# On Sufficient and Necessary Conditions in CTL with Finite States: a preliminary report

Renyan Feng and Yisong Wang

School of Computer Science and Technology, Guizhou University, Guiyang, China
yswang@gzu.edu.cn

**Abstract.** Model checking is an automatic, model-based, property-verification approach. It is intended to be used for concurrent, reactive systems and originated as a post-development methodology. Specification, which prescribes what the system has to do and what not, is used to product the properties that a system should satisfy. Computation Tree Logic (CTL) is one of the main logical formalisms for program specification and verification. In this paper, we study forgetting in CTL from the sematic forgetting point of view to distill from a set of specifications only the part that is relevant to a subset of the alphabet. And we show that a CTL system is closed under forgetting. Besides, in order to obtain the most general precondition that satisfy some properties under the given specification, we propose the concepts of strongest necessary and weakest sufficient conditions in CTL. And then explore the relation between forgetting and the conditions. It is shown that the strongest necessary and weakest sufficient conditions can be computed by using the technology of forgetting.

**Keywords:** Forgetting · CTL · Bisimulation · Strongest necessary condition · Weakest sufficient condition.

## 1 Introduction

Model checking is an automatic, model-based, property-verification approach. It is intended to be used for concurrent, reactive systems and originated as a post-development methodology. Concurrency bugs are among the most difficult to find by testing (the activity of running several simulations of important scenarios), since they tend to be non-reproducible or not covered by test cases, so it is well worth having a verification technique that can help one to find them.

Model checking (unlike Alloy) focuses explicitly on temporal properties and the temporal evolution of systems. Specification, which prescribes what the system has to do and what not, is used to product the properties that a system should satisfy. The system is considered to be correct whenever it satisfies all properties obtained from its specification. Computation Tree Logic (CTL) is one of the main logical formalisms for program specification and verification. It has been proved that model checking is P-complete [6] and satisfiability checking is EXPTIME-complete [17] in CTL. The CTL language can be used to described the properties of a system. However, with the size of the system growing, not only has the number of available (proposition) increased considerably, but they are often large in size and are becoming more complex to manage. This leads to the specification being difficult to maintain and modify, and costly to reuse for later processing, where only a specific part of an specification is of interest. Both working directly on the whole of the original specification and building a new sub-specification are unadvisable. Therefore, a strong demand for techniques and automated tools for obtaining the specific sub-specification.The *forgetting* - distilling from a knowledge base only the part that is relevant to a subset of the alphabet - is used to obtain the sub-specification.

As a logic notion, forgetting was proposed early and studied in 1994 [15], which has been used in various of fields in Artificial intelligence, such as in conflict solving [12, 18], knowledge compilation [19, 2], createing restricted views of ontologies [21], strongest and weakest definitions [11], strongest necessary and weakest sufficient conditions [13] and so on. After that, (semantic) forgetting is studied by researchers in [16, 9]. Forgetting can be defined in two closely related ways; it can be defined on the syntactic level as the dual of uniform interpolation [10] and it can be defined model-theoretically as semantic forgetting [22]. Besides, forgetting, is regarded as an abstract belief change operator, independent of the underlying logic, is reached by James P. [8]. Though forgetting has been extensively investigated from various aspects of different logical systems, in standard propositional logic, a general algorithm of forgetting and its computation-oriented investigation in CTL are still lacking. In this paper, we explore the forgetting in CTL from a semantic forgetting point of view.

The strongest necessary condition (SNC) is the most general consequent and the weakest sufficient condition (WSC) is the most general precondition. In the past decades, the SNC and WSC have been investigated in Classical Logic (CL) [14] and it has many applications in knowledge representation and reasoning. In general, it is difficult to obtain such conditions for satisfying some properties under the given specification. We explore the relation between forgetting and SNC (WSC) to find a method to compute the SNC (WSC) in CTL.

The rest of the paper is organised as follows. Section 2 defines basic notions of the problem of concept forgetting, including the syntax and semantics of CTL, the language that our proposed method is aimed for. A formal definition of concept forgetting for CTL follows in Section 3. Section 4 explores the relation between forgetting and SNC (WSC). We conclude in Section 5 with a summary of the work and an outline of directions of future work.

## 2 Preliminaries

In the following we briefly review the basic syntax and semantics of the *Computation Tree Logic* (CTL in short) [7]. 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 and U, that means 'neXt state', 'some Future state', 'all future states (Globally)' and 'Until', respectively;
- parentheses: ( and ).

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

$$\phi ::= \bot \mid p \mid \neg\phi \mid \phi \vee \phi \mid \mathrm{EX}\phi \mid \mathrm{EG}\phi \mid \mathrm{E}[\phi \;\mathrm{U}\; \phi] \tag{1}$$

where $p \in \mathcal{A}$. The formulas $\phi \wedge \psi$ and $\phi \rightarrow \psi$ are defined in a standard manner of propositional logic. Intuitively, the formula $\mathrm{EX}\phi$ means that $\phi$ holds in some immediate successor of the current program state; the formula $\mathrm{EG}\phi$ means that for some computation path $\phi$ holds at every state along the path; and the formula $\mathrm{E}[\phi \mathrm{U} \psi]$ means that for some computation path there is an initial prefix of the path such that $\psi$ holds at the last state of the prefix and $\phi$ holds at all other states along the prefix. The other form formulas of $\mathcal{L}$ are abbreviated as follows using the forms of (1):

$$\begin{array}{ll}
\top =_{\text{def}} \neg\bot, & \mathrm{A}[\phi \;\mathrm{U}\; \psi] =_{\text{def}} \neg(\mathrm{E}[\neg\psi \mathrm{U}\,(\neg\phi \wedge \neg\psi)] \vee \mathrm{EG}\neg\psi)\,, \\
\mathrm{AF}\phi =_{\text{def}} \mathrm{A}[\top \mathrm{U}\; \phi], & \mathrm{EF}\phi =_{\text{def}} \mathrm{E}[\top \mathrm{U} \phi], \\
\mathrm{AG}\phi =_{\text{def}} \neg\mathrm{EF}\neg\phi, & \mathrm{AX}\phi =_{\text{def}} \neg\mathrm{EX}\neg\phi.
\end{array}$$

Notice that, according to the above definition for formulas of CTL, each of the CTL *temporal connectives* has the form $XY$ where $X \in \{\mathrm{A}, \mathrm{E}\}$ and $Y \in \{\mathrm{X}, \mathrm{F}, \mathrm{G}, \mathrm{U}\}$. The priorities for the CTL connectives are assumed to be (from the highest to the lowest):

$$\neg, \mathrm{EX}, \mathrm{EF}, \mathrm{EG}, \mathrm{AX}, \mathrm{AF}, \mathrm{AG} \prec \wedge \prec \vee \prec \mathrm{EU}, \mathrm{AU}, \rightarrow .$$

A *transition system* $\mathcal{M}$ (of $\mathcal{L}$) is a triple $(S, R, L)$ [1] 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 $S \rightarrow 2^{\mathcal{A}}$.

We call a transition system $\mathcal{M}$ on a set $V$ of atoms if $L : S \rightarrow 2^V$, *i.e.* , the labeling function $L$ map every state to a subset of $V$ (not the $\mathcal{A}$). A *path* $\pi$ of a transition system $\mathcal{M} = (S, R, L)$ is an infinite sequence

$$(s_1, s_2, \ldots, s_i, s_{i+1}, \ldots)$$

of states in $S$ such that $(s_i, s_{i+1}) \in R$ for each $i \geq 1$. By $\pi^i$ we denote the suffix of $\pi$ starting at $s_i$, *i.e.* , $\pi_i = (s_i, s_{i+1}, \ldots)$. A sate $s \in S$ is *initial* if for any state $s' \in S$, $\mathcal{M}$ has a path starting from $s$ such that $s'$ is on the path. For $s \in S$, let $\text{Post}(s) = \{s' \in S | (s, s') \in R\}$ and $\text{Pre}(s) = \{s' \in S | (s', s) \in R\}$. For $C \subseteq S$, let $\text{Post}(C) = \{s \in S | \exists s' \in C \; s.t. \; (s', s) \in R\}$ and $\text{Pre}(C) = \{s \in S | \text{Post}(s) \cap C \neq \emptyset\}$.

For a given transition system $(S, R, L)$ 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 [4], for $n \geq 0$,

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

- $\mathrm{Tr}_0(s)$ consists of a single node $s$ with label $s$.
- $\mathrm{Tr}_{n+1}(s)$ has as its root a node $m$ with label $s$, and if $(s, s') \in R$ then the node $m$ has a subtree $\mathrm{Tr}_n(s')$[2].

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

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

- $(\mathcal{M}, s) \not\models \bot$;
- $(\mathcal{M}, s) \models p$ if $p \in L(s)$;
- $(\mathcal{M}, s) \models \phi_1 \vee \phi_2$ if $(\mathcal{M}, s) \models \phi_1$ or $(\mathcal{M}, s) \models \phi_2$;
- $(\mathcal{M}, s) \models \neg\phi$ if $(\mathcal{M}, s) \not\models \phi$;
- $(\mathcal{M}, s) \models \mathrm{EX}\phi$ if $(\mathcal{M}, s_1) \models \phi$ for some $s_1 \in S$ and $(s, s_1) \in R$;
- $(\mathcal{M}, s) \models \mathrm{EG}\phi$ if $\mathcal{M}$ has a path $(s_1 = s, s_2, \ldots)$ such that $(\mathcal{M}, s_i) \models \phi$ for each $i \geq 1$;
- $(\mathcal{M}, s) \models \mathrm{E}[\phi_1 \mathrm{U} \phi_2]$ if $\mathcal{M}$ 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 $j < i$.

Similar to the work in [4, 3], only initial Kripke structures are considered to be candidate models in the following, unless explicitly stated. Formally, an initial Kripke structure $\mathcal{K}$ is a *model* of a formula $\phi$ whenever $\mathcal{K} \models \phi$. The formula $\phi$ is *satisfiable* if $\mathit{Mod}(\phi) \neq \emptyset$. We denote $\mathit{Mod}(\phi)$ the set of models of $\phi$. Since both the underlying states in transition systems and signatures are finite, $\mathit{Mod}(\phi)$ is finite for any formula $\phi$.

Let $\phi_1$ and $\phi_2$ be two formulas. By $\phi_1 \models \phi_2$ we denote $\mathit{Mod}(\phi_1) \subseteq \mathit{Mod}(\phi_2)$. By $\phi_1 \equiv \phi_2$ we mean $\phi_1 \models \phi_2$ and $\phi_2 \models \phi_1$. In this case $\phi_1$ is *equivalent* to $\phi_2$.

Let $\phi$ be a formula. By $\mathit{Var}(\phi)$ we mean the set of atoms occurring in $\phi$. Let $V \subseteq \mathcal{A}$. The formula $\phi$ is $V$*-irrelevant*, written $\mathrm{IR}(\phi, V)$, if there is a formula $\psi$ with $\mathit{Var}(\psi) \cap V = \emptyset$ such that $\phi \equiv \psi$.

## 3 Forgetting

In this section we will propose the concept of $V$-bisimulation, which is called the set-based bisimulation, between Kripke structures and the characterizing formula of an initial Kripke structure. After that the definition of forgetting will be proposed from a sematic forgetting point of view. Besides, some properties of forgetting are also explored.

### 3.1 Set-based bisimulation

Let $V \subseteq \mathcal{A}$. We define the relations $\mathcal{B}_0, \mathcal{B}_1, \ldots$ between Kripke structures as follows: let $\mathcal{K}_i = (\mathcal{M}_i, s_i)$ and $\mathcal{M}_i = \langle S_i, R_i, L_i \rangle$ with $i \in \{1, 2\}$,

[2] Though some nodes of the tree may have the same label, they are different nodes in the tree.

– $(\mathcal{K}_1, \mathcal{K}_2) \in \mathcal{B}_0$ 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 $i \in \{1, 2\}$.

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

**Lemma 1.** *Let $\mathcal{B}_0, \mathcal{B}_1, \ldots$ be the ones in the above definition. 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.*

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

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

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

In this case, $\mathcal{K}_1$ and $\mathcal{K}_2$ are called *$V$-bisimular*. It seems that two Kripke structures $\mathcal{K}_i = (\mathcal{M}_i, s_i)$ $(i = 1, 2)$ are $V$-bisimular if $s_1$ have the same labels as $s_2$ and for each successor $s_1'$ $(s_2')$ of $s_1$ $(s_2)$ there is a successor $s_2'$ $(s_1')$ of $s_2$ $(s_1)$ such that $(\mathcal{M}_1, s_1')$ and $(\mathcal{M}_2, s_2')$ are $V$-bisimular. Formally:

**Proposition 1.** *Let $V \subseteq \mathcal{A}$, $\mathcal{M}_i = (S_i, R_i, L_i)$ $(i = 1, 2)$ be transition systems and $\mathcal{K}_i = (\mathcal{M}_i, s_i)$ $(i = 1, 2)$ with $s_i \in S_i$ be Kripke 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 $i \in \{1, 2\}$.*

On the one hand, the above set-based bisimulation is an extension of the bisimulation-equivalence of Definition 7.1 in [1] 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 [4]. But it is different in the sense that ours is defined on Kripke structures, while it is defined on states in [4]. What's more, the set-based bisimulation notion is also different from the state-based bisimulation notion of Definition 7.7 in [1], which is defined for states of a given Kripke structure.

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 transition systems 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.The next lemma easily follows from the above definition,

**Lemma 2.** *The relation $\leftrightarrow_V$ is reflexive, symmetric and transitive.*

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

Besides, the $V$-bisimulation has the union property on the sets of atoms, that is if two Kripke structures are $V_i$-bisimular ($i = 1, 2$) respectively then they are $(V_1 \cup V_2)$-bisimular. Formally:

**Proposition 2.** ***(union)*** *Let $i \in \{1, 2\}$, $V_i \subseteq \mathcal{A}$, $s_i s$ be two states and $\pi_i s$ be two pathes. 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* $\pi_1 \leftrightarrow_{V_1 \cup V_2} \pi_2$.

Except the union property, the $V$-bisimulation has also another important property, called transitivity. That is:

**Proposition 3.** ***(transitivity)*** *Let $V_1, V_2 \subseteq \mathcal{A}$ ($V_1 \cap V_2 = \emptyset$) and $\mathcal{K}_i = (\mathcal{M}_i, s_i)$ $(i = 1, 2, 3)$ be Kripke structures such that $\mathcal{K}_1 \leftrightarrow_{V_1} \mathcal{K}_2$ and $\mathcal{K}_2 \leftrightarrow_{V_2} \mathcal{K}_3$. Then:*

*(i) for each path $\pi_1 = (s_1 = s_{1,0}, s_{1,1}, s_{1,2}, \dots)$ of $\mathcal{M}_1$ there is a path $\pi_2 = (s_2 = s_{2,0}, s_{2,1}, s_{2,2}, \dots)$ of $\mathcal{M}_2$ such that $\pi_1 \leftrightarrow_{V_1} \pi_2$, and vice versa;*
*(ii)* $\mathcal{K}_1 \leftrightarrow_{V_1 \cup V_2} \mathcal{K}_3$.

This is different with the transitivity of $\leftrightarrow_V$, which show the transitivity between Kripke structures on the same set of atoms.

