Temporal logics LTL, CTL and CTL*

François SCHWARZENTRUBER

Chapter 1

The framework 'Computation Tree Logic*'

1.1 Syntax

Let ATM be a set of atomic propositions.
[vérification de logiciels, p. 33]

Definition 1 ()

$$\varphi ::= \bot \mid p \mid \neg\varphi \mid \varphi \vee \varphi \mid \bigcirc\varphi \mid F\varphi \mid G\varphi \mid \varphi U \varphi \mid \exists\varphi \mid \forall\varphi$$

where $p \in ATM$.

Remark 1 *Some people [principles of model checking, p. 422] differantiate* CTL^* *state formulae (also called* CTL^* *formulae) from Path formulae. They do as follows.* CTL^* *state formulae (also called* CTL^* *formulae) are formed according to the following grammar:*

$$\Phi ::= \bot \mid p \mid \neg\Phi \mid \Phi \vee \Phi \mid \exists\varphi \mid \forall\varphi$$

where $p \in ATM$ *and* φ *is a path formula. Path formulae are given by the following grammar:*

$$\varphi ::= \Phi \mid \neg\varphi \mid \varphi \vee \varphi \mid \bigcirc\varphi \mid \varphi U \varphi$$

.

I find it useless and prefer the point of view of [vérification de logiciels, Techniques et outils du model-checking, p. 33].

Remark 2 $F\varphi := \top U \varphi$
$G\varphi := \neg F \neg\varphi = \neg\top U \neg\varphi$

$\varphi W \psi := \varphi U \psi \vee G\varphi$
$\varphi R \psi := \neg(\neg\varphi U \neg\psi)$ (release) [p. 256]
$\forall\varphi = \neg\exists\neg\varphi$
$\bigcirc$ and U are called linear temporal operators.
$\exists$ and $\forall$ are called path quantifiers.

1.2 Semantics

We evaluate a formula in a model and a path (run) in model. A model is a transition system, that is a Kripke structure $\mathcal{M} = (W, R, V)$ that is serial (for all $w \in W$, $R(w) \neq \emptyset$). A path π is a sequence $\pi_0, \pi_1, \dots$ such that $\pi_i \in W$ and $\pi_i R \pi_{i+1}$ for all $i \geq 0$.

Definition 2 ()

- $\mathcal{M}, \pi \models p$ iff $\pi_0 \in V(p)$;
- $\mathcal{M}, \pi \models \bigcirc\varphi$ iff $\mathcal{M}, \pi[1..\infty] \models \varphi$
- $\mathcal{M}, \pi \models F\varphi$ iff there exists i such that $\mathcal{M}, \pi[i, \dots] \models \varphi$
- $\mathcal{M}, \pi \models G\varphi$ iff for all i such that $\mathcal{M}, \pi[i, \dots] \models \varphi$
- $\mathcal{M}, \pi \models \varphi U \psi$ iff there exists a integer j such that $\mathcal{M}, \pi[j..\infty] \models \psi$ and for all $i < j$, we have $\mathcal{M}, \pi[i..\infty] \models \varphi$
- $\mathcal{M}, \pi \models \exists\varphi$ iff there exists a path π' in $\mathcal{M}$ starting with π_0 such that $\mathcal{M}, \pi' \models \varphi$.
- $\mathcal{M}, \pi \models \forall\varphi$ iff for all paths π' in $\mathcal{M}$ starting with π_0 such that $\mathcal{M}, \pi' \models \varphi$.

[je diffère de principles of model-checking, mais je trouve ça plus clair avec les indices $\forall$ et $\exists$.. eux ils font universels]

- $\mathcal{M}, s \models_\forall \Phi$ iff for all path π starting from s we have $\mathcal{M}, \pi \models \Phi$;
- $\mathcal{M}, s \models_\exists \Phi$ iff there exists a path π starting from s such that $\mathcal{M}, \pi \models \Phi$.

A formula φ is satisfiable iff there exists a structure $\mathcal{M}, s$ such that $\mathcal{M}, s \models_\exists \varphi$. A formula is valid iff for all structure $\mathcal{M}, s$ we have $\mathcal{M}, s \models_\forall \varphi$.

1.2.1 Fragments

[principles of model-checking p. 422]

- LTL = only linear temporal operators (no $\exists$, no $\forall$)
- CTL = each linear temporal operator must be immediately preceded by a path quantifer.

Example 1

1.2.2 Decision problems

The $\models_\exists$ -model-checking problem of $CTL*$ (LTL, CTL) is defined as follows:

- input: a pointed model $\mathcal{M}, s$ and a formula φ of $CTL*, LTL$ or CTL
- output: yes iff $\mathcal{M}, s \models \varphi$.

