Linear Temporal Logic

Lecture #12 of Model Checking

Joost-Pieter Katoen

Lehrstuhl 2: Software Modeling & Verification

E-mail: katoen@cs.rwth-aachen.de

May 15, 2007

Overview Lecture #12

⇒ Summary of regular properties

- Syntax
- Semantics

Summary of regular properties (1)

- Languages recognized by NFA/DFA = regular languages
  - serve to represent the bad prefixes of regular safety properties
- Checking a regular safety property = invariant checking on a product
  - "never visit an accept state" in the NFA for the bad prefixes
  - amounts to solving a (DFS) reachability problem
- ω-regular languages are languages of infinite words
  - can be described by ω-regular expressions
- Languages recognized by NBA = ω-regular languages
  - serve to represent ω-regular properties

Summary of regular properties (2)

- DBA are less powerful than NBA
  - fail, e.g., to represent the persistence property ”eventually for ever a”
- Generalized NBA require repeated visits for several acceptance sets
  - the languages recognized by GNBA = ω-regular languages
- Checking an ω-regular property = checking persistency on a product
  - “eventually for ever no accept state” in the NBA for the complement property
- Persistence checking is solvable in linear time by a nested DFS
- Nested DFS = a DFS for reachable $\neg\Phi$-states + a DFS for cycle detection

Syntax

modal logic over infinite sequences [Pnueli 1977]

- Propositional logic
  - $a \in AP$ — atomic proposition
  - $\neg\phi$ and $\phi \wedge \psi$ — negation and conjunction
- Temporal operators
  - $\bigcirc \phi$ — neXt state fulfills ϕ
  - ϕ U ψ — ϕ holds Until a ψ-state is reached

linear temporal logic is a logic for describing LT properties

Derived operators

$$\phi \vee \psi \equiv \neg(\neg\phi \wedge \neg\psi)$$

$$\phi \Rightarrow \psi \equiv \neg\phi \vee \psi$$

$$\phi \Leftrightarrow \psi \equiv (\phi \Rightarrow \psi) \wedge (\psi \Rightarrow \phi)$$

$$\phi \oplus \psi \equiv (\phi \wedge \neg\psi) \vee (\neg\phi \wedge \psi)$$

$$\text{true} \equiv \phi \vee \neg\phi$$

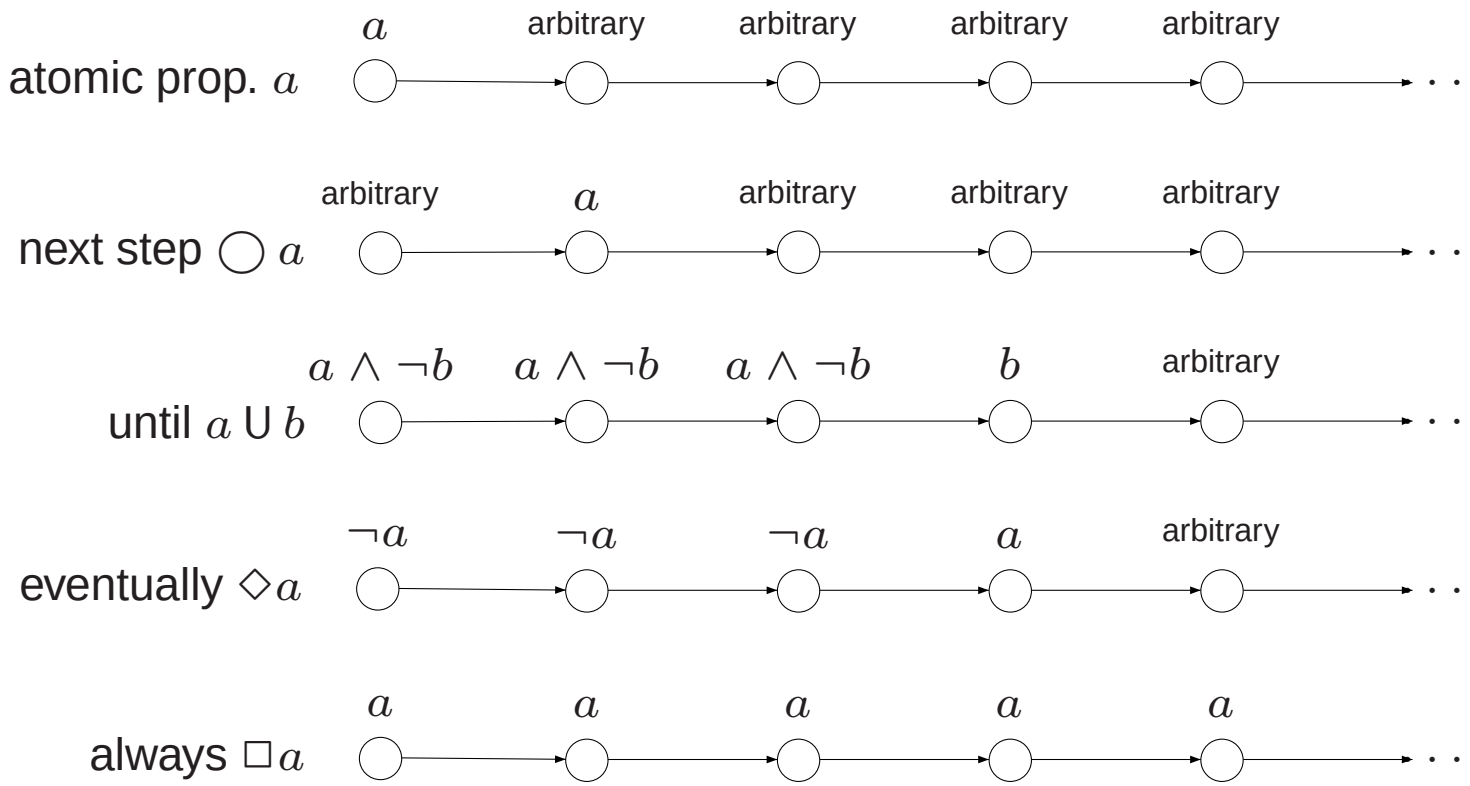
$$\text{false} \equiv \neg\,\text{true}$$