**Proposition 4.** *($V$**-bisimular expansion**). Let $V \subseteq W \subseteq \mathcal{A}$ and $\mathcal{K}_i = (\mathcal{M}_i, s_i)$ $(i = 1, 2)$ with $\mathcal{M}_i = (S_i, R_i, L_i)$ be two Kripke structures. If $\mathcal{K}_1 \leftrightarrow_V \mathcal{K}_2$ then $\mathcal{K}_1 \leftrightarrow_W \mathcal{K}_2$.*

*Proof.* We will show that $(\mathcal{K}_1, \mathcal{K}_2) \in \mathcal{B}_n$ 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 $(\mathcal{K}_1, \mathcal{K}_2) \in \mathcal{B}_{k+1}^W$.

(i) It is apparent that $L_1(s_1) - W = L_2(s_2) - W$ by base.
(ii) $\forall (s_1, s_{1,1}) \in R_1$, we will show that there is a $(s_2, s_{2,1}) \in R_2$ *s.t.* $(\mathcal{K}_{1,1}, \mathcal{K}_{2,1}) \in \mathcal{B}_k^W$. $(\mathcal{K}_{1,1}, \mathcal{K}_{2,1}) \in \mathcal{B}_{k-1}^W$ 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 $(s_{2,k}, s_{2,k+1}) \in R_2$ *s.t.* $(\mathcal{K}_{1,k+1}, \mathcal{K}_{2,k+1}) \in \mathcal{B}_0^W$ 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 $(s_{1,k}, s_{1,k+1}) \in R_1$ *s.t.* $(\mathcal{K}_{1,k+1}, \mathcal{K}_{2,k+1}) \in \mathcal{B}_0^W$ due to $(\mathcal{K}_{1,1}, \mathcal{K}_{2,1}) \in \mathcal{B}_k^V$. This can be proved as (a).

(iii) $\forall (s_2, s_{2,1}) \in R_1$, we will show that there is a $(s_1, s_{1,1}) \in R_2$ *s.t.* $(\mathcal{K}_{1,1}, \mathcal{K}_{2,1}) \in \mathcal{B}_k^W$. 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 $(s_i, s_{i,1}, s_{i,2}, \dots, s_{i,k+1}, \dots)$ $(i = 1, 2)$.

Let $\mathcal{M} = (S, R, L)$ be a transition system on a finite set $\mathcal{A}$ of atoms, $V \subseteq \mathcal{A}$ and $\mathcal{B} = \{(\mathcal{M}, s) \leftrightarrow_V (\mathcal{M}, s') | s, s' \in S\}$. For $s \in S$, $[s]_{\mathcal{B}}$ denotes the equivalence class of state $s$ under $\mathcal{B}$, *i.e.* , $[s]_{\mathcal{B}} = \{s' \in S | (s, s') \in \mathcal{B}\}$. Note that for $s' \in [s]_{\mathcal{B}}$ we have $[s']_{\mathcal{B}} = [s]_{\mathcal{B}}$. The set $[s]_{\mathcal{B}}$ is referred to as the $\mathcal{B}$-equivalence class of $s$. The $V$-quotient space of $S$ under $\mathcal{B}$, denoted by $S/\mathcal{B} = \{[s]_{\mathcal{B}} | s \in S\}$, is the consisting of all $\mathcal{B}$-equivalence classes.

**Definition 2.** *For Kripke structure $\mathcal{K} = (\mathcal{M}, s_0)$ with $\mathcal{M} = (S, R, L)$ a transition system on a finite set $\mathcal{A}$ of atoms, $V \subseteq \mathcal{A}$ and $\mathcal{B} = \{(\mathcal{M}, s) \leftrightarrow_{V'} (\mathcal{M}, s') | s, s' \in S\}$ where $V' = \mathcal{A} - V$, the $V$-quotient Kripke structure is $\mathcal{K}_{|V} = (\mathcal{M}^*, s_0^*)$ with $\mathcal{M}^* = (S^*, R^*, L^*)$ on $V$, where*

- *$s_0^*$ is an element of $[s_0]_{\mathcal{B}}$,*
- *$S^* = S/\mathcal{B}$,*
- *$R^* = \{([s]_{\mathcal{B}}, [s']_{\mathcal{B}}) | \exists s_1 \in [s]_{\mathcal{B}}$ s.t. $\exists s_2 \in [s']_{\mathcal{B}}$ and $(s_1, s_2) \in R\}$ and*
- *$L^*([s]_{\mathcal{B}}) = L([s]_{\mathcal{B}}) \cap V$.*

**Proposition 5.** *For any Kripke $\mathcal{K} = (\mathcal{M}, s_0)$ with $\mathcal{M} = (S, R, L)$ a transition system on a finite set $\mathcal{A}$ of atoms and $V \subseteq \mathcal{A}$, it holds that $\mathcal{K} \leftrightarrow_{V'} \mathcal{K}_{|V}$ where $V' = \mathcal{A} - V$.*

*Proof.* Base. It is apparent that $L(s_0) - V' = L * (s_0^*) - V'$;
Step. (i) For any $(s_0, s_1) \in R$ there is $s_1' \in [s_1]_{\mathcal{B}}$ such that $([s_0^*]_{\mathcal{A}}, [s_1']_{\mathcal{A}}) \in R^*$ and $s_1 \leftrightarrow_{V'} [s_1']_{\mathcal{B}}$ by the Definition 2;
(ii) Similarly, for any $([s_0^*]_{\mathcal{A}}, [s_1']_{\mathcal{B}}) \in R^*$ there is $(s_0, s_1) \in R$ such that $s_1 \in [s_1']_{\mathcal{B}}$ and $s_1 \leftrightarrow_{V'} [s_1']_{\mathcal{B}}$.

*Example 1.* Let $\mathcal{M}$ as Fig. 1(a), $A = \{a, b, c\}$, $V = \{b\}$ and $\mathcal{K} = (\mathcal{M}, s_0)$. Then compute $\mathcal{K}_{|V}$.

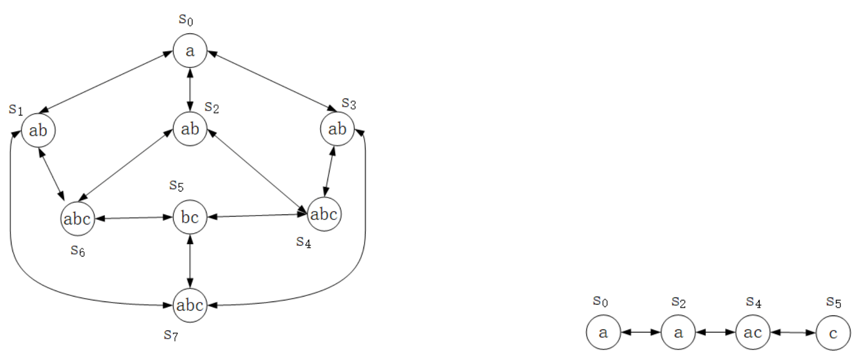
For convenience, by $(s, s') \in B_i$ for $i \geq 0$ we mean $((\mathcal{M}, s), (\mathcal{M}, s')) \in B_i$ due to $s$ and $s'$ are under the same transition system. It is easy to check that:

(a) $(s_0, s_i) \in \mathcal{B}_0$, $(s_0, s_i) \notin \mathcal{B}_1$ with $i = 1, 2, 3$, then we have $\mathrm{Tr}_0(s_0) \leftrightarrow_V \mathrm{Tr}_0(s_i)$ and $\mathrm{Tr}_1(s_0) \not\leftrightarrow_V \mathrm{Tr}_1(s_i)$. Hence, $\mathcal{M}$ is $V$-distinguished by $s_0$ and $s_i$ at the least depth 1, *i.e.* $\mathrm{dis}_V(\mathcal{M}, s_0, s_i, 1)$;
(b) $(s_0, s_5) \notin \mathcal{B}_0$, then we have $\mathrm{Tr}_0(s_0) \not\leftrightarrow_V \mathrm{Tr}_0(s_5)$ and then $\mathcal{M}$ is $V$-distinguished by $s_0$ and $s_5$ at the least depth 0, *i.e.* $\mathrm{dis}_V(\mathcal{M}, s_0, s_5, 0)$;
(c) $(s_0, s_i) \notin \mathcal{B}_0$ with $i = 4, 6, 7$, then we have $\mathrm{Tr}_0(s_0) \not\leftrightarrow_V \mathrm{Tr}_0(s_i)$ and then $\mathcal{M}$ is $V$-distinguished by $s_0$ and $s_i$ at the least depth 0, *i.e.* $\mathrm{dis}_V(\mathcal{M}, s_0, s_i, 0)$;

(d) $(s_i, s_j) \notin \mathcal{B}_0$ with $i = 1,2,3$ and $j = 4,6,7$, then we have $\text{Tr}_0(s_i) \not\leftrightarrow_V \text{Tr}_0(s_j)$ and then $\mathcal{M}$ is $V$-distinguished by $s_i$ and $s_j$ at the least depth $0$, *i.e.* $\text{dis}_V(\mathcal{M}, s_i, s_j, 0)$;
(e) $(s_i, s_5) \notin \mathcal{B}_0$ with $i = 1,2,3$, then we have $\text{Tr}_0(s_i) \not\leftrightarrow_V \text{Tr}_0(s_5)$ and then $\mathcal{M}$ is $V$-distinguished by $s_i$ and $s_5$ at the least depth $0$, *i.e.* $\text{dis}_V(\mathcal{M}, s_i, s_5, 0)$;
(f) $(s_i, s_5) \notin \mathcal{B}_0$ with $i = 4,6,7$, then we have $\text{Tr}_0(s_i) \not\leftrightarrow_V \text{Tr}_0(s_5)$ and then $\mathcal{M}$ is $V$-distinguished by $s_i$ and $s_5$ at the least depth $0$, *i.e.* $\text{dis}_V(\mathcal{M}, s_i, s_5, 0)$.

Therefore, we have that $\mathcal{M}$ is $V$-distinguishable and $ch(\mathcal{M}, V) = \max\{k \mid s, s' \in S \ \&\ \text{dis}_V(\mathcal{M}, s, s', k)\} = 1$. In order to show that $(s, s') \in \mathcal{B}$, we only need to show that $(s, s') \in \mathcal{B}_i$ with $i = 0, 1$. It is easy to show that $(s_1, s_2) \in \mathcal{B}_0$ and $(s_1, s_2) \in \mathcal{B}_1$.

Hence, we can obtain $\mathcal{B} = \{(s_1, s_2), (s_2, s_1), (s_2, s_3), (s_3, s_2), (s_1, s_3), (s_3, s_1), (s_6, s_7), (s_7, s_6), (s_6, s_4), (s_4, s_6), (s_4, s_7), (s_7, s_4)\} \cup I_S$. Then we have the $V$-quotient Kripke structure is as Fig. 1(b).




(a) Transition system $\mathcal{M}$

(b) $V$-quotient Kripke structure of $\mathcal{M}$

**Fig. 1.** Computing the $V$-quotient Kripke structure

Intuitively, if two Kripke structures are $V$-bisimular, then they satisfy the same formula $\varphi$ that dose not contain any atoms in $V$, *i.e.* $\text{IR}(\varphi, V)$.

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

Let $V \subseteq \mathcal{A}$, $\mathcal{M}_i = (S_i, R_i, L_i)$ $(i = 1,2)$ be transition systems. 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
- for every subtree $\text{Tr}_{n-1}(s_2')$ of $\text{Tr}_n(s_2)$, $\text{Tr}_n(s_1)$ has a subtree $\text{Tr}_{n-1}(s_1')$ such that $\text{Tr}_{n-1}(s_1') \leftrightarrow_V \text{Tr}_{n-1}(s_2')$.

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

**Proposition 6.** *Let $V \subseteq \mathcal{A}$ and $(\mathcal{M}_i, s_i)$ $(i = 1, 2)$ be two Kripke structures. Then*

$$(s_1, s_2) \in \mathcal{B}_n \text{ if and only if } Tr_j(s_1) \leftrightarrow_V Tr_j(s_2) \text{ for every } 0 \leq j \leq n.$$

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

**Proposition 7.** *Let $V \subseteq \mathcal{A}$, $\mathcal{M} = (S, R, L)$ be a transition system 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 transition $\mathcal{M}$ is called *$V$-distinguishable* (by states $s$ and $s'$ at the least depth $k$), which is denoted by $\mathrm{dis}_V(\mathcal{M}, s, s', k)$. It is evident that $\mathrm{dis}_V(\mathcal{M}, s, s', k)$ implies $\mathrm{dis}_V(\mathcal{M}, s, s', k')$ whenever $k' \geq k$. The *$V$-characterization number* of $\mathcal{M}$, written $ch(\mathcal{M}, V)$, is defined as

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

**Definition 3.** *Let $V \subseteq \mathcal{A}$, $\mathcal{M} = (S, R, L)$ be a transition system 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:*

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

$$\mathcal{F}_V(Tr_{k+1}(s)) = \left( \bigwedge_{(s,s') \in R} \mathrm{EX}\mathcal{F}_V(Tr_k(s')) \right) \wedge \mathrm{AX} \left( \bigvee_{(s,s') \in R} \mathcal{F}_V(Tr_k(s')) \right) \wedge \mathcal{F}_V(Tr_0(s))$$

*for $k \geq 0$.*

**Lemma 3.** *Let $V \subseteq \mathcal{A}$, $\mathcal{M} = (S, R, L)$ and $\mathcal{M}' = (S', R', L')$ be two transition systems, $s \in S$, $s' \in S'$ 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 $Tr_n(s) \leftrightarrow_{\overline{V}} Tr_n(s')$.*

A consequence of the previous lemma is:

**Lemma 4.** *Let $V \subseteq \mathcal{A}$, $\mathcal{M} = (S, R, L)$ a transition system, $k = ch(\mathcal{M}, V)$ 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 $\phi = \mathcal{F}_V(\mathrm{Tr}_k(s))$, where $c$ 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 $\mathrm{IR}(\phi, \mathcal{A} \setminus V)$. If $s \not\leftrightarrow_{\overline{V}} s'$, then $\mathrm{Tr}_c(s) \not\leftrightarrow_{\overline{V}} \mathrm{Tr}_c(s')$, and then $\mathcal{M}, s \not\models \phi$ by Lemma 3.

Let $V \subseteq \mathcal{A}$, $\mathcal{M} = (S, R, L)$ and $\mathcal{K} = (\mathcal{M}, s_0)$ be an initial Kripke structure. 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:

$$\mathcal{F}_V(\mathrm{Tr}_c(s_0)),\ \text{and}$$
$$\mathrm{AG}\left(\mathcal{F}_V(\mathrm{Tr}_c(s)) \rightarrow \bigwedge_{(s,s')\in R} \mathrm{EX}\mathcal{F}_V(\mathrm{Tr}_c(s')) \wedge \mathrm{AX} \bigvee_{(s,s')\in R} \mathcal{F}_V(\mathrm{Tr}_c(s'))\right),\ s \in S$$

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

**Lemma 5.** *Let $\varphi$ be a formula. We have*

$$\varphi \equiv \bigvee_{(\mathcal{M},s_0)\in Mod(\varphi)} \mathcal{F}_{\mathcal{A}}(\mathcal{M}, s_0). \tag{2}$$

This means 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 modes of a CTL formula is finite.

