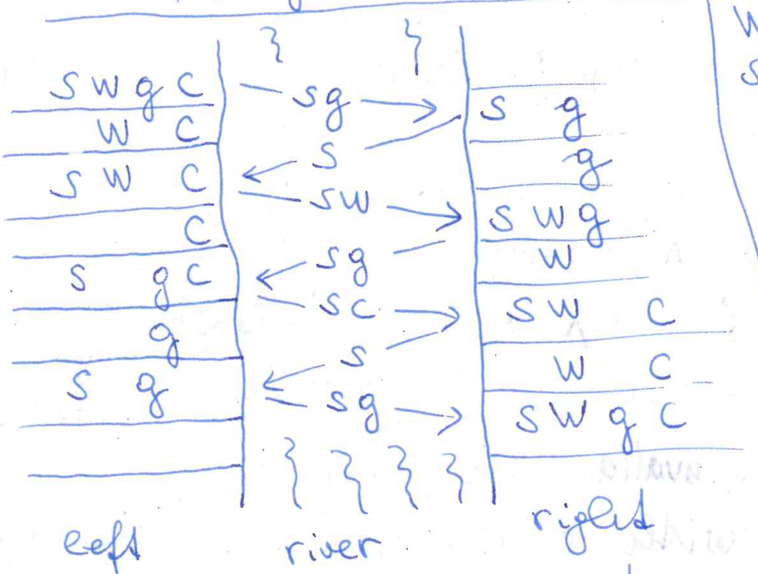
Lecture 7   05-02-2026

River Crossing

A shepherd wants to transport across a river a wolf, a goat, and a cabbage. **She** has only one boat with room for herself and another animal or item. The problem is that in the absence of the shepherd:

- the wolf would eat the goat, or
- the goat would eat the cabbage.

