Timed CTL Model Checking

Lecture #17 of Advanced Model Checking

Joost-Pieter Katoen

Lehrstuhl 2: Software Modeling & Verification

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

January 11, 2006

Timelock, time-divergence and Zenoness

- A path is *time-divergent* if its execution time is infinite

$$ExecTime(s_0 \overset{d_0}{\Longrightarrow} s_1 \overset{d_1}{\Longrightarrow} \ldots) = \sum_{i=0} d_i = \infty$$

- *TA* is *timelock-free* if no state in $Reach(TS(TA))$ contains a timelock

  a state contains a timelock whenever no time-divergent paths emanate from it

- *TA* is *non-Zeno* if there does not exist an initial Zeno path in $TS(TA)$

  a path is Zeno if it is time-convergent and performs infinitely many actions

Timed CTL

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

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

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

$$\varphi ::= \Diamond^{J} \Phi$$

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

Forms of J: $[n, m]$, $(n, m]$, $[n, m)$ or (n, m) for $n, m \in \mathbb{N}$ and $n \leqslant m$

for right-open intervals, $m = \infty$ is also allowed

TCTL-semantics for timed automata

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

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

- *TA* satisfies TCTL-formula Φ iff Φ 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 $x \in C$

Characterizing timelock

- A state is *timelock-free* if and only if it satisfies $\exists\Box$true
  - some time-divergent path satisfies $\Box$true, i.e., there is $\geqslant 1$ time-divergent path
  - note: for fair CTL, the states in which a fair path starts also satisfy $\exists\Box$true

- *TA* is timelock-free iff $\forall s \in \mathit{Reach}(\mathit{TS}(\mathit{TA}))$: $s \models \exists\Box$true

TCTL model checking

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

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

  - timelocks in TA are irrelevant as their presence can be checked

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

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

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

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*
  - $s \models \exists\Diamond^{J}\Phi$ iff *reset* z *in* $s \models z \in J \wedge \Phi$
  - deal with $\exists\Box^{J}\Phi$, $\forall\Diamond^{J}\Phi$, and $\forall\Box^{J}\Phi$ in a similar way

- 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 $z \notin C$

- E.g., $\exists\Box^{\leqslant 2}\,\Phi$ yields $\exists\Box\,((z \leqslant 2) \rightarrow \Phi)$

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

Clock equivalence

Impose an equivalence, denoted $\cong$, on the clock valuations such that:

(A) Equivalent clock valuations satisfy the same clock constraints g in *TA* and Φ:

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

- no diagonal clock constraints are considered
- all the constraints in *TA* and Φ are thus either of the form $x \leqslant c$ or $x < c$

(B) Time-divergent paths emanating from 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)

Basic recipe of TCTL model checking

Input: timed automaton *TA* and TCTL formula Φ (both over *AP* and C)
Output: $TA \models \Phi$

$\widehat{\Phi} :=$ eliminate the timing parameters from Φ;

determine the equivalence classes under $\cong$;

construct the region transition system $TS = RG(TA)$;

apply the CTL model-checking algorithm to check $TS \models \widehat{\Phi}$;

$TA \models \Phi$ if and only if $TS \models \widehat{\Phi}$

how does clock equivalence look like?

First observation

- $\eta \models x < c$ whenever $\eta(x) < c$, or equivalently, $\lfloor \eta(x) \rfloor < c$
  - $\lfloor d \rfloor = \max\{ c \in \mathbb{N} \mid c \leqslant d \}$ and $frac(d) = d - \lfloor d \rfloor$
- $\eta \models x \leqslant c$ whenever $\lfloor \eta(x) \rfloor < c$ or $\lfloor \eta(x) \rfloor = c$ and $frac(x) = 0$

$\Rightarrow$ $\eta \models g$ only depends on $\lfloor \eta(x) \rfloor$, and whether $frac(\eta(x)) = 0$

- Initial suggestion: clock valuations η and η' are equivalent if:

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

- Note: it is crucial that in $x < c$ and $x \leqslant c$, c is a natural

Example

Second observation

- Consider location ℓ with *Inv*$(\ell) =$ true and only outgoing transitions:
  - one guarded with $x \geqslant 2$ (action α) and $y > 1$ (action β)
- Let state $s = \langle \ell, \eta \rangle$ with $1 < \eta(x) < 2$ and $0 < \eta(y) < 1$
  - α and β are disabled, only time may elapse
- Transition that is enabled next depends on $x < y$ or $x \geqslant y$
  - e.g., if $frac(\eta(x)) \geqslant frac(\eta(y))$, action α is enabled first
- Suggestion for strengthening of initial proposal for all $x, y \in C$ by:

$$frac(\eta(x)) \leqslant frac(\eta(y)) \quad \text{if and only if} \quad frac(\eta'(x)) \leqslant frac(\eta'(y))$$