The satisfiability problem is defined as follows:

- input: a formula φ of $CTL*, LTL$ or CTL
- output: yes iff φ is satisfiable.

Chapter 2

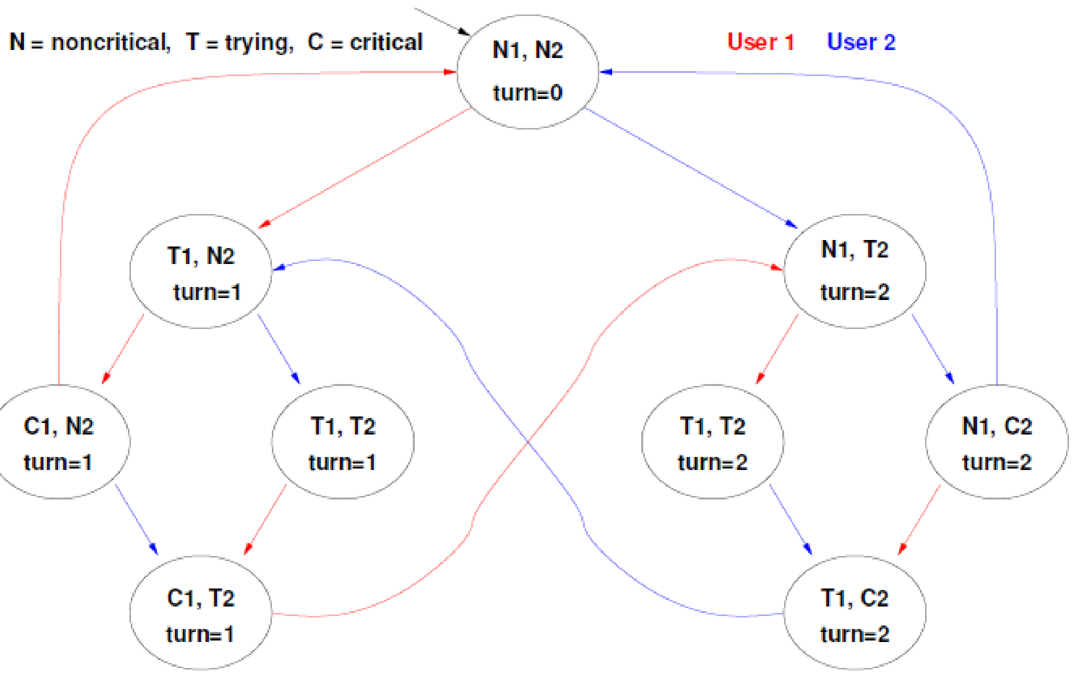
Linear Temporal Logic

2.1 Examples of properties

2.1.1 Properties LTL can express

Example 2 *'infinitely often φ': $GF\varphi$*
'eventually forever φ: $FG\varphi$

Example 3 *We consider the following Kripke structure:*



Safety (sûreté) (something bad never happens): yes. $\mathcal{M}, s \models_\forall G(\neg(c_1 \wedge c_2))$
Liveness (vivacité) (something good eventually happens): no. $\mathcal{M}, s \not\models_\forall Fc_1$

Liveness: yes. $\mathcal{M}, s \models_\forall G(t_1 \to Fc_1)$
Fairness (équité): $\mathcal{M}, s \not\models_\forall GFc_1$
Strong fairness (équité forte): $\mathcal{M}, s \models_\forall GFt_1 \to GFc_1$

Example 4 *Alternation:* $G(p \leftrightarrow \bigcirc\neg p)$
$\mathcal{M}, \pi \models G(p \leftrightarrow \bigcirc\neg p)$ *iff* $\{i \in \mathbb{N} \mid \pi_i \in V(p)\}$ = *even numbers or odd numbers*

2.1.2 Properties that LTL can not express

["temporal logic can be more expressive" from Pierre Wolper]

Theorem 1 *'p is true on each even moments' is not expressible in LTL.*

Proof.
It seems that the formula $p \wedge G(p \to \bigcirc\bigcirc p)$ expresses the property. But it is false in: $p, p, p, \neg p, p, p, p, p...$ TODO: ■

2.2 Axiomatization

[Gabbay, Pnueli, Shelah, and Stavi, 1980]
Axioms for LTL: all instances of:
[proof of completeness can be found in "Temporal logic of Programs, P; 26-...] [from Reynols paper, the axiomatization of CTL*, a bit strange with $Gp \to p \wedge \bigcirc p \wedge \bigcirc Gp$... but Wolper 83, Temporal logic can be more expressive use also this... warning, in Wolper 83, U is the weak version]

