On Sufficient and Necessary Conditions in transition system with Finite States: a preliminary report

Abstract

This article describes a method to compute the SNC (WSC) of a given property (a CTL formula) under a given transition system (expressed as a Kripke structure) by proposing a semantic forgetting for CTL. We show that the CTL system is close under our definition of forgetting, and this definition satisfies those four postulates of forgetting at first. By changing a transition system $\mathcal{M}$ into its characteristic formula, we can compute the WSC (SNC) of a property under the transition system. We also investigate some properties and algorithm for the forgetting.

1 Introduction

Weakest precondition, we also call *weakest sufficient condition* (WSC), is introduced by Dijkstra in [Dijkstra, 1978]. *Strongest postcondition* (we also call *strongest necessary condition* (SNC)), a dual concept, was introduced subsequently. WSC was widely used in program *verification*, especially in generating counterexamples [Dailler *et al.*, 2018] and refinement of system [Woodcock and Morgan, 1990]. *Computation Tree Logic* (CTL) [Clarke and Emerson, 1981] Model modification, which has been developed in [Ramdani *et al.*, 2019; Martinez-Araiza and López-Mellado, 2016; Ding and Zhang, 2006], is an extension of refinement of system. This paper explore a method to compute the WSC of a property (a CTL formula) under a given model system that may be modified for guiding CTL Model modification.

It is known that the computing of WSC for code fragment S with respect to assertion Q requires S must terminate [Tremblay, 1996] due to it just concerns relation among input values and output values. However, in the case of model checking, it concerns properties about execution runs, which may not be terminate. It can be shown by the following example.

Example 1 A Beverage Vending Machine, which has been established as standard in the field of process calculi, can be described as a Kripke structure $\mathcal{M} = (S, R, L, s_0)$ on $V_a = \{select, pay, beer, soda\}$ with $S = \{s_0, s_1, s_2\}$, $R = \{(s_0, s_1), (s_0, s_2), (s_1, s_0), (s_2, s_0)\}$, $L(s_0) = \{select\}$, $L(s_1) = \{pay, soda\}$, $L(s_2) = \{pay, beer\}$ and s_0 is an initial state. Which means that when we in s_0 if we select soda and pay for it then we change to the s_1, else if we select beer and pay for it then we change to the s_2, after takeing out the drink we transform to s_0. This is somewhat different from that in [Baier and Katoen, 2008] for simply. For convenience, we use s for $select$, p for pay, b for $beer$ and so for $soda$. Let $\varphi = \text{AGAF}(p \wedge r)$, which means $p \wedge r$ will be satisfied infinite times in the structure, be a CTL formula with r expressing orange juice.

We can decide $(\mathcal{M}, s_0) \nvDash \varphi$ easily due to this structure do not contain the atom r. In order for $(\mathcal{M}, s_0)$ satisfy φ, we should find a condition ψ such that $(\mathcal{M}, s_0) \models \psi \supset \varphi$. As we know that if this condition exists, there are many conditions that satisfy the need. In this case, if we are clever enough to judge in advance the set of possible atomic propositions that make up the condition, then we can find this condition in the set only, and the smaller the set, the easier it is to work out the condition. In this paper, we always assume that the condition is a property defined on the specified atomic proposition set V, for our example $V = \{p, r\}$, and find the weakest property (that is, the weakest sufficient condition) satisfying the condition on the set. Finding this property is called discovering theorem by Lin in [Lin, 2018]. Inspired by the forgetting-based method to compute SNC (WSC) [Lin, 2001], in this paper, we tackle this problem by proposing a semantic forgetting for CTL.

However, as we have said that $\mathcal{M}$ is a Kripke structure, which needs to be converted into a logical formula (theory), that is the characteristic formula, which is a CTL formula proposed in [Browne *et al.*, 1988]. Thanks to we find the WSC in a set V of atoms, hence a set-based bisimulation between two K-structures (a Kripke structure with a state in it), V-bisimulation, and characteristic formula on V will be proposed in this paper. Our V-bisimulation is a more general bisimulation relation than others. On the one hand, the above set-based bisimulation is an extension of the bisimulation-equivalence of Definition 7.1 in [Baier and Katoen, 2008] in the sense that if $V = \mathcal{A}$ then our bisimulation is almost same to the latter. On the other hand, the above set-based bisimulation notion is similar to the state equivalence in [Browne *et al.*, 1988]. But it is different in the sense that ours is defined on K-structures, while it is defined on states in [Browne *et al.*, 1988]. What's more, the set-based bisimulation notion is also different from the state-based bisimulation notion of

Definition 7.7 in [Baier and Katoen, 2008], which is defined for states of a given K-structure.

As a logical notion, *forgetting* was first formally defined in propostional and first order logics by Lin and Reiter [Lin and Reiter, 1994]. Over the last twenty years, researchers have developed forgetting notions and theories not only in classical logic but also in other non-classical logic systems [Eiter and Kern-Isberner, 2019], such as forgetting in logic programs under answer set/stable model semantics [Zhang and Foo, 2006; Eiter and Wang, 2008; Wong, 2009; Wang *et al.*, 2012; Wang *et al.*, 2013], forgetting in description logic [Wang *et al.*, 2010; Lutz and Wolter, 2011; Zhao and Schmidt, 2017] and knowledge forgetting in modal logic [Zhang and Zhou, 2009; Su *et al.*, 2009; Liu and Wen, 2011; Fang *et al.*, 2019]. In application, forgetting has been used in planning [Lin, 2003], conflict solving [Lang and Marquis, 2010; Zhang *et al.*, 2005], createing restricted views of ontologies [Zhao and Schmidt, 2017], strongest and weakest definitions [Lang and Marquis, 2008], SNC (WSC) [Lin, 2001] and so on.

Though forgetting has been extensively investigated from various aspects of different logical systems. However, the existing forgetting method in propositional logic, answer set programming, description logic and modal logic are not directly applicable in CTL. Similar with that in [Zhang and Zhou, 2009], we research forgetting in CTL from the semantic forgetting point of view. And it is shown that our definition of forgetting satisfies those four postulates of forgetting.

The rest of the paper is organised as follows. Section 2 introduces the related notions for forgetting in CTL, including the syntax and semantics of CTL, the language we aimed for. A formal definition of concept forgetting and its properties for CTL follows in Section 3. Section 4 explores the relation between forgetting and SNC (WSC). From the point of view of model, we propose an algorithm for computing forgetting on CTL in Section 5. Finally, we conclude this paper.

2 Preliminaries

We start with some technical and notational preliminaries. Throughout this paper, we fix a finite set $\mathcal{A}$ of propositional variables (or atoms), and use V, V' for subsets of $\mathcal{A}$. In this part, we will introduce the structure we will use for CTL and syntax and semantic of CTL.

2.1 Model structure in CTL

In general, a transition system [1] is described as a *model structure* (or *Kripke structure*), and a model structure is a triple $\mathcal{M} = (S, R, L)$, where

- S is a finite nonempty set of states,
- $R \subseteq S \times S$ and, for each $s \in S$, there is $s' \in S$ such that $(s, s') \in R$,
- L is a labeling function $S \to 2^{\mathcal{A}}$.

[1] According to [Baier and Katoen, 2008], a *transition system* TS is a tuple $(S, Act, \to, I, AP, L)$ where (1) S is a set of states, (2) Act is a set of actions, (3) $\to \subseteq S \times \text{Act} \times S$ is a transition relation, (4) $I \subseteq S$ is a set of initial states, (5) AP is a set of atomic propositions, and (6) $L : S \to 2^{\text{AP}}$ is a labeling function.

We call a model structure $\mathcal{M}$ on a set V of atoms if $L : S \to 2^V$, *i.e.* , the labeling function L map every state to V (not the $\mathcal{A}$). A *path* π_{s_i} start from s_i of $\mathcal{M}$ is a infinite sequence of states $\pi_{s_i} = (s_i, s_{i+1}s_{i+2}, \dots)$, where for each j $(0 \leq i \leq j)$, $(s_j, s_{j+1}) \in R$. By $s' \in \pi_{s_i}$ we mean that s' is a state in the path π_{s_i}. A sate $s \in S$ is *initial* if for any state $s' \in S$, there is a path π_s s.t $s' \in \pi_s$. We denote this model structure as (S, R, L, s_0), where s_0 is initial.

For a given model structure (S, R, L, s_0) and $s \in S$, the *computation tree* $\text{Tr}_n^{\mathcal{M}}(s)$ of $\mathcal{M}$(or simply $\text{Tr}_n(s)$), that has depth n and is rooted at s, is recursively defined as [Browne *et al.*, 1988], for $n \geq 0$,

- $\text{Tr}_0(s)$ consists of a single node s with label s.
- $\text{Tr}_{n+1}(s)$ has as its root a node m with label s, and if $(s, s') \in R$ then the node m has a subtree $\text{Tr}_n(s')$.

By s_n we mean a nth level node of tree $\text{Tr}_m(s)$ $(m \geq n)$.

A K-*structure* (or K-*interpretation*) is a model structure $\mathcal{M} = (S, R, L, s_0)$ associating with a state $s \in S$, which is written as $(\mathcal{M}, s)$ for convenience in the following. In the case $s = s_0$ is an initial state of $\mathcal{M}$, the K-structure is *initial*.

2.2 Syntax and semantics of CTL

In the following we briefly review the basic syntax and semantics of the CTL [Clarke *et al.*, 1986]. The *signature* of $\mathcal{L}$ includes:

- a finite set of Boolean variables, called *atoms* of $\mathcal{L}$: $\mathcal{A}$;
- the classical connectives: $\bot$, $\vee$ and $\neg$;
- the path quantifiers: A and E;
- the temporal operators: X, F, G U and W, that means 'neXt state', 'some Future state', 'all future states (Globally)', 'Until' and 'Unless', respectively;
- parentheses: (and).

The *(existential normal form or ENF in short) formulas* of $\mathcal{L}$ are inductively defined via a Backus Naur form:

$$\phi ::= \bot \mid p \mid \neg\phi \mid \phi \vee \phi \mid \text{EX}\phi \mid \text{EG}\phi \mid \text{E}[\phi \text{ U } \phi] \quad (1)$$

where $p \in \mathcal{A}$. The formulas $\phi \wedge \psi$ and $\phi \to \psi$ are defined in a standard manner of propositional logic. The other form formulas of $\mathcal{L}$ are abbreviated using the forms of (1). The priorities for the CTL connectives are assumed to be (from the highest to the lowest):

$$\neg, \text{EX}, \text{EF}, \text{EG}, \text{AX}, \text{AF}, \text{AG} \prec \wedge \prec \vee \prec \text{EU}, \text{AU}, \text{EW}, \text{AW}, \to .$$

We are now in the position to define the semantics of $\mathcal{L}$. Let $\mathcal{M} = (S, R, L, s_0)$ be an model structure, $s \in S$ and ϕ a formula of $\mathcal{L}$. The *satisfiability* relationship between $\mathcal{M}, s$ and ϕ, written $(\mathcal{M}, s) \models \phi$, is inductively defined on the structure of ϕ as follows:

- $(\mathcal{M}, s) \not\models \bot$;
- $(\mathcal{M}, s) \models p$ iff $p \in L(s)$;
- $(\mathcal{M}, s) \models \phi_1 \vee \phi_2$ iff $(\mathcal{M}, s) \models \phi_1$ or $(\mathcal{M}, s) \models \phi_2$;
- $(\mathcal{M}, s) \models \neg\phi$ iff $(\mathcal{M}, s) \not\models \phi$;
- $(\mathcal{M}, s) \models \text{EX}\phi$ iff $(\mathcal{M}, s_1) \models \phi$ for some $s_1 \in S$ and $(s, s_1) \in R$;

- $(\mathcal{M}, s) \models \text{EG}\phi$ iff $\mathcal{M}$ has a path $(s_1 = s, s_2, \ldots)$ such that $(\mathcal{M}, s_i) \models \phi$ for each $i \geq 1$;
- $(\mathcal{M}, s) \models \text{E}[\phi_1 \text{U} \phi_2]$ iff $\mathcal{M}$ has a path $(s_1 = s, s_2, \ldots)$ such that, for some $i \geq 1$, $(\mathcal{M}, s_i) \models \phi_2$ and $(\mathcal{M}, s_j) \models \phi_1$ for each $1 \leq j < i$.

Similar to the work in [Browne *et al.*, 1988; Bolotov, 1999], only initial K-structures are considered to be candidate models in the following, unless explicitly stated. Formally, an initial K-structure $\mathcal{K}$ is a *model* of a formula ϕ whenever $\mathcal{K} \models \phi$. We denote *Mod*(ϕ) the set of models of ϕ. The formula ϕ is *satisfiable* if *Mod*$(\phi) \neq \emptyset$. Since the states in model structure is finite, *Mod*(ϕ) is finite for any formula ϕ.

Let ϕ_1 and ϕ_2 be two formulas. By $\phi_1 \models \phi_2$ we denote *Mod*$(\phi_1) \subseteq$ *Mod*(ϕ_2). By $\phi_1 \equiv \phi_2$ we mean $\phi_1 \models \phi_2$ and $\phi_2 \models \phi_1$. In this case ϕ_1 is *equivalent* to ϕ_2. By *Var*(ϕ_1) we mean the set of atoms occurring in ϕ_1. ϕ_1 is V*-irrelevant*, written IR(ϕ_1, V), if there is a formula ψ with *Var*$(\psi) \cap V = \emptyset$ such that $\phi_1 \equiv \psi$.

3 Forgetting in CTL

In this section, we will define the forgetting in CTL by V-bisimulation, set-based bisimulations. Besides, some properties of forgetting are also explored. For convenience, let $\mathcal{M} = (S, R, L, s_0)$, $\mathcal{M}' = (S', R', L', s_0')$ and $\mathcal{K}_i = (\mathcal{M}_i, s_i)$ with $\mathcal{M}_i = (S_i, R_i, L_i, s_0^i)$, $s_i \in S_i$ and i is an integer.

3.1 Set-based bisimulation

To present a formal definition of forgetting, we need the concept of V-bisimulation. Inspired by the notion of bisimulation in [Browne *et al.*, 1988], we define the relations $\mathcal{B}_0, \mathcal{B}_1, \ldots$ between K-structures on V as follows: let $\mathcal{K}_i = (\mathcal{M}_i, s_i)$ with $i \in \{1, 2\}$,

- $(\mathcal{K}_1, \mathcal{K}_2) \in \mathcal{B}_0$ if $L_1(s_1) - V = L_2(s_2) - V$;
- for $n \geq 0$, $(\mathcal{K}_1, \mathcal{K}_2) \in \mathcal{B}_{n+1}$ if
  - $(\mathcal{K}_1, \mathcal{K}_2) \in \mathcal{B}_0$,
  - for every $(s_1, s_1') \in R_1$, there is $(s_2, s_2') \in R_2$ such that $(\mathcal{K}_1', \mathcal{K}_2') \in \mathcal{B}_n$, and
  - for every $(s_2, s_2') \in R_2$, there is $(s_1, s_1') \in R_1$ such that $(\mathcal{K}_1', \mathcal{K}_2') \in \mathcal{B}_n$,

  where $\mathcal{K}_i' = (\mathcal{M}_i, s_i')$ with $i \in \{1, 2\}$.

Now, we define the notion of V-bisimulation between K-structures:

Definition 1 (V-bisimulation) *Let $V \subseteq \mathcal{A}$. The V-bisimular relation $\mathcal{B}$ between K-structures is defined as:*

$$(\mathcal{K}_1, \mathcal{K}_2) \in \mathcal{B} \textit{ if and only if } (\mathcal{K}_1, \mathcal{K}_2) \in \mathcal{B}_i \textit{ for all } i \geq 0.$$

In this case, $\mathcal{K}_1$ and $\mathcal{K}_2$ are called V-bisimular.

