# Timed CTL Model Checking

## Lecture #16 of Advanced Model Checking

*Joost-Pieter Katoen*

Lehrstuhl 2: Software Modeling & Verification

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

July 1 , 2009

# Timed CTL

Syntax of TCTL *state-formulas* over *AP* and set $C$:

$$\Phi ::= \text{true} \;\Big|\; a \;\Big|\; g \;\Big|\; \Phi \wedge \Phi \;\Big|\; \neg\Phi \;\Big|\; \exists\varphi \;\Big|\; \forall\varphi$$

where $a \in AP$, $g \in ACC(C)$ and $\varphi$ is a *path-formula* defined by:

$$\varphi ::= \Phi \, \mathsf{U}^{J} \, \Phi$$

where $J \subseteq \mathbb{R}_{\geqslant 0}$ is an interval whose bounds are naturals

abbreviate $[c, \infty)$ by $x > c$, $(c_1, c_2]$ by $c_1 < x \leqslant c_2$ etc.

# TCTL-semantics for timed automata

- Let *TA* be a timed automaton with clocks $C$ and locations *Loc*
- For TCTL-state-formula $\Phi$, the *satisfaction set* *Sat*$(\Phi)$ is defined by:

$$\mathit{Sat}(\Phi) \;=\; \{\, s \in \mathit{Loc} \times \mathit{Eval}(C) \mid s \models \Phi \,\}$$

- *TA* satisfies TCTL-formula $\Phi$ iff $\Phi$ holds in all initial states of *TA*:

$$\mathit{TA} \models \Phi \quad \text{if and only if} \quad \forall \ell_0 \in \mathit{Loc}_0.\, \langle \ell_0, \eta_0 \rangle \models \Phi$$

where $\eta_0(x) = 0$ for all $x \in C$

# TCTL model checking

- TCTL model-checking problem: $\mathit{TA} \models \Phi$ for non-Zeno $\mathit{TA}$

$$\underbrace{\mathit{TA} \models \Phi}_{\text{timed automaton}} \quad \text{iff} \quad \underbrace{\mathit{TS}(\mathit{TA}) \models \Phi}_{\text{infinite transition system}}$$

- Idea: consider a finite quotient of $\mathit{TS}(\mathit{TA})$ wrt. a bisimulation
  - $\mathit{TS}(\mathit{TA}) / \cong$ is a *region* transition system and denoted $\mathit{RTS}(\mathit{TA})$
  - dependence on $\Phi$ is ignored for the moment . . .

- Transform TCTL formula $\Phi$ into an "equivalent" CTL-formula $\widehat{\Phi}$

- Then: $\mathit{TA} \models_{\text{TCTL}} \Phi \quad \text{iff} \quad \underbrace{\mathit{RTS}(\mathit{TA})}_{\text{finite transition system}} \models_{\text{CTL}} \widehat{\Phi}$

# Eliminating timing parameters

- Eliminate all intervals $J \neq [0, \infty)$ from TCTL formulas
  - introduce a fresh clock, $z$ say, that does not occur in *TA*
- Formally: for any state $s$ of $TS(TA)$ it holds:

$$s \models \exists \Diamond^{J} \Phi \quad \text{iff} \quad \underbrace{s\{z := 0\}}_{\text{state in } TS(TA \oplus z)} \models \exists \Diamond \big( (z \in J) \wedge \Phi \big)$$

  - where $TA \oplus z$ is *TA* (over $C$) extended with $z \notin C$

atomic clock constraints are atomic propositions, i.e., a CTL formula results

# Correctness

Let $\mathit{TA} = (\mathit{Loc}, \mathit{Act}, C, \hookrightarrow, \mathit{Loc}_0, \mathit{Inv}, \mathit{AP}, L)$. For clock $z \notin C$, let

$$\mathit{TA} \oplus z \;=\; (\mathit{Loc}, \mathit{Act}, C \cup \{\, z \,\}, \hookrightarrow, \mathit{Loc}_0, \mathit{Inv}, \mathit{AP}, L).$$

For any state $s$ of $\mathit{TS}(\mathit{TA})$ it holds that:

