and R. Routly, “Introduction: paraconsistent logics”, *Studia Logica* 43 (1982), 3–16.
- [28] A. Pnueli, “The temporal logic of programs”, *Proceedings of the 18th IEEE Symposium on Foundations of Computer Science*, pp. 46–57, 1977.
- [29] J.P. Quielle and J. Sifakis, “Specification and verification of concurrent systems in CESAR”, *Proceedings of the 5th International Symposium on Programming*, pp. 337–351, 1982.
- [30] R. Routley, “Semantical analysis of propositional systems of Fitch and Nelson”, *Studia Logica* 33 (1974), 283–298.
- [31] W. Rautenberg, *Klassische und nicht-klassische Aussagenlogik*, Vieweg, Braunschweig, 1979.
- [32] B. Smith, W. Ceusters, B. Klagges, J. Köhler, A. Kumar, J. Lomax, C. Mungall, F. Neuhaus, A.L. Rector, and C. Rosse, “Relations in biomedical ontologies”, *Genome Biology* 6 (2005), R46, 15 pages.

- [33] G. Wagner, “Logic programming with strong negation and inexact predicates”, *Journal of Logic and Computation* 1(6) (1991), 835–859.
- [34] H. Wansing, “The logic of information structures”, *Lecture Notes in Artificial Intelligence* 681 (1993), 163 pages.

NORIHITO KAMIDE

Research Center for Verification and Semantics

National Institute of Advanced Industrial Science and Technology

5th Floor, Mitsui Sumitomo Kaijo Bldg.

1-2-14 Shin-Senri Nishi, Toyonaka, Osaka, 560-0083 Japan

`kamide@mtc.biglobe.ne.jp`