- Propositional tautologies;
- $Fp \leftrightarrow \neg G\neg p$;
- $\bigcirc(p \to q) \to (\bigcirc p \to \bigcirc q)$;
- $G(p \to q) \to (Gp \to Gq)$;
- $\neg \bigcirc p \leftrightarrow \bigcirc\neg p$
- $Gp \to p \wedge \bigcirc p \wedge \bigcirc Gp$;
- $G(p \to \bigcirc p) \to (p \to Gp)$ (induction);
- $pUq \leftrightarrow q \vee (p \wedge \bigcirc(pUq)$;
- $pUq \to Fq$.

Rules:

- modus ponens;
- Necessitation for G;

2.3 Model-checking and satisfiability problem of LTL

2.3.1 Satisfiability problem of LTL

[Sistla and Clarke, the complexity of PLTL]

The satisfiability problem is defined as follows:

- a formula φ;
- is there a model $\mathcal{M} = (W, R)$ and a run π such that $\pi \models \varphi$.

Remark 3 *Contrary to S4, K etc. we can have an exponential branch in the model. We can force it with the following formula:*

$G \bigvee_{i=1}^{n} \neg p_i \wedge \bigwedge_{j=i+1}^{n} p_j \rightarrow X(\bigwedge_{j=i+1}^{n} \neg p_j) \wedge p_i \wedge (\bigwedge_{j=1}^{i-1}(p_j \rightarrow Xp_j) \wedge (\neg p_j \rightarrow X\neg p_j)$

Let φ the formula we want to know whether it is satisfiable of not. We are going to prove that if φ is satisfiable then it is satisfiable in a regular run, called 'ultimately periodic run'. In this section, a run π is considered as an infinite sequence of subsets of $2^{ATM}(\varphi)$ where $ATM(\varphi)$ are the atomic propositions appearing in φ.

We note $[i]_\pi$ the set of all formulas $\psi \in SF(\varphi)$ such that $\pi[i..] \models \psi$.

Lemma 1 *If $i < j$ and $[i]_\pi = [j]_\pi$ then if we define $\pi' = (\pi_0, \dots, \pi_{i-1}, \pi_j, \pi_{j+1}, \dots,$ then for all $k \in \mathbb{N} \setminus \{i, \dots, j-1\}$, $[k]_\pi = [k]_{\pi'}$.*

Proof.

By induction on ψ.

■

Let ∞_π be the set of $S \subseteq SF(\varphi)$ such that there exists an infinity of k such that $[k]_\pi = S$.

Definition 3 ()

A run π is said to be ultimately periodic with starting index i and period p if for all $k \geq i$, $\pi_k = \pi_{k+p}$.

Lemma 2 *Let $i, p \in \mathbb{N}$ such that $[i]_\pi = [i+p]_\pi$ and $\forall S \in \infty_\pi$, $\exists k \in \{i, \dots, i+p-1\}$ such that $[k]_\pi = S$.*

Let π' the ultimately periodic run with starting index i and period p defined by for all $k < i+p$, $\pi'_k = \pi_k$.

Then:

- *for all $k < i+p$, $[k]_{\pi'} = [k]_\pi$;*
- *for all $k > i$, $[k]_{\pi'} = [k+p]_\pi$.*

Proof.

By induction, we prove that for all $\psi \in SF(\varphi)$ we have:

- for all $k < i+p$, $\pi'[k..] \models \psi$ iff $\pi[k..] \models \psi$;
- for all $k > i$, $\pi'[k..] \models \psi$ iff $\pi'[k+p..] \models \psi$.

■

Theorem 2 *A formula φ is satisfiable iff it is satisfiable in an ultimately periodic path with starting index i and period p where:*

- *$i \leq 2^{|\varphi|}$;*
- *$p \leq 4^{|\varphi|}$.*

Proof.

The formula φ is satisfiable in a run π. Let j, q such that $[j]_\pi = [j+q]_\pi$ and for all $S \in \infty_\pi$, there exists $k \in \{j, \dots, j+q-1\}$ such that $[k]_\pi = S$.

We now shorten the run so that $j \leq 2^{1+|\varphi|}$ with Lemma 1: we shorten the $\pi[0..j-1]$ by removing repetitions.

We then shorten the run so that $q \leq 4^{1+|\varphi|}$ with Lemma 1: we shorten the $\pi[j, j+q+1]$ by removing repetitions between two occurrences of $[k]_\pi \in \infty_\pi$.