1. $s \models \exists(\Phi \,\mathsf{U}^J \Psi)$ iff $\underbrace{s\{z := 0\}}_{\text{state in } \mathit{TS}(\mathit{TA} \oplus z)} \models \exists\Big((\Phi \vee \Psi) \,\mathsf{U}\, \big((z \in J) \wedge \Psi\big)\Big)$

2. $s \models \forall(\Phi \,\mathsf{U}^J \Psi)$ iff $\underbrace{s\{z := 0\}}_{\text{state in } \mathit{TS}(\mathit{TA} \oplus z)} \models \forall\Big((\Phi \vee \Psi) \,\mathsf{U}\, \big((z \in J) \wedge \Psi\big)\Big)$

# Clock equivalence $\cong$

(A) Equivalent clock valuations satisfy the same clock constraints $g$:

$$\eta \cong \eta' \;\Rightarrow\; (\eta \models g \quad \text{iff} \quad \eta' \models g)$$

(B) Time-divergent paths of equivalent states are "equivalent"

- this property guarantees that equivalent states satisfy the same path formulas

(C) The number of equivalence classes under $\cong$ is finite

# Clock equivalence

- Correctness criteria (A) and (B) are ensured if equivalent states:
  - agree on the integer parts of all clock values, and
  - agree on the ordering of the fractional parts of all clocks

$\Rightarrow$ This yields a denumerable infinite set of equivalence classes

- Observe that:
  - if clocks exceed the maximal constant with which they are compared their precise value is not of interest

$\Rightarrow$ The number of equivalence classes is then finite (C)

# Clock equivalence

Clock valuations $\eta, \eta' \in \mathit{Eval}(C)$ are *equivalent*, denoted $\eta \cong \eta'$, if:

(1) for any $x \in C$: $\eta(x) > c_x$ iff $\eta'(x) > c_x$

(2) for any $x \in C$: if $\eta(x), \eta'(x) \leqslant c_x$ then:

$$\lfloor \eta(x) \rfloor = \lfloor \eta'(x) \rfloor \quad \text{and} \quad \mathit{frac}(\eta(x)) = 0 \text{ iff } \mathit{frac}(\eta'(x)) = 0$$

(3) for any $x, y \in C$: if $\eta(x), \eta'(x) \leqslant c_x$ and $\eta(y), \eta'(y) \leqslant c_y$, then:

$$\mathit{frac}(\eta(x)) \leqslant \mathit{frac}(\eta(y)) \quad \text{iff} \quad \mathit{frac}(\eta'(x)) \leqslant \mathit{frac}(\eta'(y))$$

$c_x$ is the largest constant with which $x$ is compared

# Regions

- The *clock region* of $\eta \in \mathit{Eval}(C)$, denoted $[\eta]$, is defined by:

$$[\eta] \;=\; \{\, \eta' \in \mathit{Eval}(C) \mid \eta \cong \eta' \,\}$$

- The *state region* of $s = \langle \ell, \eta \rangle \in \mathit{TS}(\mathit{TA})$ is defined by:

$$[s] \;=\; \langle \ell, [\eta] \rangle \;=\; \{\, \langle s, \eta' \rangle \mid \eta' \in [\eta] \,\}$$

# Example $c_x{=}2$, $c_y{=}1$

# Bounds on the number of regions

The *number of clock regions* is bounded from below and above by:

$$|C|! * \prod_{x \in C} c_x \;\leqslant\; \Big|\underbrace{\textit{Eval}(C)/\cong}_{\text{number of regions}}\Big| \;\leqslant\; |C|! * 2^{|C|-1} * \prod_{x \in C} (2c_x + 2)$$

where for the upper bound it is assumed that $c_x \geqslant 1$ for any $x \in C$

the number of state regions is $|\textit{Loc}|$ times larger

# Proof

# Preservation of atomic properties

1. For $\eta, \eta' \in \mathit{Eval}(C)$ such that $\eta \cong \eta'$:

$$\eta \models g \quad \text{if and only if} \quad \eta' \models g \text{ for any } g \in \mathit{ACC}(\mathit{TA} \cup \Phi)$$

2. For $s, s' \in \mathit{TS}(\mathit{TA})$ such that $s \cong s'$:

$$s \models a \quad \text{if and only if} \quad s' \models a \text{ for any } a \in \mathit{AP}'$$

where $\mathit{AP}'$ includes all propositions in *TA* and atomic clock constraints in *TA* and $\Phi$