Proposition 1 *Let $V \subseteq \mathcal{A}$ and $\mathcal{K}_i = (\mathcal{M}_i, s_i)$ $(i = 1, 2)$ be K-structures. Then $(\mathcal{K}_1, \mathcal{K}_2) \in \mathcal{B}$ if and only if*

(i) $L_1(s_1) - V = L_2(s_2) - V$,

(ii) for every $(s_1, s_1') \in R_1$, there is $(s_2, s_2') \in R_2$ such that $(\mathcal{K}_1', \mathcal{K}_2') \in \mathcal{B}$, and

(iii) for every $(s_2, s_2') \in R_2$, there is $(s_1, s_1') \in R_1$ such that $(\mathcal{K}_1', \mathcal{K}_2') \in \mathcal{B}$,

where $\mathcal{K}_i' = (\mathcal{M}_i, s_i')$ with $i \in \{1, 2\}$.

Two pathes $\pi_i = (s_{i,1}, s_{i,2}, \ldots)$ of $\mathcal{M}_i$ with $i \in \{1, 2\}$ are V*-bisimular* if

$$(\mathcal{K}_{1,j}, \mathcal{K}_{2,j}) \in \mathcal{B} \text{ for every } j \geq 0$$

where $\mathcal{K}_{i,j} = (\mathcal{M}_i, s_{i,j})$.

In the following we abbreviated $(\mathcal{K}_1, \mathcal{K}_2) \in \mathcal{B}$ by $(s_1, s_2) \in \mathcal{B}$ when the underlying model structures of states s_1 and s_2 are clear from their contexts or there is no confusion. The V-bisimular relation is uniformly abbreviated as $\leftrightarrow_V$ for convenience.

Lemma 1 *The relation $\leftrightarrow_V$ is an equivalence relation.*

Besides, we have the following properties:

Proposition 2 *Let $i \in \{1, 2\}$, $V_1, V_2 \subseteq \mathcal{A}$, s_i's be two states and π_i's be two pathes, and $\mathcal{K}_i = (\mathcal{M}_i, s_i)$ $(i = 1, 2, 3)$ be K-structures such that $\mathcal{K}_1 \leftrightarrow_{V_1} \mathcal{K}_2$ and $\mathcal{K}_2 \leftrightarrow_{V_2} \mathcal{K}_3$. Then:*

(i) $s_1' \leftrightarrow_{V_i} s_2'$ $(i = 1, 2)$ *implies* $s_1' \leftrightarrow_{V_1 \cup V_2} s_2'$;

(ii) $\pi_1' \leftrightarrow_{V_i} \pi_2'$ $(i = 1, 2)$ *implies* $\pi_1' \leftrightarrow_{V_1 \cup V_2} \pi_2'$;

(iii) for each path π_{s_1} of $\mathcal{M}_1$ there is a path π_{s_2} of $\mathcal{M}_2$ such that $\pi_{s_1} \leftrightarrow_{V_1} \pi_{s_2}$, and vice versa;

(iv) $\mathcal{K}_1 \leftrightarrow_{V_1 \cup V_2} \mathcal{K}_3$;

(v) If $V_1 \subseteq V_2$ then $\mathcal{K}_1 \leftrightarrow_{V_2} \mathcal{K}_2$.

Intuitively, if two K-structures are V-bisimular, then they satisfy the same formula φ that dose not contain any atoms in V, *i.e.* IR(φ, V).

Theorem 1 *Let $V \subseteq \mathcal{A}$, $\mathcal{K}_i$ $(i = 1, 2)$ be two K-structures such that $\mathcal{K}_1 \leftrightarrow_V \mathcal{K}_2$ and ϕ a formula with IR(ϕ, V). Then $\mathcal{K}_1 \models \phi$ if and only if $\mathcal{K}_2 \models \phi$.*

Let $V \subseteq \mathcal{A}$, $\mathcal{M}_i$ $(i = 1, 2)$ be model structures. A computation tree $\text{Tr}_n(s_1)$ of $\mathcal{M}_1$ is V*-bisimular* to a computation tree $\text{Tr}_n(s_2)$ of $\mathcal{M}_2$, written $(\mathcal{M}_1, \text{Tr}_n(s_1)) \leftrightarrow_V (\mathcal{M}_2, \text{Tr}_n(s_2))$ (or simply $\text{Tr}_n(s_1) \leftrightarrow_V \text{Tr}_n(s_2)$), if

- $L_1(s_1) - V = L_2(s_2) - V$,
- for every subtree $\text{Tr}_{n-1}(s_1')$ of $\text{Tr}_n(s_1)$, $\text{Tr}_n(s_2)$ has a subtree $\text{Tr}_{n-1}(s_2')$ such that $\text{Tr}_{n-1}(s_1') \leftrightarrow_V \text{Tr}_{n-1}(s_2')$, and

Please note that the last condition in the above definition hold trivially for $n = 0$.

Proposition 3 *Let $V \subseteq \mathcal{A}$ and $(\mathcal{M}_i, s_i)$ $(i = 1, 2)$ be two K-structures. Then*

$$(s_1, s_2) \in \mathcal{B}_n \textit{ iff } Tr_j(s_1) \leftrightarrow_V Tr_j(s_2) \textit{ for every } 0 \leq j \leq n.$$

This means that $\text{Tr}_j(s_1) \leftrightarrow_V \text{Tr}_j(s_2)$ for all $j \geq 0$ if $s_1 \leftrightarrow_V s_2$, otherwise there is some k such that $\text{Tr}_k(s_1)$ and $\text{Tr}_k(s_2)$ are not V-bisimular.

Proposition 4 *Let $V \subseteq \mathcal{A}$, $\mathcal{M}$ be a model structure and $s, s' \in S$ such that $(s, s') \notin \mathcal{B}$. There exists a least number k such that $Tr_k(s)$ and $Tr_k(s')$ are not V-bisimular.*

In this case the model structure $\mathcal{M}$ is called V-*distinguishable* (by states s and s' at the least depth k), which is denoted by $\text{dis}_V(\mathcal{M}, s, s', k)$. It is evident that $\text{dis}_V(\mathcal{M}, s, s', k)$ implies $\text{dis}_V(\mathcal{M}, s, s', k')$ whenever $k' \geq$

k. The V-*characterization number* of $\mathcal{M}$, written $ch(\mathcal{M}, V)$, is defined as

$$ch(\mathcal{M}, V) = \begin{cases} \max\{k \mid s, s' \in S \,\&\, \text{dis}_V(\mathcal{M}, s, s', k)\}, & \mathcal{M} \text{ is } V\text{-distinguishable}; \\ \min\{k \mid \mathcal{B}_k = \mathcal{B}\}, & \text{otherwise.} \end{cases}$$

Now we give the formal definition of forgetting in CTL from the semantic forgetting point view.

Definition 2 (Forgetting) *Let $V \subseteq \mathcal{A}$ and ϕ a formula. A formula ψ with $Var(\psi) \cap V = \emptyset$ is a* result of forgetting V from ϕ, *if*

$$Mod(\psi) = \{\mathcal{K} \textit{ is initial} \mid \exists \mathcal{K}' \in Mod(\phi) \,\&\, \mathcal{K}' \leftrightarrow_V \mathcal{K}\}. \quad (2)$$

Note that if both ψ and ψ' are results of forgetting V from ϕ then $Mod(\psi) = Mod(\psi')$, *i.e.* , ψ and ψ' have the same models. In the sense of equivalence the forgetting result is unique (up to equivalence).

Intuitively, forgetting an atom results in a weaker theory which entails the same set of formulae that are irrelevant to the atom. To present the representation property of forgetting in CTL and compute WSC (SNC) under an initial K-structure, we will give the Characterize formula of an initial K-structure on V in the next subsection.

3.2 Characterize formula of initial K-structure

Given a set $V \subseteq \mathcal{A}$, we can define a formula φ of V (that is $Var(\varphi) \subseteq V$) in CTL to equivalent uniquely describe a computation tree.

Definition 3 *Let $V \subseteq \mathcal{A}$, $\mathcal{M} = (S, R, L, s_0)$ be a model structure and $s \in S$. The* characterize formula *of the computation tree $Tr_n(s)$ on V, written $\mathcal{F}_V(Tr_n(s))$, is defined recursively as:*

$$\mathcal{F}_V(Tr_0(s)) = \bigwedge_{p \in V \cap L(s)} p \wedge \bigwedge_{q \in V - L(s)} \neg q,$$

$$\mathcal{F}_V(Tr_{k+1}(s)) = \bigwedge_{(s,s') \in R} \text{EX}T(s') \wedge \text{AX} \bigvee_{(s,s') \in R} T(s') \wedge \mathcal{F}_V(Tr_0(s))$$

for $k \geq 0$, where $T(s') = \mathcal{F}_V(Tr_k(s'))$.

The characterize formula of a computation tree formally exhibit the context of each node on V (atoms are true at this node if they are in V, else false) and the temporal relation between states recursively. In this way, we know:

Lemma 2 *Let $V \subseteq \mathcal{A}$, $\mathcal{M} = (S, R, L, s_0)$ and $\mathcal{M}' = (S', R', L', s'_0)$ be two model structures, $s \in S$, $s' \in S'$ and $n \geq 0$. If $Tr_n(s) \leftrightarrow_{\overline{V}} Tr_n(s')$, then $\mathcal{F}_V(Tr_n(s)) \equiv \mathcal{F}_V(Tr_n(s'))$.*

Let $s' = s$, it shows that for any formula φ of V, if φ is a characterize formula of $\text{Tr}_n(s)$ then $\varphi \equiv \mathcal{F}_V(\text{Tr}_n(s))$.

Let $V \subseteq \mathcal{A}$, $\mathcal{K} = (\mathcal{M}, s_0)$ be an initial K-structure and $T(s') = \mathcal{F}_V(\text{Tr}_c(s'))$. The *characterizing formula* of $\mathcal{K}$ on V, written $\mathcal{F}_V(\mathcal{M}, s_0)$ (or $\mathcal{F}_V(\mathcal{K})$), is defined as the conjunction of the following formulas:

$\mathcal{F}_V(\text{Tr}_c(s_0))$, and for each $s \in S$

$$\text{AG}\left(\mathcal{F}_V(\text{Tr}_c(s)) \rightarrow \bigwedge_{(s,s') \in R} \text{EX}T(s') \wedge \text{AX} \bigvee_{(s,s') \in R} T(s')\right)$$

where $c = ch(\mathcal{M}, V)$. It is apparent that $\text{IR}(\mathcal{F}_V(\mathcal{M}, s_0), \overline{V})$.

The following example show how to compute characterizing formula:

Example 2 Let $\mathcal{K} = (\mathcal{M}, s_0)$ with $\mathcal{M} = (S, R, L, s_0)$ be a initial K-structure, in which $S = \{s_0, s_1, s_2\}$, $R = \{(s_0, s_1), (s_0, s_2), (s_1, s_0), (s_2, s_0)\}$, $L(s_0) = \{a\}$, $L(s_1) = \{a, c\}$ and $L(s_2) = \{b, c\}$. Let $V = \{a, b\}$, compute the characterizing formula of $\mathcal{K}$ on V.

It is apparent that $\text{Tr}_0(s_0) \leftrightarrow_{\overline{V}} \text{Tr}_0(s_1)$ due to $L(s_0) - \overline{V} = L(s_1) - \overline{V}$, $\text{Tr}_1(s_0) \not\leftrightarrow_{\overline{V}} \text{Tr}_1(s_1)$ due to there is $(s_0, s_2) \in R$ such that for any $(s_1, s') \in R$ (there is only one immediate successor $s' = s_0$) there is $L(s_2) - \overline{V} \neq L(s') - \overline{V}$. Hence, we have that $\mathcal{M}$ is $\overline{V}$-distinguished by state s_0 and s_1 at the least depth 1, *i.e.* $\text{dis}_{\overline{V}}(\mathcal{M}, s_0, s_1, 1)$. Similarly, we have $\text{dis}_{\overline{V}}(\mathcal{M}, s_0, s_2, 0)$ and $\text{dis}_{\overline{V}}(\mathcal{M}, s_1, s_2, 0)$. Therefore, $ch(\mathcal{M}, \overline{V}) = \max\{k \mid s, s' \in S \,\&\, \text{dis}_{\overline{V}}(\mathcal{M}, s, s', k)\} = 1$. Then we have:

$\mathcal{F}_V(\text{Tr}_0(s_0)) = a \wedge \neg b$,
$\mathcal{F}_V(\text{Tr}_0(s_1)) = a \wedge \neg b$,
$\mathcal{F}_V(\text{Tr}_0(s_2)) = b \wedge \neg a$,
$\mathcal{F}_V(\text{Tr}_1(s_0)) = \text{EX}(a \wedge \neg b) \wedge \text{EX}(b \wedge \neg a) \wedge \text{AX}((a \wedge \neg b) \vee (b \wedge \neg a)) \wedge (a \wedge \neg b)$,
$\mathcal{F}_V(\text{Tr}_1(s_1)) = \text{EX}(a \wedge \neg b) \wedge \text{AX}(a \wedge \neg b) \wedge (a \wedge \neg b)$,
$\mathcal{F}_V(\text{Tr}_1(s_2)) = \text{EX}(a \wedge \neg b) \wedge \text{AX}(a \wedge \neg b) \wedge (b \wedge \neg a)$.

Then it is easy to obtain $\mathcal{F}_V(\mathcal{M}, s_0)$.

Lemma 3 *Let φ be a formula. We have*

$$\varphi \equiv \bigvee_{(\mathcal{M}, s_0) \in Mod(\varphi)} \mathcal{F}_{\mathcal{A}}(\mathcal{M}, s_0). \quad (3)$$

It follows that any CTL formula can be described by the disjunction of the characterizing formulas of all the models of itself due to the number of models of a CTL formula is finite.

Theorem 2 *Let $V \subseteq \mathcal{A}$, $\mathcal{M} = (S, R, L, s_0)$ be a model structure with initial state s_0 and $\mathcal{M}' = (S', R', L', s'_0)$ be a model structure with initial state s'_0. Then*

$(\mathcal{M}', s'_0) \models \mathcal{F}_V(\mathcal{M}, s_0)$ *if and only if* $(\mathcal{M}, s_0) \leftrightarrow_{\overline{V}} (\mathcal{M}', s'_0)$.

By the following theorem we also have that given a set $V \subseteq \mathcal{A}$, the characterizing formula of an initial K-structure is equivalent uniquely describe this initial K-structure on V.

Theorem 3 *Let $V \subseteq \mathcal{A}$, $\mathcal{M} = (S, R, L, s_0)$ a model structure with initial state s_0 and $\mathcal{M}' = (S', R', L', s'_0)$ a model structure with initial state s'_0. If $(\mathcal{M}, s_0) \leftrightarrow_{\overline{V}} (\mathcal{M}', s'_0)$ then $\mathcal{F}_V(\mathcal{M}, s_0) \equiv \mathcal{F}_V(\mathcal{M}', s'_0)$.*

3.3 Semantic properties of forgetting in CTL

In this subsection we study essential semantic properties of forgetting. We will first show that our forgetting satisfy these postulates [Zhang and Zhou, 2009] that precisely characterize the semantics of forgetting. We then discuss other semantic properties of forgetting.

By Lemma 3 and Definition 2, the result ψ of ϕ forget the set V of atoms always exists, which is equivalent to

$$\bigvee_{\mathcal{K} \in \{\mathcal{K}' \text{ is an initial interpretation} \mid \exists \mathcal{K}'' \in Mod(\phi) \,\&\, \mathcal{K}'' \leftrightarrow_V \mathcal{K}'\}} \mathcal{F}_{\overline{V}}(\mathcal{K}).$$

For this reason, the forgetting result is denoted by $\mathrm{F}_{\text{CTL}}(\phi, V)$.

In the case ψ is a result of forgetting V from ϕ, there are usually some expected properties (called *postulates*: (**W**), (**PP**), (**NP**) and (**IR**)) for it [Zhang and Zhou, 2009].