We conclude with Lemma 2.

■

Theorem 3 *LTL-SAT is PSPACE.*

Proof.

The proof starts with a definition of Hintikka set.

Definition 4 ()
A Hintikka set over Σ is a set H saturated in the following way:

- If $\neg X\psi \in H$ then $X\neg\psi \in H$;

- If $\varphi U \psi \in H$ then $\psi \in H$ or ($\varphi \in H$ and $X(\varphi U \psi)$;
- If $\neg(\varphi U \psi) \in H$, $\neg\psi \in H$ and $\neg\varphi \vee X(\neg(\varphi U \psi))$.

We design a PSPACE algorithm for satisfiability problem of LTL taking in account the fact that if a formula φ is satisfiable then it suffices to find a ultimately periodic path with starting index i and period p where:

- $i \leq 2^{|\varphi|}$;
- $p \leq 4^{|\varphi|}$.

■

2.3.2 PSPACE-hardness of model-checking of LTL

Theorem 4 *Model-checking of LTL is PSPACE-hard.*

PROOF.
We reduce the corridor tiling problem to the LTL model-checking. The model $\mathcal{M}$ encodes the horizontal conditions. The formula φ enforces a path that represents a corridor tiling.

The worlds of the model $\mathcal{M}$ are a world "begin" and pairs (t, i) where t is a tile type and $i \in \{0, \ldots, n-1\}$. Are connected:

- *begin* to $(t, 0$ for all t;
- $(t, i) \rightarrow (t', i+1)$ iff $right(t) = left(t')$ and $i < n-1$;
- $(t, n-1)$ to *begin*.

Propositions are p_t. p_t is true in only pairs (t, i). The formula φ is the conjunction of:

- $\bigwedge_{i \in \{0,\ldots,n-1\}} \bigcirc^{i+1} b_i$;
- $F(begin \wedge \bigwedge_{i \in \{0,\ldots,n-1\}} \bigcirc^{i+1} e_i)$;
- $G \bigwedge_{t \in T} p_t \rightarrow \bigcirc^{n+1} \bigvee_{t' \in T | up(t') = down(t')} p_{t'}$

■ As the model-checking of LTL consists in encoding the problem into the satisfiability problem of LTL! That is why we study the satisfiability problem!

2.3.3 Encoding the model-checking of LTL into the LTL-satisfiability problem

Theorem 5 *The model-checking of LTL is reductible (est réductible en temps polynomial) to the LTL-satisfiability problem.*

Proof.
Let $\mathcal{M} = (W, R, V)$ and φ.

We extend the set of atomic propositions with propositions in_s for all $s \in W$ meaning that the current point is the world $s \in W$.

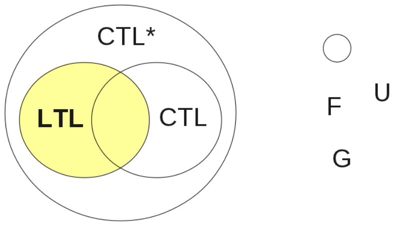
For all $w \in W$, we define:

- $here_w = p_w \wedge \bigwedge_{v \in W \setminus \{w\}} \neg p_v$;
- $val_w = \bigwedge_{p | w \in V(p)} p \wedge \bigwedge_{p | w \notin V(p)} \neg p$;
- $succ_w = \bigcirc \bigvee_{u \in R(w)} in_w$;
- $\varphi_w = here_w \wedge val_w \wedge succ_w$.

We have $\mathcal{M}, w \models \varphi$ iff the formula $\varphi \wedge in_w \wedge G \bigvee_{u \in W} in_u$ is LTL-satisfiable. ■

Theorem 6 *Model-checking of LTL is PSPACE.*

Theorem 7 *Satisfiability problem of LTL is PSPACE-hard.*



Satisfiability and model-checking of LTL

in PSPACE

Non-deterministic algorithm for satisfiability of a LTL-formula φ

function <u>satLTL</u>(φ)
  choose $i \in \{1, \dots, 2^{|\varphi|}\}$
  choose $p \in \{1, \dots, 4^{|\varphi|}\}$
  $state := hintikkaSaturate(\{\varphi\})$
  for $j := 1$ to $i - 1$
    $state := hintikkaSaturate(\{\psi \mid \bigcirc\psi \in state\})$
  endFor
  $state_i = state$
  $formulaToFulFill = \{\psi \mid \psi' U \psi \in state_i$ and $\psi \notin state_i\}$
  for $j := i + 1$ to $i + p - 1$
    $state := hintikkaSaturate(\{\psi \mid \bigcirc\psi \in state\})$
    $formulaToFulFill = state \cap formulaToFulFill$
  endFor
  if $state \not\subseteq state_i$ **reject**
  if $formulaToFullFill \neq \emptyset$ **reject**
  accept
function

where $hintikkaSaturate(\Sigma)$ non-deterministically returns a Hintikka set over Σ. If there is no such Hintikka set, $hintikkaSaturate(\Sigma)$ fails.