# Clock equivalence is a bisimulation

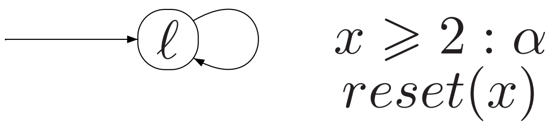
Clock equivalence is a bisimulation equivalence over $AP'$

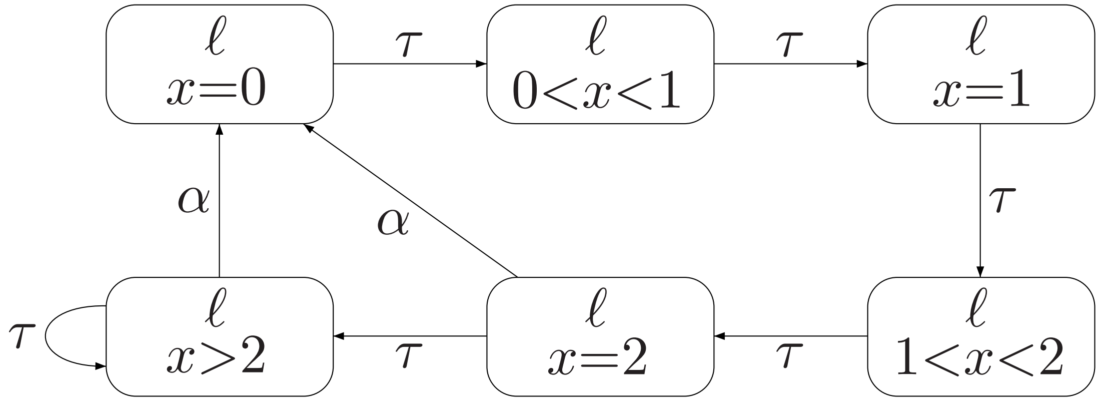
# Proof

# Region automaton: intuition

- Region automaton = quotient of $\mathit{TS}(\mathit{TA})$ under $\cong$
- State regions are states in quotient transition system under $\cong$
- Transitions in region automaton "mimic" those in $\mathit{TS}(\mathit{TA})$
- Delays are abstract
  - the exact delay is not recorded, only that some delay took place
  - if any clock $x$ exceeds $c_x$, delays are self-loops
- Discrete transitions correspond to actions

# A simple example



# Unbounded and successor regions

- Clock region $r_\infty = \{\, \eta \in \mathit{Eval}(C) \mid \forall x \in C.\, \eta(x) > c_x \,\}$ is *unbounded*
- $r'$ is the *successor (clock) region* of $r$, denoted $r' = \mathit{succ}(r)$, if either:
  1. $r = r_\infty$ and $r = r'$, or
  2. $r \neq r_\infty$, $r \neq r'$ and $\forall \eta \in r$:

$$\exists d \in \mathbb{R}_{>0}.\, (\eta{+}d \in r' \quad \text{and} \quad \forall 0 \leqslant d' \leqslant d.\, \eta{+}d' \in r \cup r')$$

- The *successor region*: $\mathit{succ}(\langle \ell, r \rangle) \;=\; \langle \ell, \mathit{succ}(r) \rangle$
- Note: the location invariants are ignored so far!

# Example

# Time convergence (without proof)

For non-zeno *TA* and $\pi = s_0\, s_1\, s_2 \ldots$ a path in $TS(TA)$:

(a) $\pi$ is *time-convergent* $\Rightarrow$ $\exists$ state region $\langle \ell, r \rangle$ such that for some $j$:

$$s_i \in \langle \ell, r \rangle \;\; \text{for all } i \geqslant j$$

(b) If $\exists$ state region $\langle \ell, r \rangle$ with $r \neq r_\infty$ and an index $j$ such that:

$$s_i \in \langle \ell, r \rangle \;\; \text{for all } i \geqslant j$$

then $\pi$ is *time-convergent*

time-convergent paths are paths that only perform delays from some time instant on