$$\Diamond\phi \equiv \text{true}\,\mathsf{U}\,\phi \qquad \text{“sometimes in the future”}$$

$$\Box\phi \equiv \neg\Diamond\neg\phi \qquad \text{“from now on for ever”}$$

precedence order: the unary operators bind stronger than the binary ones.
$\neg$ and $\bigcirc$ bind equally strong. U takes precedence over $\wedge$, $\vee$, and $\rightarrow$

Intuitive semantics



Traffic light properties

- Once red, the light cannot become green immediately:

$$\Box\,(red \;\Rightarrow\; \neg \bigcirc\; green)$$

- The green light becomes green eventually: $\Diamond\, green$
- Once red, the light becomes green eventually: $\Box\,(red \;\Rightarrow\; \Diamond\, green)$
- Once red, the light always becomes green eventually after being yellow for some time inbetween:

$$\Box(red \rightarrow \bigcirc\,(red\,\mathsf{U}\,(yellow \;\wedge\; \bigcirc\;(yellow\,\mathsf{U}\,green))))$$

Practical properties in LTL

- Reachability
  - negated reachability $\Diamond \neg\psi$
  - conditional reachability $\phi \,\mathsf{U}\, \psi$
  - reachability from any state not expressible
- Safety
  - simple safety $\Box \neg\phi$
  - conditional safety $(\phi \,\mathsf{U}\, \psi) \vee \Diamond \phi$
- Liveness $\Box (\phi \Rightarrow \Diamond \psi)$ and others
- Fairness $\Box \Diamond \phi$ and others

Semantics over words

The LT-property induced by LTL formula φ over *AP* is:

$\mathit{Words}(\varphi) = \left\{\sigma \in \left(2^{AP}\right)^{\omega} \mid \sigma \models \varphi\right\}$, where $\models$ is the smallest relation satisfying:

$$
\begin{array}{lcll}
\sigma & \models & \text{true} & \\
\sigma & \models & a & \text{iff} \quad a \in A_0 \quad (\text{i.e., } A_0 \models a) \\
\sigma & \models & \varphi_1 \wedge \varphi_2 & \text{iff} \quad \sigma \models \varphi_1 \text{ and } \sigma \models \varphi_2 \\
\sigma & \models & \neg\, \varphi & \text{iff} \quad \sigma \not\models \varphi \\
\sigma & \models & \bigcirc \varphi & \text{iff} \quad \sigma[1..] = A_1 A_2 A_3 \ldots \models \varphi \\
\sigma & \models & \varphi_1 \,\mathsf{U}\, \varphi_2 & \text{iff} \quad \exists j \geqslant 0.\ \sigma[j..] \models \varphi_2 \text{ and } \sigma[i..] \models \varphi_1,\ 0 \leqslant i < j
\end{array}
$$

for $\sigma = A_0 A_1 A_2 \ldots$ we have $\sigma[i..] = A_i A_{i+1} A_{i+2} \ldots$ is the suffix of σ from index i on

Semantics of □, ◇, □◇ and ◇□

Semantics over paths and states

Let $TS = (S, Act, \rightarrow, I, AP, L)$ be a transition system without terminal states, and let φ be an LTL-formula over AP.