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

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

We will give an example to show the computing of characterizing formula:

*Example 2.* Let $\mathcal{K} = (\mathcal{M}, s_0)$ with $\mathcal{M} = (S, R, L)$ be a initial Kripke structure (in Fig. 2), 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 $\mathrm{Tr}_0(s_0) \leftrightarrow_{\overline{V}} \mathrm{Tr}_0(s_1)$ due to $L(s_0) - \overline{V} = L(s_1) - \overline{V}$, $\mathrm{Tr}_1(s_0) \not\leftrightarrow_{\overline{V}} \mathrm{Tr}_1(s_1)$ 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.* $\mathrm{dis}_{\overline{V}}(\mathcal{M}, s_0, s_1, 1)$. Similarly, we have $\mathrm{dis}_{\overline{V}}(\mathcal{M}, s_0, s_2, 0)$ and $\mathrm{dis}_{\overline{V}}(\mathcal{M}, s_1, s_2, 0)$. Therefore, $ch(\mathcal{M}, \overline{V}) = \max\{k \mid s, s' \in S \ \&\ \mathrm{dis}_{\overline{V}}(\mathcal{M}, s, s', k)\} = 1$. Then we have:

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

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

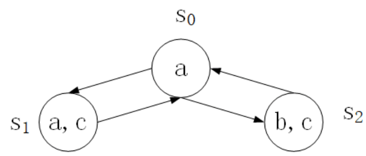



**Fig. 2.** A simple Kripke structure

### 3.2 Forgetting

Having talked about the $V$-bisimulation between two Kripke structures and the characterizing formula of a Kripke structure, we will give the definition of forgetting under CTL from the sematic forgetting point of view.

**Definition 4 (Forgetting).** *Let $V \subseteq \mathcal{A}$ and $\phi$ a formula. A formula $\psi$ with $Var(\psi) \cap V = \emptyset$ is a* result of forgetting $V$ from $\phi$*, if*

$$Mod(\psi) = \{\mathcal{K} \text{ is initial} \mid \exists \mathcal{K}' \in Mod(\phi) \ \& \ \mathcal{K}' \leftrightarrow_V \mathcal{K}\}. \tag{3}$$

Note that if both $\psi$ and $\psi'$ are results of forgetting $V$ from $\phi$ then $Mod(\psi) = Mod(\psi')$, *i.e.* , $\psi$ and $\psi'$ have the same models. In the sense of equivalence the forgetting result is unique (up to equivalence). By Lemma 5, such a formula $\psi$ always exists, which is equivalent to

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

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

In the case $\psi$ is a result of forgetting $V$ from $\phi$, there are usually some expected properties (called *postulates*) for them [20]:

- Weakening (**W**): $\varphi \models \psi$;
- Positive Persistence (**PP**): if $\mathrm{IR}(\eta, V)$ and $\varphi \models \eta$, then $\psi \models \eta$;
- Negative Persistence (**NP**): if $\mathrm{IR}(\eta, V)$ and $\varphi \not\models \eta$, then $\psi \not\models \eta$;
- Irrelevance (**IR**): $\mathrm{IR}(\psi, V)$.

**Theorem 3.** *Let $\varphi$ and $\psi$ be two formulas and $V \subseteq \mathcal{A}$. Then the following statements are equivalent:*

*(i)* $\psi \equiv \mathrm{F}_{\mathrm{CTL}}(\varphi, V)$,
*(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 the forgetting under CTL is closed, *i.e.* for any CTL formula the result of forgetting is also a CTL formula.

**Lemma 6.** *Let $\varphi$ and $\alpha$ 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 8.** *Let $\varphi$ be a formula, $V$ a set of atoms and $p$ an atom such that $p \notin V$. Then:*

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

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

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

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

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

**Proposition 9.** *Let $\varphi$, $\varphi_i$, $\psi_i$ $(i = 1, 2)$ be formulas and $V \subseteq \mathcal{A}$. We have*

*(i)* $\mathrm{F}_{\mathrm{CTL}}(\varphi, V)$ *is satisfiable iff $\varphi$ is;*
*(ii)* *If $\varphi_1 \equiv \varphi_2$, then* $\mathrm{F}_{\mathrm{CTL}}(\varphi_1, V) \equiv \mathrm{F}_{\mathrm{CTL}}(\varphi_2, V)$*;*
*(iii)* *If $\varphi_1 \models \varphi_2$, then* $\mathrm{F}_{\mathrm{CTL}}(\varphi_1, V) \models \mathrm{F}_{\mathrm{CTL}}(\varphi_2, V)$*;*
*(iv)* $\mathrm{F}_{\mathrm{CTL}}(\psi_1 \vee \psi_2, V) \equiv \mathrm{F}_{\mathrm{CTL}}(\psi_1, V) \vee \mathrm{F}_{\mathrm{CTL}}(\psi_2, V)$*;*
*(v)* $\mathrm{F}_{\mathrm{CTL}}(\psi_1 \wedge \psi_2, V) \models \mathrm{F}_{\mathrm{CTL}}(\psi_1, V) \wedge \mathrm{F}_{\mathrm{CTL}}(\psi_2, V)$.

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

**Proposition 10.** *Let $V \subseteq \mathcal{A}$ and $\phi$ a formula.*

*(i)* $\mathrm{F}_{\mathrm{CTL}}(\mathrm{AX}\phi, V) \equiv \mathrm{AX}\mathrm{F}_{\mathrm{CTL}}(\phi, V)$.
*(ii)* $\mathrm{F}_{\mathrm{CTL}}(\mathrm{EX}\phi, V) \equiv \mathrm{EX}\mathrm{F}_{\mathrm{CTL}}(\phi, V)$.
*(iii)* $\mathrm{F}_{\mathrm{CTL}}(\mathrm{AF}\phi, V) \equiv \mathrm{AF}\mathrm{F}_{\mathrm{CTL}}(\phi, V)$.
*(iv)* $\mathrm{F}_{\mathrm{CTL}}(\mathrm{EF}\phi, V) \equiv \mathrm{EF}\mathrm{F}_{\mathrm{CTL}}(\phi, V)$.
*(v)* $\mathrm{F}_{\mathrm{CTL}}(\mathrm{AG}\phi, V) \equiv \mathrm{AG}\mathrm{F}_{\mathrm{CTL}}(\phi, V)$.

## 4 Computing Temporal Forgetting

The computing of Forgetting under a lots of logics, such as classical propositional logic (CPL), first-order logic, modal logic S5 (S5 in short) and so on, have been explored. The Forgetting of CPL formula $\varphi$ is easy, *i.e.* , *Forget*$(\varphi, \{q\}) = \varphi[q/\top] \vee \varphi[q/\bot]$, where $q$ is an atom. It is known that the method of Knowledge Forgetting of S5 is an extend of that of classical propositional logic. In this case, every S5 formula is translated a set of term and then computing the Knowledge Forgetting of each term by the method under CPL. For example, a term is the form $\varphi_0 \wedge \mathrm{K}\varphi_1 \wedge \mathrm{B}\varphi_2 \wedge ... \wedge \mathrm{B}\varphi_n$, where each $\varphi_i (0 \leq i \leq n)$ is a propositional formula, and any of $\varphi_i$ may be absent. Then *KForget*$(\varphi, \{q\}) =$ *Forget*$(\varphi_0, \{q\}) \wedge \mathrm{K}($*Forget*$(\varphi_1, \{q\})) \wedge \mathrm{B}($*Forget*$(\varphi_1 \wedge \varphi_2, \{q\})) \wedge ... \wedge \mathrm{B}($*Forget*$(\varphi_1 \wedge \varphi_n, \{q\}))$.

In this section, we will explore the method of computing temporal forgetting under CTL.

### 4.1 AG fragment

We will explore the method of computing temporal forgetting of AG fragment of CTL formula from the angle of classical propositional logic forgetting. The AG fragment of CTL can be defined recursively as follows:

$$\phi ::= \bot \mid p \mid \neg p \mid \phi \vee \phi \mid \phi \wedge \phi \mid \mathrm{AG}\phi \tag{4}$$

where $p$ is an atom. We can see from this definition that the AG fragment is a subset of CTL such that each formula is in negation normal form (NNF), that is the negation operator is only applied to atoms, and there is only AG temporal operator appearing in this formula. For convenience, we call the formula in AG fragment AG formula.

**Proposition 11.** *Every* AG *formula can be equivalently transformed into a formula without nested temporal operator in polynomial time.*

*Proof.* By applying the following rules, each AG formula can be transformed into a equivalent formula without nested temporal operators. These rules are as follows:

(1) $\mathrm{AG}(\varphi \wedge \mathrm{AG}\psi) \equiv \mathrm{AG}\varphi \wedge \mathrm{AG}\psi \equiv \mathrm{AG}(\varphi \wedge \psi)$,
(2) $\mathrm{AG}(\varphi \vee \mathrm{AG}\psi) \equiv \mathrm{AG}\varphi \vee \mathrm{AG}\psi$.

**Proposition 12.** *Every* AG *formula can be equivalently transformed into a disjunction of terms of the following form:*

$$\phi_0 \wedge \mathrm{AG}(\phi_1 \wedge \ldots \wedge \phi_n), \tag{5}$$

*where each $\phi_i (0 \leq i \leq n)$ is proposition formula, and any of $\phi_i$ may be absent.*

*Proof.* By Proposition 11, every formula can be transformed into a AG formula without nested modal operators. By replacing each temporal sub-formula with a new atom, we can view this formula as a propositional formula. Then this formula can be transformed into a disjunctive normal form (DNF). Notice that every term in the DNF is actually a AG formula, which can be transformed into the form (5) using rules illustrated in the proof of Proposition 11.

**Proposition 13.** *Let $\mathrm{AG}\varphi$ be a* AG *formula with $\varphi$ be a proposition formula and $q$ an atom. Then we have:*

$$\mathrm{F}_{\mathrm{CTL}}(\mathrm{AG}\varphi, \{q\}) \equiv \mathrm{AG}(\mathit{Forget}(\varphi, \{q\})).$$

*Proof.* Let $\mathcal{M} = (S, R, L), s_0$ and $\mathcal{M}' = (S', R', L'), s_0'$ be two initial structure.
($\Rightarrow$) $\forall \mathcal{M}', s_0' \models \mathrm{F}_{\mathrm{CTL}}(\mathrm{AG}\varphi, \{q\})$
$\Rightarrow \exists \mathcal{M}, s_0$, *s.t.* , $\mathcal{M}, s_0 \models \mathrm{AG}\varphi$ and $\mathcal{M}, s_0 \leftrightarrow_{\{q\}} \mathcal{M}', s_0'$
$\Rightarrow \forall s \in S$ there is $\mathcal{M}, s \models \varphi$ and then $\mathcal{M}, s \models \mathit{Forget}(\varphi, \{q\})$
$\Rightarrow \mathcal{M}, s_0 \models \mathrm{AG}(\mathit{Forget}(\varphi, \{q\}))$
$\Rightarrow \mathcal{M}', s_0' \models \mathrm{AG}(\mathit{Forget}(\varphi, \{q\}))$ (IR(AG($\mathit{Forget}(\varphi, \{q\})), \{q\}$))
($\Leftarrow$) $\forall \mathcal{M}, s_0 \models \mathrm{AG}(\mathit{Forget}(\varphi, \{q\}))$
$\Rightarrow \forall s \in S$ there is $\mathcal{M}, s \models \mathit{Forget}(\varphi, \{q\})$
$\Rightarrow \exists \mathcal{M}', s_0'$ *s.t.* $\mathcal{M}'.s_0' \leftrightarrow_{\{q\}} \mathcal{M}, s_0$ and $\forall s' \in S'$ there is $\mathcal{M}', s' \models \varphi$
$\Rightarrow \mathcal{M}', s_0' \models \mathrm{AG}\varphi$
$\Rightarrow \mathcal{M}, s_0 \models \mathrm{F}_{\mathrm{CTL}}(\mathrm{AG}\varphi, \{q\})$

**Theorem 4.** *Let* $\mathrm{AG}\varphi \wedge \psi$ *be a* $\mathrm{AG}$ *formula with* $\varphi$ *and* $\psi$ *are a proposition formulas and* $q$ *an atom. If* $\mathrm{AG}\varphi \wedge \psi$ *is satisfiable, then we have:*

$$\mathrm{F}_{\mathrm{CTL}}(\mathrm{AG}\varphi\wedge\psi, \{q\}) \equiv \mathrm{AG}(\mathit{Forget}(\varphi, \{q\}))\wedge \mathit{Forget}(\psi\wedge\varphi, \{q\}) \equiv \mathrm{AG}(\mathit{Forget}(\varphi, \{q\}))\wedge \mathit{Forget}(\psi, \{q\}).$$

*Proof.* For convenience, let $\mathcal{M} = (S, R, L)$, $s_0$ and $\mathcal{M}' = (S', R', L')$, $s_0'$ be two initial structure.

($\Rightarrow$) $\forall \mathcal{M}, s_0 \models \mathrm{F}_{\mathrm{CTL}}(\mathrm{AG}\varphi \wedge \psi, \{q\})$
$\Rightarrow \exists \mathcal{M}', s_0'$ *s.t.* $\mathcal{M}', s_0' \leftrightarrow_{\{q\}} \mathcal{M}, s_0$ and $\mathcal{M}', s_0' \models \mathrm{AG}\varphi \wedge \psi$
$\Rightarrow \mathcal{M}', s_0' \models \mathrm{F}_{\mathrm{CTL}}(\mathrm{AG}\varphi, \{q\}) \wedge \mathrm{F}_{\mathrm{CTL}}(\psi\wedge\varphi, \{q\})$ since $\mathrm{AG}\varphi\wedge\psi \equiv \mathrm{AG}\varphi\wedge\psi\wedge\varphi$ and (v)
$\Rightarrow \mathcal{M}, s_0 \models \mathrm{F}_{\mathrm{CTL}}(\mathrm{AG}\varphi, \{q\})$ and $\mathcal{M}, s_0 \models \mathrm{F}_{\mathrm{CTL}}(\psi\wedge\varphi, \{q\})$ (IR($\mathrm{F}_{\mathrm{CTL}}(\mathrm{AG}\varphi, \{q\}), \{q\}$), IR($\mathrm{F}_{\mathrm{CTL}}(\psi \wedge \varphi, \{q\}), \{q\}$), Theorem 1 or Def. of forgetting)
$\Rightarrow \mathcal{M}, s_0 \models \mathrm{AG}(\mathit{Forget}(\varphi, \{q\})) \wedge \mathit{Forget}(\psi \wedge \varphi, \{q\})$ (Proposition 13)
($\Leftarrow$) $\forall \mathcal{M}, s_0 \models \mathrm{AG}(\mathit{Forget}(\varphi, \{q\})) \wedge \mathit{Forget}(\psi \wedge \varphi, \{q\})$
$\Rightarrow \mathcal{M}, s_0 \models \mathit{Forget}(\psi \wedge \varphi, \{q\})$ and $\forall s \in S$ there is $\mathcal{M}, s \models \mathit{Forget}(\varphi, \{q\})$
$\Rightarrow \exists \mathcal{M}_1, s_1$ *s.t.* $\forall s' \in S_1$ there is $\mathcal{M}_1, s' \models \varphi$ and $\mathcal{M}, s_0 \leftrightarrow_{\{q\}} \mathcal{M}_1, s_1$
$\Rightarrow \exists \mathcal{M}_2, s_2 \models \psi$ s.t. $\mathcal{M}_2, s_2 \models \psi \wedge \varphi$ and $\mathcal{M}, s_0 \leftrightarrow_{\{q\}} \mathcal{M}_2, s_2$

We construct $\mathcal{M}' = (S', R', L', s_0')$ as follows:

- $S' = S_1$,
- $R' = R_1$,
- $L'(s) \equiv \begin{cases} L_2(s_2), & \text{if } s = s_0'; \\ L'(s), & \text{otherwise.} \end{cases}$

Then it is easy to check that $\mathcal{M}, s_0 \leftrightarrow_{\{q\}} \mathcal{M}', s_0'$.
$\Rightarrow \mathcal{M}', s_0' \models \mathrm{AG}\varphi$ and $\mathcal{M}', s_0' \models \psi$
$\Rightarrow \mathcal{M}', s_0' \models \mathrm{AG}\varphi \wedge \psi$
$\Rightarrow \mathcal{M}, s_0 \models \mathrm{F}_{\mathrm{CTL}}(\mathrm{AG}\varphi \wedge \psi, \{q\})$

**Proposition 14.** *Let* $\mathit{\Pi} = \psi \wedge \mathrm{AG}\varphi$ *with* $\psi$ *and* $\varphi$ *are DNF formula,* $\mathit{\Sigma}$ *be two formula in* $\mathrm{AG}$*-fragment and* $V \subset \mathcal{A}$*. Then have:*

*(i) deciding wether* $\mathit{\Sigma} \models \mathrm{F}_{\mathrm{CTL}}(\mathit{\Pi}, V)$ *is PSPACE-complete;*
*(ii) deciding wether* $\mathrm{F}_{\mathrm{CTL}}(\mathit{\Pi}, V) \models \mathit{\Sigma}$ *is PSPACE-complete;*
*(iii) deciding wether* $\mathrm{F}_{\mathrm{CTL}}(\mathit{\Pi}, V) \equiv \mathit{\Sigma}$ *is PSPACE-complete.*

*Proof.* (i) As $\mathrm{F}_{\mathrm{CTL}}(\mathit{\Pi}, V)$ is computable in polynomial time by Theorem 4 and the result of $\mathrm{F}_{\mathrm{CTL}}(\mathit{\Pi}, V)$ is also in $\mathrm{AG}$-fragment. Hence this problem is to decide the validity of $\mathit{\Sigma} \wedge \neg\mathrm{F}_{\mathrm{CTL}}(\mathit{\Pi}, V)$, which is the complement problem. It has proved that CTL-SAT($\{\mathrm{AG}\}, BF$) is PSPACE-complete [17], and then we obtain this result.