Algorithm for model-checking of a LTL-formula φ based on a reduction from the LTL-$\models_\exists$-model-checking problem to the LTL-satisfiability problem

input: $\mathcal{M} = (W, R, V)$, $w \in W$ and φ.
output: the set of worlds w such that $\mathcal{M}, w \models_\exists \varphi$
function <u>mc$_\exists$LTL</u>($\mathcal{M}, \varphi$)
  oneworld says 'at each step, we are in at most one world of model $\mathcal{M}$' with extra-propositions in_u saying that 'the current world is u'.
  $oneworld := G\left(\bigvee_{u \in W} in_u \rightarrow \bigwedge_{v \in W \setminus \{u\}} \neg in_v\right);$
  valuations says 'at each step, if we are in world u we copy the corresponding valuation from $\mathcal{M}$'
  $valuations := G\left(\bigvee_{u \in W} in_u \rightarrow \left(\bigwedge_{p \mid u \in V(p)} p \wedge \bigwedge_{p \mid u \notin V(p)} \neg p\right)\right)$
  path says 'we are following a path in $\mathcal{M}$
  $path := G\left(\bigvee_{u \in W} in_u \rightarrow \bigcirc \bigvee_{v \in R(u)} in_v\right);$
  return $\{w \in W \mid$ <u>satLTL</u>($\varphi \wedge in_w \wedge oneworld \wedge path \wedge valuations)\}$
endFunction

Chapter 3

Branching-time logic : CTL*, CTL

3.1 Motivation

[p. 315]

$\forall G \exists F start$: it is always possible to return in the initial state.

$\forall G \bigcirc \bigcirc start$: it is always possible to return in the initial state in two steps.

$\forall G \exists F start$ is a CTL formula but $\forall G \bigcirc \bigcirc start$ is not.

$\forall X G \neg p \wedge \exists F G (p \vee \forall (q U p))$

[p. 422, section about CTL*]

3.2 Model-checking

3.2.1 Model-checking of CTL* in PSPACE

The model-checking of CTL* is dynamic programming: it consists in applying model-checking of LTL on subformulas without path quantification.

3.2.2 Model-checking of CTL in P

The benefit of the fragment of CTL is that for all formulas φ of CTL we have $\mathcal{M}, \pi \models \varphi$ iff $\mathcal{M}, \pi_0 \models_{\exists} \varphi$ iff $\mathcal{M}, \pi_0 \models_{\forall} \varphi$. That is: we do not care about the path.

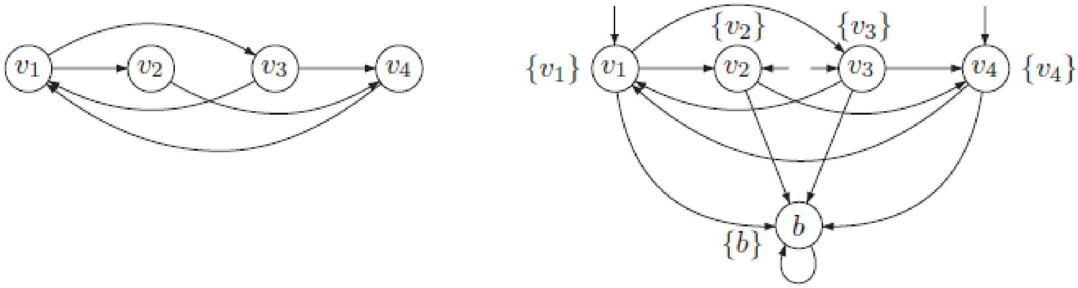
Easy! As for K! [p. 341] [vérification de logiciels. Schnoebelen et al., p. 40]

[Vérification des Systèmes Réactifs Temps-Réel, cours école polytechnique, p. 66(pdf)]

Theorem 8 *$O(|\varphi| * (|W| + |R|))$ in time.*

Example 5 (p. 356) *Let $G = (V, E)$ be a connected and directed graph. We say that $v_1, \ldots, v_n$ is an hamiltonian path iff $V = \{v_1, \ldots, v_n\}$.*

In order to solve the Hamiltonian path we define a Kripke model $\mathcal{M}_G$ where the worlds are the vertices of G plus an extra state b as depicted in



(a) (b)

The purpose of b is to ensure that $\mathcal{M}_G$ is serial.