Theorem 4 *(Representation theorem). Let φ and ψ be two formulas and $V \subseteq \mathcal{A}$. Then the following statements are equivalent:*

(i) $\psi \equiv \mathrm{F}_{\text{CTL}}(\varphi, V)$*,*

(ii) $\psi \equiv \{\phi | \varphi \models \phi \& IR(\phi, V)\}$*,*

*(iii) Postulates (**W**), (**PP**), (**NP**) and (**IR**) hold.*

We can see from this theorem that our definition of forgetting under CTL is close, *i.e.* for any CTL formula the result of forgetting is also a CTL formula, and captures the four postulates that forgetting should satisfy.

Lemma 4 *Let φ and α be two CTL formulae and $q \in \overline{Var(\varphi \cup \{\alpha\})}$. Then $\mathrm{F}_{CTL}(\varphi \cup \{q \leftrightarrow \alpha\}, q) \equiv \varphi$.*

Proposition 5 *Let φ be a formula, V a set of atoms and p an atom such that $p \notin V$. Then:*

$$\mathrm{F}_{\text{CTL}}(\varphi, \{p\} \cup V) \equiv \mathrm{F}_{\text{CTL}}(\mathrm{F}_{\text{CTL}}(\varphi, p), V).$$

This means that the result of forgetting V from φ can be obtained by forgetting atom in V one by one. Similarly, a consequence of the previous proposition is:

Corollary 5 *Let φ be a formula and $V_i \subseteq \mathcal{A}$ $(i = 1, 2)$. Then:*

$$\mathrm{F}_{\text{CTL}}(\varphi, V_1 \cup V_2) \equiv \mathrm{F}_{\text{CTL}}(\mathrm{F}_{\text{CTL}}(\varphi, V_1), V_2).$$

The following results, which are satisfied in both classical proposition logic and modal logic **S5**, further illustrate other essential semantic properties of forgetting.

Proposition 6 *Let φ, φ_i, ψ_i $(i = 1, 2)$ be formulas and $V \subseteq \mathcal{A}$. We have*

(i) $\mathrm{F}_{\text{CTL}}(\varphi, V)$ *is satisfiable iff φ is;*

(ii) If $\varphi_1 \equiv \varphi_2$, then $\mathrm{F}_{\text{CTL}}(\varphi_1, V) \equiv \mathrm{F}_{\text{CTL}}(\varphi_2, V)$;

(iii) If $\varphi_1 \models \varphi_2$, then $\mathrm{F}_{\text{CTL}}(\varphi_1, V) \models \mathrm{F}_{\text{CTL}}(\varphi_2, V)$;

(iv) $\mathrm{F}_{\text{CTL}}(\psi_1 \vee \psi_2, V) \equiv \mathrm{F}_{\text{CTL}}(\psi_1, V) \vee \mathrm{F}_{\text{CTL}}(\psi_2, V)$*;*

(v) $\mathrm{F}_{\text{CTL}}(\psi_1 \wedge \psi_2, V) \models \mathrm{F}_{\text{CTL}}(\psi_1, V) \wedge \mathrm{F}_{\text{CTL}}(\psi_2, V)$*;*

Another interest result is that the forgetting of $PT\varphi$ ($P \in \{\mathrm{E}, \mathrm{A}\}$, $T \in \{\mathrm{F}, \mathrm{X}\}$) on $V \subseteq \mathcal{A}$ can be computed by $PT\mathrm{F}_{\text{CTL}}(\varphi, V)$. This give a convenient method to compute forgetting.

Proposition 7 *Let $V \subseteq \mathcal{A}$ and ϕ a formula.*

(i) $\mathrm{F}_{\text{CTL}}(\mathrm{AX}\phi, V) \equiv \mathrm{AX}\mathrm{F}_{\text{CTL}}(\phi, V)$.

(ii) $\mathrm{F}_{\text{CTL}}(\mathrm{EX}\phi, V) \equiv \mathrm{EX}\mathrm{F}_{\text{CTL}}(\phi, V)$.

(iii) $\mathrm{F}_{\text{CTL}}(\mathrm{AF}\phi, V) \equiv \mathrm{AF}\mathrm{F}_{\text{CTL}}(\phi, V)$.

(iv) $\mathrm{F}_{\text{CTL}}(\mathrm{EF}\phi, V) \equiv \mathrm{EF}\mathrm{F}_{\text{CTL}}(\phi, V)$.

3.4 Main complexity

In the following we consider the main complexities of reasoning problems on forgetting in CTL.

Proposition 8 *Let $\mathcal{M}, s_0$ be an initial* K*-structure, φ be a CTL formula and V a set of atoms. Deciding whether $\mathcal{M}, s_0$ is a model of $\mathrm{F}_{CTL}(\varphi, V)$ is NP-complete.*

By this proposition, we have:

Theorem 6 *Let φ and ψ be two CTL(*AF*) (a fragment of CTL, in which each formula contains only* AF *temporal connective) formulas and V a set of atoms. Then we have the results:*

(i) deciding if $\mathrm{F}_{CTL}(\varphi, V) \models \psi$ is co-NP-complete,

(ii) deciding if $\psi \models \mathrm{F}_{CTL}(\varphi, V)$ is Π_2^P-complete,

(iii) deciding if $\mathrm{F}_{CTL}(\varphi, V) \models \mathrm{F}_{CTL}(\psi, V)$ is Π_2^P-complete.

The theorem implies:

Corollary 7 *Let φ and ψ be two CTL(*AF*) formulas and V a set of atoms. Then*

(i) deciding if $\psi \equiv \mathrm{F}_{CTL}(\varphi, V)$ is Π_2^P-complete,

(ii) deciding if $\mathrm{F}_{CTL}(\varphi, V) \equiv \varphi$ is co-NP-complete,

(iii) deciding if $\mathrm{F}_{CTL}(\varphi, V) \equiv \mathrm{F}_{CTL}(\psi, V)$ is Π_2^P-complete.

4 SNC and WSC

In this section, we will give the definition of SNC (WSC) and show that the SNC (WSC) of a specification (a CTL formula) under a given initial K-structure and set V of atoms can be obtained from forgetting in CTL. The SNC (WSC) of a proposition will be given at first:

Definition 4 (sufficient and necessary condition) *Let ϕ be a formula or an initial* K*-structure, ψ be a formula, $V \subseteq Var(\phi)$, $q \in Var(\phi) - V$ and $Var(\psi) \subseteq V$.*

- *ψ is a* necessary condition *(NC in short) of q on V under ϕ if $\phi \models q \rightarrow \psi$.*
- *ψ is a* sufficient condition *(SC in short) of q on V under ϕ if $\phi \models \psi \rightarrow q$.*
- *ψ is a* strongest necessary condition *(SNC in short) of q on V under ϕ if it is a NC of q on V under ϕ and $\phi \models \psi \rightarrow \psi'$ for any NC ψ' of q on V under ϕ.*
- *ψ is a* weakest sufficient condition *(SNC in short) of q on V under ϕ if it is a SC of q on V under ϕ and $\phi \models \psi' \rightarrow \psi$ for any SC ψ' of q on V under ϕ.*

Note that if both ψ and ψ' are SNC (WSC) of q on V under ϕ then $Mod(\psi) = Mod(\psi')$, *i.e.* ψ and ψ' have the same models. In the sense of equivalence the SNC (WSC) is unique (up to equivalence).

Proposition 9 *(**dual**) Let V, q, φ and ψ are the ones in Definition 4. The ψ is a SNC (WSC) of q on V under φ iff $\neg\psi$ is a WSC (SNC) of $\neg q$ on V under φ.*

This show that the SNC and WSC are in fact dual conditions. Under the dual property, we can consider the SNC party only in sometimes, while the WSC part can be talked similarly.

For the case of formula, we have that the SCN (WSC) of any formula can be defined as follows:

Definition 5 *Let Γ be a formula or an initial* K*-structure, α be a formula and $P \subseteq (Var(\Gamma) \cup Var(\alpha))$. A formula φ of P is said to be a NC (SC) of α on P under Γ iff $\Gamma \models \alpha \rightarrow \varphi$. It is said to be a SNC (WSC) if it is a NC (SC), and for any other NC (SC) φ', we have that $\Gamma \models \varphi \rightarrow \varphi'$ ($\Gamma \models \varphi' \rightarrow \varphi$).*

It is seems that the SNC (WSC) of any formula can be obtained by changing to that of a proposition. Formally:

Proposition 10 *Let Γ be a formula, P, and α be as in Definition 5. A formula φ of P is the SNC (WSC) of α on P under Γ iff it is the SNC (WSC) of q on P under $\Gamma' = \Gamma \cup \{q \equiv \alpha\}$, where q is a new proposition not in Γ and α.*

We propose the theorem of computing the SNC (WSC) of an atom due to the SNC (WSC) of a formula can be changed to the SNC (WSC) of an atom by Proposition 10.

Theorem 8 *Let φ be a formula, $V \subseteq Var(\varphi)$ and $q \in Var(\varphi) - V$.*

(i) $\mathrm{F}_{\text{CTL}}(\varphi \wedge q, (Var(\varphi) \cup \{q\}) - V)$ *is a SNC of q on V under φ.*

(ii) $\neg\mathrm{F}_{\text{CTL}}(\varphi \wedge \neg q, (Var(\varphi) \cup \{q\}) - V)$ *is a WSC of q on V under φ.*

As we have said before that any initial K-structure can be characterized by a CTL formula, we can obtain the SNC (WSC) of an initial K-structure for satisfy some needed property (formula) by forgetting.

Theorem 9 *Let $\mathcal{K} = (\mathcal{M}, s)$ be an initial* K*-structure with $\mathcal{M} = (S, R, L, s_0)$ on the finite set $\mathcal{A}$ of atoms, $V \subseteq \mathcal{A}$ and $q \in V'$ ($V' = \mathcal{A} - V$). Then:*

(i) the SNC of q on V under $\mathcal{K}$ is $\mathrm{F}_{\text{CTL}}(\mathcal{F}_{\mathcal{A}}(\mathcal{K}) \wedge q, V')$.

(ii) the WSC of q on V under $\mathcal{K}$ is $\neg\mathrm{F}_{\text{CTL}}(\mathcal{F}_{\mathcal{A}}(\mathcal{K}) \wedge \neg q, V')$.

Example 3 For the Example 1, the WSC of φ on V under $\mathcal{K} = (\mathcal{M}, s_0)$ is $\neg\mathrm{F}_{\text{CTL}}(\mathcal{F}_{\mathcal{A}}(\mathcal{K}) \wedge (q \equiv \varphi) \wedge \neg q, \mathcal{A} \setminus V)$.

5 Algorithm to compute forgetting

To compute the forgetting in CTL, we propose a model-based method in this part. Literally speaking, the model-based method means that we can obtain the result of forgetting in CTL by obtain all the possible finite models of this result. By the definition of forgetting in CTL, the set of models of the result of forgetting is also a finite set of initial K-structures.

Then we have the following model-based Algorithm 1 to computing the forgetting under CTL. By Lemma 3 and Theorem 3 we can prove the correctness of this algorithm.

Example 4 Let $\varphi = \text{AGAF}(p \wedge r)$, $\mathcal{A} = \{p, r\}$ and $V = \{r\}$. For convenience, we use the label of a state to express the state and then remove the label function in a model structure. Let $\mathcal{M}_1 = (\{\{p, r\}\}, \{(\{p, r\}, \{p, r\})\}, \{p, r\})$ and $\mathcal{M}_2 = (\{\emptyset, \{p, r\}\}, \{(\emptyset, \{p, r\}), (\{p, r\}, \{p, r\})\}, \emptyset)$. The set of models of$\varphi$ is $Mod(\varphi) = \{(\mathcal{M}_1, \{p\}), (\mathcal{M}_2, \emptyset), \dots\}$.

Let $\mathcal{M}_1' = (\{\{p\}\}, \{(\{p\}, \{p\})\}, \{p\})$ and $\mathcal{M}_2' = (\{\emptyset, \{p\}\}, \{(\emptyset, \{p\}), (\{p\}, \{p\})\}, \emptyset)$ Then we can obtain all the possible initial K-structure that is a model of $\mathrm{F}_{\text{CTL}}(\varphi, V)$, *i.e.* $Mod(\mathrm{F}_{\text{CTL}}(\varphi, V)) = \{\mathcal{K}_1 = (\mathcal{M}_1', \{p\}), \mathcal{K}_2 = (\mathcal{M}_2', \emptyset), \dots\}$.

Algorithm 1: Model-based: Computing forgetting

```
Input: A CTL formula φ and a set V of atoms
Output: F_CTL(φ, V)
T = ∅ // the set of models of φ ;
T' = ∅ // the set of possible initial K-structures ;
n = |A|;
for i=1, ..., 2^n do
    for s_j ∈ {s_1, ..., s_i} do
        Let s_j be an initial state, construct
        M = (S, R, L, s_j) by the definition of model
        structure with S = {s_1, ..., s_i};
        if for each K ∈ T', (M, s_j) ↮_{Var(φ)} K then
            Let T' := T' ∪ {(M, s_j)};
        end
    end
    for (M, s_0) ∈ T' do
        if (M, s_0) ⊨ φ then
            T := T ∪ {(M, s_0)};
        end
    end
end
return ⋁_{(M', s'_0) ∈ T} F_{V̄}(M', s'_0).
```

Let $V' = \{p\}$, then $\mathcal{F}_{V'}(\mathcal{K}_1) = p \wedge \text{AG}(p \supset \text{EX}p \wedge \text{AX}p)$, and $\mathcal{F}_{V'}(\mathcal{K}_2) = \neg p \wedge \text{AG}(p \supset \text{EX}\neg p \wedge \text{AX}\neg p) \wedge \text{AG}(\neg p \supset \text{EX}p \wedge \text{AX}p)$. Similarly, we can obtain the characteristic formula of other models and then the $\mathrm{F}_{\text{CTL}}(\varphi, V)$.

Proposition 11 *Let φ be a CTL formula and $V \subseteq \mathcal{A}$. The time complexity of Algorithm 1 is $O(2^{2^m})$ and the space complexity is $O(2^{2m})$, where $|\mathcal{A}| = m$.*

6 Concluding Remark

Based on the proposed V-bisimulation between K-structures, forgetting in CTL and characteristic formula on V on an initial K-structure $\mathcal{K}$, a method compute the WSC (SNC) of a property φ (a CTL formula) on $\mathcal{K}$ and V has been introduced by computing forgetting in CTL. Besides, we have shown that the CTL system is close under our definition of forgetting, and this definition satisfies those four postulates of forgetting. As we have said the complexity of Algorithm 1 is $O(2^{2^m})$ (very inefficient), a future work is to find an efficient algorithm to compute forgetting in CTL and then WSC (SNC).

7 Proof

Proposition 1.

Proof: ($\Rightarrow$) (a) It is apparent that $L_1(s_1) - V = L_2(s_2) - V$; (b) $(\mathcal{K}_1, \mathcal{K}_2) \in \mathcal{B}$ iff $(\mathcal{K}_1, \mathcal{K}_2) \in \mathcal{B}_i$ for all $i \geq 0$, then for each $(s_1, s_1') \in R_1$, there is a $(s_2, s_2') \in R_2$ such that $(\mathcal{K}_1', \mathcal{K}_2') \in \mathcal{B}_{i-1}$ for all $i > 0$ and then $L_1(s_1') - V = L_2(s_2') - V$. Therefore, $(\mathcal{K}_1', \mathcal{K}_2') \in \mathcal{B}$. (c) This is similar with (b).