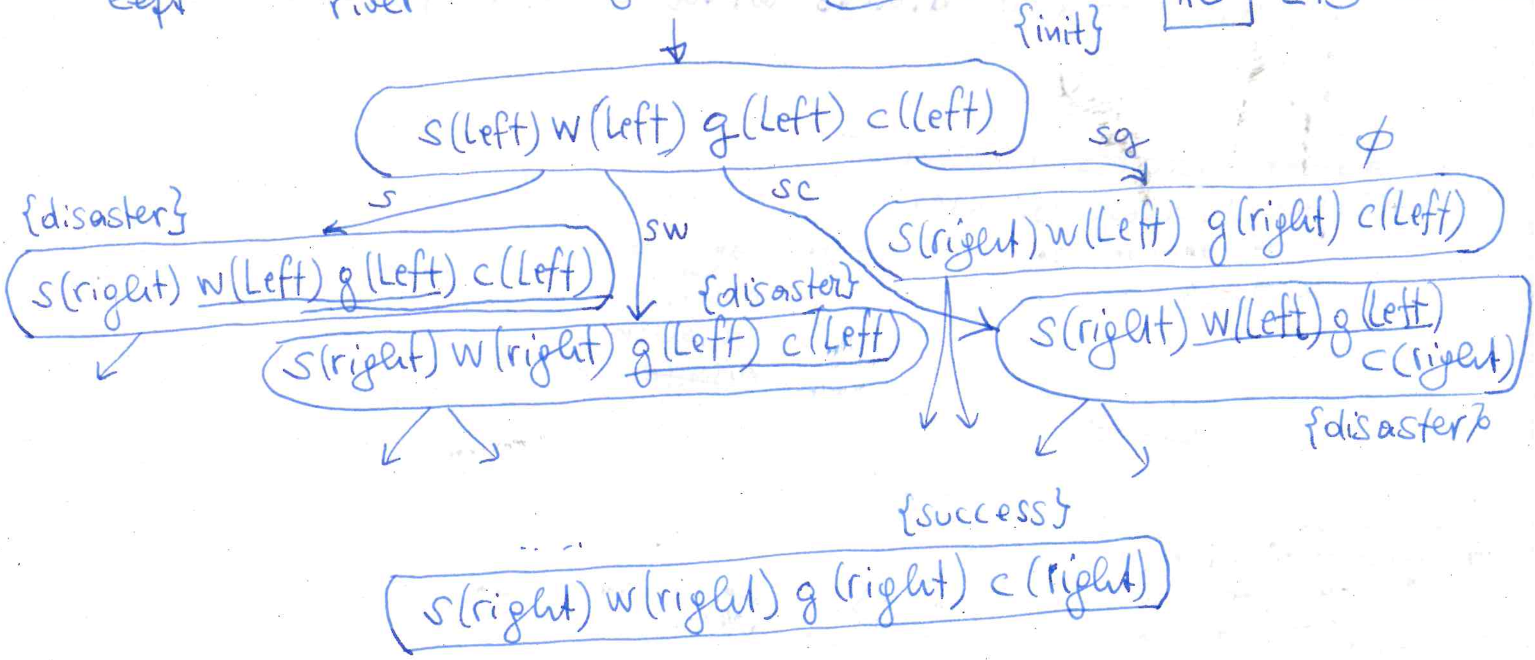


We search for a formula φ in LTL such that for all paths π in RC:

$$\pi \not\models \varphi \iff \pi \text{ successful river crossing}$$

In this situation, the model-checker will give us such a path π **as a counterexample** to $RC \models \varphi$, if indeed $RC \not\models \varphi$ holds

$\boxed{RC}$ LTS



$\pi \not\models \varphi \iff \pi \underbrace{}_{\text{starts with}}$ a successful river crossing

$A_0 \; A_1 \; A_2 \quad \ldots \quad A_{i-1} \; A_i$

$\{init\}$ $\qquad$ $\{success\}$ $\neg disaster$

$\underbrace{}$

disaster $\notin A_0, \ldots, A_{i-1}$

$\neg$success $\in A_0 \ldots A_{i-1}$

$\neg disaster \; U (success \wedge \neg disaster)$ $\equiv$ $(\neg disaster \wedge \neg success) \; W \; disaster$

~~$\neg(\neg disaster \; U (success \wedge \neg disaster))$~~

$\langle\rangle success$

~~$\langle\rangle disaster \wedge (\neg success \; U \; disaster)$~~

~~$\langle\rangle success \rightarrow (\langle\rangle disaster \wedge (\neg success \; U \; disaster))$~~

Then φ can be chosen as

$$\varphi := \begin{cases} \neg(\neg disaster \; U (success \wedge \neg disaster)) \\ \langle\rangle success \rightarrow (\langle\rangle disaster \wedge (\neg success \; U \; disaster)) \\ \equiv \langle\rangle success \rightarrow (\neg success \; U \; disaster) \quad \text{(Maude book)} \end{cases}$$

$\boxed{\begin{matrix}180 \\ 420\end{matrix}}$

because both formulas being invalid for a path π implies that π starts with a successful river crossing

---

~~$\neg(\neg disaster \; U (success \wedge \neg disaster))$~~

$\neg disaster \; U (success \wedge \neg disaster)$

$A_0 \; A_1 \; A_2 \; A_3 \; A_4 \; \ldots \; A_{n-1} \; A_n$

success

$\neg$disaster

~~disaster~~

$\underbrace{}_{\cancel{\text{disaster}}}$

$\langle\rangle success$

~~$\neg success \; U \; disaster$~~

$\boxed{\text{Case 1}}$

Succ Succ Succ .... Succ Succ success

$\underbrace{}_{\neg disaster}$

Case 2

Succ Succ Succ ...... Succ Succ Succ .... disaster

$$\neg(\neg disaster\ U\ (success \wedge \neg disaster))$$

$$\equiv (\neg disaster \wedge \neg(success \wedge \neg disaster))\ W\ (disaster \wedge \neg(success \wedge \neg disaster))$$

$\neg disaster \wedge (\neg success \vee disaster)$
$\equiv (\neg disaster \wedge \neg success) \vee (\neg disaster \wedge disaster)$
$\equiv \neg disaster \wedge \neg success$

$$\equiv (\neg disaster \wedge \neg success)\ W\ \underbrace{((disaster \wedge \neg success) \vee disaster)}_{\equiv\ disaster}$$

$$\equiv (\neg disaster \wedge \neg success)\ W\ disaster \quad \equiv \text{(A)}$$

$$\Diamond success \rightarrow \underbrace{\Diamond disaster \wedge (\neg success\ U\ disaster)}_{\equiv}$$

$$\Diamond success \longrightarrow \neg success\ U\ disaster \quad \equiv \text{(B)}$$

(A) $\equiv$ (B)

$\Rightarrow$: Suppose (A).