In order to know whether G has a Hamiltonian path, we define $\varphi = \bigvee_{\sigma \text{ permutation of } \{1,\ldots n\}} \varphi_\sigma$ where $\varphi_\sigma = v_{\sigma_1} \wedge \exists \bigcirc (v_{\sigma_2} \wedge \exists \bigcirc (v_{\sigma_3} \wedge \ldots))$.

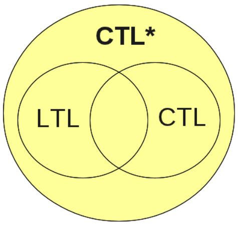
The length φ is exponential in the size of G.

The problem Hamiltonian path problem is defined as:

- input: a graph G;
- output: yes iff the graph G contains a Hamiltonian path.

is NP-complete.

Theorem 9 *If for all graph G we find a polynomial sized CTL-formula φ_G such that G is Hamiltonian iff there exists a world w of $\mathcal{M}_G$ such that $\mathcal{M}, w \models \varphi_G$... then $P = NP$.*



$\exists$ $\bigcirc$ U $\forall$ F G

Model-checking of CTL^*

PSPACE

Algorithm using the model-checking of LTL as a subroutine

input: $\mathcal{M} = (W, R, V)$, φ CTL^*-formula without $\forall$

output: the set of worlds s where $\mathcal{M}, s \models_\exists \varphi$.

function $\underline{\mathrm{mc}_\exists\mathsf{CTL}^*}(\mathcal{M}, \varphi)$

 if φ does not contain any $\exists$

  return $\underline{\mathrm{mc}_\exists\mathrm{LTL}}(\mathcal{M}, \varphi)$

 else

  $\psi := \exists\psi'$ a subformula of φ such that ψ' is $\exists$-free.

  $\mathcal{M}' := (W, R, V')$ where:

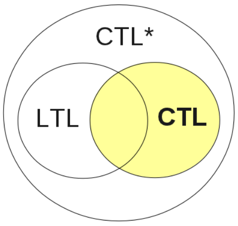
- $V' := V$ extended with $V'(p_\psi) = \underline{\mathrm{mc}_\exists\mathrm{LTL}}(\mathcal{M}, \psi')$ where p_ψ is a fresh atomic proposition

  $\psi' := \psi$ where we replaced subformulas ψ by p_ψ

  return $\underline{\mathrm{mc}_\exists\mathsf{CTL}^*}(\mathcal{M}', \varphi')$

 endIf

endFunction



$\exists\bigcirc$ $\forall\mathsf{F}$ $\exists\mathsf{U}$ $\forall\bigcirc$ $\exists\mathsf{G}$ $\forall\mathsf{F}$

Model-checking of CTL

$O((|W| + |A|)|\varphi|)$

input: $\mathcal{M} = (W, R, V)$, φ CTL-formula
output: the set of worlds s where $\mathcal{M}, s \models_{\exists} \varphi$.

function $\underline{mcCTL}(\mathcal{M}, \varphi)$
    match φ
        p: **return** $V(p)$
        $\neg\psi$: **return** $W \setminus \underline{mcCTL}(\mathcal{M}, \psi)$
        $\psi_1 \vee \psi_2$: **return** $\underline{mcCTL}(\mathcal{M}, \psi_1) \cup \underline{mcCTL}(\mathcal{M}, \psi_2)$
        $\exists \bigcirc \psi$: **return** $\{w \in W \mid \exists u \in R(w) \mid u \in \underline{mcCTL}(\mathcal{M}, \psi)\}$
        $\forall \bigcirc \psi$: **return** $\{w \in W \mid \forall u \in R(w) \mid u \in \underline{mcCTL}(\mathcal{M}, \psi)\}$
        $\exists\psi_1 U \psi_2$:
            $S_{\psi_1} := \underline{mcCTL}(\mathcal{M}, \psi_1)$
            $L_{totreat} := \underline{mcCTL}(\mathcal{M}, \psi_2)$
            $result := L_{treated} := []$
            while $L_{totreat} \neq \emptyset$
                $q := L_{totreat}.defiler$
                $result := result \cup \{q\}$
                for $u \rightarrow q$
                    if $u \notin L_{treated}$ **then**
                        $L_{treated} := L_{treated} \cup \{u\}$
                        if $u \in S_{\psi_1}$ **then**
                            $L_{totreat} := L_{totreat} \cup \{u\}$
                    endIf
                endFor
            endWhile
            return result
        $\forall\psi_1 U \psi_2$:
            $S_{\psi_1} := \underline{mcCTL}(\mathcal{M}, \psi_1)$
            for $w \in W$, $deg[w] :=$ number of successors of w
            $L_{totreat} := \underline{mcCTL}(\mathcal{M}, \psi_2)$
            $result := []$
            while $L_{totreat} \neq \emptyset$
                $q := L_{totreat}.defiler$
                $result := result \cup \{q\}$
                for $u \rightarrow q$
                    $deg[u] := deg[u] - 1$
                    if $deg[u] = 0$ and $u \notin result$ and $u \in S_{\psi_1}$ **then**
                        $L_{totreat} := L_{totreat} \cup \{u\}$
                    endIf
                endFor
            endWhile
            return result
    endMatch
endFunction