($\Leftarrow$) (a) $L_1(s_1) - V = L_2(s_2) - V$ implies that $(s_1, s_2) \in \mathcal{B}_0$; (b) Condition (ii) implies that for every $(s_1, s_1') \in R_1$, there is $(s_2, s_2') \in R_2$ such that $(\mathcal{K}_1', \mathcal{K}_2') \in \mathcal{B}_i$ for all $i \geq 0$; (c) Condition (iii) implies that

for every $(s_2, s_2') \in R_2$, there is $(s_1, s_1') \in R_1$ such that $(\mathcal{K}_1', \mathcal{K}_2') \in \mathcal{B}_i$ for all $i \geq 0$
$\Rightarrow (\mathcal{K}_1, \mathcal{K}_2) \in \mathcal{B}_i$ for all $i \geq 0$
$\Rightarrow (\mathcal{K}_1, \mathcal{K}_2) \in \mathcal{B}$. ∎

Lemma 5 *Let $\mathcal{B}_0, \mathcal{B}_1, \dots$ be the ones in the definition of section 3.1. Then, for each $i \geq 0$,*

(i) $\mathcal{B}_{i+1} \subseteq \mathcal{B}_i$;

(ii) there is the leat number $k \geq 0$ such that $\mathcal{B}_{k+1} = \mathcal{B}_k$;

(iii) $\mathcal{B}_i$ is reflexive, symmetric and transitive.

Proof: (i) Base: it is clear for $i = 0$ by the above definition.

Step: suppose it holds for $i = n$, *i.e.* $\mathcal{B}_{n+1} \subseteq \mathcal{B}_n$.
$(s, s') \in \mathcal{B}_{n+2}$
$\Rightarrow$ (a) $(s, s') \in \mathcal{B}_0$, (b) for every $(s, s_1) \in R$, there is $(s', s_1') \in R'$ such that $(s_1, s_1') \in \mathcal{B}_{n+1}$, and (c) for every $(s', s_1') \in R'$, there is $(s, s_1) \in R$ such that $(s_1, s_1') \in \mathcal{B}_{n+1}$
$\Rightarrow$ (a) $(s, s') \in \mathcal{B}_0$, (b) for every $(s, s_1) \in R$, there is $(s', s_1') \in R'$ such that $(s_1, s_1') \in \mathcal{B}_n$ by inductive assumption, and (c) for every $(s', s_1') \in R'$, there is $(s, s_1) \in R$ such that $(s_1, s_1') \in \mathcal{B}_n$ by inductive assumption
$\Rightarrow (s, s') \in \mathcal{B}_{n+1}$.

(ii) and (iii) are evident by the above definition. ∎

Lemma 1. **Proof:** It is clear from Lemma 5 due to there is the leat number $k \geq 0$ such that $\mathcal{B}_k = \mathcal{B}$. ∎

Proposition 2. **Proof:** In order to distinguish the relations $\mathcal{B}_0, \mathcal{B}_1, \dots$ for different set $V \subseteq \mathcal{A}$, by $\mathcal{B}_i^V$ we mean the relation $\mathcal{B}_1, \mathcal{B}_2, \dots$ for $V \subseteq \mathcal{A}$. Denote as $\mathcal{B}_0, \mathcal{B}_1, \dots$ when the underlying set V is clear from their contexts or there is no confusion.

(i) Base: it is clear for $n = 0$.
Step: For $n > 0$, supposing if $(\mathcal{K}_1, \mathcal{K}_2) \in \mathcal{B}_i^{V_1}$ and $(\mathcal{K}_1, \mathcal{K}_2) \in \mathcal{B}_i^{V_2}$ then $(\mathcal{K}_1, \mathcal{K}_2) \in \mathcal{B}_i^{V_1 \cup V_2}$ for all $0 \leq i \leq n$. We will show that if $(\mathcal{K}_1, \mathcal{K}_2) \in \mathcal{B}_{n+1}^{V_1}$ and $(\mathcal{K}_1, \mathcal{K}_2) \in \mathcal{B}_{n+1}^{V_2}$ then $(\mathcal{K}_1, \mathcal{K}_2) \in \mathcal{B}_{n+1}^{V_1 \cup V_2}$.
(a) It is evident that $L_1(s_1) - (V_1 \cup V_2) = L_2(s_2) - (V_1 \cup V_2)$.
(b) We will show that for each $(s_1, s_1^1) \in R_1$ there is a $(s_2, s_2^1) \in R_2$ such that $(s_1^1, s_2^1) \in \mathcal{B}_n^{V_1 \cup V_2}$. There is $(\mathcal{K}_1^1, \mathcal{K}_2^1) \in \mathcal{B}_{n-1}^{V_1 \cup V_2}$ due to $(\mathcal{K}_1, \mathcal{K}_2) \in \mathcal{B}_n^{V_1 \cup V_2}$ by inductive assumption. Then we only need to prove for each $(s_1^1, s_1^2) \in R_1$ there is a $(s_2^1, s_2^2) \in R_2$ such that $(\mathcal{K}_1^2, \mathcal{K}_2^2) \in \mathcal{B}_{n-2}^{V_1 \cup V_2}$ and for each $(s_2^1, s_2^2) \in R_2$ there is a $(s_1^1, s_1^2) \in R_1$ such that $(\mathcal{K}_1^2, \mathcal{K}_2^2) \in \mathcal{B}_{n-2}^{V_1 \cup V_2}$. Therefore, we only need to prove that for each $(s_1^n, s_1^{n+1}) \in R_1$ there is a $(s_2^n, s_2^{n+1}) \in R_2$ such that $(\mathcal{K}_1^{n+1}, \mathcal{K}_2^{n+1}) \in \mathcal{B}_0^{V_1 \cup V_2}$ and for each $(s_2^n, s_2^{n+1}) \in R_2$ there is a $(s_1^n, s_1^{n+1}) \in R_1$ such that $(\mathcal{K}_1^{n+1}, \mathcal{K}_2^{n+1}) \in \mathcal{B}_0^{V_1 \cup V_2}$. It is apparent that $L_1(s_1^{n+1}) - (V_1 \cup V_2) = L_1(s_2^{n+1}) - (V_1 \cup V_2)$ due to $(\mathcal{K}_1, \mathcal{K}_2) \in \mathcal{B}_{n+1}^{V_1}$ and $(\mathcal{K}_1, \mathcal{K}_2) \in \mathcal{B}_{n+1}^{V_2}$. Where $\mathcal{K}_i^j = (\mathcal{M}_i, s_i^j)$ with $i \in \{1, 2\}$ and $0 < j \leq n+1$.
(c) It is similar with (b).

(ii) It is clear from (i).

(iii) It is clear from Proposition 1.

(iv) Let $\mathcal{M}_i = (S_i, R_i, L_i, s_i)$ $(i = 1, 2, 3)$, $s_1 \leftrightarrow_{V_1} s_2$ via a binary relation $\mathcal{B}$, and $s_2 \leftrightarrow_{V_2} s_3$ via a binary relation $\mathcal{B}''$. Let $\mathcal{B}' \subseteq S_1 \times S_3$ and $\mathcal{B}' = \{(w_1, w_3) | (w_1, w_2) \in \mathcal{B}$ and $(w_2, w_3) \in \mathcal{B}_2\}$. It's apparent that $(s_1, s_3) \in \mathcal{B}'$. We prove $\mathcal{B}'$ is a $V_1 \cup V_2$-bisimulation between s_1 and s_3 from the three points of Proposition 1 of X-bisimulation (where X is a set of atoms). For all $(w_1, w_3) \in \mathcal{B}'$:

(a) there is $w_2 \in S_2$ such that $(w_1, w_2) \in \mathcal{B}$ and $(w_2, w_3) \in \mathcal{B}''$, and $\forall q \notin V_1$, $q \in L_1(w_1)$ iff $q \in L_2(w_2)$ by $w_1 \leftrightarrow_{V_1} w_2$ and $\forall q' \notin V_2$, $q' \in L_2(w_2)$ iff $q' \in L_3(w_3)$ by $w_2 \leftrightarrow_{V_2} w_3$. Then we have $\forall r \notin V_1 \cup V_2, r \in L_1(w_1)$ iff $r \in L_3(w_3)$.

(b) if $(w_1, u_1) \in \mathcal{R}_1$, then $\exists u_2 \in S_2$ such that $(w_2, u_2) \in \mathcal{R}_2$ and $(u_1, u_2) \in \mathcal{B}$ (due to $(w_1, w_2) \in \mathcal{B}$ and $(w_2, w_3) \in \mathcal{B}''$ by the definition of $\mathcal{B}'$); and then $\exists u_3 \in S_3$ such that $(w_3, u_3) \in \mathcal{R}_3$ and $(u_2, u_3) \in \mathcal{B}''$, hence $(u_1, u_3) \in \mathcal{B}'$ by the definition of $\mathcal{B}'$.

(c) if $(w_3, u_3) \in \mathcal{R}_3$, then $\exists u_2 \in S_2$ such that $(w_2, u_2) \in \mathcal{R}_2$ and $(u_2, u_3) \in \mathcal{B}_2$; and then $\exists u_1 \in S_1$ such that $(w_1, u_1) \in \mathcal{R}_1$ and $(u_1, u_2) \in \mathcal{B}$, hence $(u_1, u_3)\mathcal{B}'$ by the definition of $\mathcal{B}'$.

(v) We will show that $(\mathcal{K}_1, \mathcal{K}_2) \in \mathcal{B}_n^W$ for all $n \geq 0$ inductively.

Base: $L_1(s_1) - V = L_2(s_2) - V$
$\Rightarrow \forall q \in A - V$ there is $q \in L_1(s_1)$ iff $q \in L_2(s_2)$
$\Rightarrow \forall q \in A - W$ there is $q \in L_1(s_1)$ iff $q \in L_2(s_2)$ due to $V \subseteq W$
$\Rightarrow L_1(s_1) - W = L_2(s_2) - W$, *i.e.* $(\mathcal{K}_1, \mathcal{K}_2) \in \mathcal{B}_0^W$.

Step: Supposing that $(\mathcal{K}_1, \mathcal{K}_2) \in \mathcal{B}_i^W$ for all $0 \leq i \leq k$ $(k > 0)$, we will show $(\mathcal{K}_1, \mathcal{K}_2) \in \mathcal{B}_{k+1}^W$.

(a) It is apparent that $L_1(s_1) - W = L_2(s_2) - W$ by base.

(b) $\forall (s_1, s_{1,1}) \in R_1$, we will show that there is a $(s_2, s_{2,1}) \in R_2$ $(\mathcal{K}_{1,1}, \mathcal{K}_{2,1}) \in \mathcal{B}_k^W$. $(\mathcal{K}_{1,1}, \mathcal{K}_{2,1}) \in \mathcal{B}_{k-1}^W$ by inductive assumption, we need only to prove the following points:
(a) $\forall (s_{1,k}, s_{1,k+1}) \in R_1$ there is a $(s_{2,k}, s_{2,k+1}) \in R_2$ $(\mathcal{K}_{1,k+1}, \mathcal{K}_{2,k+1}) \in \mathcal{B}_0^W$ due to $(\mathcal{K}_{1,1}, \mathcal{K}_{2,1}) \in \mathcal{B}_k^V$. It is easy to see that $L_1(s_{1,k+1}) - V = L_1(s_{2,k+1}) - V$, then there is $L_1(s_{1,k+1}) - W = L_1(s_{2,k+1}) - W$. Therefore, $(\mathcal{K}_{1,k+1}, \mathcal{K}_{2,k+1}) \in \mathcal{B}_0^W$.
(b) $\forall (s_{2,k}, s_{2,k+1}) \in R_1$ there is a $(s_{1,k}, s_{1,k+1}) \in R_1$ $(\mathcal{K}_{1,k+1}, \mathcal{K}_{2,k+1}) \in \mathcal{B}_0^W$ due to $(\mathcal{K}_{1,1}, \mathcal{K}_{2,1}) \in \mathcal{B}_k^V$. This can be proved as (a).

(c) $\forall (s_2, s_{2,1}) \in R_1$, we will show that there is a $(s_1, s_{1,1}) \in R_2$ $(\mathcal{K}_{1,1}, \mathcal{K}_{2,1}) \in \mathcal{B}_k^W$. This can be proved as (ii).

Where $\mathcal{K}_{i,j} = (\mathcal{M}_i, s_{i,j})$ and $(s_{i,k}, s_{i,k+1}) \in R_i$ means that $s_{i,k+1}$ is the $(k+2)$-th node in the path $(s_i, s_{i,1}, s_{i,2}, \dots, s_{i,k+1}, \dots)$ $(i = 1, 2)$. ∎

Theorem1 **Proof:** This theorem can be proved by inducting on the formula φ and supposing $Var(\varphi) \cap V = \emptyset$.

Here we only prove the only-if direction. The other direction can be similarly proved.

Case $\varphi = p$ where $p \in \mathcal{A} - V$:
$(\mathcal{M}, s) \models \varphi$ iff $p \in L(s)$ (by the definition of satisfiability)
$\Leftrightarrow p \in L'(s')$ $(s \leftrightarrow_V s')$
$\Leftrightarrow (\mathcal{M}', s') \models \varphi$

Case $\varphi = \neg\psi$:
$(\mathcal{M}, s) \models \varphi$ iff $(\mathcal{M}, s) \not\models \psi$
$\Leftrightarrow (\mathcal{M}', s') \not\models \psi$ (induction hypothesis)
$\Leftrightarrow (\mathcal{M}', s') \models \varphi$

Case $\varphi = \psi_1 \vee \psi_2$:
$(\mathcal{M}, s) \models \varphi$
$\Leftrightarrow (\mathcal{M}, s) \models \psi_1$ or $(\mathcal{M}, s) \models \psi_2$
$\Leftrightarrow (\mathcal{M}', s') \models \psi_1$ or $(\mathcal{M}', s') \models \psi_2$ (induction hypothesis)
$\Leftrightarrow (\mathcal{M}', s') \models \varphi$

Case $\varphi = \mathrm{EX}\psi$:
$\mathcal{M}, s \models \varphi$
$\Leftrightarrow$ There is a path $\pi = (s, s_1, ...)$ such that $\mathcal{M}, s_1 \models \psi$
$\Leftrightarrow$ There is a path $\pi' = (s', s'_1, ...)$ such that $\pi \leftrightarrow_V \pi'$ ($s \leftrightarrow_V s'$, Proposition 2)
$\Leftrightarrow s_1 \leftrightarrow_V s'_1$ ($\pi \leftrightarrow_V \pi'$)
$\Leftrightarrow (\mathcal{M}', s'_1) \models \psi$ (induction hypothesis)
$\Leftrightarrow (\mathcal{M}', s') \models \varphi$

Case $\varphi = \mathrm{EG}\psi$:
$\mathcal{M}, s \models \varphi$
$\Leftrightarrow$ There is a path $\pi = (s = s_0, s_1, ...)$ such that for each $i \geq 0$ there is $(\mathcal{M}, s_i) \models \psi$
$\Leftrightarrow$ There is a path $\pi' = (s' = s'_0, s'_1, ...)$ such that $\pi \leftrightarrow_V \pi'$ ($s \leftrightarrow_V s'$, Proposition 2)
$\Leftrightarrow s_i \leftrightarrow_V s'_i$ for each $i \geq 0$ ($\pi \leftrightarrow_V \pi'$)
$\Leftrightarrow (\mathcal{M}', s'_i) \models \psi$ for each $i \geq 0$ (induction hypothesis)
$\Leftrightarrow (\mathcal{M}', s') \models \varphi$

Case $\varphi = \mathrm{E}[\psi_1 \mathrm{U} \psi_2]$:
$\mathcal{M}, s \models \varphi$
$\Leftrightarrow$ There is a path $\pi = (s = s_0, s_1, ...)$ such that there is $i \geq 0$ such that $(\mathcal{M}, s_i) \models \psi_2$, and for all $0 \leq j < i$, $(\mathcal{M}, s_j) \models \psi_1$
$\Leftrightarrow$ There is a path $\pi' = (s = s'_0, s'_1, ...)$ such that $\pi \leftrightarrow_V \pi'$ ($s \leftrightarrow_V s'$, Proposition 2)
$\Leftrightarrow (\mathcal{M}', s'_i) \models \psi_2$, and for all $0 \leq j < i$ $(\mathcal{M}', s'_j) \models \psi_1$ (induction hypothesis)
$\Leftrightarrow (\mathcal{M}', s') \models \varphi$ ■

Proposition 3. **Proof:** We will prove this from two aspects:

($\Rightarrow$) If $s\mathcal{B}_n s'$, then $Tr_j(s) \leftrightarrow_V Tr_j(s')$ for all $0 \leq j \leq n$. $s\mathcal{B}_n s'$ implies both roots of $Tr_n(s)$ and $Tr_n(s')$ have the same atoms except those atoms in V. Besides, for any s_1 with $s \rightarrow s_1$, there is a s'_1 with $s' \rightarrow s'_1$ s.t. $s_1 \mathcal{B}_{n-1} s'_1$ and vice versa. Then we have $Tr_1(s) \leftrightarrow_V Tr_1(s')$. Therefore, $Tr_n(s) \leftrightarrow_V Tr_n(s')$ by use such method recursively, and then $Tr_j(s) \leftrightarrow_V Tr_j(s')$ for all $0 \leq j \leq n$.