(ii) and (iii) can be proved as (i).

For convenience, by $\beta(\mathrm{AG}q)$ denote the formula that all the occurrence of $q$ in $\beta$ are of the form $\mathrm{AG}q$ and that $\mathrm{AG}q$ does not occur within a scope of a temporal operator. Then we have the following theorem:

**Theorem 5.** *Let* $\text{AG}(q \to \alpha) \wedge \beta(\text{AG}q)$ *be a* AG *formula with* $\alpha$ *does not contain* $q$ *and* $q$ *an atom. Then*

$$\text{F}_{\text{CTL}}(\text{AG}(q \to \alpha) \wedge \beta(\text{AG}q), \{q\}) \equiv \beta[q/\alpha],$$

*where* $\beta[q/\alpha]$ *denotes the formula obtained from* $\beta$ *by substituting* $\alpha$ *for every occurrence of* $q$ *in* $\beta$.

*Proof.* $(\Rightarrow)$ $\forall \mathcal{M}, s_0 \models \text{F}_{\text{CTL}}(\text{AG}(q \to \alpha) \wedge \beta(\text{AG}q), \{q\})$
$\Rightarrow \exists \mathcal{M}', s_0'$ *s.t.* $\mathcal{M}, s_0 \leftrightarrow_{\{q\}} \mathcal{M}', s_0'$ and $\mathcal{M}', s_0' \models \text{AG}(q \to \alpha) \wedge \beta(\text{AG}q)$
$\Rightarrow \mathcal{M}', s_0' \models \beta(\text{AG}q)$
$\Rightarrow \mathcal{M}', s_0' \models \beta[q/\alpha]$ due to $\beta$ is positive w.r.t. $\text{AG}q$
$\Rightarrow \mathcal{M}, s_0 \models \beta[q/\alpha]$ $\qquad (IR(\beta[q/\alpha], \{q\}))$

$(\Leftarrow)$ $\forall \mathcal{M}, s_0 \models \beta[q/\alpha]$ we can construct a initial structure $\mathcal{M}', s_0'$ as follows:

1. $R' = R$, $S' = S$ and $s_0' = s_0$;
2. $\forall s' \in S'$ there is $L'(s') = L(s')$ if $\mathcal{M}, s' \not\models \alpha$, else $L'(s') = L(s') \cup \{q\}$.

It is easy to see that $\mathcal{M}', s_0' \leftrightarrow_{\{q\}} \mathcal{M}, s_0$, and then we have $\mathcal{M}', s_0' \models \text{AG}(q \to \alpha) \wedge \beta(\text{AG}q)$, therefore $\mathcal{M}, s_0 \models \text{F}_{\text{CTL}}(\text{AG}(q \to \alpha) \wedge \beta(\text{AG}q), \{q\})$.

### 4.2 AF-fragment

- $\text{AF}(\varphi \vee \text{AF}\psi) \equiv \text{AF}\varphi \vee \text{AF}\psi$
- $\text{AF}(\varphi \wedge \text{AF}\psi) \equiv \text{AF}\varphi \wedge \text{AF}\psi \not\equiv \text{AF}(\varphi \wedge \psi)$

**Proposition 15.** *Let* $\text{AF}\varphi$ *be a* AF *formula with* $\varphi$ *be a proposition formula and* $q$ *an atom. Then we have:*

$$\text{F}_{\text{CTL}}(\text{AF}\varphi, \{q\}) \equiv \text{AF}(\textit{Forget}(\varphi, \{q\})).$$

*Proof.* Let $(\mathcal{M} = (S, R, L), s_0)$ and $(\mathcal{M}' = (S', R', L'), s_0')$ be two initial structure.
$(\Rightarrow)$ $\forall (\mathcal{M}', s_0') \models \text{F}_{\text{CTL}}(\text{AF}\varphi, \{q\})$
$\Rightarrow \exists (\mathcal{M}, s_0)$, *s.t.* , $(\mathcal{M}, s_0) \models \text{AF}\varphi$ and $(\mathcal{M}, s_0) \leftrightarrow_{\{q\}} (\mathcal{M}', s_0')$
$\Rightarrow$ for all path $\pi_{s_0}$, $\exists s \in \pi_{s_0}$ such that $(\mathcal{M}, s) \models \varphi$ and then $(\mathcal{M}, s) \models \textit{Forget}(\varphi, \{q\})$
$\Rightarrow (\mathcal{M}, s_0) \models \text{AF}(\textit{Forget}(\varphi, \{q\}))$
$\Rightarrow (\mathcal{M}', s_0') \models \text{AF}(\textit{Forget}(\varphi, \{q\}))$ $\qquad$ (IR($\text{AF}(\textit{Forget}(\varphi, \{q\})), \{q\}$))
$(\Leftarrow)$ $\forall (\mathcal{M}, s_0) \models \text{AF}(\textit{Forget}(\varphi, \{q\}))$
$\Rightarrow$ for all path $\pi_{s_0}$ there is $s \in \pi_{s_0}$ such that $(\mathcal{M}, s) \models \textit{Forget}(\varphi, \{q\})$
$\Rightarrow \exists (\mathcal{M}', s_0')$ *s.t.* $(\mathcal{M}', s_0') \leftrightarrow_{\{q\}} (\mathcal{M}, s_0)$ and for all path $\pi_{s_0'}$ there is $s' \in \pi_{s_0'}$ s.t. $(\mathcal{M}', s') \models \varphi$
$\Rightarrow (\mathcal{M}', s_0') \models \text{AF}\varphi$
$\Rightarrow (\mathcal{M}, s_0) \models \text{F}_{\text{CTL}}(\text{AF}\varphi, \{q\})$

**Theorem 6.** *Let* $\text{AF}\varphi \wedge \psi$ *be a* AF *formula with* $\varphi$ *and* $\psi$ *are a proposition formulas and* $q$ *an atom. If* $\text{AF}\varphi \wedge \psi$ *is satisfiable, then we have:*

$$\text{F}_{\text{CTL}}(\text{AF}\varphi \wedge \psi, \{q\}) \equiv \text{AF}(\textit{Forget}(\varphi, \{q\})) \wedge \textit{Forget}(\psi, \{q\}).$$

*Proof.* For convenience, let $\mathcal{M} = (S, R, L)$, $s_0$ and $\mathcal{M}' = (S', R', L')$, $s_0'$ be two initial structure.

($\Rightarrow$) $\forall(\mathcal{M}, s_0) \models \mathrm{F}_{\mathrm{CTL}}(\mathrm{AF}\varphi \wedge \psi, \{q\})$
$\Rightarrow \exists(\mathcal{M}', s_0')$ *s.t.* $(\mathcal{M}', s_0') \leftrightarrow_{\{q\}} (\mathcal{M}, s_0)$ and $(\mathcal{M}', s_0') \models \mathrm{AF}\varphi \wedge \psi$
$\Rightarrow (\mathcal{M}', s_0') \models \mathrm{F}_{\mathrm{CTL}}(\mathrm{AF}\varphi, \{q\}) \wedge \mathrm{F}_{\mathrm{CTL}}(\psi, \{q\})$
$\Rightarrow (\mathcal{M}, s_0) \models \mathrm{F}_{\mathrm{CTL}}(\mathrm{AF}\varphi, \{q\})$ and $(\mathcal{M}, s_0) \models \mathrm{F}_{\mathrm{CTL}}(\psi, \{q\})$ (IR($\mathrm{F}_{\mathrm{CTL}}(\mathrm{AF}\varphi, \{q\}), \{q\}$), IR($\mathrm{F}_{\mathrm{CTL}}(\psi, \{q\}), \{q\}$), Theorem 1)
$\Rightarrow (\mathcal{M}, s_0) \models \mathrm{AF}(\mathit{Forget}(\varphi, \{q\})) \wedge \mathit{Forget}(\psi, \{q\})$ (Proposition 15)

($\Leftarrow$) $\forall(\mathcal{M}, s_0) \models \mathrm{AF}(\mathit{Forget}(\varphi, \{q\})) \wedge \mathit{Forget}(\psi, \{q\})$
$\Rightarrow (\mathcal{M}, s_0) \models \mathit{Forget}(\psi, \{q\})$ and for all path $\pi_{s_0}$, $\exists s \in \pi_{s_0}$ such that $(\mathcal{M}, s) \models \mathit{Forget}(\varphi, \{q\})$
$\Rightarrow \exists(\mathcal{M}', s_0')$ *s.t.* for all path $\pi_{s_0'}$ there is $s' \in \pi_{s_0'}$ s.t. $(\mathcal{M}', s') \models \varphi$, $(\mathcal{M}', s_0') \models \psi$ and $(\mathcal{M}, s_0) \leftrightarrow_{\{q\}} (\mathcal{M}', s_0')$
$\Rightarrow (\mathcal{M}', s_0') \models \mathrm{AF}\varphi$ and $(\mathcal{M}', s_0') \models \psi$
$\Rightarrow (\mathcal{M}', s_0') \models \mathrm{AF}\varphi \wedge \psi$
$\Rightarrow (\mathcal{M}, s_0) \models \mathrm{F}_{\mathrm{CTL}}(\mathrm{AF}\varphi \wedge \psi, \{q\})$

**Proposition 16.** *Let $\Pi = \psi \wedge \mathrm{AF}\varphi$ with $\psi$ and $\varphi$ are DNF formula, $\Sigma$ be two formula in* AF*-fragment and $V \subset \mathcal{A}$. Then have:*

*(i) deciding wether $\Sigma \models \mathrm{F}_{\mathrm{CTL}}(\Pi, V)$ is* co-NP*-complete;*
*(ii) deciding wether $\mathrm{F}_{\mathrm{CTL}}(\Pi, V) \models \Sigma$ is* co-NP*-complete;*
*(iii) deciding wether $\mathrm{F}_{\mathrm{CTL}}(\Pi, V) \equiv \Sigma$ is* co-NP*-complete.*

*Proof.* (i) As $\mathrm{F}_{\mathrm{CTL}}(\Pi, V)$ is computable in polynomial time by Theorem ? and the result of $\mathrm{F}_{\mathrm{CTL}}(\Pi, V)$ is also in AF-fragment. Hence this problem change into deciding the validity of $\Sigma \wedge \neg\mathrm{F}_{\mathrm{CTL}}(\Pi, V)$, which is the complement problem. It has proved that CTL-SAT($\{\mathrm{AF}\}, BF$) is NP-complete [17], and then we obtain this result.

(ii) and (iii) can be proved as (i).

### 4.3 AX-fragment

Emerson and Sistla [**?**] showed by a simple argument that any CTLformula $F$ can be transformed into a normal form where the degree of nesting of path quantifiers is at most 2. That means any CTL formula can be changed to a CTL formula with at most 2 nested temporal operators, this can be computed by the operator Red in [**?**].

In this part we assume the degree of nesting of path quantifiers is **at most one**. Then we have:

$$\mathrm{AX}(\varphi \wedge \psi) \equiv \mathrm{AX}\varphi \wedge \mathrm{AX}\psi.$$

**Proposition 17.** *Any CTL formula with the degree of nesting of path quantifiers is at most one in* AX*-fragment can be transformed into the following form:*

$$\bigvee(\varphi_0 \wedge \mathrm{AX}\varphi_1).$$

**Proposition 18.** *Let $\mathrm{AX}\varphi$ be a* AF *formula with $\varphi$ be a proposition formula and $q$ an atom. Then we have:*

$$\mathrm{F}_{\mathrm{CTL}}(\mathrm{AX}\varphi, \{q\}) \equiv \mathrm{AX}(\mathit{Forget}(\varphi, \{q\})).$$

*Proof.* Let $(\mathcal{M} = (S, R, L), s_0)$ and $(\mathcal{M}' = (S', R', L'), s_0')$ be two initial structure.

$(\Rightarrow)$ $\forall(\mathcal{M}', s_0') \models \mathrm{F}_{\text{CTL}}(\text{AX}\varphi, \{q\})$
$\Rightarrow \exists(\mathcal{M}, s_0)$, *s.t.* , $(\mathcal{M}, s_0) \models \text{AX}\varphi$ and $(\mathcal{M}, s_0) \leftrightarrow_{\{q\}} (\mathcal{M}', s_0')$
$\Rightarrow$ for all $(s_0, s) \in R$ there is $(\mathcal{M}, s) \models \varphi$ and then $(\mathcal{M}, s) \models$ *Forget*$(\varphi, \{q\})$
$\Rightarrow (\mathcal{M}, s_0) \models \text{AX}($*Forget*$(\varphi, \{q\}))$
$\Rightarrow (\mathcal{M}', s_0') \models \text{AX}($*Forget*$(\varphi, \{q\}))$ (IR(AX(*Forget*$(\varphi, \{q\})), \{q\}$))

$(\Leftarrow)$ $\forall(\mathcal{M}, s_0) \models \text{AX}($*Forget*$(\varphi, \{q\}))$
$\Rightarrow$ for all $(s_0, s) \in R$ there is $(\mathcal{M}, s) \models$ *Forget*$(\varphi, \{q\})$
$\Rightarrow \exists(\mathcal{M}', s_0')$ *s.t.* $(\mathcal{M}', s_0') \leftrightarrow_{\{q\}} (\mathcal{M}, s_0)$ and $\forall(s_0', s') \in R$ there is $(\mathcal{M}', s') \models \varphi$
$\Rightarrow (\mathcal{M}', s_0') \models \text{AX}\varphi$
$\Rightarrow (\mathcal{M}, s_0) \models \mathrm{F}_{\text{CTL}}(\text{AX}\varphi, \{q\})$

**Theorem 7.** *Let* $\text{AX}\varphi \wedge \psi$ *be a* AX *formula with* $\varphi$ *and* $\psi$ *are a proposition formulas and* $q$ *an atom. If* $\text{AX}\varphi \wedge \psi$ *is satisfiable, then we have:*

$$\mathrm{F}_{\text{CTL}}(\text{AX}\varphi \wedge \psi, \{q\}) \equiv \text{AX}(\textit{Forget}(\varphi, \{q\})) \wedge \textit{Forget}(\psi, \{q\}).$$

*Proof.* For convenience, let $\mathcal{M} = (S, R, L), s_0$ and $\mathcal{M}' = (S', R', L'), s_0'$ be two initial structure.