Case 1: $\sigma \models \Box \neg disaster \wedge (\neg disaster \wedge \neg success)$
$\equiv \Box \neg disaster \wedge \neg success$
$\Rightarrow \sigma \not\models \Diamond success$
$\Rightarrow \sigma \models$ (B)

Case 2: $\sigma \models (\neg disaster \wedge \neg success)\ U\ disaster.$
$\Rightarrow \sigma \models \neg success\ U\ disaster$
$\Rightarrow \sigma \models$ (B)

In both cases we conclude (B).
$\sigma \models A \Rightarrow \sigma \models B$
for all $\sigma \in (2^{AP})^\omega$

$\Leftarrow$: Suppose (B)

Case 1. $\sigma \not\models \Diamond success$
Then $\sigma \models \Box \neg success.$
It follows $(\neg disaster \wedge \neg success)\ W\ disaster. \equiv$ (A)

Case 2. $\sigma \models \Diamond success$
Then (B) implies that disaster must occur before success happens for the first time.
But then $\sigma \models \neg success\ W\ disaster$, and
hence $\sigma \models (\neg disaster \wedge \neg success)\ W\ disaster \equiv$ (A)

In both cases we conclude (B).

Complexity of LTL Model-Checking

$TS \models \varphi$ Instance: $TS = \langle S, Act, \rightarrow, I, AP, L \rangle$

$\varphi \in LTL(AP)$

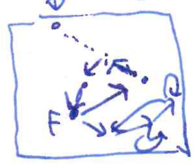
Question: Does $TS \models \varphi$ hold?

$TS \models \varphi \iff Traces(TS) \subseteq Words(\varphi)$

$\iff Traces(TS) \cap \underbrace{(2^{AP})^\omega \setminus Words(\varphi)}_{Words(\neg\varphi) = \mathcal{L}_\omega(\mathcal{A}_{\neg\varphi})} = \emptyset$

$\iff TS \otimes \mathcal{A}_{\neg\varphi} \models \Diamond\Box\neg F$

$TS \otimes \mathcal{A}_{\neg\varphi}$

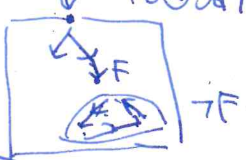


$TS \otimes \mathcal{A}_{\neg\varphi} \not\models \Diamond\Box\neg F$

$\Downarrow$

$TS \not\models \varphi$

$TS \otimes \mathcal{A}_{\neg\varphi}$



$TS \otimes \mathcal{A}_{\neg\varphi} \models \Box\Diamond\neg F$

$TS \models \varphi$

$\mathcal{A}_{\neg\varphi}$ may have size $O(2^{|\varphi|})$

$\varphi \mapsto \mathcal{A}_{\neg\varphi}$ takes exp. time $O(2^{|\varphi|} \cdot |\varphi|) = O(2^{|\varphi| + \log|\varphi|}) = O(2^{|\varphi|})$

$|TS \otimes \mathcal{A}_{\neg\varphi}| = O(|TS| \cdot \underbrace{|\mathcal{A}_{\neg\varphi}|}_{\in O(2^{|\varphi|})}) = O(|TS| \cdot 2^{|\varphi|})$

Checking $\Diamond\Box\neg F$ on $TS \otimes \mathcal{A}_{\neg\varphi}$ takes $\sim O(|TS \otimes \mathcal{A}_{\neg\varphi}|)$ time

$\Longrightarrow$ Overall time $\leq O(|TS| \cdot 2^{|\varphi|})$

Prop. LTL-MODEL-CHECKING is co-NP-hard

Proof. We reduce the Hamiltonian Path Problem to the complement of the LTL-model-checking problem.

HPP: Instance: $G = \langle V, E \rangle$ graph
Question: Does G have a Hamiltonian path?
(visiting every vertex precisely once)

$\overline{\text{LTL-MC}}$: Instance: φ LTL formula, TS Labeled trans. system
Complement of LTL-MC: Question: Does $TS \not\models \varphi$ hold?

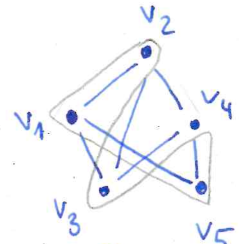
$\text{HPP} \leq_{\text{polyred}} \overline{\text{LTL-MC}} \implies \overline{\text{HPP}} \leq_{\text{polyred}} \text{LTL-MC}$.