($\Leftarrow$) If $Tr_j(s) \leftrightarrow_V Tr_j(s')$ for all $j \leq n$, then $s\mathcal{B}_n s'$. $Tr_0(s) \leftrightarrow_V Tr_0(s')$ implies $L(s) - V = L'(s') - V$ and then $s\mathcal{B}_0 s'$. $Tr_1(s) \leftrightarrow_V Tr_1(s')$ implies $L(s) - V = L'(s') - V$ and for every successors s_1 of the root of one, it is possible to find a successor of the root of the other s'_1 such that $s_1 \mathcal{B}_0 s'_1$. Therefore $s\mathcal{B}_1 s'$, and then we will have $s\mathcal{B}_n s'$ by use such method recursively. ■

Proposition 4 **Proof:** If $(s, s') \notin \mathcal{B}$, then there exists a least constant k such that $(s_i, s_j) \notin \mathcal{B}_k$, and then there is a least constant m ($m \leq k$) such that $\mathrm{Tr}_m(s_i)$ and $\mathrm{Tr}_m(s_j)$ are not V-corresponding by Proposition 3. Let $c = m$, the lemma is proved. ■

Lemma2 **Proof:** This result can be proved by inducting on n.

Base. It is apparent that for any $s_n \in S$ and $s'_n \in S'$, if $\mathrm{Tr}_0(s_n) \leftrightarrow_{\overline{V}} \mathrm{Tr}_0(s'_n)$ then $\mathcal{F}_V(\mathrm{Tr}_0(s_n)) \equiv \mathcal{F}_V(\mathrm{Tr}_0(s'_n))$ due to $L(s_n) - \overline{V} = L'(s'_n) - \overline{V}$ by known.

Step. Supposing that for $k = m$ ($0 < m \leq n$) there is if $\mathrm{Tr}_{n-k}(s_k) \leftrightarrow_{\overline{V}} \mathrm{Tr}_{n-k}(s'_k)$ then $\mathcal{F}_V(\mathrm{Tr}_{n-k}(s_k)) \equiv \mathcal{F}_V(\mathrm{Tr}_{n-k}(s'_k))$, then we will show if $\mathrm{Tr}_{n-k+1}(s_{k-1}) \leftrightarrow_{\overline{V}} \mathrm{Tr}_{n-k+1}(s'_{k-1})$ then $\mathcal{F}_V(\mathrm{Tr}_{n-k+1}(s_{k-1})) \equiv \mathcal{F}_V(\mathrm{Tr}_{n-k+1}(s'_{k-1}))$. Apparent that:
$\mathcal{F}_V(\mathrm{Tr}_{n-k+1}(s_{k-1})) =$
$\left(\bigwedge_{(s_{k-1}, s_k) \in R} \mathrm{EX}\mathcal{F}_V(\mathrm{Tr}_{n-k}(s_k))\right) \wedge$
$\mathrm{AX}\left(\bigvee_{(s_{k-1}, s_k) \in R} \mathcal{F}_V(\mathrm{Tr}_{n-k}(s_k))\right) \wedge \mathcal{F}_V(\mathrm{Tr}_0(s_{k-1}))$
$\mathcal{F}_V(\mathrm{Tr}_{n-k+1}(s'_{k-1})) =$
$\left(\bigwedge_{(s'_{k-1}, s'_k) \in R} \mathrm{EX}\mathcal{F}_V(\mathrm{Tr}_{n-k}(s'_k))\right) \wedge$
$\mathrm{AX}\left(\bigvee_{(s'_{k-1}, s'_k) \in R} \mathcal{F}_V(\mathrm{Tr}_{n-k}(s'_k))\right) \wedge \mathcal{F}_V(\mathrm{Tr}_0(s'_{k-1}))$
by the definition of characterize formula of the computation tree. Then we have for any $(s_{k-1}, s_k) \in R$ there is $(s'_{k-1}, s'_k) \in R'$ such that $\mathrm{Tr}_{n-k}(s_k) \leftrightarrow_{\overline{V}} \mathrm{Tr}_{n-k}(s'_k)$ by $\mathrm{Tr}_{n-k+1}(s_{k-1}) \leftrightarrow_{\overline{V}} \mathrm{Tr}_{n-k+1}(s'_{k-1})$. Besides, for any $(s'_{k-1}, s'_k) \in R'$ there is $(s_{k-1}, s_k) \in R$ such that $\mathrm{Tr}_{n-k}(s_k) \leftrightarrow_{\overline{V}} \mathrm{Tr}_{n-k}(s'_k)$ by $\mathrm{Tr}_{n-k+1}(s_{k-1}) \leftrightarrow_{\overline{V}} \mathrm{Tr}_{n-k+1}(s'_{k-1})$. Therefore, we have $\mathcal{F}_V(\mathrm{Tr}_{n-k+1}(s_{k-1})) \equiv \mathcal{F}_V(\mathrm{Tr}_{n-k+1}(s'_{k-1}))$ by induction hypothesis. ■

Lemma 3. **Proof:** Let $(\mathcal{M}', s'_0)$ be a model of φ. Then $(\mathcal{M}', s'_0) \models \bigvee_{(\mathcal{M}, s_0) \in Mod(\varphi)} \mathcal{F}_\mathcal{A}(\mathcal{M}, s_0)$ due to $(\mathcal{M}', s'_0) \models \mathcal{F}_\mathcal{A}(\mathcal{M}', s'_0)$. On the other hand, suppose that $(\mathcal{M}', s'_0)$ is a model of $\bigvee_{(\mathcal{M}, s_0) \in Mod(\varphi)} \mathcal{F}_\mathcal{A}(\mathcal{M}, s_0)$. Then there is a $(\mathcal{M}, s_0) \in Mod(\varphi)$ such that $(\mathcal{M}', s'_0) \models \mathcal{F}_\mathcal{A}(\mathcal{M}, s_0)$. And then $(\mathcal{M}, s_0) \leftrightarrow_\emptyset (\mathcal{M}', s'_0)$ by Theorem 2. Therefore, $(\mathcal{M}, s_0)$ is also a model of φ by Theorem 1. ■

Theorem 2
In order to prove Theorem 2, we prove the following two lemmas at first.

Lemma 6 *Let* $V \subseteq \mathcal{A}$, $\mathcal{M} = (S, R, L, s_0)$ *and* $\mathcal{M}' = (S', R', L', s'_0)$ *be two model structures,* $s \in S$, $s' \in S'$ *and* $n \geq 0$.

(i) $(\mathcal{M}, s) \models \mathcal{F}_V(Tr_n(s))$.

(ii) If $(\mathcal{M}, s) \models \mathcal{F}_V(Tr_n(s'))$ *then* $Tr_n(s) \leftrightarrow_{\overline{V}} Tr_n(s')$.

Proof: (i) It is apparent from the definition of $\mathcal{F}_V(\mathrm{Tr}_n(s))$.
Base. It is apparent that $(\mathcal{M}, s) \models \mathcal{F}_V(\mathrm{Tr}_0(s))$.
Step. For $k \geq 0$, supposing the result talked in (i) is correct in $k - 1$, we will show that $(\mathcal{M}, s) \models \mathcal{F}_V(\mathrm{Tr}_{k+1}(s))$, *i.e.* :

$$(\mathcal{M}, s) \models \left(\bigwedge_{(s,s') \in R} \mathrm{EX}T(s')\right) \wedge \mathrm{AX}\left(\bigvee_{(s,s') \in R} T(s')\right) \wedge \mathcal{F}_V(\mathrm{Tr}_0(s)).$$

Where $T(s') = \mathcal{F}_V(\mathrm{Tr}_k(s'))$. It is apparent that $(\mathcal{M}, s) \models \mathcal{F}_V(\mathrm{Tr}_0(s))$ by Base. It is apparent that for any $(s, s') \in R$, there is $(\mathcal{M}, s') \models \mathcal{F}_V(\text{Tr}_k(s'))$ by inductive assumption. Then we have $(\mathcal{M}, s) \models \text{EX}\mathcal{F}_V(\text{Tr}_k(s'))$, and then $(\mathcal{M}, s) \models \left(\bigwedge_{(s,s')\in R} \text{EX}\mathcal{F}_V(\text{Tr}_k(s'))\right)$. Similarly, we have that for any $(s, s') \in R$, there is $(\mathcal{M}, s') \models \bigvee_{(s,s'')\in R} \mathcal{F}_V(\text{Tr}_k(s''))$. Therefore, $(\mathcal{M}, s) \models \text{AX}\left(\bigvee_{(s,s'')\in R} \mathcal{F}_V(\text{Tr}_k(s''))\right)$.

(ii) **Base**. If $n = 0$, then $(\mathcal{M}, s) \models \mathcal{F}_V(\text{Tr}_0(s'))$ implies $L(s) - \overline{V} = L'(s') - \overline{V}$. Hence, $\text{Tr}_0(s) \leftrightarrow_{\overline{V}} \text{Tr}_0(s')$.

Step. Supposing $n > 0$ and the result talked in (ii) is correct in $n - 1$.

(a) It is easy to see that $L(s) - \overline{V} = L'(s') - \overline{V}$.

(b) We will show that for each $(s, s_1) \in R$, there is a $(s', s_1') \in R'$ such that $\text{Tr}_{n-1}(s_1) \leftrightarrow_{\overline{V}} \text{Tr}_{n-1}(s_1')$. Since $(\mathcal{M}, s) \models \mathcal{F}_V(\text{Tr}_n(s'))$, then $(\mathcal{M}, s) \models \text{AX}\left(\bigvee_{(s',s_1')\in R} \mathcal{F}_V(\text{Tr}_{n-1}(s_1'))\right)$. Therefore, for each $(s, s_1) \in R$ there is a $(s', s_1') \in R'$ such that $(\mathcal{M}, s_1) \models \mathcal{F}_V(\text{Tr}_{n-1}(s_1'))$. Hence, $\text{Tr}_{n-1}(s_1) \leftrightarrow_{\overline{V}} \text{Tr}_{n-1}(s_1')$ by inductive hypothesis.

(c) We will show that for each $(s', s_1') \in R'$ there is a $(s, s_1) \in R$ such that $\text{Tr}_{n-1}(s_1') \leftrightarrow_{\overline{V}} \text{Tr}_{n-1}(s_1)$. Since $(\mathcal{M}, s) \models \mathcal{F}_V(\text{Tr}_n(s'))$, then $(\mathcal{M}, s) \models \bigwedge_{(s',s_1')\in R'} \text{EX}\mathcal{F}_V(\text{Tr}_{n-1}(s_1'))$. Therefore, for each $(s', s_1') \in R'$ there is a $(s, s_1) \in R$ such that $(\mathcal{M}, s_1) \models \mathcal{F}_V(\text{Tr}_{n-1}(s_1')$. Hence, $\text{Tr}_{n-1}(s_1) \leftrightarrow_{\overline{V}} \text{Tr}_{n-1}(s_1')$ by inductive hypothesis. ■

A consequence of the previous lemma is:

Lemma 7 *Let $V \subseteq \mathcal{A}$, $\mathcal{M} = (S, R, L, s_0)$ a model structure, $k = ch(\mathcal{M}, V)$ and $s \in S$.*

- $(\mathcal{M}, s) \models \mathcal{F}_V(Tr_k(s))$, *and*
- *for each $s' \in S$, $(\mathcal{M}, s) \leftrightarrow_{\overline{V}} (\mathcal{M}, s')$ if and only if $(\mathcal{M}, s') \models \mathcal{F}_V(Tr_k(s))$.*

Proof: Let $\phi = \mathcal{F}_V(\text{Tr}_k(s))$, where k is the V-characteristic number of $\mathcal{M}$. $(\mathcal{M}, s) \models \phi$ by the definition of $\mathcal{F}$, and then $\forall s' \in S$, if $s \leftrightarrow_{\overline{V}} s'$ there is $(\mathcal{M}, s') \models \phi$ by Theorem 1 due to $\text{IR}(\phi, \mathcal{A} \setminus V)$. Supposing $(\mathcal{M}, s') \models \phi$, if $s \not\leftrightarrow_{\overline{V}} s'$, then $\text{Tr}_k(s) \not\leftrightarrow_{\overline{V}} \text{Tr}_k(s')$, and then $(\mathcal{M}, s') \not\models \phi$ by Lemma 6, a contradiction. ■

Now we are in the position of proving Theorem 2.

Proof: Let $\mathcal{F}_V(\mathcal{M}, s_0)$ be the characterizing formula of $(\mathcal{M}, s_0)$ on V. It is apparent that $\text{IR}(\mathcal{F}_V(\mathcal{M}, s_0), \overline{V})$. We will show that $(\mathcal{M}, s_0) \models \mathcal{F}_V(\mathcal{M}, s_0)$ at first.

It is apparent that $(\mathcal{M}, s_0) \models \mathcal{F}_V(\text{Tr}_c(s_0))$ by Lemma 6. We must show that $(\mathcal{M}, s_0) \models \bigwedge_{s\in S} G(\mathcal{M}, s)$. Let $\mathcal{X} = \mathcal{F}_V(\text{Tr}_c(s)) \rightarrow \left(\bigwedge_{(s,s_1)\in R} \text{EX}\mathcal{F}_V(\text{Tr}_c(s_1))\right) \wedge \text{AX}\left(\bigvee_{(s,s_1)\in R} \mathcal{F}_V(\text{Tr}_c(s_1))\right)$, we will show $\forall s \in S$, $(\mathcal{M}, s_0) \models G(\mathcal{M}, s)$. Where $G(\mathcal{M}, s) = \text{AG}\mathcal{X}$. There are two cases we should consider:

- If $(\mathcal{M}, s_0) \not\models \mathcal{F}_V(\text{Tr}_c(s))$, it is apparent that $(\mathcal{M}, s_0) \models \mathcal{X}$;
- If $(\mathcal{M}, s_0) \models \mathcal{F}_V(\text{Tr}_c(s))$:
  $(\mathcal{M}, s_0) \models \mathcal{F}_V(\text{Tr}_c(s))$
  $\Rightarrow s_0 \leftrightarrow_{\overline{V}} s$ by the definition of characteristic number and Lemma 7
  for each $(s, s_1) \in R$ there is $(\mathcal{M}, s_1) \models \mathcal{F}_V(\text{Tr}_c(s_1))$ $(s_1 \leftrightarrow_{\overline{V}} s_1)$
  $\Rightarrow (\mathcal{M}, s) \models \bigwedge_{(s,s_1)\in R} \text{EX}\mathcal{F}_V(\text{Tr}_c(s_1))$
  $\Rightarrow (\mathcal{M}, s_0) \models \bigwedge_{(s,s_1)\in R} \text{EX}\mathcal{F}_V(\text{Tr}_c(s_1))$ $(\text{IR}(\bigwedge_{(s,s_1)\in R} \text{EX}\mathcal{F}_V(\text{Tr}_c(s_1)), \overline{V})$, $s_0 \leftrightarrow_{\overline{V}} s)$
  for each (s, s_1) there is $\mathcal{M}, s_1 \models \bigvee_{(s,s_2)\in R} \mathcal{F}_V(\text{Tr}_c(s_2))$
  $\Rightarrow (\mathcal{M}, s) \models \text{AX}\left(\bigvee_{(s,s_2)\in R} \mathcal{F}_V(\text{Tr}_c(s_2))\right)$
  $\Rightarrow (\mathcal{M}, s_0) \models \text{AX}\left(\bigvee_{(s,s_2)\in R} \mathcal{F}_V(\text{Tr}_c(s_2))\right)$ $(\text{IR}(\text{AX}\left(\bigvee_{(s,s_2)\in R} \mathcal{F}_V(\text{Tr}_c(s_2))\right), \overline{V})$, $s_0 \leftrightarrow_{\overline{V}} s)$
  $\Rightarrow (\mathcal{M}, s_0) \models \mathcal{X}$.