$(\Rightarrow)$ $\forall(\mathcal{M}, s_0) \models \mathrm{F}_{\text{CTL}}(\text{AX}\varphi \wedge \psi, \{q\})$
$\Rightarrow \exists(\mathcal{M}', s_0')$ *s.t.* $(\mathcal{M}', s_0') \leftrightarrow_{\{q\}} (\mathcal{M}, s_0)$ and $(\mathcal{M}', s_0') \models \text{AX}\varphi \wedge \psi$
$\Rightarrow (\mathcal{M}', s_0') \models \mathrm{F}_{\text{CTL}}(\text{AX}\varphi, \{q\}) \wedge \mathrm{F}_{\text{CTL}}(\psi, \{q\})$
$\Rightarrow (\mathcal{M}, s_0) \models \mathrm{F}_{\text{CTL}}(\text{AX}\varphi, \{q\})$ and $(\mathcal{M}, s_0) \models \mathrm{F}_{\text{CTL}}(\psi, \{q\})$ (IR($\mathrm{F}_{\text{CTL}}(\text{AX}\varphi, \{q\}), \{q\}$), IR($\mathrm{F}_{\text{CTL}}(\psi, \{q\}), \{q\}$), Theorem 1)
$\Rightarrow (\mathcal{M}, s_0) \models \text{AX}($*Forget*$(\varphi, \{q\})) \wedge$ *Forget*$(\psi, \{q\})$ (Proposition 15)

$(\Leftarrow)$ $\forall(\mathcal{M}, s_0) \models \text{AX}($*Forget*$(\varphi, \{q\})) \wedge$ *Forget*$(\psi, \{q\})$
$\Rightarrow (\mathcal{M}, s_0) \models$ *Forget*$(\psi, \{q\})$ and for all $(s_0, s) \in R$ there is $(\mathcal{M}, s) \models$ *Forget*$(\varphi, \{q\})$
$\Rightarrow \exists(\mathcal{M}', s_0')$ *s.t.* for all $(s_0', s') \in R$ there is $(\mathcal{M}', s') \models \varphi$, $(\mathcal{M}', s_0') \models \psi$ and $(\mathcal{M}, s_0) \leftrightarrow_{\{q\}} (\mathcal{M}', s_0')$
$\Rightarrow (\mathcal{M}', s_0') \models \text{AX}\varphi$ and $(\mathcal{M}', s_0') \models \psi$
$\Rightarrow (\mathcal{M}', s_0') \models \text{AX}\varphi \wedge \psi$
$\Rightarrow (\mathcal{M}, s_0) \models \mathrm{F}_{\text{CTL}}(\text{AX}\varphi \wedge \psi, \{q\})$

**Theorem 8.** $\mathrm{F}_{\text{CTL}}(\varphi_0 \wedge \text{AX}\varphi_1, p) \equiv \textit{Forget}(\varphi_0, p) \wedge \text{AX}(\textit{Forget}(\varphi_1, p))$.

*Proof.* For convenience, let $\varphi = \varphi_0 \wedge \text{AX}\varphi_1$.

$(\Rightarrow)$ $\forall(\mathcal{M}, s_0) \in$ *Mod*$(\varphi)$
$\Rightarrow (\mathcal{M}, s_0) \models \varphi_0, (\mathcal{M}, s_0) \models$

**Proposition 19.** *Let* $\Pi = \psi \wedge \text{AX}\varphi$ *with* $\psi$ *and* $\varphi$ *are DNF formula,* $\Sigma$ *be two formula in* AX*-fragment and* $V \subset \mathcal{A}$*. Then have:*

*(i) deciding wether* $\Sigma \models \mathrm{F}_{\text{CTL}}(\Pi, V)$ *is PSPACE-complete;*
*(ii) deciding wether* $\mathrm{F}_{\text{CTL}}(\Pi, V) \models \Sigma$ *is PSPACE-complete;*
*(iii) deciding wether* $\mathrm{F}_{\text{CTL}}(\Pi, V) \equiv \Sigma$ *is PSPACE-complete.*

*Proof.* (i) As $\mathrm{F}_{\text{CTL}}(\Pi, V)$ is computable in polynomial time by Theorem 8 and the result of $\mathrm{F}_{\text{CTL}}(\Pi, V)$ is also in AX-fragment. Hence this problem change into deciding the validity of $\Sigma \wedge \neg\mathrm{F}_{\text{CTL}}(\Pi, V)$, which is the complement problem. It has proved that CTL-SAT($\{\text{AX}\}, BF$) is PSPACE-complete [17], and then we obtain this result.

(ii) and (iii) can be proved as (i).

### 4.4 $\{\text{AF}, \text{AG}\}$-fragment

Therefore, in this part we assume that the any formula is at most one nested temporal operator. Then we have the follow result: Any formula in $\{\text{AF}, \text{AG}\}$-fragment can be changed into a formula with the following form:

$$\bigvee(\varphi_0 \wedge \text{AF}\varphi_1 \wedge \text{AG}\varphi_2).$$

Where $\varphi_i$ ($i \in \{0, 1, 2\}$) are CPL formulae.

We have:

**Theorem 9.** *Let $\varphi_0 \wedge \text{AF}\varphi_1 \wedge \text{AG}\varphi_2$ be a $\{\text{AF}, \text{AG}\}$ formula with $\varphi_i$ $0 \le i \le 2$ are a proposition formulas and $q$ an atom. If $\varphi_0 \wedge \text{AF}\varphi_1 \wedge \text{AG}\varphi_2$ is satisfiable, then we have:*

$$\text{F}_{\text{CTL}}(\varphi_0 \wedge \text{AF}\varphi_1 \wedge \text{AG}\varphi_2, p) \equiv \textit{Forget}(\varphi_0, p) \wedge \text{AG}(\textit{Forget}(\varphi_2, p)) \wedge \text{AF}(\textit{Forget}(\varphi_1 \wedge \varphi_2, p)).$$

*Proof.* For convenience, let $\varphi = \varphi_0 \wedge \text{AF}\varphi_1 \wedge \text{AG}\varphi_2$.

$(\Rightarrow)$ $\forall(\mathcal{M}, s_0) \in \textit{Mod}(\text{F}_{\text{CTL}}(\varphi, p))$
$\Rightarrow \exists(\mathcal{M}', s_0')$ *s.t.* $(\mathcal{M}', s_0') \leftrightarrow_{\{p\}} (\mathcal{M}, s_0)$ and $(\mathcal{M}', s_0') \models \varphi$
$\Rightarrow (\mathcal{M}', s_0') \models \varphi_0$, $(\mathcal{M}', s_0') \models \text{AF}\varphi_1$ and $(\mathcal{M}', s_0') \models \text{AG}\varphi_2$
$\Rightarrow (\mathcal{M}', s_0') \models \varphi_0$, for all $s \in S'$ there is $(\mathcal{M}', s) \models \varphi_2$, and for all path $\pi_{s_0'}$ there is a $s' \in \pi_{s_0'}$ such that $(\mathcal{M}', s') \models \varphi_1 \wedge \varphi_2$
$\Rightarrow (\mathcal{M}', s_0') \models \textit{Forget}(\varphi_0, p)$, $(\mathcal{M}', s_0') \models \text{AF}(\textit{Forget}(\varphi_1 \wedge \varphi_2, p))$ and $(\mathcal{M}', s_0') \models \text{AG}(\textit{Forget}(\varphi_2, p))$
$\Rightarrow (\mathcal{M}, s_0) \models \textit{Forget}(\varphi_0, p) \wedge \text{AG}(\textit{Forget}(\varphi_2, p)) \wedge \text{AF}(\textit{Forget}(\varphi_1 \wedge \varphi_2, p))$

$(\Leftarrow)$ $\forall(\mathcal{M}, s_0) \in \textit{Mod}(\textit{Forget}(\varphi_0, p) \wedge \text{AG}(\textit{Forget}(\varphi_2, p)) \wedge \text{AF}(\textit{Forget}(\varphi_1 \wedge \varphi_2, p)))$
$\Rightarrow \exists(\mathcal{M}', s_0')$ such that $(\mathcal{M}', s_0') \leftrightarrow_{\{p\}} (\mathcal{M}, s_0)$ and $(\mathcal{M}', s_0') \models \varphi_0$, for all $s \in S'$ there is $(\mathcal{M}', s) \models \varphi_2$, and for all path $\pi_{s_0'}$ there is a $s' \in \pi_{s_0'}$ such that $(\mathcal{M}', s') \models \varphi_1 \wedge \varphi_2$
$\Rightarrow (\mathcal{M}', s_0) \models \varphi_0 \wedge \text{AF}\varphi_1 \wedge \text{AG}\varphi_2$
$\Rightarrow (\mathcal{M}, s_0) \models \text{F}_{\text{CTL}}(\varphi_0 \wedge \text{AF}\varphi_1 \wedge \text{AG}\varphi_2, p)$

**Proposition 20.** *Let $\Pi = \varphi_0 \wedge \text{AF}\varphi_1 \wedge \text{AG}\varphi_2$ with $\varphi_0$ is a DNF and at most one of $\varphi_i$ ($i \in \{1, 2\}$) is DNF and another one is a clause or both are clauses, $\Sigma$ be two formula in $\{\text{AF}, \text{AG}\}$-fragment and $V \subset \mathcal{A}$. Then have:*

*(i) deciding wether $\Sigma \models \text{F}_{\text{CTL}}(\Pi, V)$ is PSPACE-complete;*
*(ii) deciding wether $\text{F}_{\text{CTL}}(\Pi, V) \models \Sigma$ is PSPACE-complete;*
*(iii) deciding wether $\text{F}_{\text{CTL}}(\Pi, V) \equiv \Sigma$ is PSPACE-complete.*

*Proof.* (i) As $\text{F}_{\text{CTL}}(\Pi, V)$ is computable in polynomial time by Theorem 9 and the result of $\text{F}_{\text{CTL}}(\Pi, V)$ is also in AG-fragment. Hence this problem change into deciding the validity of $\Sigma \wedge \neg\text{F}_{\text{CTL}}(\Pi, V)$, which is the complement problem. It has proved that CTL-SAT($\{\text{AF}, \text{AG}\}, BF$) is PSPACE-complete [17], and then we obtain this result.

(ii) and (iii) can be proved as (i).

# 5 applications

## 5.1 Sufficient and Necessary Conditions

In this section, we will give the definition of SNC (WSC) and explore the relation between forgetting and SNC (WSC). The SNC (WSC) of a proposition will be given at first:

**Definition 5 (sufficient and necessary condition).** *Let $\phi$ be a formulas or an initial structure, $\psi$ be formulas, $V \subseteq Var(\phi)$, $q \in Var(\phi) - V$ and $Var(\psi) \subseteq V$.*

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

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

**Proposition 21.** *(**dual**) Let $V, q, \varphi$ and $\psi$ are the ones in Definition 6.*

*(i)* *$\psi$ is a SNC of $q$ on $V$ under $\varphi$ iff $\neg\psi$ is a WSC of $q$ on $V$ under $\varphi$.*
*(ii)* *$\psi$ is a WSC of $q$ on $V$ under $\varphi$ iff $\neg\psi$ is a SNC of $q$ on $V$ under $\varphi$.*

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.

As the propositional formula, the SCN (WSC) of any formula can be defined as follows:

**Definition 6.** *Let $\Gamma$ be a formula or an initial structure, $\alpha$ be a formula and $P \subseteq (Var(\Gamma) \cup Var(\alpha))$. A formula $\varphi$ of $P$ is said to be a NC (SC) of $\alpha$ on $P$ under $\Gamma$ 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) $\varphi'$, 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 22.** *Let $\Gamma$ be a formula, $P$, and $\alpha$ be as in Definition 7. A formula $\varphi$ of $P$ is the SNC (WSC) of $\alpha$ on $P$ under $\Gamma$ 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 $\Gamma$ and $\alpha$.*

We propose the theorem of computing the SNC (WSC) of an atom due to the SNC (WSC) of a formula can be change to the SNC (WSC) of an atom by Proposition 22.

**Theorem 10.** *Let $\varphi$ be a formula, $V \subseteq Var(\varphi)$ and $q \in Var(\varphi) - V$.*

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

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

Then we have the following consequence:

**Theorem 11.** *Let $\mathcal{K} = (\mathcal{M}, s)$ be a initial Kripke structure with $\mathcal{M} = (S, R, L)$ 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* $\mathrm{F}_{\text{CTL}}(\mathcal{F}_{V\cup\{q\}}(\mathcal{K}_{|V\cup\{q\}}) \wedge q, q)$.
*(ii) the WSC of $q$ on $V$ under $\mathcal{K}$ is* $\neg\mathrm{F}_{\text{CTL}}(\mathcal{F}_{V\cup\{q\}}(\mathcal{K}_{|V\cup\{q\}}) \wedge \neg q, q)$.

*Proof.* (i) As we know that any initial Kripke 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 $\mathrm{F}_{\text{CTL}}(\mathcal{F}_{\mathcal{A}}(\mathcal{K}) \wedge q, \mathcal{A} - V)$. We will prove that $\mathrm{F}_{\text{CTL}}(\mathcal{F}_{V\cup\{q\}}(\mathcal{K}_{|V\cup\{q\}}) \wedge q, q) \equiv \mathrm{F}_{\text{CTL}}(\mathcal{F}_{\mathcal{A}}(\mathcal{K}) \wedge q, \mathcal{A} - V)$.