TS_G

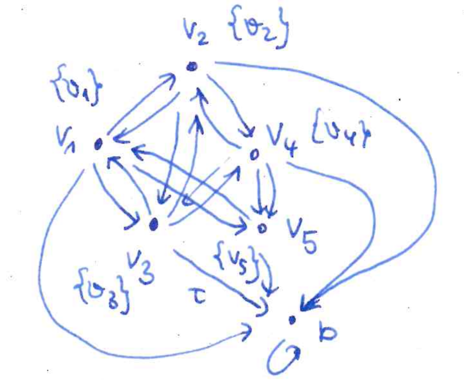
We define $f : G = \langle V, E \rangle \longmapsto \varphi_G$, such that

(i) G has a Hamiltonian path $\iff TS \not\models \varphi_G$

(ii) f is computable in polynomial time.



G
$\shortparallel$
$\langle V, E \rangle$



$TS_G = \langle V \uplus \{b\}, \{\tau\}, \rightarrow, V, V, L \rangle$

$$\frac{\langle v, w \rangle \in E}{v \xrightarrow{\tau} w} \qquad \frac{v \in V \cup \{b\}}{v \xrightarrow{\tau} b}$$

$$\varphi_G := \neg \bigwedge_{i=1}^{5} \left(\Diamond v_i \wedge \Box (v_i \rightarrow \bigcirc \Box \neg v_i) \right)$$

$$\pi \not\models \varphi_G \iff \pi \models \bigwedge_{i=1}^{5} \left(\Diamond v_i \wedge \Box (v_i \rightarrow \bigcirc \Box \neg v_i) \right)$$

$$\overset{\Rightarrow}{\Leftarrow} \quad \pi = v_{\sigma(1)} \dots v_{\sigma(5)}\, b\, b \dots$$

for some permutation σ on $\{1 \dots 5\}$

Aspect	Linear time	Branching time
"behaviour" in a state s	path-based: trace(s)	state-based computation tree of s
temporal logic	LTL: path formula φ $s \models \varphi \iff$ $\iff \forall \pi \in Paths(s) : \pi \models \varphi$	CTL: state formulae existential path quantification universal path quantification
Complexity of model checking problems	PSPACE-complete $O(\|TS\| \cdot exp(\|\varphi\|))$	PTIME $O(\|TS\| \cdot \|\Phi\|)$
adequate subsumption and equivalence relations	trace inclusion and trace equivalence (can be checked in PSPACE-complete)	bisimulation subsumption bisimulation equivalence (can be checked in polynomial time)
fairness	no special techniques needed	special techniques needed

CTL-formulas Φ and Ψ over AP are *equivalent* (denoted $\Phi \equiv \Psi$) if $Sat(\Phi) = Sat(\Psi)$ for all transition systems TS over AP.

Normal Forms

Existential Normal Form (ENF)

$$\Phi ::= true \mid a \mid \Phi \wedge \Phi \mid \neg \Phi \mid \exists \bigcirc \Phi \mid \exists(\Phi \, U \, \Phi) \mid \exists \Box \Phi \qquad (a \in AP)$$

Thm. For every CTL-formula there is an equivalent CTL-formula in ENF.

Positive Normal Form

$$\Phi ::= true \mid false \mid a \mid \neg a \mid \Phi_1 \wedge \Phi_2 \mid \Phi_1 \vee \Phi_2 \mid \exists \varphi \mid \forall \varphi$$

$$\varphi ::= \bigcirc \Phi \mid \Phi_1 \, U \, \Phi_2 \mid \Phi_1 \, W \, \Phi_2$$

W: weak until

Thm. For each CTL-formula there is an equivalent CTL-formula in PNF.