For any other states s' which can reach from s_0 can be proved similarly, *i.e.* , $(\mathcal{M}, s') \models \mathcal{X}$. Therefore, $\forall s \in S$, $(\mathcal{M}, s_0) \models G(\mathcal{M}, s)$, and then $(\mathcal{M}, s_0) \models \mathcal{F}_V(\mathcal{M}, s_0)$.

We will prove this theorem from the following two aspects:

$(\Leftarrow)$ If $s_0 \leftrightarrow_{\overline{V}} s_0'$, then $(\mathcal{M}', s_0') \models \mathcal{F}_V(M, s_0)$. Since $(\mathcal{M}, s_0) \models \mathcal{F}_V(\mathcal{M}, s_0)$ and $\text{IR}(\mathcal{F}_V(\mathcal{M}, s_0), \overline{V})$, hence $(\mathcal{M}', s_0') \models \mathcal{F}_V(M, s_0)$ by Theorem 1.

$(\Rightarrow)$ If $(\mathcal{M}', s_0') \models \mathcal{F}_V(M, s_0)$, then $s_0 \leftrightarrow_{\overline{V}} s_0'$. We will prove this by showing that $\forall n \geq 0, Tr_n(s_0) \leftrightarrow_{\overline{V}} Tr_n(s_0')$.

Base. It is apparent that $Tr_0(s_0) \equiv Tr_0(s_0')$.

Step. Supposing $\text{Tr}_k(s_0) \leftrightarrow_{\overline{V}} \text{Tr}_k(s_0')$ $(k > 0)$, we will prove $\text{Tr}_{k+1}(s_0) \leftrightarrow_{\overline{V}} \text{Tr}_{k+1}(s_0')$. We should only show that $\text{Tr}_1(s_k) \leftrightarrow_{\overline{V}} \text{Tr}_1(s_k')$. Where $(s_0, s_1), (s_1, s_2), \ldots, (s_{k-1}, s_k) \in R$ and $(s_0', s_1'), (s_1', s_2'), \ldots, (s_{k-1}', s_k') \in R'$, *i.e.* s_{i+1} (s_{i+1}') is an immediate successor of s_i (s_i') for all $0 \leq i \leq k - 1$.

(i) It is apparent that $L(s_k) - \overline{V} = L'(s_k') - \overline{V}$ by inductive assumption.

Before talking about the other points, note the following fact that:

$(\mathcal{M}', s_0') \models \mathcal{F}_V(\mathcal{M}, s_0)$
$\Rightarrow \forall s' \in S'$, $(\mathcal{M}', s') \models \mathcal{F}_V(\text{Tr}_c(s)) \rightarrow \left(\bigwedge_{(s,s_1)\in R} \text{EX}\mathcal{F}_V(\text{Tr}_c(s_1))\right) \wedge \text{AX}\left(\bigvee_{(s,s_1)\in R} \mathcal{F}_V(\text{Tr}_c(s_1))\right)$ for any $s \in S$. **(fact)**

(I) $(\mathcal{M}', s_0') \models \mathcal{F}_V(\text{Tr}_c(s_0)) \rightarrow \left(\bigwedge_{(s_0,s_1)\in R} \text{EX}\mathcal{F}_V(\text{Tr}_c(s_1))\right) \wedge \text{AX}\left(\bigvee_{(s_0,s_1)\in R} \mathcal{F}_V(\text{Tr}_c(s_1))\right)$ **(fact)**

(II) $(\mathcal{M}', s_0') \models \mathcal{F}_V(\text{Tr}_c(s_0)))$ (known)

(III) $(\mathcal{M}', s_0') \models \left(\bigwedge_{(s_0,s_1)\in R} \text{EX}\mathcal{F}_V(\text{Tr}_c(s_1))\right) \wedge \text{AX}\left(\bigvee_{(s_0,s_1)\in R} \mathcal{F}_V(\text{Tr}_c(s_1))\right)$ ((I),(II))

(ii) We will show that for each $(s_k, s_{k+1}) \in R$ there is a $(s_k', s_{k+1}') \in R'$ such that $L(s_{k+1}) - \overline{V} = L'(s_{k+1}') - \overline{V}$.

(1) $(\mathcal{M}', s_0') \models \bigwedge_{(s_0,s_1)\in R} \text{EX}\mathcal{F}_V(\text{Tr}_c(s_1))$ (III)

(2) $\forall (s_0, s_1) \in R$, $\exists (s_0', s_1') \in R'$ $(\mathcal{M}', s_1') \models \mathcal{F}_V(\text{Tr}_c(s_1))$ (2)

(3) $\text{Tr}_c(s_1) \leftrightarrow_{\overline{V}} \text{Tr}_c(s_1')$ ((2), Lemma 6)

(4) $L(s_1) - \overline{V} = L'(s'_1) - \overline{V}$ ((3), $c \geq 0$)
(5) $(\mathcal{M}', s'_1) \models \mathcal{F}_V(\text{Tr}_c(s_1)) \rightarrow \left(\bigwedge_{(s_1,s_2)\in R} \text{EX}\mathcal{F}_V(\text{Tr}_c(s_2))\right) \wedge \text{AX}\left(\bigvee_{(s_1,s_2)\in R} \mathcal{F}_V(\text{Tr}_c(s_2))\right)$ **(fact)**
(6) $(\mathcal{M}', s'_1) \models \left(\bigwedge_{(s_1,s_2)\in R} \text{EX}\mathcal{F}_V(\text{Tr}_c(s_2))\right) \wedge \text{AX}\left(\bigvee_{(s_1,s_2)\in R} \mathcal{F}_V(\text{Tr}_c(s_2))\right)$ ((2), (5))
(7) ......
(8) $(\mathcal{M}', s'_k) \models \left(\bigwedge_{(s_k,s_{k+1})\in R} \text{EX}\mathcal{F}_V(\text{Tr}_c(s_{k+1}))\right) \wedge \text{AX}\left(\bigvee_{(s_k,s_{k+1})\in R} \mathcal{F}_V(\text{Tr}_c(s_{k+1}))\right)$ (similar with (6))
(9) $\forall(s_k, s_{k+1}) \in R$, $\exists(s'_k, s'_{k+1}) \in R'$ $(\mathcal{M}', s'_{k+1}) \models \mathcal{F}_V(\text{Tr}_c(s_{k+1}))$ (8)
(10) $\text{Tr}_c(s_{k+1}) \leftrightarrow_{\overline{V}} \text{Tr}_c(s'_{k+1})$ ((9), Lemma 6)
(11) $L(s_{k+1}) - \overline{V} = L'(s'_{k+1}) - \overline{V}$ ((10), $c \geq 0$)

(iii) We will show that for each $(s'_k, s'_{k+1}) \in R'$ there is a $(s_k, s_{k+1}) \in R$ such that $L(s_{k+1}) - \overline{V} = L'(s'_{k+1}) - \overline{V}$.
(1) $(\mathcal{M}', s'_k) \models \text{AX}\left(\bigvee_{(s_k,s_{k+1})\in R} \mathcal{F}_V(\text{Tr}_c(s_{k+1}))\right)$ (by (8) talked above)
(2) $\forall(s'_k, s'_{k+1}) \in R'$, $\exists(s_k, s_{k+1}) \in R$ $(\mathcal{M}', s'_{k+1}) \models \mathcal{F}_V(\text{Tr}_c(s'_{k+1}))$ (1)
(3) $\text{Tr}_c(s_{k+1}) \leftrightarrow_{\overline{V}} \text{Tr}_c(s'_{k+1})$ ((2), Lemma 6)
(4) $L(s_{k+1}) - \overline{V} = L'(s'_{k+1}) - \overline{V}$ ((3), $c \geq 0$)

■

Theorem3. **Proof:** This is following Lemma 2 and the definition of the characterizing formula of initial K-structure $\mathcal{K}$ on V. ■

Theorem 4:
Proof: $(i) \Leftrightarrow (ii)$. To prove this, we will show that:

$$\mathit{Mod}(\text{F}_{\text{CTL}}(\varphi, V)) = \mathit{Mod}(\{\phi|\varphi \models \phi, \text{IR}(\phi, V)\})$$
$$= \mathit{Mod}(\bigvee_{\mathcal{M},s_0\in \mathit{Mod}(\varphi)} \mathcal{F}_{\mathcal{A}\setminus V}(\mathcal{M}, s_0)).$$

Firstly, suppose that $(\mathcal{M}', s'_0)$ is a model of $\text{F}_{\text{CTL}}(\varphi, V)$. Then there exists an an initial K-structure $(\mathcal{M}, s_0)$ such that $(\mathcal{M}, s_0)$ is a model of φ and $(\mathcal{M}, s_0) \leftrightarrow_V (\mathcal{M}', s'_0)$. By Theorem 1, we have $(\mathcal{M}', s'_0) \models \phi$ for all ϕ that $\varphi \models \phi$ and $\text{IR}(\phi, V)$. Thus, $(\mathcal{M}', s'_0)$ is a model of $\{\phi|\varphi \models \phi, \text{IR}(\phi, V)\}$.

Secondly, suppose that $(\mathcal{M}', s'_0)$ is a models of $\{\phi|\varphi \models \phi, \text{IR}(\phi, V)\}$. Thus, $(\mathcal{M}', s'_0) \models \bigvee_{(\mathcal{M},s_0)\in \mathit{Mod}(\varphi)} \mathcal{F}_{\mathcal{A}\setminus V}(\mathcal{M}, s_0)$ due to $\bigvee_{(\mathcal{M},s_0)\in \mathit{Mod}(\varphi)} \mathcal{F}_{\mathcal{A}\setminus V}(\mathcal{M}, s_0)$ is irrelevant to V.

Finally, suppose that $(\mathcal{M}', s'_0)$ is a model of $\bigvee_{\mathcal{M},s_0\in \mathit{Mod}(\varphi)} \mathcal{F}_{\mathcal{A}\setminus V}(\mathcal{M}, s_0)$. Then there exists $(\mathcal{M}, s_0) \in \mathit{Mod}(\varphi)$ such that $(\mathcal{M}', s'_0) \models \mathcal{F}_{\mathcal{A}\setminus V}(\mathcal{M}, s_0)$. Hence, $(\mathcal{M}, s_0) \leftrightarrow_V (\mathcal{M}', s'_0)$ by Theorem 2. Thus $(\mathcal{M}', s'_0)$ is also a model of $\text{F}_{\text{CTL}}(\varphi, V)$.

$(ii) \Rightarrow (iii)$. It is not difficult to prove it.

$(iii) \Rightarrow (ii)$. Suppose that all postulates hold. By Positive Persistence, we have $\psi \models \{\phi|\varphi \models \phi, \text{IR}(\phi, V)\}$. Now we show that $\{\phi|\varphi \models \phi, \text{IR}(\phi, V)\} \models \psi$. Otherwise, there exists formula ϕ' such that $\psi \models \phi'$ but $\{\phi|\varphi \models \phi, \text{IR}(\phi, V)\} \not\models \phi'$. There are three cases:

- ϕ' is relevant to V. Thus, ψ is also relevant to V, a contradiction to Irrelevance.
- ϕ' is irrelevant to V and $\varphi \models \phi'$. This contradicts to our assumption.
- ϕ' is irrelevant to V and $\varphi \not\models \phi'$. By Negative Persistence, $\psi \not\models \phi'$, a contradiction.

Thus, ψ is equivalent to $\{\phi|\varphi \models \phi, \text{IR}(\phi, V)\}$. ■

Lemma 4 **Proof:** Let $\varphi' = \varphi \cup \{q \leftrightarrow \alpha\}$. For any model $(\mathcal{M}, s)$ of $\text{F}_{\text{CTL}}(\Gamma', q)$ there is an initial K-structure $(\mathcal{M}', s')$ s.t. $(\mathcal{M}, s) \leftrightarrow_{\{q\}} (\mathcal{M}', s')$ and $(\mathcal{M}', s') \models \varphi'$. It's apparent that $(\mathcal{M}', s') \models \varphi$, and then $(\mathcal{M}, s) \models \varphi$ since $\text{IR}(\varphi, \{q\})$ and $(\mathcal{M}, s) \leftrightarrow_{\{q\}} (\mathcal{M}', s')$ by Theorem 1.

Let $(\mathcal{M}, s) \in \mathit{Mod}(\varphi)$ with $\mathcal{M} = (S, R, L, s)$. We construct $(\mathcal{M}', s)$ with $\mathcal{M}' = (S, R, L', s)$ as follows:

$L' : S \rightarrow \mathcal{A}$ *and* $\forall s^* \in S, L'(s^*) = L(s^*)$ *if* $(\mathcal{M}, s^*) \not\models \alpha$, *else* $L'(s^*) = L(s^*) \cup \{q\}$,
$L'(s) = L(s) \cup \{q\}$ *if* $(\mathcal{M}, s) \models \alpha$, *and* $L'(s) = L(s)$ *otherwise*.

It is clear that $(\mathcal{M}', s) \models \varphi$, $(\mathcal{M}', s) \models q \leftrightarrow \alpha$ and $(\mathcal{M}', s) \leftrightarrow_{\{q\}} (\mathcal{M}, s)$. Therefore $(\mathcal{M}', s) \models \varphi \cup \{q \leftrightarrow \alpha\}$, and then $(\mathcal{M}, s) \models \text{F}_{\text{CTL}}(\varphi \cup \{q \leftrightarrow \alpha\}, q)$ by $(\mathcal{M}', s) \leftrightarrow_{\{q\}} (\mathcal{M}, s)$. ■

Proposition 5. **Proof:** Let $(\mathcal{M}_1, s_1)$ with $\mathcal{M}_1 = (S_1, R_1, L_1, s_1)$ be a model of $\text{F}_{\text{CTL}}(\varphi, \{p\} \cup V)$. By the definition, there exists a model $(\mathcal{M}, s)$ with $\mathcal{M} = (S, R, L, s)$ of φ, such that $(\mathcal{M}_1, s_1) \leftrightarrow_{\{p\}\cup V} (\mathcal{M}, s)$ via a binary relation $\mathcal{B}$. We construct an initial K-structure $(\mathcal{M}_2, s_2)$ with $\mathcal{M}_2 = (S_2, R_2, L_2, s_2)$ as follows:

(1) for s_2: let s_2 be the state such that:
  - $p \in L_2(s_2)$ iff $p \in L_1(s_1)$,
  - for all $q \in V$, $q \in L_2(s_2)$ iff $q \in L(s)$,
  - for all other atoms q', $q' \in L_2(s_2)$ iff $q' \in L_1(s_1)$ iff $q' \in L(s)$.

(2) for another:
  (i) for all pairs $w \in S$ and $w_1 \in S_1$ such that $w\mathcal{B}w_1$, let $w_2 \in S_2$ and
    - $p \in L_2(w_2)$ iff $p \in L_1(w_1)$,
    - for all $q \in V$, $q \in L_2(w_2)$ iff $q \in L(w)$,
    - for all other atoms q', $q' \in L_2(w_2)$ iff $q' \in L_1(w_1)$ iff $q' \in L(w)$.

  (ii) if $w'_1\mathcal{R}_1 w_1$, w_2 is constructed based on w_1 and $w'_2 \in S_2$ is constructed based on w'_1, then $w'_2\mathcal{R}_2 w_2$.

(3) delete duplicated states in S_2 and pairs in R_2.

