

Emergency kit for CTL

François Laroussinie

14 janvier 2025

draft !

1 CTL

Let AP be a finite set of atomic propositions. CTL formulas are defined as follows :

$$\text{CTL} \ni \varphi, \psi ::= p \mid \varphi \vee \psi \mid \neg \varphi \mid \mathbf{EX} \varphi \mid \mathbf{AX} \varphi \mid \mathbf{E} \varphi \mathbf{U} \psi \mid \mathbf{A} \varphi \mathbf{U} \psi$$

with $p \in \text{AP}$.

A CTL formula is interpreted over a state of a Kripke structure $\mathcal{S} = \langle S, s_0, \rightarrow, \ell \rangle$:

- $\mathcal{S}, s \models p$ iff $p \in \ell(s)$
- $\mathcal{S}, s \models \varphi \vee \psi$ iff $\mathcal{S}, s \models \varphi$ or $\mathcal{S}, s \models \psi$
- $\mathcal{S}, s \models \neg \varphi$ iff $\mathcal{S}, s \not\models \varphi$
- $\mathcal{S}, s \models \mathbf{EX} \varphi$ iff there exists $s \rightarrow s'$ s.t. $\mathcal{S}, s' \models \varphi$
- $\mathcal{S}, s \models \mathbf{E} \varphi \mathbf{U} \psi$ iff there exists some $\rho \in \text{Exec}(s)$ s.t. $\exists j \geq 0 : \mathcal{S}, \rho(j) \models \psi$ and $\forall 0 \leq k < j$, we have : $\mathcal{S}, \rho(k) \models \varphi$
- $\mathcal{S}, s \models \mathbf{A} \varphi \mathbf{U} \psi$ iff for any $\rho \in \text{Exec}(s)$, we have $\exists j \geq 0 : \mathcal{S}, \rho(j) \models \psi$ and $\forall 0 \leq k < j$, we have : $\mathcal{S}, \rho(k) \models \varphi$

where $\text{Exec}(s)$ denotes the set of infinite executions from s .

2 Model checking algorithm

Model-checking : $\mathcal{S} \models \varphi$

input : $\mathcal{S} = \langle S, s_0, \rightarrow, \ell \rangle$ a Kripke structure, $\varphi \in \text{CTL}$

output : Yes iff there exists $w \in W(\mathcal{S})$ such that $\mathcal{D}, s_0 \models \varphi$.

The model-checking algorithm for CTL consists in **marking** every state of the structure by the subformulas it satisfies. In the following we use $s.\psi$ to denote the variable storing the truth value of ψ at $s \in S$. We proceed inductively over the formula as described in Algorithms 1 and 2.

Let Φ be a CTL formula and $\mathcal{S} = \langle S, s_0, \rightarrow, \ell \rangle$ be a Kripke structure. The complexity of this algorithm is stated as follows : for Boolean cases, we get procedures in $O(|S|)$ and for temporal operators, we have procedures in $O(|S| + |\rightarrow|)$. We use $|\mathcal{S}|$ to denote $|S| + |\rightarrow|$. The overall complexity of the model-checking algorithm is in $O(|\Phi| \cdot |\mathcal{S}|)$.

The satisfiability of CTL is also decidable but the complexity is higher. Finally we have :

Procedure Mark(ψ) (part 1.)

```

case  $\psi = p$  : do
  | foreach  $s \in S$  do
  |   | if  $p \in \ell(s)$  then  $s.\psi := \text{true}$ ;
  |   | else  $q.\psi := \text{false}$ ;
  |   end
  end
case  $\psi = \neg\psi_1$  : do
  | Mark( $\psi_1$ );
  | foreach  $s \in S$  do
  |   |  $s.\psi := \neg s.\psi_1$ ;
  |   end
  end
case  $\psi = \psi_1 \vee \psi_2$  : do
  | Mark( $\psi_1$ ); Mark( $\psi_2$ );
  | foreach  $s \in S$  do
  |   |  $s.\psi := \neg s.\psi_1 \vee \neg s.\psi_2$ ;
  |   end
  end
case  $\psi = \text{EX } \psi_1$  : do
  | Mark( $\psi_1$ );
  | foreach  $s \in S$  do  $s.\psi := \text{false}$ ;
  | foreach  $s \rightarrow s'$  do
  |   | if  $s'.\psi_1$  then  $s.\psi := \text{true}$ ;
  |   end
  end
case  $\psi = \text{AX } \psi_1$  : do
  | Mark( $\psi_1$ );
  | foreach  $s \in S$  do  $s.\psi := \text{true}$ ;
  | foreach  $s \rightarrow s'$  do
  |   | if  $\neg s'.\psi_1$  then  $s.\psi := \text{false}$ ;
  |   end
  end