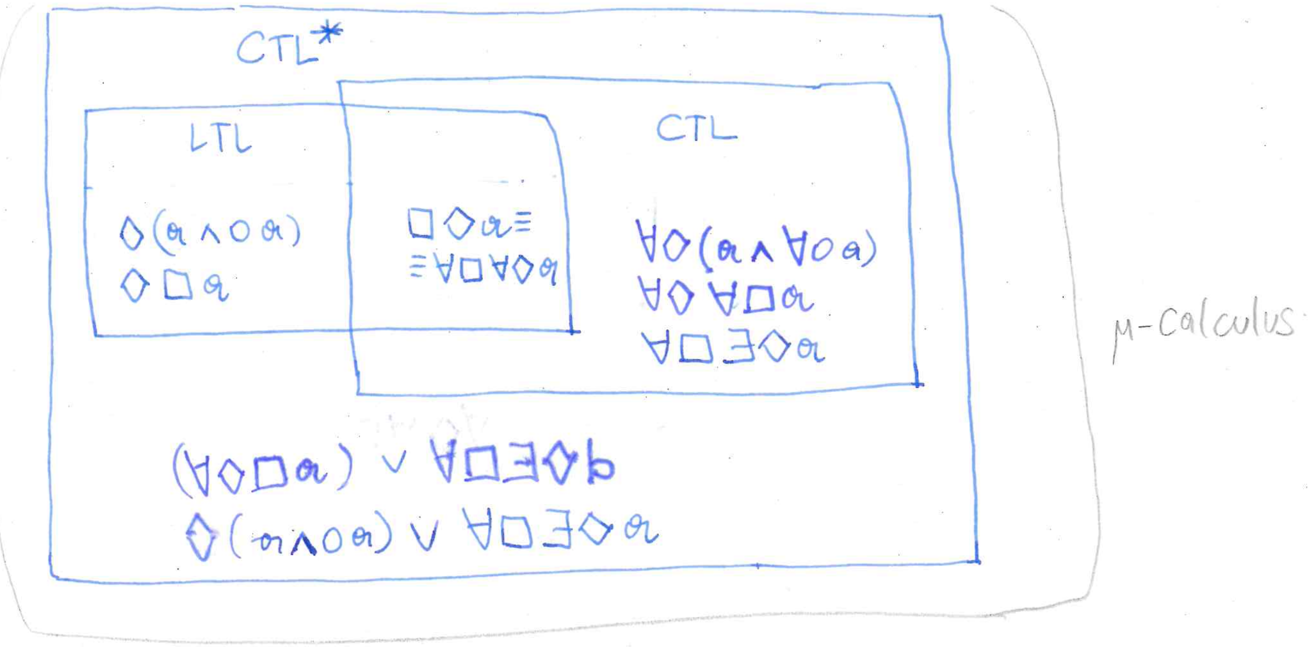
Weak Until:

intuitively not yet a definition:

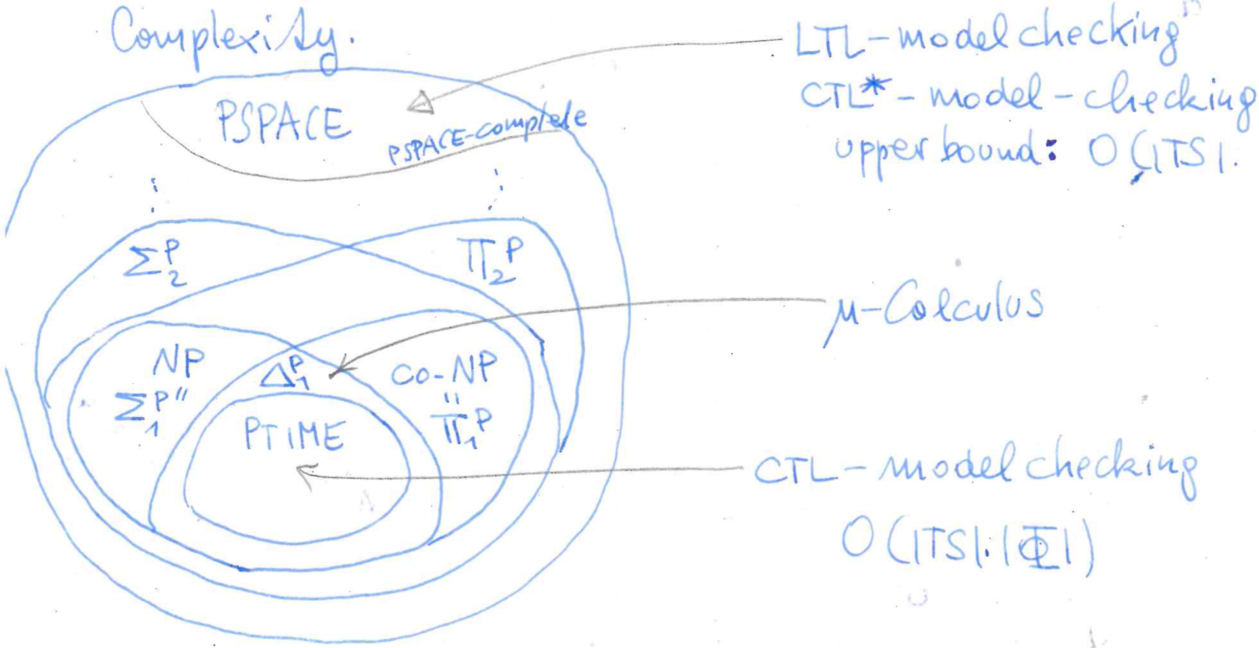
$$\pi \models \Phi W \Psi \iff \pi \models \Phi U \Psi \text{ or } \pi \models \Box(\Phi \wedge \neg \Psi)$$
$$\iff \pi \models \Phi U \Psi \text{ or } \pi \models \Box \Phi$$

Can be obtained by defining:

$$\exists(\Phi W \Psi) := \neg \forall((\Phi \wedge \neg \Psi) \, U \, (\neg \Phi \wedge \neg \Psi))$$
$$\forall(\Phi W \Psi) := \neg \exists((\Phi \wedge \neg \Psi) \, U \, (\neg \Phi \wedge \neg \Psi))$$



Complexity.