# Region automaton

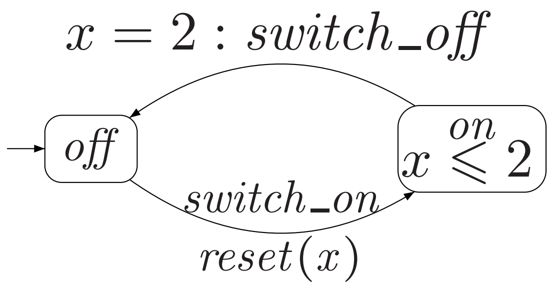
For non-zeno *TA* with $\mathit{TS}(\mathit{TA}) = (S, \mathit{Act}, \rightarrow, I, \mathit{AP}, L)$ let:

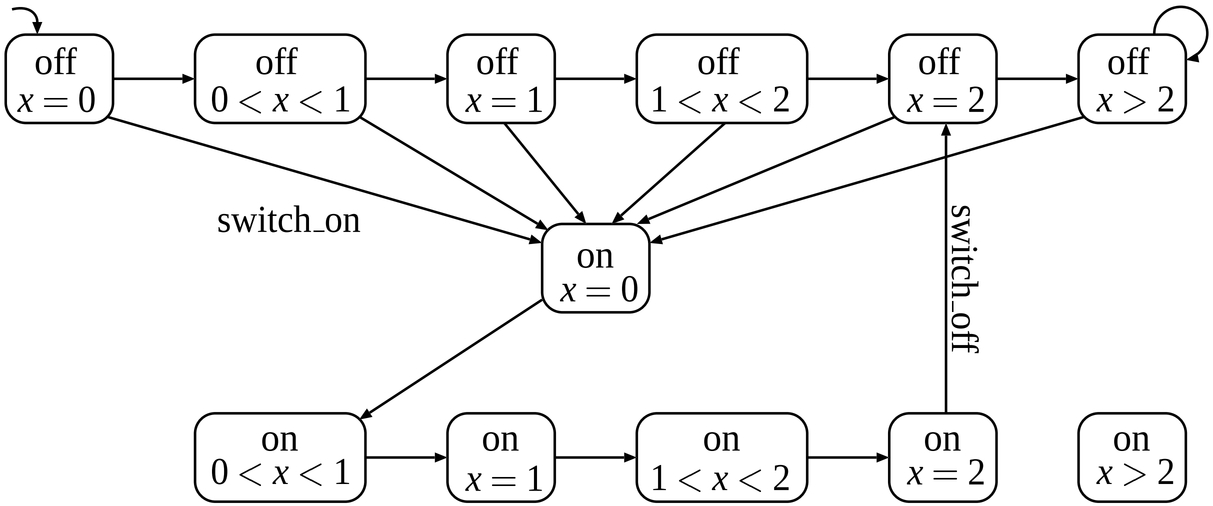
$$\mathit{RTS}(\mathit{TA}, \Phi) = (S', \mathit{Act} \cup \{\, \tau \,\}, \rightarrow', I, \mathit{AP}', L') \quad \text{with}$$

- $S' = S/\cong = \{\, [s] \mid s \in S \,\}$ and $I' = \{\, [s] \mid s \in I \,\}$, the state regions
- $L'(\langle \ell, r \rangle) = L(\ell) \cup \{\, g \in \mathit{AP}' \setminus \mathit{AP} \mid r \models g \,\}$
- $\rightarrow'$ is defined by: $\dfrac{\ell \xhookrightarrow{g:\alpha, D} \ell' \quad r \models g \quad \text{reset } D \text{ in } r \models \mathit{Inv}(\ell')}{\langle \ell, r \rangle \xrightarrow{\alpha}' \langle \ell', \text{reset } D \text{ in } r \rangle}$

  and $\dfrac{r \models \mathit{Inv}(\ell) \quad \mathit{succ}(r) \models \mathit{Inv}(\ell)}{\langle \ell, r \rangle \xrightarrow{\tau}' \langle \ell, \mathit{succ}(r) \rangle}$

# Example: simple light switch



# Correctness theorem [Alur and Dill, 1989]

For non-Zeno timed automaton *TA* and TCTL$_\diamond$ formula $\Phi$:

$$\underbrace{\mathit{TA} \models \Phi}_{\text{TCTL semantics}} \quad \text{iff} \quad \underbrace{\mathit{RTS}(\mathit{TA}, \Phi) \models \Phi}_{\text{CTL semantics}}$$