```

Algorithme 1 : Model-checking for CTL (part 1)

Theorem 1. — *CTL Model-checking is P-complete.*
 — *CTL satisfiability is EXPTIME-complete.*

Procedure Mark(ψ) (part 1.)

```

case  $\psi = \mathbf{E}\psi_1 \mathbf{U} \psi_2$  : do
  Mark( $\psi_1$ ); Mark( $\psi_2$ );  $L := \emptyset$ ;
  foreach  $s \in S$  do
    if  $s.\psi_2$  then  $s.\psi := \text{true}$ ;  $L := L \cup \{s\}$ ;
    else  $s.\psi := \text{false}$ ;
  end
  while  $L \neq \emptyset$  do
    pick a state  $s$  in  $L$ ;
    foreach  $s' \rightarrow s$  do
      if  $s'.\psi_1 \wedge \neg s'.\psi$  then  $L := L \cup \{s'\}$ ;  $s'.\psi := \text{true}$ ;
    end
  end
case  $\psi = \mathbf{A}\psi_1 \mathbf{U} \psi_2$  : do
  Mark( $\psi_1$ ); Mark( $\psi_2$ );  $L := \emptyset$ ;
  foreach  $s \in S$  do
     $s.\text{nb} := \text{deg}^-(s)$ ; //  $\text{deg}^-(s)$  is the out-degree of  $s$ 
    if  $s.\psi_2$  then  $s.\psi := \text{true}$ ;  $L := L \cup \{s\}$ ;
    else  $s.\psi := \text{false}$ ;
  end
  while  $L \neq \emptyset$  do
    pick a state  $s$  in  $L$ ;
    foreach  $s' \rightarrow s$  do
       $s'.\text{nb} := s'.\text{nb} - 1$ ;
      if  $s'.\psi_1 \wedge s'.\text{nb} = 0 \wedge \neg s'.\psi$  then  $L := L \cup \{s'\}$ ;  $s'.\psi := \text{true}$ ;
    end
  end

```

Algorithm 2 : Model-checking for CTL (part 2)

3 Expressivity

It is easy to see that some CTL properties cannot be expressed with LTL. For example, the CTL formula $\mathbf{AG}(\mathbf{EF} p)$ (*i.e.* "from any reachable state, one can reach a state satisfying p ") has no equivalent in LTL. We can consider the two structures \mathcal{S} and \mathcal{S}' at Figure 1 : they clearly have the same set of "traces" (*i.e.* the set labeled executions) with $(\neg p)^+ \cdot p^\omega \cup (\neg p)^\omega$, and then they satisfy the same LTL formulas, but $s_0 \models \mathbf{AG}(\mathbf{EF} p)$ and $s'_0 \not\models \mathbf{AG}(\mathbf{EF} p)$ (there is no way to reach a p state from s'_2).

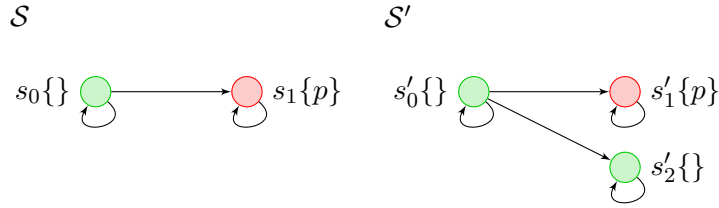


FIGURE 1 – Kripke structures \mathcal{S} and \mathcal{S}' for $\mathbf{AX}(\mathbf{EF} p)$.