- For infinite path fragment π of TS:

$$\pi \models \varphi \qquad \text{iff} \qquad trace(\pi) \models \varphi$$

- For state $s \in S$:

$$s \models \varphi \qquad \text{iff} \qquad (\forall \pi \in Paths(s).\ \pi \models \varphi)$$

- TS satisfies φ, denoted $TS \models \varphi$, if $Traces(TS) \subseteq Words(\varphi)$

Semantics for transition systems

$$TS \models \varphi$$

iff (* transition system semantics *)

$$\mathit{Traces}(TS) \subseteq \mathit{Words}(\varphi)$$

iff (* definition of $\models$ for LT-properties *)

$$TS \models \mathit{Words}(\varphi)$$

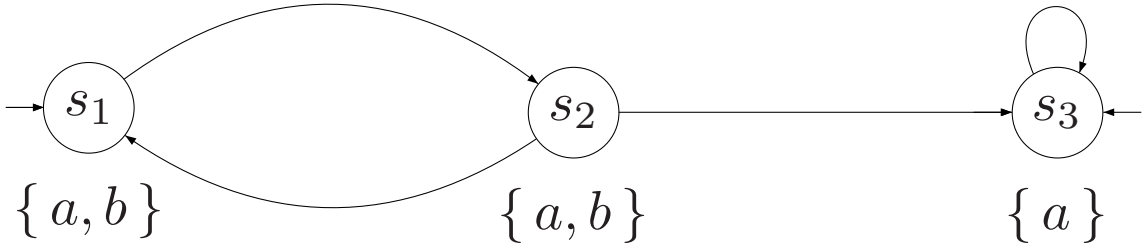
iff (* Definition of $\mathit{Words}(\varphi)$ *)

$$\pi \models \varphi \text{ for all } \pi \in \mathit{Paths}(TS)$$

iff (* semantics of $\models$ for states *)

$$s_0 \models \varphi \text{ for all } s_0 \in I \quad .$$

Example



Semantics of negation

For paths, it holds $\pi \models \varphi$ if and only if $\pi \not\models \neg\varphi$ since:

$$\mathit{Words}(\neg\varphi) = \left(2^{AP}\right)^{\omega} \setminus \mathit{Words}(\varphi) \quad .$$

But: $\mathit{TS} \not\models \varphi$ and $\mathit{TS} \models \neg\varphi$ are *not* equivalent in general

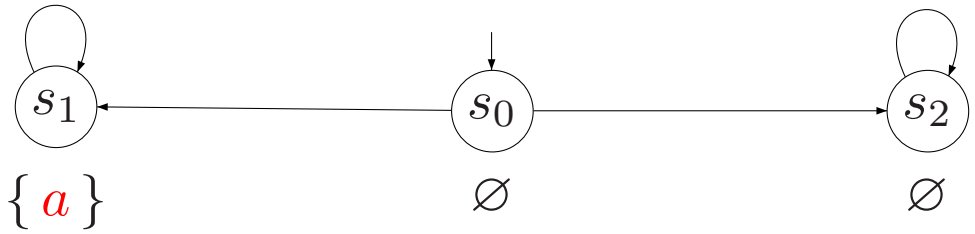
It holds: $\mathit{TS} \models \neg\varphi$ implies $\mathit{TS} \not\models \varphi$. Not always the reverse!

Note that:

$$\begin{aligned} \mathit{TS} \not\models \varphi \quad & \text{iff } \mathit{Traces}(\mathit{TS}) \not\subseteq \mathit{Words}(\varphi) \\ & \text{iff } \mathit{Traces}(\mathit{TS}) \setminus \mathit{Words}(\varphi) \neq \varnothing \\ & \text{iff } \mathit{Traces}(\mathit{TS}) \cap \mathit{Words}(\neg\varphi) \neq \varnothing \quad . \end{aligned}$$

TS neither satisfies φ nor $\neg\varphi$ if there are
paths π_1 and π_2 in TS such that $\pi_1 \models \varphi$ and $\pi_2 \models \neg\varphi$

Example



A transition system for which *TS* $\not\models \Diamond a$ and *TS* $\not\models \neg\Diamond a$

A communication channel