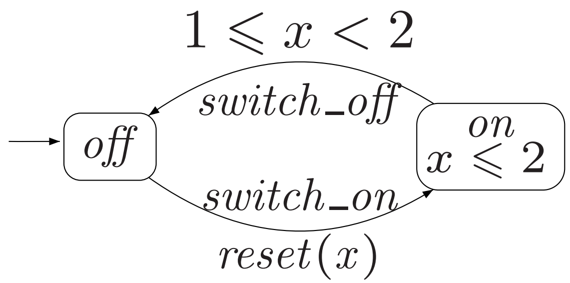
# Proof

# Timelock freedom

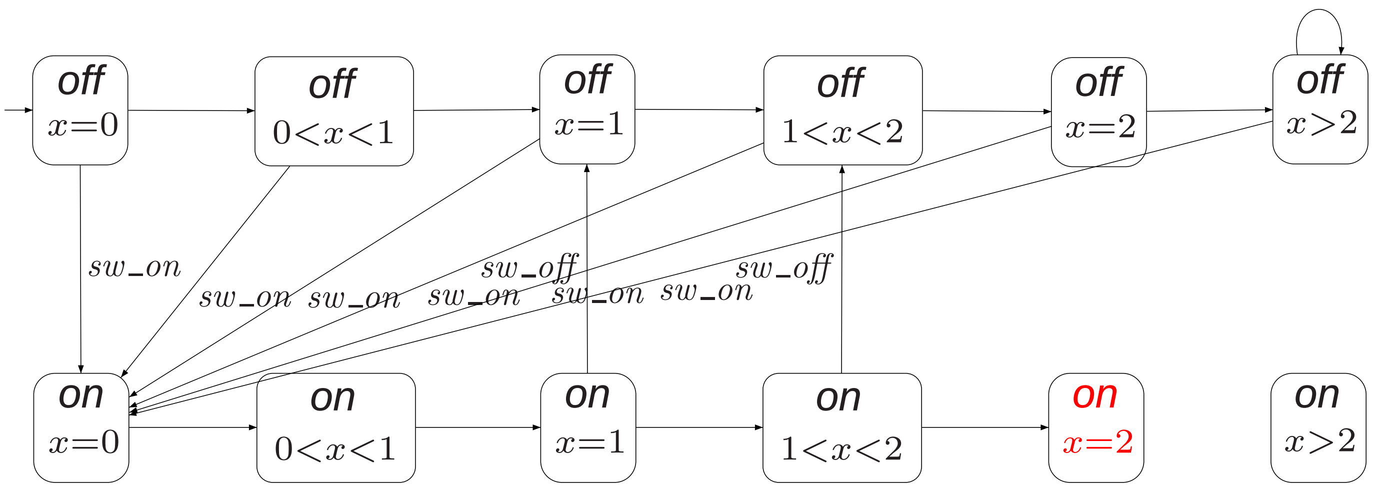
Non-zeno $TA$ is timelock-free iff no reachable state in $RTS(TA)$ is terminal

timelocks can thus be checked by a reachability analysis of $RTS(TA)$

# Example



$1 \leqslant x < 2$
$switch\_off$
$off$
$on$
$x \leqslant 2$
$switch\_on$
$reset(x)$

off $x{=}0$
off $0{<}x{<}1$
off $x{=}1$
off $1{<}x{<}2$
off $x{=}2$
off $x{>}2$
$sw\_on$
$sw\_on$ $sw\_on$
$sw\_off$
$sw\_on$ $sw\_on$ $sw\_on$
$sw\_off$
on $x{=}0$
on $0{<}x{<}1$
on $x{=}1$
on $1{<}x{<}2$
on $x{=}2$
on $x{>}2$

# TCTL model-checking algorithm

Main ideas:

- Equip *TA* with a single clock
  - as opposed to a single clock for each (timed) subformula $\Phi \, \mathsf{U}^{J} \, \Psi$
- Introduce atomic proposition for each timed subformula
- Convert timed CTL formula $\Phi$ into $\widehat{\Phi}$
- And check $\widehat{\Phi}$ on $RTS(TA)$
  - using standard CTL model checking