In conclusion, LTL is not "at least as expressive as" CTL. But the converse is also true. There are LTL properties that cannot be expressed with CTL. Consider the problem $\mathcal{S} \models_{\exists} \mathbf{GF} p$ which states the existence of a path along which p is true infinitely many times. This property cannot be expressed with CTL. But we cannot proceed as before to prove this result because CTL has a stronger *distinguishing power* than LTL and it entails that for any Kripke structures \mathcal{S} and \mathcal{S}' , if $\mathcal{S} \models_{\exists} \mathbf{GF} p$ and $\mathcal{S}' \not\models_{\exists} \mathbf{GF} p$, then there exists a CTL formula φ such $\mathcal{S} \models \varphi$ and $\mathcal{S}' \not\models \varphi$ (see below). The proof ($\mathbf{GF} p$ cannot be expressed with CTL) is thus more complex and it is not presented here.

3.1 Distinguishing power

The distinguishing power is the ability of a logic to distinguish two models (Kripke structures) : \mathcal{S} and \mathcal{S}' are distinguished by a formula φ if $\mathcal{S} \models \varphi$ and $\mathcal{S}' \not\models \varphi$. In the following we use $\mathcal{S} \equiv_{\mathcal{L}} \mathcal{S}'$ to denote that $\mathcal{S} \models \varphi \Leftrightarrow \mathcal{S}' \models \varphi$ for any $\varphi \in \mathcal{L}$.

We say that a logic \mathcal{L} distinguishes *at least as* \mathcal{L}' iff for any models \mathcal{S} and \mathcal{S}' , we have $\mathcal{S} \equiv_{\mathcal{L}} \mathcal{S}'$ implies $\mathcal{S} \equiv_{\mathcal{L}'} \mathcal{S}'$.

Strong bisimulation. We will see that the distinguishing power of CTL coincides with the (strong) bisimulation. We have the following definition :

Definition 2. Let $\mathcal{S}_1 = \langle S_1, s_0^1, \rightarrow_1, \ell_1 \rangle$ and $\mathcal{S}_2 = \langle S_2, s_0^2, \rightarrow_2, \ell_2 \rangle$ be two Kripke structures. A relation $\mathcal{R} \subseteq S_1 \times S_2$ is a *bisimulation* iff for any $(s_1, s_2) \in \mathcal{R}$, we have :

1. $\ell_1(s_1) = \ell_2(s_2)$,
2. $\forall s_1 \rightarrow_1 s'_1, \exists s_2 \rightarrow_2 s'_2$ such that $(s'_1, s'_2) \in \mathcal{R}$, and
3. $\forall s_2 \rightarrow_2 s'_2, \exists s_1 \rightarrow_1 s'_1$ such that $(s'_1, s'_2) \in \mathcal{R}$.

We say that two states s_1 and s_2 are bisimilar (denoted $s_1 \sim s_2$) iff there exists a bisimulation relation \mathcal{R} such that $(s_1, s_2) \in \mathcal{R}$. And two KS \mathcal{S}_1 and \mathcal{S}_2 are bisimilar (denoted $\mathcal{S}_1 \sim \mathcal{S}_2$) iff their initial states are bisimilar.

We have the following theorem :

Theorem 3 (Hennessy 1980). Let $\mathcal{S}_1 = \langle S_1, s_0^1, \rightarrow_1, \ell_1 \rangle$ and $\mathcal{S}_2 = \langle S_2, s_0^2, \rightarrow_2, \ell_2 \rangle$ be two finitely branching¹ Kripke structures. Let $s_1 \in S_1$ and $s_2 \in S_2$, we have :

$$s_1 \sim s_2 \quad \Leftrightarrow \quad s_1 \equiv_{\text{CTL}} s_2$$

Proof. \Rightarrow : Consider a bisimulation relation \mathcal{R} . We prove that for any $(s_1, s_2) \in \mathcal{R}$, we have $s_1 \models \varphi \Leftrightarrow s_2 \models \varphi$ for any $\varphi \in \text{CTL}$. The proof is done by structural induction over the formula :