($\Rightarrow$) $\forall \mathcal{K}_1 \in \mathit{Mod}(\mathrm{F}_{\text{CTL}}(\mathcal{F}_{V\cup\{q\}}(\mathcal{K}_{|V\cup\{q\}}) \wedge q, q))$
$\Rightarrow$ there is an initial structure $\mathcal{K}'$ such that $\mathcal{K}' \models \mathcal{F}_{V\cup\{q\}}(\mathcal{K}_{|V\cup\{q\}}) \wedge q$ and $\mathcal{K}_1 \leftrightarrow_{\{q\}} \mathcal{K}'$
$\Rightarrow \mathcal{K}' \leftrightarrow_{\mathcal{A}-(V\cup\{q\})} \mathcal{K}_{|V\cup\{q\}}$ (Theorem 2)
$\Rightarrow \mathcal{K}_1 \leftrightarrow_{\mathcal{A}-V} \mathcal{K}_{|V\cup\{q\}}$ (Proposition 3)
$\Rightarrow \mathcal{K}_{|V\cup\{q\}} \leftrightarrow_{\mathcal{A}-(V\cup\{q\})} \mathcal{K}$ (Proposition 5)
$\Rightarrow \mathcal{K}' \leftrightarrow_{\mathcal{A}-(V\cup\{q\})} \mathcal{K}$ (Proposition 3)
$\Rightarrow \mathcal{K} \models \mathcal{F}_{\mathcal{A}}(\mathcal{K}) \wedge q$
$\Rightarrow \mathcal{K}_1 \leftrightarrow_{\mathcal{A}-V} \mathcal{K}$
$\Rightarrow \mathcal{K}_1 \models \mathrm{F}_{\text{CTL}}(\mathcal{F}_{\mathcal{A}}(\mathcal{K}) \wedge q, \mathcal{A} - V)$

($\Leftarrow$) $\forall \mathcal{K}_1 \in \mathit{Mod}(\mathrm{F}_{\text{CTL}}(\mathcal{F}_{\mathcal{A}}(\mathcal{K}) \wedge q, \mathcal{A} - V))$
$\Rightarrow$ there an initial structure $\mathcal{K}_2$ *s.t.* $\mathcal{K}_2 \models \mathcal{F}_{\mathcal{A}}(\mathcal{K}) \wedge q$ and $\mathcal{K}_1 \leftrightarrow_{\mathcal{A}-V} \mathcal{K}_2$
$\Rightarrow \mathcal{K}_2 \leftrightarrow_{\emptyset} \mathcal{K}$ (Theorem 2)
$\Rightarrow \mathcal{K}_2 \leftrightarrow_{\mathcal{A}-(V\cup\{q\})} \mathcal{K}_{|V\cup\{q\}}$ due to $\mathcal{K}_{|V\cup\{q\}} \leftrightarrow_{\mathcal{A}-(V\cup\{q\})} \mathcal{K}$ (Proposition 5)
$\Rightarrow \mathcal{K}_2 \models \mathcal{F}_{V\cup\{q\}}(\mathcal{K}_{|V\cup\{q\}}) \wedge q$
$\Rightarrow \mathcal{K}_1 \models \mathrm{F}_{\text{CTL}}(\mathcal{F}_{V\cup\{q\}}(\mathcal{K}_{|V\cup\{q\}}) \wedge q, q)$.

(ii) This is proved by the dual property.

### 5.2 Strongest and weakest definitions

Definability is acknowledged as an important logical concept when reasoning about knowledge represented in propositional logic. Informally speaking, an atom $p$ can be "defined" in a given formula $\Sigma$ in terms of a set $V$ of atoms whenever the knowledge of the truth values of $V$ enables concluding about the truth value of $p$, under the condition of $\Sigma$ [].

## 6 Complexity

In this part we will give an algorithm to compute the V-quotient space, and then check the $V$-bisimulation between Kripke structures by this.

**Definition 7.** *Let $\mathcal{M} = (S, R, L)$ be a finite transition system with $S \neq \emptyset$ and $V \subseteq \mathcal{A}$. A* partition *for $S$ is a set $\Pi = \{B_1, ..., B_n\}$ such that $B_i \neq \emptyset$ (for $0 < i \leq n$), $B_i \cap B_j = \emptyset$ (for $0 < i, j \leq n$ and $i \neq j$), and $S = \bigcup_{0<i\leq n} B_i$. A $V$-*partition *for $S$ is a partition $\Pi$ for $S$ such that $\forall B \in \Pi$ there is $\forall s_1, s_2 \in B$, $L(s_1) \cap V = L(s_2) \cap V$.*

$B_i \in \Pi$ is called a *block*. $C \subseteq S$ is a *superblock* of $\Pi$ is $C = B_{i_1} \cup ... \cup B_{i_m}$ for some $B_{i_1} \cup ... \cup B_{i_m} \in \Pi$. Let $[s]_\Pi$ denote the unique block of partition $\Pi$ containing $s$.

Let $\Pi_1$ and $\Pi_2$ be two partitions for $S$, $\Pi_1$ is called *finer* than $\Pi_2$, or $\Pi_2$ is called *coarser* than $\Pi_1$, if:

$$\forall B_1 \in \Pi_1 \; \exists B_2 \in \Pi_2 \; s.t. \; B_1 \subseteq B_2.$$

In this case, every block (the element of $\Pi_2$) of $\Pi_2$ can be written as a disjoint union of blocks in $\Pi_1$. $\Pi_1$ is *strictly finer* than $\Pi_2$ (and $\Pi_2$ is *strictly coarser* than $\Pi_1$) if $\Pi_1$ is finer than $\Pi_2$ and $\Pi_1 = \Pi_2$.

There is a close connection between equivalence relations and partitions. For equivalence relation $R$ on $S$, the set $S/R$ is a partition for $S$. Vice versa, partition $\Pi$ for $S$ induces the equivalence relation:

$$R_\Pi = \{(s_1, s_2) | \exists B \in \Pi \; s.t. \; s_1 \in B \;\&\; s_2 \in B\} = \{(s_1, s_2) | [s_1]_\Pi = [s_2]_\Pi\}.$$

Given a transition system $\mathcal{M} = (S, R, L)$ on $\mathcal{A}$ and a set $V \subseteq \mathcal{A}$. In the initial $V$-partition $\Pi_0 = \Pi_V$, each group of equally labeled except those atoms in $\overline{V}$ states forms a block. $\Pi_V$ can be computed as follows. The basic idea is to generate a *decision tree* for $q \in V$. A decision tree $\text{Tr}_V$ for $V = \{q_1, ..., q_n\}$ is a full binary tree such that the height of the decision tree for $V$ is $n$ and the vertices at depth $i < n$ represent the decision "is $a_{i+1} \in L(s)$". The left branch of vertex $v$ at depth $i < n$ represents the case "$a_{i+1} \notin L(s)$", while the right branch represents "$a_{i+1} \in L(s)$". Leaf $v$ represents a set of states such that $\forall s_1, s_2 \in v$ there is $L(s_1) \cap V = L(s_2) \cap V$.

*Example 3.* Let $\mathcal{M} = (S, R, L)$ be a transition system on $\mathcal{A} = \{a, b, c\}$ (Fig. 2 and $V = \{a, b\} \subseteq \mathcal{A}$. Then the decision $\text{Tr}_V$ on $V$ is as Fig. 2(hau tu).

The decision tree for $V$ is constructed successively by considering all states in $S$ separately. The initial decision tree consists only of the root $v_0$. On considering state $s$, the tree is traversed in a top-down manner, and new vertices are inserted when necessary, i.e., when $s$ is the first encountered state with labeling $L(s) \cap V$. Once the tree traversal for state $s$ reaches leaf $w$, states($w$) is extended with $s$. The essential steps are outlined in Algorithm 1.

**Algorithm 1:** An initial $V$-partition algorithm

**Input**: finite transition system $\mathcal{M} = (S, R, L)$ on $\mathcal{A} = \{q_1, ..., q_n\}$ and $V = \{p_1, ..., p_m\} \subseteq \mathcal{A}$

**Output**: Initial $V$-partition $\Pi_V$

```
new($v_0$);                                            //Create a new node
//part the states in $S$ to different blocks.
for $s \in S$ do
    $v \to v_0$;                                       //Let $v$ point $v_0$
    for i=1, ..., m do
        //If the right node is NULL, i.e. nil, then create a new node
        if $p_i \in L(s)$ then
            if right(v) = nil then
                new(right($v$));                       //Create a new right node.
            end
            $v \to$ right($v$);
        end
        else
            //If the left node is NULL, i.e. nil, then create a new node
            if left(v) = nil then
                new(left($v$));
            end
            $v \to$ left($v$);
        end
    end
    //Computing the elements of leaf nodes.
    if $p_m \in L(s)$ then
        if rigth(v) = nil then
            new(right($v$));
            states(right($v$)) := states(right($v$))$\cup\{s\}$;
        end
    end
    else
        if right(v) = nil then
            new(left($v$));
            states(left($v$)) := states(left($v$)) $\cup\{s\}$;
        end
    end
end
return $\{$states(w) $|w$ is a leaf$\}$
```

For each state $s$ in S, the decision tree has to be traversed from root to leaf. This takes $O(|V|)$ time. The overall time complexity for computing initial $V$-partition $\Pi_V$ is $O(|S| * |V|)$.

**Lemma 7.** *Let $\mathcal{M} = (S, R, L)$ be a transition system on $\mathcal{A}$, $V \subseteq \mathcal{A}$ and Let $\Pi$ be a $V$-partition of $S$ and $R_\Pi$ the equivalence relation on $S$ induced by $\Pi$. Then $R_\Pi$ is a $\overline{V}$-bisimulation (i.e. $\mathcal{B}$) if and only if*

*(**Finer**) $\Pi$ is finer than $\Pi_V$;*
*(**Par**) for all $B, C \in \Pi$ there is $B \cap Pre(C) = \emptyset$ or $B \subseteq Pre(C)$.*

*Where $\mathcal{B} = \{(s, s')|(\mathcal{M}, s) \leftrightarrow_{\overline{V}} (\mathcal{M}, s')\}$.*

*Proof.* ($\Leftarrow$) Assume that $\Pi$ satisfies (**Finer**) and (**Par**). We prove that $R_\Pi$ induced by $\Pi$ is a $\overline{V}$-bisimulation. Let $(s_1, s_2) \in R_\Pi$ and $B = [s_1]_\Pi = [s_2]_\Pi$.

(a) Since $\Pi$ is finer than $\Pi_V$ by (**Finer**), there exists a block $B'$ of $\Pi_V$ containing $B$. Thus, $s_1, s_2 \in B \subseteq B' \in \Pi_V$, and therefore, $L(s_1) \cap V = L(s_2) \cap V$.
(b) Let $s_1' \in$ Post$(s_1)$ and $C = [s_1']_\Pi$. Then, $s_1 \in B \cap$ Pre$(C)$. By condition (**Par**), we obtain $B \subseteq$ Pre$(C)$. Hence, $s_2 \in$ Pre$(C)$. So, there exists a state $s_2' \in$ Post$(s_2) \cap C$. Since $s_2' \in C = [s_1']_\Pi$, it follows that $(s_1', s_2') \in R_\Pi$.

($\Rightarrow$) Assume $R_\Pi$ is a $\overline{V}$-bisimulation. We will show that (**Finer**) and (**Par**) are satisfied.

(i) (**Finer**) Assume that $\Pi$ is not finer than $\Pi_V$. Then, there exist a block $B \in \Pi$ and states $s_1, s_2 \in B$ with $[s_1]_{\Pi_V} \neq [s_2]_{\Pi_V}$. Then, $L(s_1) \cap V \neq L(s_2) \cap V$. Hence, $R_\Pi$ is not a $\overline{V}$-bisimulation. Contradiction.
(ii) (**Par**) Let $B, C$ be blocks of $\Pi$. We assume that $B \cap$ Pre$(C) \neq \emptyset$ and show that $B \subseteq$ Pre$(C)$. $B \cap$ Pre$(C) \neq \emptyset$ means there exists a state $s_1 \in B$ with Post$(s_1) \cap C \neq \emptyset$. Let $s_1' \in$ Post$(s_1) \cap C$ and $s_2$ be an arbitrary state of $B$. We demonstrate that $s_2 \in$ Pre$(C)$. Since $s_1, s_2 \in B$, we get that $(s_1, s_2) \in R_\Pi$. Due to $(s_1, s_1') \in R$, there exists a transition $(s_2, s_2') \in R$ *s.t.* $(s_1', s_2') \in R_\Pi$ by $R_\Pi$ is a $\overline{V}$-bisimulation. But then $s_1' \in C$ yields $s_2' \in C$. Hence, $s_2' \in$ Post$(s_2) \cap C$. Thus, $s_2 \in$ Pre$(C)$.

**Theorem 12.** *Let $\mathcal{M}$, $\Pi$ and $\mathcal{B}$ be defined as Lemma 7. Then*

*(i) if $\Pi$ fulfills (ii), then $B \cap Pre(C) = \emptyset$ or $B \subseteq Pre(C)$ for all $B \in \Pi$ and all superlociks $C$ of $\Pi$;*
*(ii) $S/\mathcal{B}$ is the partition satisfying (**Finer**) and (**Par**).*

*Proof.* (i) Assume $\Pi$ satisfies (**Par**). We will show this by contraposition. Let $B \in \Pi$ and $C$ a superblock, *i.e.* , $C$ is of the form $C = C_1 \cup ... \cup C_n$ for blocks $C_1, ..., C_n$ of $\Pi$. Assume that $B \cap$ Pre$(C) \neq \emptyset$ and $B \not\subseteq$ Pre$(C)$. Then, there exists some $C_i$ ($i \in \{1, ..., n\}$) *s.t.* $B \cap$ Pre$(C_i) \neq \emptyset$. It is clear that $B \not\subseteq$ Pre$(C_i)$, since otherwise $B \subseteq$ Pre$(C_i) \subseteq$ Pre$(C)$. Thus, condition (**Par**) is not satisfied for block $C_i \in \Pi$. Contradiction.

(ii) This immediately follows from the fact that $\mathcal{B}$ is a $\overline{V}$-bisimulation.

Therefore, the process of computing V-quotient space is to refine $\Pi_V$ to satisfy condition (**Par**). To that end, a superblock $C$ of current partition $\Pi$ is considered and every block $B$ of $\Pi$ is decomposed ("splitted") into $B \cap \text{Pre}(C)$ and $B \setminus \text{Pre}(C)$. Formally, the refinement operator is defined:

$$\textsc{Refine}(\Pi, C) = \bigcup_{B \in \Pi} \textsc{Refine}(B, C)$$

where $\textsc{Refine}(B, C) = \{B \cap \text{Pre}(C), B \setminus \text{Pre}(C)\} \setminus \{\emptyset\}$. $B$ is not decomposed with respect to $C$, if all states in $B$ have a direct $C$-successor or if no state in $B$ has a direct $C$-successor.

The Lemma 7.36 in [1] means that successive refinements, starting with partition $\Pi_V$, yield a series of partitions $\Pi_0 = \Pi_V, \Pi_1, \Pi_2, \Pi_3, ...$, which become increasingly finer and which all are coarser than $S/\mathcal{B}$. And the termination (to $S/\mathcal{B}$) of this process follows from Lemma 7.38 in [1].

An improvement refinement operator, which decomposes each block into subblocks, is defined:

$$\textsc{Refine}(\Pi, C, C' \backslash C) = \textsc{Refine}(\textsc{Refine}(\Pi, C), C' \backslash C) = \bigcup_{B \in \Pi} \textsc{Refine}(B, C, C' \backslash C).$$

Where, $C \in \Pi$ and $C'$ is a superblock of $\Pi$ such that $C \subseteq C'$ and $|C| \leq \frac{C'}{2}$. $\textsc{Refine}(B, C, C' \setminus C) = \{B_1, B_2, B_3\} \setminus \emptyset$, where:

(i) $B_1 = B \cap \text{Pre}(C) \cap \text{Pre}(C' \setminus C)$, the set of states that have direct successors in both $C$ and $C'$,
(ii) $B_2 = (B \cap \text{Pre}(C)) \setminus \text{Pre}(C' \setminus C)$, the set of states that only have direct successors in $C$,
(iii) $B_3 = (B \cap \text{Pre}(C' \setminus C)) \setminus \text{Pre}(C)$, the set of states that only have direct successors in $C'$.

Note that the blocks $B_1, B_2, B_3$ can not be splitted by $C$ and $C' \setminus C$. Then The Algorithm 2, which is similar with Algorithm 31 in [1], is used to compute $\overline{V}$-quotient space $S/\mathcal{B}$.