$\Sigma^P_{n+1} = \{\exists^P L \mid L \in \Pi^P_n\}$ $\qquad$ $\Pi^P_{n+1} = \{\forall^P L \mid L \in \Sigma^P_n\}$

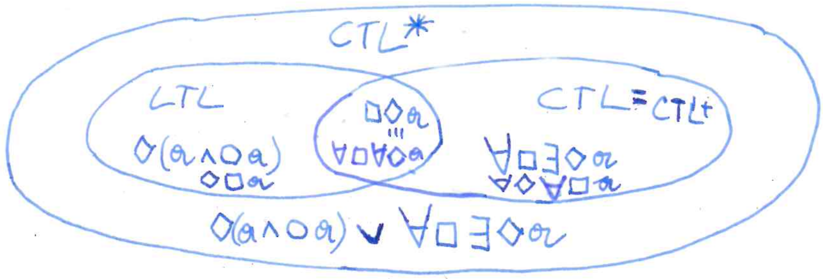
$\exists^p L = \{x \in \{0,1\}^* / \forall w \in \{0,1\}^{* \le p(|x|)} . \langle x, w\rangle \in L\}$ for $p \in Poly$

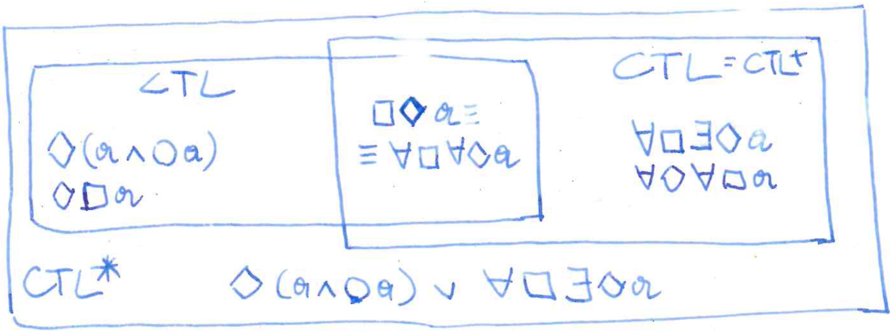
$\exists^P L = \{\exists^p L / p \in Poly\}$ for all $L \subseteq \{0,1\}^*$

$\Pi^p L = \{x \in \{0,1\}^* / \forall w \in \{0,1\}^{* \le p(|x|)} . \langle x, w\rangle \in L\}$ for all $p \in Poly$

$\Pi^P L = \{\forall^p L \mid p \in Poly\}$ for all $L \subseteq \{0,1\}^*$

Relationship between LTL, CTL, and CTL*





Thm. For the CTL*-formula $\Diamond(a \wedge \bigcirc a) \vee \forall\Box\exists\Diamond a$ there does not exist any equivalent LTL or CTL-formula.

	CTL	LTL	CTL*
model checking	PTIME	PSPACE-complete	PSPACE-Complete
without fairness	$size(TS) \cdot \lvert\Phi\rvert$	$size(TS) \cdot exp(\lvert\Phi\rvert)$	$size(TS) \cdot exp(\lvert\Phi\rvert)$
with fairness	$size(TS) \cdot \lvert\Phi\rvert \cdot \lvert fair\rvert$	$size(TS) \cdot exp(\lvert\Phi\rvert) \cdot \lvert fair\rvert$	$size(TS) \cdot exp(\lvert\Phi\rvert) \cdot \lvert fair\rvert$
for fixed specifications	$O(size(TS))$	$O(size(TS))$	$O(size(TS))$
Satisfiability check	EXPTIME	PSPACE-complete	2EXPTIME
best known technique upper bound	$O(exp(\lvert\Phi\rvert))$	$exp(\lvert\Phi\rvert)$	$exp(exp(\lvert\Phi\rvert))$

CTL Model Checking

CTL Model Checking Problem
Input: a transition system TS, and a CTL formula Φ
Question: does $TS \models \Phi$ hold?