Then we have $(\mathcal{M}, s) \leftrightarrow_{\{p\}} (\mathcal{M}_2, s_2)$ and $(\mathcal{M}_2, s_2) \leftrightarrow_V (\mathcal{M}_1, s_1)$. Thus, $(\mathcal{M}_2, s_2) \models \mathrm{F}_{\mathrm{CTL}}(\varphi, p)$. And therefore $(\mathcal{M}_1, s_1) \models \mathrm{F}_{\mathrm{CTL}}(\mathrm{F}_{\mathrm{CTL}}(\varphi, p), V)$.

On the other hand, suppose that $(\mathcal{M}_1, s_1)$ be a model of $\mathrm{F}_{\mathrm{CTL}}(\mathrm{F}_{\mathrm{CTL}}(\varphi, p), V)$, then there exists an initial Kripke structure $(\mathcal{M}_2, s_2)$ such that $(\mathcal{M}_2, s_2) \models \mathrm{F}_{\mathrm{CTL}}(\varphi, p)$ and $(\mathcal{M}_2, s_2) \leftrightarrow_V (\mathcal{M}_1, s_1)$, and there exists $(\mathcal{M}, s)$ such that $(\mathcal{M}, s) \models \varphi$ and $(\mathcal{M}, s) \leftrightarrow_{\{p\}} (\mathcal{M}_2, s_2)$. Therefore, $(\mathcal{M}, s) \leftrightarrow_{\{p\} \cup V} (\mathcal{M}_1, s_1)$ by Proposition 2, and consequently, $(\mathcal{M}_1, s_1) \models \mathrm{F}_{\mathrm{CTL}}(\varphi, \{p\} \cup V)$. ■

Proposition 6. **Proof:** (i) ($\Rightarrow$) Supposing $(\mathcal{M}, s)$ is a model of $\mathrm{F}_{\mathrm{CTL}}(\varphi, V)$, then there is a model $(\mathcal{M}', s')$ of φ s.t. $(\mathcal{M}, s) \leftrightarrow_V (\mathcal{M}', s')$ by the definition of $\mathrm{F}_{\mathrm{CTL}}$.

($\Leftarrow$) Supposing $(\mathcal{M}, s)$ is a model of φ, then there is an initial Kripke structure $(\mathcal{M}', s')$ s.t. $(\mathcal{M}, s) \leftrightarrow_V (\mathcal{M}', s')$, and then $(\mathcal{M}', s') \models \mathrm{F}_{\mathrm{CTL}}(\varphi, V)$ by the definition of $\mathrm{F}_{\mathrm{CTL}}$.

The (ii) and (iii) can be proved similarly.

(iv) ($\Rightarrow$) $\forall (\mathcal{M}, s) \in \mathit{Mod}(\mathrm{F}_{\mathrm{CTL}}(\psi_1 \vee \psi_2, V))$, $\exists (\mathcal{M}', s') \in \mathit{Mod}(\psi_1 \vee \psi_2)$ s.t. $(\mathcal{M}, s) \leftrightarrow_V (\mathcal{M}', s')$ and $(\mathcal{M}', s') \models \psi_1$ or $(\mathcal{M}', s') \models \psi_2$
$\Rightarrow \exists (\mathcal{M}_1, s_1) \in \mathit{Mod}(\mathrm{F}_{\mathrm{CTL}}(\psi_1, V))$ s.t. $(\mathcal{M}', s') \leftrightarrow_V (\mathcal{M}_1, s_1)$ or $\exists (\mathcal{M}_2, s_2) \in \mathit{Mod}(\mathrm{F}_{\mathrm{CTL}}(\psi_2, V))$ s.t. $(\mathcal{M}', s') \leftrightarrow_V (\mathcal{M}_2, s_2)$
$\Rightarrow (\mathcal{M}, s) \models \mathrm{F}_{\mathrm{CTL}}(\psi_1, V) \vee \mathrm{F}_{\mathrm{CTL}}(\psi_2, V)$ by Theorem 1.

($\Leftarrow$) $\forall (\mathcal{M}, s) \in \mathit{Mod}(\mathrm{F}_{\mathrm{CTL}}(\psi_1, V) \vee \mathrm{F}_{\mathrm{CTL}}(\psi_2, V))$
$\Rightarrow (\mathcal{M}, s) \models \mathrm{F}_{\mathrm{CTL}}(\psi_1, V)$ or $(\mathcal{M}, s) \models \mathrm{F}_{\mathrm{CTL}}(\psi_2, V)$
$\Rightarrow$ there is an initial K-structure $(\mathcal{M}_1, s_1)$ s.t. $(\mathcal{M}, s) \leftrightarrow_V (\mathcal{M}_1, s_1)$ and $(\mathcal{M}_1, s_1) \models \psi_1$ or $(\mathcal{M}_1, s_1) \models \psi_2$
$\Rightarrow (\mathcal{M}_1, s_1) \models \psi_1 \vee \psi_2$
$\Rightarrow$ there is an initial K-structure $(\mathcal{M}_2, s_2)$ s.t. $(\mathcal{M}_1, s_1) \leftrightarrow_V (\mathcal{M}_2, s_2)$ and $(\mathcal{M}_2, s_2) \models \mathrm{F}_{\mathrm{CTL}}(\psi_1 \vee \psi_2, V)$
$\Rightarrow (\mathcal{M}, s) \leftrightarrow_V (\mathcal{M}_2, s_2)$ and $(\mathcal{M}, s) \models \mathrm{F}_{\mathrm{CTL}}(\psi_1 \vee \psi_2, V)$.

The (v) can be proved as (iV). ■

Proposition 7 **Proof:** Let $\mathcal{M} = (S, R, L, s_0)$ with initial state s_0 and $\mathcal{M}' = (S', R', L', s_0')$ with initial state s_0', then we call $\mathcal{M}', s_0'$ be a sub-structure of $\mathcal{M}, s_0$ if:

- $S' = \{s' | s'$ is reachable from $s_0'\}$ and $S' \subseteq S$,
- $R' = \{(s_1, s_2) | s_1, s_2 \in S'$ and $(s_1, s_2) \in R\}$,
- $L' : S' \to \mathcal{A}$ and $\forall s_1 \in S'$ there is $L'(s_1) = L(s_1)$, and
- there is a stete $s \in S$ reachable from s_0 such that $(\mathcal{M}, s) \leftrightarrow_\varnothing (\mathcal{M}', s_0')$.

(i) In order to prove $\mathrm{F}_{\mathrm{CTL}}(\mathrm{AX}\phi, V) \equiv \mathrm{AX}(\mathrm{F}_{\mathrm{CTL}}(\phi, V))$, we only need to prove $\mathit{Mod}(\mathrm{F}_{\mathrm{CTL}}(\mathrm{AX}\phi, V)) = \mathit{Mod}(\mathrm{AX}\mathrm{F}_{\mathrm{CTL}}(\phi, V))$:

($\Rightarrow$) $\forall (\mathcal{M}', s') \in \mathit{Mod}(\mathrm{F}_{\mathrm{CTL}}(\mathrm{AX}\phi, V))$ there exists an initial K-structure $(\mathcal{M}, s)$ s.t. $(\mathcal{M}, s) \models \mathrm{AX}\phi$ and $(\mathcal{M}, s) \leftrightarrow_V (\mathcal{M}', s')$
$\Rightarrow$ for any sub-structure $(\mathcal{M}_1, s_1)$ of $(\mathcal{M}, s)$ there is $(\mathcal{M}_1, s_1) \models \phi$, where s_1 is a directed successor of s
$\Rightarrow$ there is an initial K-structure $(\mathcal{M}_2, s_2)$ s.t. $(\mathcal{M}_2, s_2) \models \mathrm{F}_{\mathrm{CTL}}(\phi, V)$ and $(\mathcal{M}_2, s_2) \leftrightarrow_V (\mathcal{M}_1, s_1)$
$\Rightarrow$ it is easy to construct an initial K-structure $(\mathcal{M}_3, s_3)$ by $(\mathcal{M}_2, s_2)$ s.t. $(\mathcal{M}_2, s_2)$ is a sub-structure of $(\mathcal{M}_3, s_3)$ that s_2 is a direct successor of s_3 and $(\mathcal{M}_3, s_3) \leftrightarrow_V (\mathcal{M}, s)$
$\Rightarrow \mathcal{M}_3, s_3 \models \mathrm{AX}(\mathrm{F}_{\mathrm{CTL}}(\phi, V))$, especially, let $\mathcal{M}_3, s_3 = \mathcal{M}', s'$, we have $\mathcal{M}', s' \models \mathrm{AX}(\mathrm{F}_{\mathrm{CTL}}(\phi, V))$.

($\Leftarrow$) $\forall$ $(\mathcal{M}_3, s_3) \in \mathit{Mod}(\mathrm{AX}(\mathrm{F}_{\mathrm{CTL}}(\phi, V)))$, then for any sub-structure $(\mathcal{M}_2, s_2)$ whit s_2 is a directed successor s_3 of $(\mathcal{M}_3, s_3)$ s.t. $(\mathcal{M}_2, s_2) \models \mathrm{F}_{\mathrm{CTL}}(\phi, V)$
$\Rightarrow$ there is an initial K-structure $(\mathcal{M}_1, s_1)$ s.t. $(\mathcal{M}_1, s_1) \models \phi$ and $(\mathcal{M}_1, s_1) \leftrightarrow_V (\mathcal{M}_2, s_2)$
$\Rightarrow$ it is easy to construct an initial structure $(\mathcal{M}, s)$ by $(\mathcal{M}_1, s_1)$ s.t. $(\mathcal{M}_1, s_1)$ is a sub-structure of $(\mathcal{M}, s)$ that s_1 is a direct successor of s and $(\mathcal{M}, s) \leftrightarrow_V (\mathcal{M}_3, s_3)$
$\Rightarrow (\mathcal{M}, s) \models \mathrm{AX}\phi$ and then $(\mathcal{M}_3, s_3) \models \mathrm{F}_{\mathrm{CTL}}(\mathrm{AX}\phi, V)$.

(ii) In order to prove $\mathrm{F}_{\mathrm{CTL}}(\mathrm{EX}\phi, V) \equiv \mathrm{EX}\mathrm{F}_{\mathrm{CTL}}(\phi, V)$, we only need to prove Mod $(\mathrm{F}_{\mathrm{CTL}}(\mathrm{EX}\phi, V)) = \mathit{Mod}(\mathrm{EX}\mathrm{F}_{\mathrm{CTL}}(\phi, V))$:

($\Rightarrow$) $\forall \mathcal{M}', s' \in \mathit{Mod}(\mathrm{F}_{\mathrm{CTL}}(\mathrm{EX}\phi, V))$ there exists an initial K-structure $(\mathcal{M}, s)$ s.t. $(\mathcal{M}, s) \models \mathrm{EX}\phi$ and $(\mathcal{M}, s) \leftrightarrow_V (\mathcal{M}', s')$
$\Rightarrow$ there is a sub-structure $(\mathcal{M}_1, s_1)$ of $(\mathcal{M}, s)$ s.t. $(\mathcal{M}_1, s_1) \models \phi$, where s_1 is a directed successor of s
$\Rightarrow$ there is an initial K-structure $(\mathcal{M}_2, s_2)$ s.t. $(\mathcal{M}_2, s_2) \models \mathrm{F}_{\mathrm{CTL}}(\phi, V)$ and $(\mathcal{M}_2, s_2) \leftrightarrow_V (\mathcal{M}_1, s_1)$
$\Rightarrow$ it is easy to construct an initial K-structure $(\mathcal{M}_3, s_3)$ by $(\mathcal{M}_2, s_2)$ s.t. $(\mathcal{M}_2, s_2)$ is a sub-structure of $(\mathcal{M}_3, s_3)$ that s_2 is a direct successor of s_3 and $(\mathcal{M}_3, s_3) \leftrightarrow_V (\mathcal{M}, s)$
$\Rightarrow (\mathcal{M}_3, s_3) \models \mathrm{EX}(\mathrm{F}_{\mathrm{CTL}}(\phi, V))$, especially, let $(\mathcal{M}_3, s_3) = (\mathcal{M}', s')$, we have $(\mathcal{M}', s') \models \mathrm{EX}(\mathrm{F}_{\mathrm{CTL}}(\phi, V))$.

($\Leftarrow$) $\forall$ $(\mathcal{M}_3, s_3) \in \mathit{Mod}(\mathrm{EX}(\mathrm{F}_{\mathrm{CTL}}(\phi, V)))$, then there exists a sub-structure $(\mathcal{M}_2, s_2)$ of $(\mathcal{M}_3, s_3)$ s.t. $(\mathcal{M}_2, s_2) \models \mathrm{F}_{\mathrm{CTL}}(\phi, V)$
$\Rightarrow$ there is an initial K-structure $(\mathcal{M}_1, s_1)$ s.t. $(\mathcal{M}_1, s_1) \models \phi$ and $(\mathcal{M}_1, s_1) \leftrightarrow_V (\mathcal{M}_2, s_2)$
$\Rightarrow$ it is easy to construct an initial K-structure $(\mathcal{M}, s)$ by $(\mathcal{M}_1, s_1)$ s.t. $(\mathcal{M}_1, s_1)$ is a sub-structure of $(\mathcal{M}, s)$ that s_1 is a direct successor of s and $(\mathcal{M}, s) \leftrightarrow_V (\mathcal{M}_3, s_3)$
$\Rightarrow (\mathcal{M}, s) \models \mathrm{EX}\phi$ and then $(\mathcal{M}_3, s_3) \models \mathrm{F}_{\mathrm{CTL}}(\mathrm{EX}\phi, V)$.

(iii) and (iV) can be proved as (i) and (ii) respectively. ■

Proposition8. **Proof:** The problem can be determined by the following two things: (1) guessing an I-structure $\mathcal{M}', s_0'$ satisfying φ; and (2) checking if $(\mathcal{M}, s_0) \leftrightarrow_V (\mathcal{M}', s_0')$. Both two steps can be done in polynomial time. Hence, the problem is in NP. The hardness follows that the model checking for propositional variable forgetting is NP-hard. ■

Theorem 6 **Proof:** (1) It is proved that deciding whether ψ is satisfiable is NP-Complete [Meier *et al.*, 2015]. The hardness is easy to see by setting $\mathrm{F}_{\mathrm{CTL}}(\varphi, \mathit{Var}(\varphi))$, *i.e.* deciding whether ψ is valid. For membership, from Theorem 3, we have $\mathrm{F}_{\mathrm{CTL}}(\varphi, V) \models \psi$ iff $\varphi \models \psi$ and $IR(\psi, V)$. Clearly, in CTL(AF), deciding $\varphi \models \psi$ is in co-NP. We show that deciding whether $IR(\psi, V)$ is also in co-NP. Without loss of generality, we assume that ψ is satisfiable. We consider the complement of the problem: deciding whether ψ is not irrelevant to V. It is easy to see that ψ is not irrelevant to V iff there exist a model $\mathcal{M}, s_0$ of ψ and an initial K-structure $\mathcal{M}', s_0'$ such that $\mathcal{M}, s_0 \leftrightarrow_V \mathcal{M}', s_0'$ and $\mathcal{M}', s_0' \not\models \psi$. So checking whether ψ is not irrelevant to V can be achieved in the following steps: (1) guess two initial K-structures $\mathcal{M}, s_0$ and $\mathcal{M}', s_0'$, (2) check if $\mathcal{M}, s_0 \models \psi$ and $\mathcal{M}', s_0' \not\models \psi$, and

(3) check $\mathcal{M}, s_0 \leftrightarrow_V \mathcal{M}', s_0'$. Obviously (1) can be done in polynomial time with a non-deterministic Turing machine while (2) and (3) can be done in polynomial time.