- $\varphi = p \in \text{AP}$: $(s_1, s_2) \in \mathcal{R}$ implies that $\ell_1(s_1) = \ell_2(s_2)$, and then $p \in \ell_1(s_1) \Leftrightarrow p \in \ell_2(s_2)$.
- $\varphi = \psi_1 \wedge \psi_2$: Assume $s_1 \models \psi_1 \wedge \psi_2$. Thus by def., we have $s_1 \models \psi_1$ and $s_1 \models \psi_2$, and by i.h. we get $s_2 \models \psi_1$ and $s_2 \models \psi_2$, and then $s_2 \models \psi_1 \wedge \psi_2$.
- $\varphi = \neg\psi_1$: Assume $s_1 \models \neg\psi_1$. Then $s_1 \not\models \psi_1$ and by i.h. we have $s_2 \not\models \psi_1$ and then $s_2 \models \neg\psi_1$.
- $\varphi = \mathbf{EX} \psi_1$. Assume $s_1 \models \mathbf{EX} \psi_1$. By def., we know there exists $s_1 \rightarrow_1 s'_1$ such that $s'_1 \models \psi_1$. As $(s_1, s_2) \in \mathcal{R}$, there exists $s_2 \rightarrow_2 s'_2$ such that $(s'_1, s'_2) \in \mathcal{R}$. By i.h. we deduce $s'_2 \models \psi_1$, and then (by def. of \mathbf{EX}) we have $s_2 \models \mathbf{EX} \psi_1$.
- $\varphi = \mathbf{E}\psi_1 \mathbf{U} \psi_2$: Assume $s_1 \models \mathbf{E}\psi_1 \mathbf{U} \psi_2$. Then there exists some execution $\rho \in \text{Exec}(s_1)$ and $i \geq 0$ such that (1) $\rho(i) \models \psi_2$ and (2) for any $0 \leq j < i$, we have $\rho(j) \models \psi_1$. We proceed exactly as in the previous case, and we can deduce (from the definition of the bisimulation) that there exists an execution $\rho' \in \text{Exec}(s_2)$ issued from s_2 such that (1) $\rho(k) \sim \rho'(k)$ for any $0 \leq k \leq i$, and then the i.h. allows us to deduce that (1) $\rho'(i) \models \psi_2$ and (2) for any $0 \leq j < i$, we have $\rho'(j) \models \psi_1$, and therefore we have $s_2 \models \mathbf{E}\psi_1 \mathbf{U} \psi_2$.
- $\varphi = \mathbf{EG} \psi_1$: as in the previous case².

\Leftarrow : Now we aim at proving that $s_1 \equiv_{\text{CTL}} s_2$ implies $s_1 \sim s_2$. For this, it is sufficient to prove that there exists a bisimulation relation. We consider the relation $\mathcal{R} = \{(s_1, s_2) \mid s_1 \equiv_{\text{CTL}} s_2\}$. We now prove that it is a bisimulation. Consider $(s_1, s_2) \in \mathcal{R}$.

We clearly have $\ell_1(s_1) = \ell_2(s_2)$ because $s_1 \equiv_{\text{CTL}} s_2$ and CTL contains atomic propositions. Now assume $s_1 \rightarrow_1 s'_1$. Can we find some $s_2 \rightarrow_2 s'_2$ s.t. $(s'_1, s'_2) \in \mathcal{R}$? Assume there is no such a state s'_2 , that is every successor of s_2 can be distinguished by a CTL formula from s'_1 . Let $\{r_1, \dots, r_k\}$ be the set of (immediate) successors of s_2 (this set is finite thanks to the finitely branching hypothesis). Thus we know that for any $1 \leq i \leq k$, there exists some CTL formula ψ_i such that $s'_1 \not\models \psi_i$ and $r_i \models \psi_i$. Therefore we have :

$$s_1 \models \mathbf{EX} \left(\bigwedge_{1 \leq i \leq k} \neg\psi_i \right) \quad \text{and} \quad s'_1 \models \mathbf{AX} \left(\bigvee_{1 \leq i \leq k} \psi_i \right)$$

And then $s_1 \not\equiv_{\text{CTL}} s_2$ and this contradicts the initial hypothesis. \square

1. every state has a finite number of successors.

2. remember that $\mathbf{A}\psi_1 \mathbf{U} \psi_2 \equiv \neg \mathbf{EG} \neg\psi_2 \wedge \neg \mathbf{E}(\psi_2) \mathbf{U} (\neg\psi_1 \wedge \neg\psi_2)$