3.3 Expressivity

3.3.1 Comparison of LTL, CTL and CTL*

[p. 237] [p. 334, Def 6.17 remanié pour CTL*]

Definition 5 ()
Two CTL^* formulae Φ_1 and Φ_2 are equivalent iff for all $\mathcal{M}, s$ we have $\mathcal{M}, s \models_\forall \Phi_1$ iff $\mathcal{M}, s \models_\forall \Phi_2$.

Theorem 10 *Let Φ be a CTL* formula.*

- *Either Φ is equivalent to $\forall\varphi$, where φ is the LTL formula obtained by eliminating all path quantifiers in Φ;*
- *Or there is no LTL formula that is equivalent to Φ.*

PROOF.

[Clarke and Draghicescu 1985, Th. 1, P. 5 of the article] Suppose that Φ is equivalent to a formula $\forall\chi$ where χ is an LTL-formula. Let us prove that Φ is equivalent to $\forall\varphi$.

Let $\mathcal{M}, s$ be a pointed-model such that $\mathcal{M}, s \models_\forall \Phi$.

for all path π (that begins with s) we have $\mathcal{M}, \pi \models \chi$

for all path π of the form xy^w we have $\mathcal{M}, \pi \models \chi$

for all path π of the form xy^w we have $\mathcal{M}^\pi, s' \models_\forall \chi$ where $\mathcal{M}^\pi, s'$ is the model that contains only the path π

for all path π of the form xy^w we have $\mathcal{M}^\pi, s' \models_\forall \Phi$

for all path π of the form xy^w we have $\mathcal{M}^\pi, s' \models_\forall \varphi$ (because $\mathcal{M}^\pi$ is deterministic)

for all path π of the form xy^w we have $\mathcal{M}, \pi \models \varphi$

for all path π we have $\mathcal{M}, \pi \models \varphi$

$\mathcal{M}, s \models_\forall \forall\varphi$

■

Question 1 *What is the complexity of knowing if Φ has an LTL equivalent formula?*

Example 6 *$\forall G \forall F p$ is equivalent to GFp.[p. 335, and p. 326, remark 6.8]*

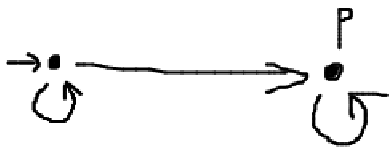
We prove that $\mathcal{M}, s \models \forall G \forall F p$ implies that in all paths π, p is true infinitely often. So $\mathcal{M}, s \models GFp$.

Conversely (yes, we have to prove it) $\mathcal{M}, s \models GFp$ implies that ...

[p. 424]

Theorem 11 *There exist CTL formulae for which no equivalent LTL formula exists. For instance, $\forall G \exists F p$ has no equivalent in LTL.*

PROOF.



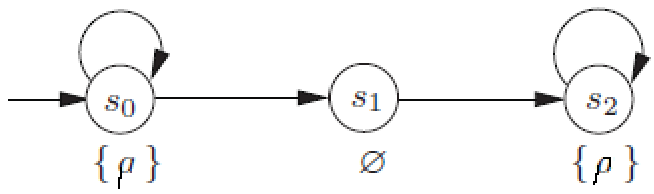
$\forall G \exists F p$ is not equivalent to GFp, see the model above:

- GFp: false in the path where we stay in the state of the left;
- $\forall G \exists F p$: true.

By the previous theorem, the CTL formula $\forall F \forall G p$ has no LTL equivalent.
■

Theorem 12 *Other example: $\forall F \forall G p$ has no equivalent in LTL.*

PROOF.



