

Introduction to Model Checking

René Thiemann

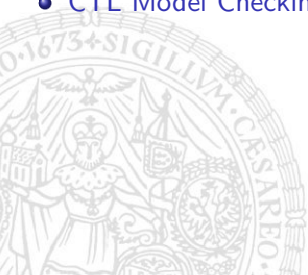
Institute of Computer Science
University of Innsbruck

WS 2007/2008



Outline

- Last Lecture
- CTL Model Checking



Computation Tree Logic

- Formulas over states (capital greek letters)

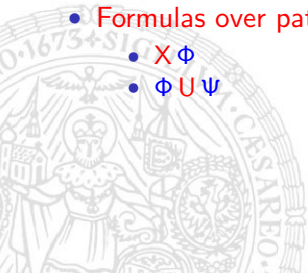
- true
- $a \in AP$
- $\neg \Phi$ and $\Phi \wedge \Psi$
- $E \varphi$
- $A \varphi$

atomic proposition
negation and conjunction
there exists a **path** fulfilling φ
all **paths** fulfill φ

- Formulas over paths (lower case greek letters)

- $X \Phi$
- $\Phi U \Psi$

the next **state** fulfills Φ
 Φ holds until a **Ψ -state** is reached



Semantics of CTL

A **state-formula** Φ holds in state s (written $s \models \Phi$) iff

$$s \models \text{true}$$

$$s \models a \quad \text{iff } a \in L(s)$$

$$s \models \neg \Phi \quad \text{iff } s \not\models \Phi$$

$$s \models \Phi \wedge \Psi \quad \text{iff } s \models \Phi \text{ and } s \models \Psi$$

$$s \models E \varphi \quad \text{iff } \pi \models \varphi \text{ for some path } \pi \text{ that starts in } s$$

$$s \models A \varphi \quad \text{iff } \pi \models \varphi \text{ for all paths } \pi \text{ that start in } s$$

A **path-formula** φ holds for path π (written $\pi \models \varphi$) iff

$$\pi \models X \Phi \quad \text{iff } \pi[1] \models \Phi$$

$$\pi \models \Phi U \Psi \quad \text{iff } (\exists j \geq 0. \pi[j] \models \Psi \text{ and } (\forall 0 \leq k < j. \pi[k] \models \Phi))$$

where $\pi[i]$ denotes the state s_i in the path $\pi = s_0 s_1 s_2 \dots$

Transition System Semantics

- For CTL-state-formula Φ , the **satisfaction set** $Sat(\Phi)$ is defined by:

$$Sat(\Phi) = \{s \in S \mid s \models \Phi\}$$

- TS satisfies CTL-state-formula Φ iff Φ holds in all its initial states:

$$I \subseteq Sat(\Phi)$$



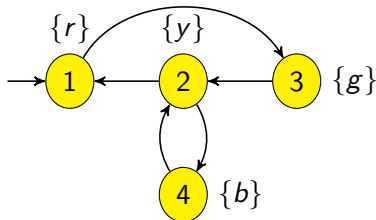
Exercise

Provide two transition systems TS_1, TS_2 (same atomic propositions, every state has at least one outgoing edge) and a CTL-formula Φ such that $Traces(TS_1) = Traces(TS_2)$ and $TS_1 \models \Phi$, but $TS_2 \not\models \Phi$.



Exercise

Consider the following transition system which models a traffic light which can blink.



Compute the set $Sat(\Phi)$ for the following formulas.

- AFy
- AGy
- $AGAFy$
- AFg
- EFg
- EGg
- $A(\neg bU b)$
- $E(\neg bU b)$
- $AGAFb$

Main Idea

- Semantics of CTL-state formula ϕ is defined inductively on sub-formulas ($Sat(\phi)$ is set of satisfying states)
- \Rightarrow compute $Sat(\psi)$ inductively for all sub-state-formulas ψ of ϕ
- \Rightarrow afterwards model checking can be done by checking $I \subseteq Sat(\phi)$

CTL Model checking boils down to simple set operations
no Büchi automata required



CTL Model Checking: Everything but Until

Let $TS = (S, \rightarrow, I, AP, L)$

- $s \models \text{true}$

$$\Rightarrow \text{Sat}(\text{true}) = S$$

- $s \models a$ iff $a \in L(s)$

$$\Rightarrow \text{Sat}(a) = \{s \mid a \in L(s)\}$$

- $s \models \neg \Phi$ iff $s \not\models \Phi$

$$\Rightarrow \text{Sat}(\neg \Phi) = S \setminus \text{Sat}(\Phi)$$

- $s \models \Phi \wedge \Psi$ iff $s \models \Phi$ and $s \models \Psi$

$$\Rightarrow \text{Sat}(\Phi \wedge \Psi) = \text{Sat}(\Phi) \cap \text{Sat}(\Psi)$$