TS is assumed to be finite, with no terminal states.

Recall: $Sat(\Phi) := \{ s \in S \mid s \models \Phi \}$ ... states of S in which Φ is satisfied.

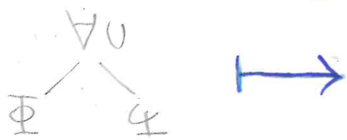
We will use CTL-formulas in ENF existential normal form

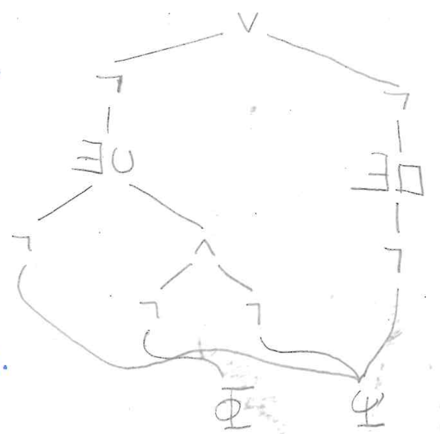
$\boxed{\text{CTL-ENF}}$ $\Phi ::= true \mid \underset{\in Act}{a} \mid \Phi_1 \wedge \Phi_2 \mid \neg\Phi \mid \exists \bigcirc \Phi \mid \exists(\Phi_1 \cup \Phi_2) \mid \exists \Box \Phi$

Recall:
every CTL-formula can be transformed into an (equivalent) CTL-ENF formula
(although with an exponential overhead)

e.g. $\forall(\Phi \cup \Psi) \longmapsto \neg \exists(\neg\Psi \cup (\neg\Phi \wedge \neg\Psi)) \vee \neg \exists \Box \neg \Psi$
(3 occurrences of Ψ)

This overhead could be avoided by using dag-representations of formulas.





The overhead is relevant for the **determination** of the complexity of the CTL-model checking algorithm.

A different approach to avoid the complexity increase due to this transformation, is to extend the CTL-model checking algorithm to deal also with formulas $\forall \bigcirc \Phi$, $\forall(\Phi \cup \Psi)$, and $\forall \Box \Phi$.

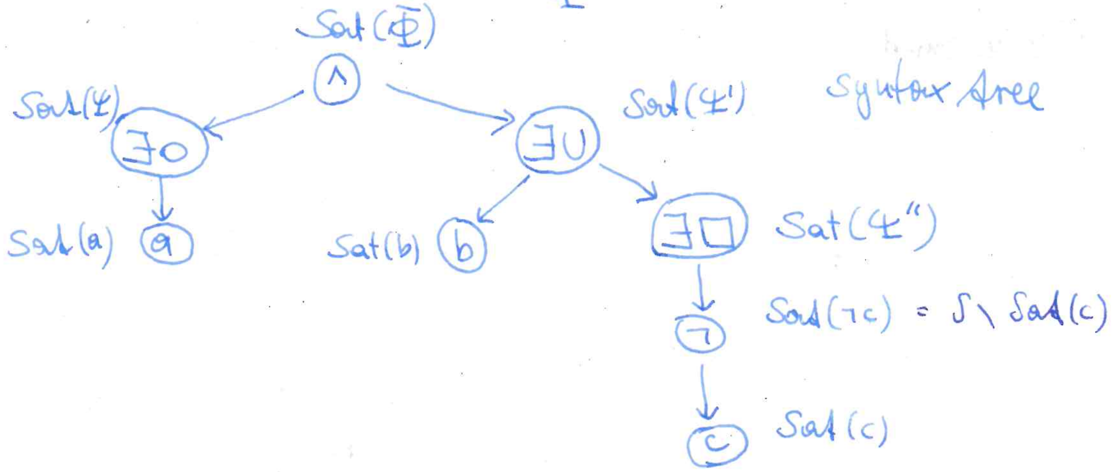
Basic idea of the model-checking algorithm for CTL:

(i) Compute $Sat(\Phi)$ by induction on subformulas of Φ

(ii) Then $TS \models \Phi \iff \underbrace{I}_{\text{initial states of } \Phi} \subseteq Sat(\Phi)$

Example: $AP = \{a, b, c\}$

$$\Phi = \underbrace{\exists \bigcirc a}_{\Psi} \wedge \underbrace{\exists (b \, U \, \underbrace{\exists \Box \neg c}_{\Psi''})}_{\Psi'}$$



we calculate the satisfaction sets $Sat(\tilde{\Psi})$ for all subformulas $\tilde{\Psi}$ of Φ by induction over the syntax tree

Characterization of Sat(·) for CTL formulae in ENF

Theorem. Let $TS = \langle S, Act, \rightarrow, I, AP, L \rangle$ be a transition system. For all CTL-formulae Φ, Ψ over AP:

(a) $Sat(true) = S$,

(b) $Sat(a) = \{s \in S \mid a \in L(s)\}$ for all $a \in Act$.