$\forall F \forall G p$ is not equivalent to FGp, see the model above [p. 335]:

- FGp: true in all paths;
- $\forall F \forall G p$: false because in the path $(s_0)^\omega$ we do not have $F \forall G p$. Indeed, at each point we can decide to change the path for $s_1 s_2^\omega$ and $s_1 \models \neg p$.

By the previous theorem, the CTL formula $\forall F \forall G p$ has no LTL equivalent.
■

TD : $\forall F(p \wedge \forall \bigcirc p)$ is not equivalent to $F(p \wedge \bigcirc p)$. [p. 336-337] $\forall G \exists F p$ is not equivalent to GFp

[p. 424]

Theorem 13 (p. 337. th. 6.21) *There exist LTL formulae for which no equivalent CTL formula exists. For instance, the LTL formula FGp has no equivalent in CTL.*

PROOF.
Let $\mathcal{M}_n, s$ be the following model:

Let $\mathcal{M}'_n, s$ be the following model:

For all $n \in \mathbb{N}$, we have $\mathcal{M}'_n, s \models_\forall FGp$.
For all $n \in \mathbb{N}$, we have $\mathcal{M}_n, s \not\models_\forall FGp$.

Lemma 3 *For all $n \in \mathbb{N}$, for all CTL formula φ such that $|\varphi| \leq n$, we have $\mathcal{M}_n \models \varphi$ iff $\mathcal{M}'_n \models \varphi$.*

PROOF.
TODO: ■
■
TD: or $F(p \wedge \bigcirc p)$
[p. 424, th. 6.84]

Theorem 14 *There exists a CTL* formula that is not expressible in CTL and also not expressible in LTL. For instance $\forall FGp \vee \forall G \exists Fp$.*

PROOF.

■

3.3.2 Comparison of CTL with K and S4

Proposition 1 *Let $\mathcal{M} = (W, R, V)$ be a Kripke model. Let tr be the following from K to CTL translation: $tr() = A \bigcirc tr(\varphi)$. Then we have $\mathcal{M}, w \models \varphi$ iff $\mathcal{M}, w \models_\exists tr(\varphi)$.*

Proposition 2 *Let $\mathcal{M} = (W, R, V)$ be a Kripke model. Let $\mathcal{M}* = (W, R*, V)$ where $R*$ is the reflexive and transitive closure of R. Let tr be the following from S4 to* CTL *translation: $tr() = AGtr(\varphi)$. Then we have $\mathcal{M}*, w \models \varphi$ iff $\mathcal{M}, w \models_{\exists} tr(\varphi)$.*

3.3.3 Bissimilation

Definition 6 ()
We say that $\mathcal{M}, \pi$ and $\mathcal{M}', \pi'$ are bissimilar (noted $\mathcal{M}, \pi \underline{\leftrightarrow} \mathcal{M}', \pi'$) iff for all $n \in \mathbb{N}$, $\mathcal{M}, \pi_n$ and $\mathcal{M}', \pi'_n$ are bissimilar.

Proposition 3 *If $\mathcal{M}, w \underline{\leftrightarrow} \mathcal{M}', w'$ and let π be a path in $\mathcal{M}$ such that $\pi_0 = w$. Then there exists a path π' in $\mathcal{M}'$ such that $\pi'_0 = w'$ such that $\mathcal{M}, \pi \underline{\leftrightarrow} \mathcal{M}', \pi'$.*

Proof.

■
[p. 473]

Proposition 4 *If $\mathcal{M}, \pi \underline{\leftrightarrow} \mathcal{M}', \pi'$ then for all* CTL^**-formula φ we have $\mathcal{M}, \pi \models \varphi$ iff $\mathcal{M}', \pi' \models \varphi$.*

Proof.
By induction on φ. ■

Theorem 15 *Let $\mathcal{M}, w$ and $\mathcal{M}', w'$ two image-finite models. We have equivalence between:*

1. *$\mathcal{M}, w \underline{\leftrightarrow} \mathcal{M}', w'$;*
2. *for all* CTL^**-formula, $\mathcal{M}, w \models_{\exists} \varphi$ iff $\mathcal{M}', w' \models_{\exists} \varphi$;*
3. *for all* CTL*-formula, $\mathcal{M}, w \models \varphi$ iff $\mathcal{M}', w' \models \varphi$.*

Proof.
$\boxed{1 \Rightarrow 2}$ Done.
$\boxed{2 \Rightarrow 3}$ Trivial.
$\boxed{3 \Rightarrow 1}$ Because CTL embeds logic K. Hence $\mathcal{M}, w$ and $\mathcal{M}, w'$ satisfies the same formulas of K. As they are image-finite models, they are bissimilar.

■