**Algorithm 2:** An V-partition refinement algorithm

**Input**: finite transition system $\mathcal{M} = (S, R, L)$ on $\mathcal{A}$, $s \in S$ and $V \subseteq \mathcal{A}$
**Output**: $\overline{V}$-quotient space $S/\mathcal{B}$

```
Π' = {S};
Π = Π_V ;
repeat
    choose block C' ∈ Π' − Π and block C ∈ Π with C ⊆ C' and |C| ≤ |C'|/2;
    Π' = Π;
    Π = REFINE(Π, C, C' \ C) ;
until Π = Π';
```

Due to the fact that $S$ is finite, a partition $\Pi$ with $\Pi = \Pi'$ is reached after at most $|S|$ iteration. After $|S|$ proper refinements, any block in $\Pi$ is a singleton, and a further refinement is thus impossible. The time complexity of Algorithm 2 is $O(|S| * |V| + |R| * \log|S|)$.

Let $\mathcal{M}_i = (S_i, R_i, L_i)$ with $i = 1, 2$ be two transition systems. Then

$$\mathcal{M} = \mathcal{M}_1 \oplus \mathcal{M}_2 = (S_1 \uplus S_2, R_1 \cup R_2, L)$$

where $\uplus$ stands for disjoint union and where $L(s) = L_i(s)$ if $s \in S_i$. Then $(\mathcal{M}_1, s_1) \leftrightarrow_V (\mathcal{M}_2, s_2)$ with $s_1 \in S_1$ and $s_2 \in S_2$ for some $V \subseteq \mathcal{A}$ if and only if $(\mathcal{M}, s_1) \leftrightarrow_V (\mathcal{M}, s_2)$, *i.e.* there is $C \in S_1 \uplus S_2/\mathcal{B}$ such that $s_1, s_2 \in C$, where $\mathcal{B} = \{(s_1, s_2) | (\mathcal{M}, s_1) \leftrightarrow_V (\mathcal{M}, s_2)\}$. Then we have:

**Corollary 2.** *Let $\mathcal{M}_i = (S_i, R_i, L_i)$ with $i = 1, 2$ be two transition systems on $\mathcal{A}$ and $V \subseteq \mathcal{A}$. Then checking wether $(\mathcal{M}_1, s_1) \leftrightarrow_{\overline{V}} (\mathcal{M}_2, s_2)$ for $s_1 \in S_1$ and $s_2 \in S_2$ can be performed in time*

$$O((|S_1| + |S_2|) * |V| + (|R_1| + |R_2|) * log(|S_1| + |S_2|)).$$

## 7 Examples of minimal set

The following example shows that the atoms appearing in the counterexample do not help to computing the minimal set.

*Example 4.* Let $\varphi = \mathrm{AX}((tl1.state = green) \wedge (tl2.state = green)) \wedge (tl1.state = red) \wedge (tl1.state = green)$, $\mathcal{M}$ is a system of traffic light in the figure. There is $\mathcal{M} \not\models \varphi$ and the NuSMV products a counterexample as follows:

- Trace Description: CTL Counterexample:
  - $\rightarrow$ State: 1.1 $\leftarrow$
  - tl1.state = red
  - tl2.state = red

We have that the minimal set $V$ is:

$$V = \{tl1.state = green, tl1.state = green\}$$

and

$$V \cap \{tl1.state = red, tl2.state = red\} = \emptyset$$

.

## 8 Related work

As we know that transition systems can model a piece of software or hardware at various abstraction levels. The lower the abstraction level, the more implementation details are present. At high abstraction levels, such details are deliberately left unspecified. Bisimulation equivalence aims to identify transition systems with the same branching structure but possibly at different abstraction levels, and which thus can simulate each other in a stepwise manner [1].

**Definition 8.** ***(Bisimulation** [1]) Let $\mathcal{M}_i = (S_i, Act_i, R_i, \mathcal{A}, L_i)$, $i = 1, 2$, be transition systems on $\mathcal{A}$. A bisimulation for $\mathcal{M}_1$ and $\mathcal{M}_2$ is a binary relation $\mathcal{R} \subseteq S_1 \times S_2$ such that*

*(a)* $\forall s_1 \in I_1\ \exists s_2 \in I_2$ *s.t.* $(s_1, s_2) \in \mathcal{R}$ *and* $\forall s_2 \in I_2\ \exists s_1 \in I_1$ *s.t.* $(s_1, s_2) \in \mathcal{R}$
*(b) for all* $(s_1, s_2) \in \mathcal{R}$ *it holds:*
- $L_1(s_1) = L_2(s_2)$,
- *if* $(s_1, s_1') \in R_1$ *then there exists* $(s_2, s_2') \in R_2$ *with* $(s_1', s_2') \in \mathcal{R}$, *and*
- *if* $(s_2, s_2') \in R_2$ *then there exists* $(s_1, s_1') \in R_1$ *with* $(s_1', s_2') \in \mathcal{R}$.

$\mathcal{M}_1$ and $\mathcal{M}_2$ are bisimulation-equivalent if there exists a bisimulation $\mathcal{R}$ for $\mathcal{M}_1$ and $\mathcal{M}_2$.

Besides, the similar equivalence relation based on states is defined in [5].

**Definition 9.** ***(equivalence** [5]) Let $\mathcal{M}_i = (S_i, R_i, L_i)$, $i = 1, 2$, be transition systems on $\mathcal{A}$, a sequence of equivalence relations $E_0, E_1, \ldots$ on $S_1 \times S_2$ as follows:*

*(a)* $(s_1, s_2) \in E_0$ *if and only if* $L_1(s_1) = L_2(s_2)$;
*(b)* $(s_1, s_2) \in E_{n+1}$ *if and only if*
- $L_1(s_1) = L_2(s_2)$,
- $\forall (s_1, s_1') \in R_1$ *there is* $(s_2, s_2') \in R_2$ *such that* $(s_1', s_2') \in E_n$, *and*
- $\forall (s_2, s_2') \in R_2$ *there is* $(s_1, s_1') \in R_1$ *such that* $(s_1', s_2') \in E_n$.

Then $s$ and $s'$ is equivalence if and only if $(s, s') \in E_i$ for all $i \geq 0$.

Both of the two equivalences are different with our $V$-bisimulation, which is a set-based bisimulation and is defined on Kripke structures. Binary relations between states (henceforth implementation relations) are useful to relate or to compare transition systems, possibly at different abstraction levels. But it cannot compare transition systems on different set of atoms. By $V$-bisimulation we can relate Kripke structures on different set of atoms and obtain an abstract Kripke structure on a small set of atoms from a Kripke structure on bigger set of atoms as talked in Proposition 5.

## 9 Concluding Remarks

We have given the definition of forgetting in CTL from the sematic forgetting point of view. To investigate whether a CTL system is closed under forgetting, that is a result of forgetting some set $V \subseteq \mathcal{A}$ from a CTL formula $\varphi$ is also a CTL formula $\psi$. We resorted characterizing formula introduced by Browne, and we have extend this characterizing formula of an initial Kripke structure on $\mathcal{A}$ into that on $V \subseteq \mathcal{A}$ from the V-bisimulation point of view proposed in this paper. An arbitrary CTL formula is equivalent to the disjunction of the set of characterizing formula of the models of this formula. This show that the CTL system is closed under forgetting thanks to the fact that the characterizing formula of an initial Kripke structure on $V$ is in CTL. Another interest result is that the forgetting of the fragment $PT\varphi$ ($P \in \{\mathrm{E}, \mathrm{A}\}$, $T \in \{\mathrm{F}, \mathrm{X}\}$) on $V \subseteq \mathcal{A}$ can be computed by $PT\mathrm{F}_{\mathrm{CTL}}(\varphi, V)$. Besides, we have prove that the SNC (WSC) can be computed by using the technology of forgetting.

Extending the method to handle model checking going expressively further than propositional CTL is a direction of ongoing research. And an Algorithm for computing forgetting will be explored.

## 10 Appendix Proofs

In this section, we give proofs for main results in the paper in the order they appear in the paper.

**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 $i \geq 0$
$\Rightarrow (\mathcal{K}_1, \mathcal{K}_2) \in \mathcal{B}$.

**Proposition** 3.

*Proof.* (i) It is clear from Proposition 1.

(ii) Let $\mathcal{M}_i = (S_i, R_i, L_i)$ $(i = 1, 2, 3)$, $s_1 \leftrightarrow_{V_1} s_2$ 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 $(w_1, w_3) \in \mathcal{B}'$:

(1) 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 $r \in L_3(w_3)$.
(2) 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 $\mathcal{B}'$.
(3) 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 $\mathcal{B}'$.

**Theorem** 1.

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

**Lemma** 3.

*Proof.* (i) It is apparent from the definition of $\mathcal{F}_V(\mathrm{Tr}_n(s))$.

(ii) **Base**. If $n = 0$, then $(\mathcal{M}, s) \models \mathcal{F}_V(\mathrm{Tr}_0(s'))$ implies $L(s) - \overline{V} = L'(s') - \overline{V}$. Hence, $\mathrm{Tr}_0(s) \leftrightarrow_{\overline{V}} \mathrm{Tr}_0(s')$.
**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 $\mathrm{Tr}_{n-1}(s_1) \leftrightarrow_{\overline{V}} \mathrm{Tr}_{n-1}(s_1')$. Since $(\mathcal{M}, s) \models \mathcal{F}_V(\mathrm{Tr}_n(s'))$, then $(\mathcal{M}, s) \models \mathrm{AX}\left(\bigvee_{(s',s_1')\in R} \mathcal{F}_V(\mathrm{Tr}_{n-1}(s_1'))\right)$. Therefore, for each $(s, s_1) \in R$ there is a $(s', s_1') \in R'$ such that $(\mathcal{M}, s_1) \models \mathcal{F}_V(\mathrm{Tr}_{n-1}(s_1'))$. Hence, $\mathrm{Tr}_{n-1}(s_1) \leftrightarrow_{\overline{V}} \mathrm{Tr}_{n-1}(s_1')$ 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 $\mathrm{Tr}_{n-1}(s_1') \leftrightarrow_{\overline{V}} \mathrm{Tr}_{n-1}(s_1)$. Since $(\mathcal{M}, s) \models \mathcal{F}_V(\mathrm{Tr}_n(s'))$, then $(\mathcal{M}, s) \models \bigwedge_{(s',s_1')\in R'} \mathrm{EX}\mathcal{F}_V(\mathrm{Tr}_{n-1}(s_1'))$. Therefore, for each $(s', s_1') \in R'$ there is a $(s, s_1) \in R$ such that $(\mathcal{M}, s_1) \models \mathcal{F}_V(\mathrm{Tr}_{n-1}(s_1')$. Hence, $\mathrm{Tr}_{n-1}(s_1) \leftrightarrow_{\overline{V}} \mathrm{Tr}_{n-1}(s_1')$ by inductive hypothesis.

**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 $\mathrm{IR}(\mathcal{F}_V(\mathcal{M}, s_0), \overline{V})$. We will show that $(\mathcal{M}, s_0) \models \mathcal{F}_V(\mathcal{M}, s_0)$ at first.

It is apparent that $(\mathcal{M}, s_0) \models \mathcal{F}_V(\mathrm{Tr}_c(s_0))$ by Lemma 3. We must show that $(\mathcal{M}, s_0) \models \bigwedge_{s\in S} G(\mathcal{M}, s)$. Let $\mathcal{X} = \mathcal{F}_V(\mathrm{Tr}_c(s)) \rightarrow \left(\bigwedge_{(s,s_1)\in R} \mathrm{EX}\mathcal{F}_V(\mathrm{Tr}_c(s_1))\right)$ $\wedge \mathrm{AX}\left(\bigvee_{(s,s_1)\in R} \mathcal{F}_V(\mathrm{Tr}_c(s_1))\right)$, we will show $\forall s \in S$, $(\mathcal{M}, s_0) \models G(\mathcal{M}, s)$. There are two cases we should consider:

- If $(\mathcal{M}, s_0) \not\models \mathcal{F}_V(\mathrm{Tr}_c(s))$, it is apparent that $(\mathcal{M}, s_0) \models \mathcal{X}$;
- If $(\mathcal{M}, s_0) \models \mathcal{F}_V(\mathrm{Tr}_c(s))$:
  $(\mathcal{M}, s_0) \models \mathcal{F}_V(\mathrm{Tr}_c(s))$
  $\Rightarrow s_0 \leftrightarrow_{\overline{V}} s$ by the definition of characteristic number and Lemma 4
  for each $(s, s_1) \in R$ there is $(\mathcal{M}, s_1) \models \mathcal{F}_V(\mathrm{Tr}_c(s_1))$ $\quad (s_1 \leftrightarrow_{\overline{V}} s_1)$
  $\Rightarrow (\mathcal{M}, s) \models \bigwedge_{(s,s_1)\in R} \mathrm{EX}\mathcal{F}_V(\mathrm{Tr}_c(s_1))$
  $\Rightarrow (\mathcal{M}, s_0) \models \bigwedge_{(s,s_1)\in R} \mathrm{EX}\mathcal{F}_V(\mathrm{Tr}_c(s_1))$ $\quad (\mathrm{IR}(\bigwedge_{(s,s_1)\in R} \mathrm{EX}\mathcal{F}_V(\mathrm{Tr}_c(s_1)), \overline{V})$, $s_0 \leftrightarrow_{\overline{V}} s)$
  for each $(s, s_1)$ there is $\mathcal{M}, s_1 \models \bigvee_{(s,s_2)\in R} \mathcal{F}_V(\mathrm{Tr}_c(s_2))$
  $\Rightarrow (\mathcal{M}, s) \models \mathrm{AX}\left(\bigvee_{(s,s_2)\in R} \mathcal{F}_V(\mathrm{Tr}_c(s_2))\right)$
  $\Rightarrow (\mathcal{M}, s_0) \models \mathrm{AX}\left(\bigvee_{(s,s_2)\in R} \mathcal{F}_V(\mathrm{Tr}_c(s_2))\right)$ $(\mathrm{IR}(\mathrm{AX}\left(\bigvee_{(s,s_2)\in R} \mathcal{F}_V(\mathrm{Tr}_c(s_2))\right), \overline{V})$, $s_0 \leftrightarrow_{\overline{V}} s)$
  $\Rightarrow (\mathcal{M}, s_0) \models \mathcal{X}$.