(2) Membership. We consider the complement of the problem. We may guess an initial K-structure $\mathcal{M}, s_0$ and check whether $\mathcal{M}, s_0 \models \psi$ and $\mathcal{M}, s_0 \not\models \mathrm{F_{CTL}}(\varphi, V)$. From Proposition 8, we know that this is in Σ_2^P. So the original problem is in Π_2^P. Hardness. Let $\psi \equiv \top$. Then the problem is reduced to decide $\mathrm{F_{CTL}}(\varphi, V)$'s validity. Since a propositional variable forgetting is a special case temporal forgetting, the hardness is directly followed from the proof of Proposition 24 in [Lang *et al.*, 2003].

(3) Membership. If $\mathrm{F_{CTL}}(\varphi, V) \not\models \mathrm{F_{CTL}}(\psi, V)$ then there exist an initial K-structure $\mathcal{M}, s$ such that $\mathcal{M}, s \models \mathrm{F_{CTL}}(\varphi, V)$ but $\mathcal{M}, s \not\models \mathrm{F_{CTL}}(\psi, V)$, *i.e.*, there is $\mathcal{M}_1, s_1 \leftrightarrow_V \mathcal{M}, s$ with $\mathcal{M}_1, s_1 \models \varphi$ but $\mathcal{M}_2, s_2 \not\models \psi$ for every $\mathcal{M}_2, s_2$ with $\mathcal{M}, s \leftrightarrow_V \mathcal{M}_2, s_2$. It is evident that guessing such $\mathcal{M}, s$, $\mathcal{M}_1, s_1$ with $\mathcal{M}_1, s_1 \leftrightarrow_V \mathcal{M}, s$ and checking $\mathcal{M}_1, s_1 \models \varphi$ are feasible while checking $\mathcal{M}_2, s_2 \not\models \psi$ for every $\mathcal{M}, s \leftrightarrow_V \mathcal{M}_2, s_2$ can be done in polynomial time by call a nondeterministic Turing machine. Thus the problem is in Π_2^P.

Hardness. It follows from (2) due to the fact that $\mathrm{F_{CTL}}(\varphi, V) \models \mathrm{F_{CTL}}(\psi, V)$ iff $\varphi \models \mathrm{F_{CTL}}(\psi, V)$ thanks to $IR(\mathrm{F_{CTL}}(\psi, V), V)$.

■

Proposition 9 **Proof:** (i) Suppose ψ is the SNC of q. Then $\varphi \models q \to \psi$. Thus $\varphi \models \neg\psi \to \neg q$. So $\neg\psi$ is a SC of $\neg q$. Suppose ψ' is any other SC of $\neg q$: $\varphi \models \psi' \to \neg q$. Then $\varphi \models q \to \neg\psi'$, this means $\neg\psi'$ is a NC of q on P under φ. Thus $\varphi \models \psi \to \neg\psi'$ by assumption. So $\varphi \models \psi' \to \neg\psi$. This proves that $\neg\psi$ is the WSC of $\neg q$. The proof of the other part of the proposition is similar.

(ii) The WSC case can be proved similarly with SNC case.
■

Proposition 10 **Proof:** We prove this for SNC. The case for WSC is similar. Let $\mathit{SNC}(\varphi, \alpha, P, \Gamma)$ denote that φ is the SNC of α on P under Γ, and $\mathit{NC}(\varphi, \alpha, P, \Gamma)$ denote that φ is the NC of α on P under Γ.

($\Rightarrow$) if $\mathit{SNC}(\varphi, \alpha, P, \Gamma)$ holds, then $\mathit{SNC}(\varphi, q, P, \Gamma')$ will be true. According to $\mathit{SNC}(\varphi, \alpha, P, \Gamma)$ and $\alpha \equiv q$, we have $\Gamma' \models q \to \varphi$, which means φ is a NC of q on P under Γ'. Suppose φ' is any NC of q on P under Γ', then $\mathrm{F_{CTL}}(\Gamma', q) \models \alpha \to \varphi'$ due to $\alpha \equiv q$, $\mathit{IR}(\alpha \to \varphi', \{q\})$ and **(pp)**, *i.e.* $\Gamma \models \alpha \to \varphi'$ by Lemma 4, this means $\mathit{NC}(\varphi', \alpha, P, \Gamma)$. Therefore, $\Gamma \models \varphi \to \varphi'$ by the definition of SNC and $\Gamma' \models \varphi \to \varphi'$. Hence, $\mathit{SNC}(\varphi, q, P, \Gamma')$ holds.

($\Leftarrow$) if $\mathit{SNC}(\varphi, q, P, \Gamma')$ holds, then $\mathit{SNC}(\varphi, \alpha, P, \Gamma)$ will be true. According to $\mathit{SNC}(\varphi, q, P, \Gamma')$, it's not difficult to know that $\mathrm{F_{CTL}}(\Gamma', \{q\}) \models \alpha \to \varphi$ due to $\alpha \equiv q$, $\mathit{IR}(\alpha \to \varphi, \{q\})$ and **(pp)**, *i.e.* $\Gamma \models \alpha \to \varphi$ by Lemma 4, this means $\mathit{NC}(\varphi, \alpha, P, \Gamma)$. Suppose φ' is any NC of α on P under Γ. Then $\Gamma' \models q \to \varphi'$ since $\alpha \equiv q$ and $\Gamma' = \Gamma \cup \{q \equiv \alpha\}$, which means $\mathit{NC}(\varphi', q, P, \Gamma')$. According to $\mathit{SNC}(\varphi, q, P, \Gamma')$, $\mathit{IR}(\varphi \to \varphi', \{q\})$ and **(pp)**, we have $\mathrm{F_{CTL}}(\Gamma', \{q\}) \models \varphi \to \varphi'$, and $\Gamma \models \varphi \to \varphi'$ by Lemma 4. Hence, $\mathit{SNC}(\varphi, \alpha, P, \Gamma)$ holds. ■

Theorem 8 **Proof:** We will prove the SNC part, while it is not difficult to prove the WSC part according to Proposition 9. Let $\mathcal{F} = \mathrm{F_{CTL}}(\varphi \wedge q, (\mathit{Var}(\varphi) \cup \{q\}) - V)$.

The "NC" part: It's easy to see that $\varphi \wedge q \models \mathcal{F}$ by **(W)**. Hence, $\varphi \models q \to \mathcal{F}$, this means $\mathcal{F}$ is a NC of q on P under φ.

The "SNC" part: for all ψ', ψ' is the NC of q on V under φ, s.t. $\varphi \models \mathcal{F} \to \psi'$. Suppose that there is a NC ψ of q on V under φ and ψ is not logic equivalence with $\mathcal{F}$ under φ, s.t. $\varphi \models \psi \to \mathcal{F}$. We know that $\varphi \wedge q \models \psi$ iff $\mathcal{F} \models \psi$ by **(PP)**, since $\mathit{IR}(\psi, (\mathit{Var}(\varphi) \cup \{q\}) - V)$. Hence, $\varphi \wedge \mathcal{F} \models \psi$ by $\varphi \wedge q \models \psi$ (by suppose). We can see that $\varphi \wedge \psi \models \mathcal{F}$ by suppose. Therefore, $\varphi \models \psi \leftrightarrow \mathcal{F}$, which means ψ is logic equivalence with $\mathcal{F}$ under φ. This is contradict with the suppose. Then $\mathcal{F}$ is the SNC of q on P under φ. ■

Theorem 9:
Proof: (i) As we know that any initial K-structure $\mathcal{K}$ can be described as a characterizing formula $\mathcal{F}_{\mathcal{A}}(\mathcal{K})$, then the SNC of q on V under $\mathcal{F}_{\mathcal{A}}(\mathcal{K})$ is $\mathrm{F_{CTL}}(\mathcal{F}_{\mathcal{A}}(\mathcal{K}) \wedge q, \mathcal{A} - V)$.

(ii) This is proved by the dual property. ■

Proposition11 **Proof:** The time and space spend by Algorithm 1 is mainly the **for** cycles from sentence 4 to 16. Under a given number i of states, there are i^i number of relations, i^m number of label functions and i number of possible initial states. In the case, we need the memory for the initial K-model in each time is $(i + i^2 + i * m + 1)$. Therefore, in the worst case is $i = 2^m$, that is we need $(2^m + 2^{2m} + m * 2^m + 1)$ memory to store the initial K-model.

For the time complexity, for each $1 \le i \le 2^m$, there is at most $i * i^i * i^m * i = i^2 * 2^{(i+m)}$ possible initial K-models. If we suppose that we can obtain an initial K-models in unit time, then in there will spend $(2^m)^2 * 2^{2^m + m}$ unit time in the worst case. Therefore, the time complexity is $O(2^{2^m})$. ■

References

[Baier and Katoen, 2008] Christel Baier and JoostPieter Katoen. *Principles of Model Checking*. The MIT Press, 2008.

[Bolotov, 1999] Alexander Bolotov. A clausal resolution method for ctl branching-time temporal logic. *Journal of Experimental & Theoretical Artificial Intelligence*, 11(1):77–93, 1999.

[Browne *et al.*, 1988] Michael C. Browne, Edmund M. Clarke, and Orna Grumberg. Characterizing finite kripke structures in propositional temporal logic. *Theor. Comput. Sci.*, 59:115–131, 1988.

[Clarke and Emerson, 1981] Edmund M Clarke and E Allen Emerson. Design and synthesis of synchronization skeletons using branching time temporal logic. In *Workshop on Logic of Programs*, pages 52–71. Springer, 1981.

[Clarke *et al.*, 1986] Edmund M. Clarke, E. Allen Emerson, and A. Prasad Sistla. Automatic verification of finite-state concurrent systems using temporal logic specifications. *ACM Trans. Program. Lang. Syst.*, 8(2):244–263, 1986.

[Dailler *et al.*, 2018] Sylvain Dailler, David Hauzar, Claude Marché, and Yannick Moy. Instrumenting a weakest precondition calculus for counterexample generation. *Journal of logical and algebraic methods in programming*, 99:97–113, 2018.

[Dijkstra, 1978] Edsger W Dijkstra. Guarded commands, nondeterminacy, and formal derivation of programs. In *Programming Methodology*, pages 166–175. Springer, 1978.

[Ding and Zhang, 2006] Yulin Ding and Yan Zhang. Ctl model update: Semantics, computations and implementation. In *ECAI*, pages 362–366, 2006.

[Eiter and Kern-Isberner, 2019] Thomas Eiter and Gabriele Kern-Isberner. A brief survey on forgetting from a knowledge representation and reasoning perspective. *KI-Künstliche Intelligenz*, 33(1):9–33, 2019.

[Eiter and Wang, 2008] Thomas Eiter and Kewen Wang. *Semantic forgetting in answer set programming*. Elsevier Science Publishers Ltd., 2008.

[Fang *et al.*, 2019] Liangda Fang, Yongmei Liu, and Hans Van Ditmarsch. Forgetting in multi-agent modal logics. *Artificial Intelligence*, 266:51–80, 2019.

[Lang and Marquis, 2008] Jerome Lang and Pierre Marquis. On propositional definability. *Artificial Intelligence*, 172(8):991–1017, 2008.

[Lang and Marquis, 2010] Jerome Lang and Pierre Marquis. *Reasoning under inconsistency: a forgetting-based approach*. Elsevier Science Publishers Ltd, 2010.

[Lang *et al.*, 2003] Jérôme Lang, Paolo Liberatore, and Pierre Marquis. Propositional independence: Formula-variable independence and forgetting. *J. Artif. Intell. Res.*, 18:391–443, 2003.

[Lin and Reiter, 1994] Fangzhen Lin and Ray Reiter. Forget it. In *Working Notes of AAAI Fall Symposium on Relevance*, pages 154–159, 1994.

[Lin, 2001] Fangzhen Lin. On strongest necessary and weakest sufficient conditions. *Artif. Intell.*, 128(1-2):143–159, 2001.

[Lin, 2003] Fangzhen Lin. Compiling causal theories to successor state axioms and strips-like systems. *Journal of Artificial Intelligence Research*, 19:279–314, 2003.

[Lin, 2018] Fangzhen Lin. Machine theorem discovery. *AI Magazine*, 39(2):53–59, 2018.

[Liu and Wen, 2011] Yongmei Liu and Ximing Wen. On the progression of knowledge in the situation calculus. In *IJCAI 2011, Proceedings of the 22nd International Joint Conference on Artificial Intelligence*, pages 976–982, Barcelona, Catalonia, Spain, 2011. IJCAI/AAAI.

[Lutz and Wolter, 2011] Carsten Lutz and Frank Wolter. Foundations for uniform interpolation and forgetting in expressive description logics. In *IJCAI 2011, Proceedings of the 22nd International Joint Conference on Artificial Intelligence*, pages 989–995, Barcelona, Catalonia, Spain, 2011. IJCAI/AAAI.

[Martinez-Araiza and López-Mellado, 2016] Ulises Martinez-Araiza and Ernesto López-Mellado. Ctl model repair for inter-organizational business processes modelled as owfn. *IFAC-PapersOnLine*, 49(2):6–11, 2016.

[Meier *et al.*, 2015] Arne Meier, Michael Thomas, Heribert Vollmer, and Martin Mundhenk. Erratum: The complexity of satisfiability for fragments of CTL and ctl*. *Int. J. Found. Comput. Sci.*, 26(8):1189, 2015.

[Ramdani *et al.*, 2019] Mohamed Ramdani, Laid Kahloul, Mohamed Khalgui, and Yousra Hafidi. R-tnces rebuilding: A new method of ctl model update for reconfigurable systems. In *Proceedings of the 14th International Conference on Evaluation of Novel Approaches to Software Engineering*, pages 159–168. SCITEPRESS-Science and Technology Publications, Lda, 2019.

[Su *et al.*, 2009] Kaile Su, Abdul Sattar, Guanfeng Lv, and Yan Zhang. Variable forgetting in reasoning about knowledge. *Journal of Artificial Intelligence Research*, 35:677–716, 2009.

[Tremblay, 1996] Jean-Paul Tremblay. *Logic and Discrete Mathematics: a Computer Science Perspective*. Upper Saddle River, NJ: Prentice Hall, 1996.

[Wang *et al.*, 2010] Zhe Wang, Kewen Wang, Rodney W. Topor, and Jeff Z. Pan. Forgetting for knowledge bases in DL-Lite. *Annuals of Mathematics and Artificial Intelligence*, 58(1-2):117–151, 2010.

[Wang *et al.*, 2012] Yisong Wang, Yan Zhang, Yi Zhou, and Mingyi Zhang. Forgetting in logic programs under strong equivalence. In *Principles of Knowledge Representation and Reasoning: Proceedings of the Thirteenth International Conference*, pages 643–647, Rome, Italy, 2012. AAAI Press.

[Wang *et al.*, 2013] Yisong Wang, Kewen Wang, and Mingyi Zhang. Forgetting for answer set programs revisited. In *IJCAI 2013, Proceedings of the 23rd International Joint Conference on Artificial Intelligence*, pages 1162–1168, Beijing, China, 2013. IJCAI/AAAI.

[Wong, 2009] Ka-Shu Wong. *Forgetting in Logic Programs*. PhD thesis, The University of New South Wales, 2009.

[Woodcock and Morgan, 1990] Jim CP Woodcock and Carroll Morgan. Refinement of state-based concurrent systems. In *International Symposium of VDM Europe*, pages 340–351. Springer, 1990.

[Zhang and Foo, 2006] Yan Zhang and Norman Y. Foo. Solving logic program conflict through strong and weak forgettings. *Artificial Intelligence*, 170(8-9):739–778, 2006.

[Zhang and Zhou, 2009] Yan Zhang and Yi Zhou. Knowledge forgetting: Properties and applications. *Artificial Intelligence*, 173(16-17):1525–1537, 2009.

[Zhang *et al.*, 2005] Yan Zhang, Norman Y. Foo, and Kewen Wang. Solving logic program conflict through strong and weak forgettings. In *Ijcai-05, Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence, Edinburgh, Scotland, Uk, July 30-August*, pages 627–634, 2005.

[Zhao and Schmidt, 2017] Yizheng Zhao and Renate A Schmidt. Role forgetting for alcoqh (δ)-ontologies using an ackermann-based approach. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pages 1354–1361. AAAI Press, 2017.