(c) $Sat(\Phi \wedge \Psi) = Sat(\Phi) \cap Sat(\Psi)$.

(d) $Sat(\neg \Phi) = S \setminus Sat(\Phi)$

(e) $Sat(\exists \bigcirc \Phi) = \{s \in S / Post(s) \cap Sat(\Phi) \neq \phi\}$

(f) $Sat(\exists \Phi \cup \Psi) =$ the smallest subset $T \subseteq S$ such that

(i) $Sat(\Psi) \subseteq T$, and (ii) $s \in Sat(\Phi)$ and $Post(s) \cap T \neq \phi \Rightarrow s \in T$

(g) $Sat(\exists \square \Phi) =$ the largest subset $T \subseteq S$ such that

(i) $T \subseteq Sat(\Phi)$, and (ii) $s \in T \Rightarrow Post(s) \cap T \neq \phi$.

Theorem. Time Complexity of CTL-model checking

For transition system TS with N states and K transitions, and CTL-formula Φ, the CTL-model checking problem $TS \models \Phi$ can be determined in time $O((N+K) \cdot |\Phi|)$.

Derived characterizations for CTL-formulas of the forms $\forall \bigcirc \Phi$, $\forall \Phi \cup \Psi$, $\forall \square \Phi$:

(h) $Sat(\forall \bigcirc \Phi) = \{s \in S / Post(s) \subseteq Sat(\Phi)\}$

(i) $Sat(\forall \Phi \cup \Psi)$ is the smallest set $T \subseteq S$ such that

$$Sat(\Psi) \cup \{s \in Sat(\Phi) / Post(s) \subseteq T\} \subseteq T$$

(j) $Sat(\forall \square \Phi)$ is the largest set $T \subseteq S$ such that

$$T \subseteq \{s \in Sat(\Phi) \mid Post(s) \subseteq T\}.$$

Alternative Formulation of Sat($\exists \Phi U \Psi$) and Sat($\exists \Box \Psi$)

$$\exists \Phi U \Psi \equiv \Psi \vee (\Phi \wedge \exists \bigcirc (\exists \Phi U \Psi)).$$

Thus $\exists \Phi U \Psi$ is a fixed point of:

$$F \equiv \Psi \vee (\Phi \wedge \exists \bigcirc (F)). \qquad (*)$$

But also $\exists (\Phi W \Psi)$ is a solution, but it is larger in the sense that $Sat(\exists \Box (\Phi W \Psi)) \supseteq Sat(\exists \Box (\Phi U \Psi))$.

However: $\exists (\Phi U \Psi)$ is the <u>least</u> solution of $(*)$:

(f)' $Sat(\exists(\Phi U \Psi))$ is the smallest set $T \subseteq S$ such that

$$Sat(\Psi) \cup \{s \in Sat(\Phi) \,/\, Post(s) \cap T \neq \emptyset\} \subseteq T.$$

with μ-Calculus notation:

$$\exists (\Phi U \Psi) \simeq \underbrace{\mu F. (\Psi \vee (\Phi \wedge \exists \bigcirc \Psi))}_{\mu\text{-Calculus notation.}}$$

Also:

$$\exists \Box \Phi \equiv \Phi \wedge \exists \bigcirc (\exists \Box \Phi))$$

Hence $\exists \Box \Phi$ is a fixed point of

$$F \equiv \Phi \wedge \exists \bigcirc F.$$

Indeed it is the Largest fixed point w.r.t. "measure" $Sat(\cdot)$.

(g)' $Sat(\exists \Box \Phi)$ is the Largest set $T \subseteq S$ such that

$$T \subseteq \{s \in Sat(\Phi) \,/\, Post(s) \cap T \neq \emptyset\}.$$

with μ-Calculus notation:

$$\exists \Box \Phi \simeq \underbrace{\nu F. (\Phi \wedge \exists \bigcirc \Phi)}_{\mu\text{-Calculus notation.}}$$