# Extra atomic propositions

$$\text{TCTL formula } \Phi \;=\; \forall\Box^{\leqslant 3}\Big(\underbrace{\underbrace{\exists\Diamond^{[2,6]}a}_{=\Psi_1} \wedge \underbrace{\exists\Box^{]2,5[}\underbrace{\forall\Diamond^{\geqslant 3}\underbrace{(b \wedge (x=9))}_{=\Psi_2}}_{=\Psi_3}}_{=\Psi_4}}_{=\Psi_5}\Big)$$

The set of propositions of $R$ contains:

- the propositions $a$ and $b$, and the clock constraint $x{=}9$
- the propositions $a_{\Psi_1}$ through $a_{\Psi_5}$, and $a_\Phi$
- the clock constraints $z \leqslant 3$, $z \in [2,6]$, $z \in ]2,5[$ and $z \geqslant 3$

*Input:* non-zeno, timelock-free timed automaton *TA* and TCTL formula $\Phi$
*Output:* "yes" if *TA* $\models \Phi$, "no" otherwise.

---

$R := \mathit{RTS}(\mathit{TA} \oplus z, \Phi)$; (* with state space $S_{rts}$ and labeling $L_{rts}$ *)
**for all** $i \leqslant |\,\Phi\,|$ **do**
  **for all** $\Psi \in \mathit{Sub}(\Phi)$ with $|\,\Psi\,| = i$ **do**

    **switch**($\Psi$):
      true : $\mathit{Sat}_R(\Psi) := S_{rts}$;
      $a$ : $\mathit{Sat}_R(\Psi) := \{\, s \in S_{rts} \mid a \in L_{rts}(s) \,\}$;
      $\Psi_1 \wedge \Psi_2$ : $\mathit{Sat}_R(\Psi) := \{\, s \in S_{rts} \mid \{a_{\Psi_1}, a_{\Psi_2}\} \subseteq L_{rts}(s) \,\}$;
      $\neg\Psi'$ : $\mathit{Sat}_R(\Psi) := \{\, s \in S_{rts} \mid a_{\Psi'} \notin L_{rts}(s) \,\}$;
      $\exists(\Psi_1 \,\mathsf{U}^J\, \Psi_2)$ : $\mathit{Sat}_R(\Psi) := \mathit{Sat}_{\mathrm{CTL}}\Big( \exists(\, (a_{\Psi_1} \vee a_{\Psi_2}) \,\mathsf{U}\, ((z \in J) \wedge a_{\Psi_2})\,) \Big)$;
      $\forall(\Psi_1 \,\mathsf{U}^J\, \Psi_2)$ : $\mathit{Sat}_R(\Psi) := \mathit{Sat}_{\mathrm{CTL}}\Big( \forall(\, (a_{\Psi_1} \vee a_{\Psi_2}) \,\mathsf{U}\, (\, (z \in J) \wedge a_{\Psi_2})\,) \Big)$;
    **end switch**

(* add $a_\Psi$ to the labeling of all state regions where $\Psi$ holds *)
    **forall** $s \in S_{rts}$ with $s\{z := 0\} \in \mathit{Sat}_R(\Psi)$ **do** $L_{rts}(s) := L_{rts}(s) \cup \{\, a_\Psi \,\}$ **od**;
  **od**
**od**
**if** $I_{rts} \subseteq \mathit{Sat}_R(\Phi)$ **then return** "yes" **else return** "no" **fi**

# Time complexity

For timed automaton *TA* and TCTL formula $\Phi$, the model-checking problem

*TA* $\models \Phi$ can be determined in time $\mathcal{O}\left( (N+K) \cdot | \Phi | \right)$,

where $N$ and $K$ are the number of states and transitions in *RTS*(*TA*, $\Phi$)

# Other verification problems

1. The TCTL model-checking problem is PSPACE-complete
2. Model checking safety, reachability, or $\omega$-regular properties in TA is PSPACE-complete
3. Model checking LTL and CTL against TA is PSPACE-complete
4. The model-checking problem for timed LTL is undecidable
5. The satisfaction problem for TCTL is undecidable

*all facts without proof*