- $s \models \text{EX } \Phi$ iff there is a path $s \ s' \ s'' \dots$ such that $s' \models \Phi$

$$\Rightarrow \text{Sat}(\text{EX } \Phi) = \{s \mid \exists s' : s \rightarrow s', s' \in \text{Sat}(\Phi)\}$$

- $s \models \text{AX } \Phi$ iff for all paths $s \ s' \ s'' \dots$ it holds: $s' \models \Phi$

$$\Rightarrow \text{Sat}(\text{AX } \Phi) = \{s \mid \forall s' : s \rightarrow s' \Rightarrow s' \in \text{Sat}(\Phi)\}$$

CTL Model Checking: Until

Remember LTL equivalence:

$$\Phi U \Psi \equiv \Psi \vee (\Phi \wedge X(\Phi U \Psi))$$

In CTL:

$$E\Phi U \Psi \equiv \Psi \vee (\Phi \wedge EX(E\Phi U \Psi))$$

Hence:

$$\begin{aligned} \text{Sat}(E\Phi U \Psi) &\equiv \text{Sat}(\Psi) \cup (\text{Sat}(\Phi) \cap \text{Sat}(EX(E\Phi U \Psi))) \\ &\equiv \text{Sat}(\Psi) \cup (\text{Sat}(\Phi) \cap \{s \mid \exists s' : s \rightarrow s', s' \in \text{Sat}(E\Phi U \Psi)\}) \end{aligned}$$

One can prove that $\text{Sat}(E\Phi U \Psi)$ is **least** set T satisfying

$$T = \text{Sat}(\Psi) \cup (\text{Sat}(\Phi) \cap \{s \mid \exists s' : s \rightarrow s', s' \in T\})$$

CTL Model Checking: Until (2)

$Sat(E \Phi U \Psi)$ is least set T satisfying

$$T = Sat(\Psi) \cup (Sat(\Phi) \cap \{s \mid \exists s' : s \rightarrow s', s' \in T\})$$

Corresponding operation $o : 2^S \rightarrow 2^S$:

$$o(T) = Sat(\Psi) \cup (Sat(\Phi) \cap \{s \mid \exists s' : s \rightarrow s', s' \in T\})$$



Computation of $Sat(E \Phi U \Psi)$

Define

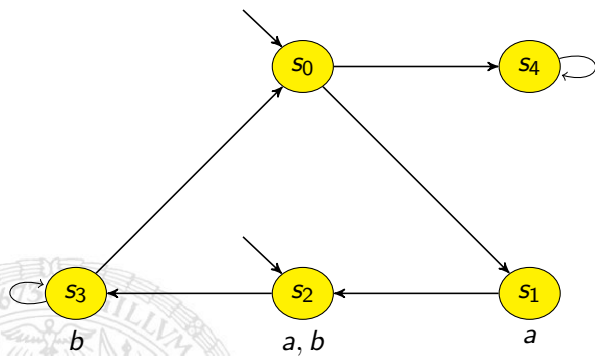
$$o(T) = Sat(\Psi) \cup (Sat(\Phi) \cap \{s \mid \exists s' : s \rightarrow s', s' \in T\})$$

Computation of $Sat(A \Phi U \Psi)$

just replace o by o' :

$$o'(T) = Sat(\Psi) \cup (Sat(\Phi) \cap \{s \mid \forall s' : s \rightarrow s' \Rightarrow s' \in T\})$$

Example



Complexity of CTL Model Checking

Model-Checking $TS = (S, \rightarrow, I, AP, L) \models \Phi$:

- Number of sets $Sat(\Psi)$ to be computed: $|\Phi|$
 - Complexity of computing a single set $Sat(\Psi)$:
 $\mathcal{O}(|S| + |\rightarrow|) \leq \mathcal{O}(|TS|)$
 - Complexity of checking $I \subseteq Sat(\Psi)$: $\mathcal{O}(|I|) \leq \mathcal{O}(|TS|)$
- \Rightarrow overall complexity: $\mathcal{O}(|\Phi| \cdot |TS|)$



Exercises

- Consider the transition system on Slide 15 where $L(s_4)$ is changed to $\{b\}$. Does the modified model satisfy the formula Φ ?

$$\Phi = A(a \cup b) \vee EX(AGb)$$

Use the algorithm of this lecture with an additional rule for disjunction.

$$Sat(\Psi \vee \Psi') = Sat(\Psi) \cup Sat(\Psi')$$

- Provide a direct fixpoint characterization of $Sat(EG\Phi)$ without using the following equivalence.

$$EG\Phi \equiv \neg AF\neg\Phi \equiv \neg A \text{ true} \cup \neg\Phi$$