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 $\mathrm{IR}(\mathcal{F}_V(\mathcal{M}, s_0), \overline{V})$, 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 $\mathrm{Tr}_k(s_0) \leftrightarrow_{\overline{V}} \mathrm{Tr}_k(s_0')$ $(k > 0)$, we will prove $\mathrm{Tr}_{k+1}(s_0) \leftrightarrow_{\overline{V}} \mathrm{Tr}_{k+1}(s_0')$. We should only show that $\mathrm{Tr}_1(s_k) \leftrightarrow_{\overline{V}} \mathrm{Tr}_1(s_k')$. 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:

$(\mathcal{M}', s'_0) \models \mathcal{F}_V(\mathcal{M}, s_0)$

$\Rightarrow \forall s' \in S', (\mathcal{M}', s') \models \mathcal{F}_V(\mathrm{Tr}_c(s)) \to \left(\bigwedge_{(s,s_1)\in R} \mathrm{EX}\mathcal{F}_V(\mathrm{Tr}_c(s_1))\right) \wedge \mathrm{AX}\left(\bigvee_{(s,s_1)\in R} \mathcal{F}_V(\mathrm{Tr}_c(s_1))\right)$

for any $s \in S$. **(fact)**

(I) $(\mathcal{M}', s'_0) \models \mathcal{F}_V(\mathrm{Tr}_c(s_0)) \to \left(\bigwedge_{(s_0,s_1)\in R} \mathrm{EX}\mathcal{F}_V(\mathrm{Tr}_c(s_1))\right) \wedge \mathrm{AX}\left(\bigvee_{(s_0,s_1)\in R} \mathcal{F}_V(\mathrm{Tr}_c(s_1))\right)$ **(fact)**

(II) $(\mathcal{M}', s'_0) \models \mathcal{F}_V(\mathrm{Tr}_c(s_0)))$ (known)

(III) $(\mathcal{M}', s'_0) \models \left(\bigwedge_{(s_0,s_1)\in R} \mathrm{EX}\mathcal{F}_V(\mathrm{Tr}_c(s_1))\right) \wedge \mathrm{AX}\left(\bigvee_{(s_0,s_1)\in R} \mathcal{F}_V(\mathrm{Tr}_c(s_1))\right)$ ((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) $(\mathcal{M}', s'_0) \models \bigwedge_{(s_0,s_1)\in R} \mathrm{EX}\mathcal{F}_V(\mathrm{Tr}_c(s_1))$ (III)

(2) $\forall (s_0, s_1) \in R$, $\exists (s'_0, s'_1) \in R'$ *s.t.* $(\mathcal{M}', s'_1) \models \mathcal{F}_V(\mathrm{Tr}_c(s_1))$ (2)

(3) $\mathrm{Tr}_c(s_1) \leftrightarrow_{\overline{V}} \mathrm{Tr}_c(s'_1)$ ((2), Lemma 3)

(4) $L(s_1) - \overline{V} = L'(s'_1) - \overline{V}$ ((3), $c \geq 0$)

(5) $(\mathcal{M}', s'_1) \models \mathcal{F}_V(\mathrm{Tr}_c(s_1)) \to \left(\bigwedge_{(s_1,s_2)\in R} \mathrm{EX}\mathcal{F}_V(\mathrm{Tr}_c(s_2))\right) \wedge \mathrm{AX}\left(\bigvee_{(s_1,s_2)\in R} \mathcal{F}_V(\mathrm{Tr}_c(s_2))\right)$ **(fact)**

(6) $(\mathcal{M}', s'_1) \models \left(\bigwedge_{(s_1,s_2)\in R} \mathrm{EX}\mathcal{F}_V(\mathrm{Tr}_c(s_2))\right) \wedge \mathrm{AX}\left(\bigvee_{(s_1,s_2)\in R} \mathcal{F}_V(\mathrm{Tr}_c(s_2))\right)$ ((2), (5))

(7) ......

(8) $(\mathcal{M}', s'_k) \models \left(\bigwedge_{(s_k,s_{k+1})\in R} \mathrm{EX}\mathcal{F}_V(\mathrm{Tr}_c(s_{k+1}))\right) \wedge \mathrm{AX}\left(\bigvee_{(s_k,s_{k+1})\in R} \mathcal{F}_V(\mathrm{Tr}_c(s_{k+1}))\right)$ (similar with (6))

(9) $\forall (s_k, s_{k+1}) \in R$, $\exists (s'_k, s'_{k+1}) \in R'$ *s.t.* $(\mathcal{M}', s'_{k+1}) \models \mathcal{F}_V(\mathrm{Tr}_c(s_{k+1}))$ (8)

(10) $\mathrm{Tr}_c(s_{k+1}) \leftrightarrow_{\overline{V}} \mathrm{Tr}_c(s'_{k+1})$ ((9), Lemma 3)

(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) $(\mathcal{M}', s'_k) \models \mathrm{AX}\left(\bigvee_{(s_k,s_{k+1})\in R} \mathcal{F}_V(\mathrm{Tr}_c(s_{k+1}))\right)$ (by (8) talked above)

(2) $\forall (s'_k, s'_{k+1}) \in R'$, $\exists (s_k, s_{k+1}) \in R$ *s.t.* $(\mathcal{M}', s'_{k+1}) \models \mathcal{F}_V(\mathrm{Tr}_c(s'_{k+1}))$ (1)

(3) $\mathrm{Tr}_c(s_{k+1}) \leftrightarrow_{\overline{V}} \mathrm{Tr}_c(s'_{k+1})$ ((2), Lemma 3)

(4) $L(s_{k+1}) - \overline{V} = L'(s'_{k+1}) - \overline{V}$ ((3), $c \geq 0$)

**Theorem** 3.

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

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

Firstly, suppose that $(\mathcal{M}', s'_0)$ is a model of $\mathrm{F}_{\mathrm{CTL}}(\varphi, V)$. Then there exists an an initial Kripke structure $(\mathcal{M}, s_0)$ such that $(\mathcal{M}, s_0)$ is a model of $\varphi$ and $(\mathcal{M}, s_0) \leftrightarrow_V (\mathcal{M}', s'_0)$.

By Theorem 1, we have $(\mathcal{M}', s_0') \models \phi$ for all $\phi$ that $\varphi \models \phi$ and 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, $(\mathcal{M}', s_0') \models \bigvee_{(\mathcal{M}, s_0) \in Mod(\varphi)} \mathcal{F}_{\mathcal{A} \setminus V}(\mathcal{M}, s_0)$ due to $\bigvee_{(\mathcal{M}, s_0) \in Mod(\varphi)} \mathcal{F}_{\mathcal{A} \setminus V}(\mathcal{M}, s_0)$ is irrelevant to $V$.

Finally, suppose that $(\mathcal{M}', s_0')$ is a model of $\bigvee_{\mathcal{M}, s_0 \in Mod(\varphi)} \mathcal{F}_{\mathcal{A} \setminus V}(\mathcal{M}, s_0)$. Then there exists $(\mathcal{M}, s_0) \in Mod(\varphi)$ 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 $\phi'$ such that $\psi \models \phi'$ but $\{\phi | \varphi \models \phi, \text{IR}(\phi, V)\} \not\models \phi'$. There are three cases:

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

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

**Proposition** 8.

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

On the other hand, suppose that $(\mathcal{M}_1, s_1)$ be a model of $\text{F}_{\text{CTL}}(\text{F}_{\text{CTL}}(\varphi, p), V)$, then there exists an initial Kripke structure $(\mathcal{M}_2, s_2)$ such that $(\mathcal{M}_2, s_2) \models \text{F}_{\text{CTL}}(\varphi, p)$ 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 3, and consequently, $(\mathcal{M}_1, s_1) \models \text{F}_{\text{CTL}}(\varphi, \{p\} \cup V)$.

**Proposition** 10

*Proof.* Let $\mathcal{M} = (S, R, L)$ with initial state $s_0$ and $\mathcal{M}' = (S', R', L')$ 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 $(\mathcal{M}, s) \leftrightarrow_{\emptyset} (\mathcal{M}', s_0')$.

(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 *Mod*$(\mathrm{F}_{\mathrm{CTL}}(\mathrm{AX}\phi, V)) =$ *Mod*$(\mathrm{AXF}_{\mathrm{CTL}}(\phi, V))$:

$(\Rightarrow)$ $\forall(\mathcal{M}', s') \in$ *Mod*$(\mathrm{F}_{\mathrm{CTL}}(\mathrm{AX}\phi, V))$ there exists an initial 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 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 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 $\mathcal{M}', s' \models \mathrm{AX}(\mathrm{F}_{\mathrm{CTL}}(\phi, V))$.

$(\Leftarrow)$ $\forall$ $(\mathcal{M}_3, s_3) \in$ *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 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 $\mathrm{F}_{\mathrm{CTL}}(\mathrm{EX}\phi, V) \equiv \mathrm{EXF}_{\mathrm{CTL}}(\phi, V)$, we only need to prove *Mod* $(\mathrm{F}_{\mathrm{CTL}}(\mathrm{EX}\phi, V)) =$ *Mod*$(\mathrm{EXF}_{\mathrm{CTL}}(\phi, V))$:

$(\Rightarrow)$ $\forall \mathcal{M}', s' \in$ *Mod*$(\mathrm{F}_{\mathrm{CTL}}(\mathrm{EX}\phi, V))$ there exists an initial 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 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 I-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 $(\mathcal{M}', s') \models \mathrm{EX}(\mathrm{F}_{\mathrm{CTL}}(\phi, V))$.

$(\Leftarrow)$ $\forall$ $(\mathcal{M}_3, s_3) \in$ *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 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 $(\mathcal{M}, s) \leftrightarrow_V (\mathcal{M}_3, s_3)$
$\Rightarrow (\mathcal{M}, s) \models \mathrm{EX}\phi$ 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.

**Proposition** 22

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

($\Rightarrow$) 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 $\Gamma' \models q \rightarrow \varphi$, which means $\varphi$ is a NC of $q$ on $P$ under $\Gamma'$. Suppose $\varphi'$ is any NC of $q$ on $P$ under $\Gamma'$, then $\mathrm{F}_{\mathrm{CTL}}(\Gamma', q) \models \alpha \rightarrow \varphi'$ due to $\alpha \equiv q$, $\mathit{IR}(\alpha \rightarrow \varphi', \{q\})$ and **(pp)**, *i.e.* $\Gamma \models \alpha \rightarrow \varphi'$ by Lemma 6, this means $\mathit{NC}(\varphi', \alpha, P, \Gamma)$. Therefore, $\Gamma \models \varphi \rightarrow \varphi'$ by the definition of SNC and $\Gamma' \models \varphi \rightarrow \varphi'$. 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 $\mathrm{F}_{\mathrm{CTL}}(\Gamma', \{q\}) \models \alpha \rightarrow \varphi$ due to $\alpha \equiv q$, $\mathit{IR}(\alpha \rightarrow \varphi, \{q\})$ and **(pp)**, *i.e.* $\Gamma \models \alpha \rightarrow \varphi$ by Lemma 6, this means $\mathit{NC}(\varphi, \alpha, P, \Gamma)$. Suppose $\varphi'$ is any NC of $\alpha$ on $P$ under $\Gamma$. Then $\Gamma' \models q \rightarrow \varphi'$ since $\alpha \equiv q$ and $\Gamma' = \Gamma \cup \{q \equiv \alpha\}$, which means $\mathit{NC}(\varphi', q, P, \Gamma')$. According to $\mathit{SNC}(\varphi, q, P, \Gamma')$, $\mathit{IR}(\varphi \rightarrow \varphi', \{q\})$ and **(pp)**, we have $\mathrm{F}_{\mathrm{CTL}}(\Gamma', \{q\}) \models \varphi \rightarrow \varphi'$, and $\Gamma \models \varphi \rightarrow \varphi'$ by Lemma 6. Hence, $\mathit{SNC}(\varphi, \alpha, P, \Gamma)$ holds.

**Theorem** 10

*Proof.* We will prove the SNC part, while it is not difficult to prove the WSC part according to Proposition 21. Let $\mathcal{F} = \mathrm{F}_{\mathrm{CTL}}(\varphi \wedge q, \mathit{Var}(\varphi) \cup \{q\}) - V)$.

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

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

## References


1. Baier, C., Katoen, J.: Principles of Model Checking. The MIT Press (2008)
2. Bienvenu, M., Fargier, H., Marquis, P.: Knowledge compilation in the modal logic s5. In: Twenty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2010, Atlanta, Georgia, Usa, July. pp. 261–266 (2010)
3. Bolotov, A.: A clausal resolution method for ctl branching-time temporal logic. Journal of Experimental & Theoretical Artificial Intelligence **11**(1), 77–93 (1999). https://doi.org/10.1080/095281399146625
4. Browne, M.C., Clarke, E.M., Grumberg, O.: Characterizing finite kripke structures in propositional temporal logic. Theor. Comput. Sci. **59**, 115–131 (1988). https://doi.org/10.1016/0304-3975(88)90098-9, `https://doi.org/10.1016/0304-3975(88)90098-9`
5. Browne, M.C., Clarke, E.M., Grümberg, O.: Characterizing finite kripke structures in propositional temporal logic. Theoretical Computer Science **59**(1-2), 115–131 (1988)
6. Clarice, E., Emerson, E.: Synthesis of synchronization skeletons for branching time temporal logic. ACM Transactions on Programming Languages and Systems **8**(2), 244–263 (1986)

7. Clarke, E.M., Emerson, E.A., Sistla, A.P.: Automatic verification of finite-state concurrent systems using temporal logic specifications. ACM Trans. Program. Lang. Syst. **8**(2), 244–263 (1986). https://doi.org/10.1145/5397.5399, https://doi.org/10.1145/5397.5399
8. Delgrande, J.P.: A knowledge level account of forgetting. J. Artif. Intell. Res. **60**, 1165–1213 (2017). https://doi.org/10.1613/jair.5530, https://doi.org/10.1613/jair.5530
9. Eiter, T., Wang, K.: Semantic forgetting in answer set programming. Elsevier Science Publishers Ltd. (2008)
10. Konev, B., Walther, D., Wolter, F.: Forgetting and uniform interpolation in large-scale description logic terminologies. In: Twenty-First International Joint Conference on Artificial Intelligence (2009)
11. Lang, J., Marquis, P.: On propositional definability. Artificial Intelligence **172**(8), 991–1017 (2008)
12. Lang, J., Marquis, P.: Reasoning under inconsistency: a forgetting-based approach. Elsevier Science Publishers Ltd (2010)
13. Lin, F.: On strongest necessary and weakest sufficient conditions. Artificial Intelligence **128**(1), 143–159 (2001)
14. Lin, F.: On strongest necessary and weakest sufficient conditions. Artif. Intell. **128**(1-2), 143–159 (2001). https://doi.org/10.1016/S0004-3702(01)00070-4, https://doi.org/10.1016/S0004-3702(01)00070-4
15. Lin, F., Reiter, R.: Forget it. In: Working Notes of AAAI Fall Symposium on Relevance. pp. 154–159 (1994)
16. Marquis, P.: Propositional independence: formula-variable independence and forgetting. AI Access Foundation (2003)
17. Meier, A., Thomas, M., Vollmer, H., Mundhenk, M.: The complexity of satisfiability for fragments of ctl and ctl$^\star$. International Journal of Foundations of Computer Science **20**(05), 901–918 (2009)
18. Zhang, Y., Foo, N.Y., Wang, K.: 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. pp. 627–634 (2005)
19. Zhang, Y., Zhou, Y.: Knowledge forgetting: Properties and applications. Artificial Intelligence **173**(16), 1525–1537 (2009)
20. Zhang, Y., Zhou, Y.: Knowledge forgetting: Properties and applications. Artificial Intelligence **173**(16-17), 1525–1537 (2009)
21. Zhao, Y., Schmidt, R.A.: Fame: An automated tool for semantic forgetting in expressive description logics. In: Galmiche, D., Schulz, S., Sebastiani, R. (eds.) Automated Reasoning (IJCAR 2018). Lecture Notes in Artificial Intelligence, vol. 10900, pp. 19–27. Springer (2018). https://doi.org/https://doi.org/10.1007/978-3-319-94205-6_2, http://www.cs.man.ac.uk/~schmidt/publications/ZhaoSchmidt18a.html
22. Zhao, Y., Schmidt, R.A.: Role forgetting for alcoqh ($\delta$)-ontologies using an ackermann-based approach. In: Proceedings of the 26th International Joint Conference on Artificial Intelligence. pp. 1354–1361. AAAI Press (2017)