Example

Final observation

- So far, clock equivalence yield a denumerable though not finite quotient
- For *TA* $\models \Phi$ only the clock constraints in *TA* and Φ are relevant
  - let $c_x \in \mathbb{N}$ the *largest constant* with which x is compared in *TA* or Φ

$\Rightarrow$ If $\eta(x) > c_x$ then the actual value of x is irrelevant

- constraints on $\cong$ so far are only relevant for clock values of x (y) up to c_x (c_y)

Clock equivalence

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

(1) for any $x \in C$: $(\eta(x) > c_x) \wedge (\eta'(x) > c_x)$ or $(\eta(x) \leqslant c_x) \wedge (\eta'(x) \leqslant c_x)$

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

$$\lfloor \eta(x) \rfloor = \lfloor \eta'(x) \rfloor \quad \text{and} \quad \mathit{frac}(\eta(x)) = 0 \text{ iff } \mathit{frac}(\eta_2(x)) = 0$$

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

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

$$s \cong s' \quad \text{iff} \quad \ell = \ell' \quad \text{and} \quad \eta \cong \eta'$$

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

Number of regions

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

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

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

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

Proof

Preservation of atomic properties

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

$$\eta \models g \quad \text{if and only if} \quad \eta' \models g \text{ for any } g \in \mathit{AP}' \setminus \mathit{AP}$$

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 AP' includes all atomic propositions in *TA* and atomic clock constraints

Clock equivalence is a bisimulation

Clock equivalence is a bisimulation equivalence over AP'

Proof

Region automaton: intuition

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 (no proof)

For non-Zeno *TA* and $\pi = s_0\, s_1\, s_2 \ldots$ an initial, infinite path in *TS*(*TA*):

(a) π 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 π is *time-convergent*

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

Region automaton

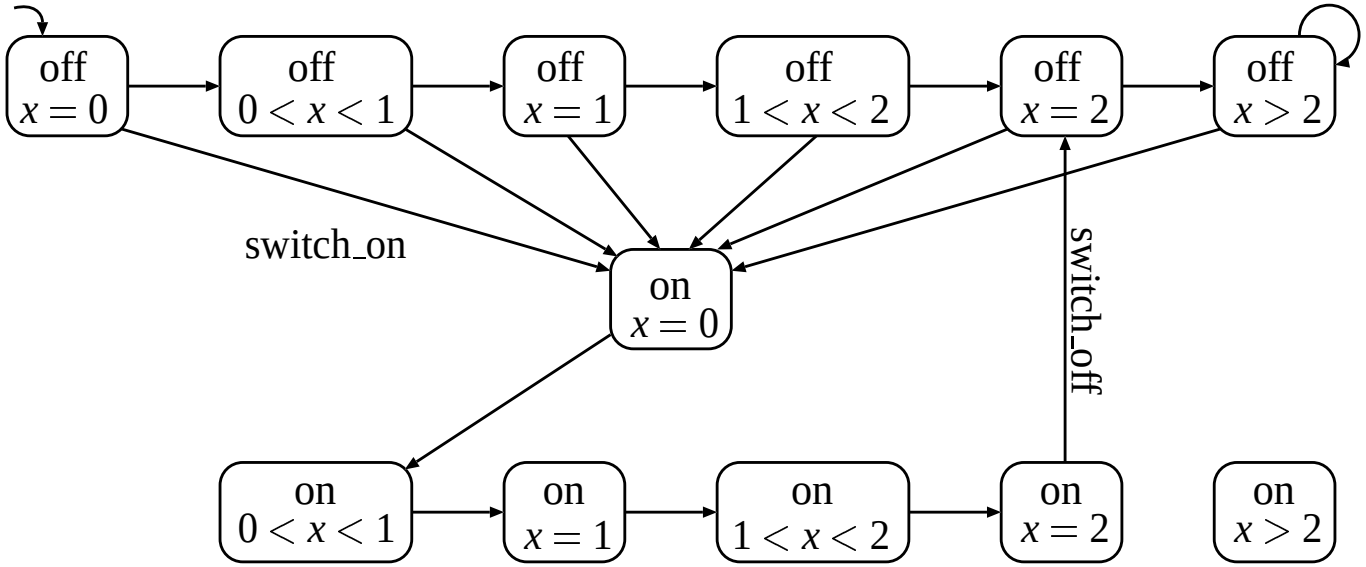
For non-Zeno *TA* with $TS(TA) = (S, Act, \rightarrow, I, AP, L)$ let:

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

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

$$\frac{r \models \textit{Inv}(\ell) \quad \textit{succ}(r) \models \textit{Inv}(\ell)}{\langle \ell, r \rangle \xrightarrow{\tau}{}' \langle \ell, \textit{succ}(r) \rangle}$$

Example: simple light switch



Correctness theorem

Let *TA* be a non-Zeno timed automaton and Φ a TCTL$_\Diamond$ formula. Then:

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

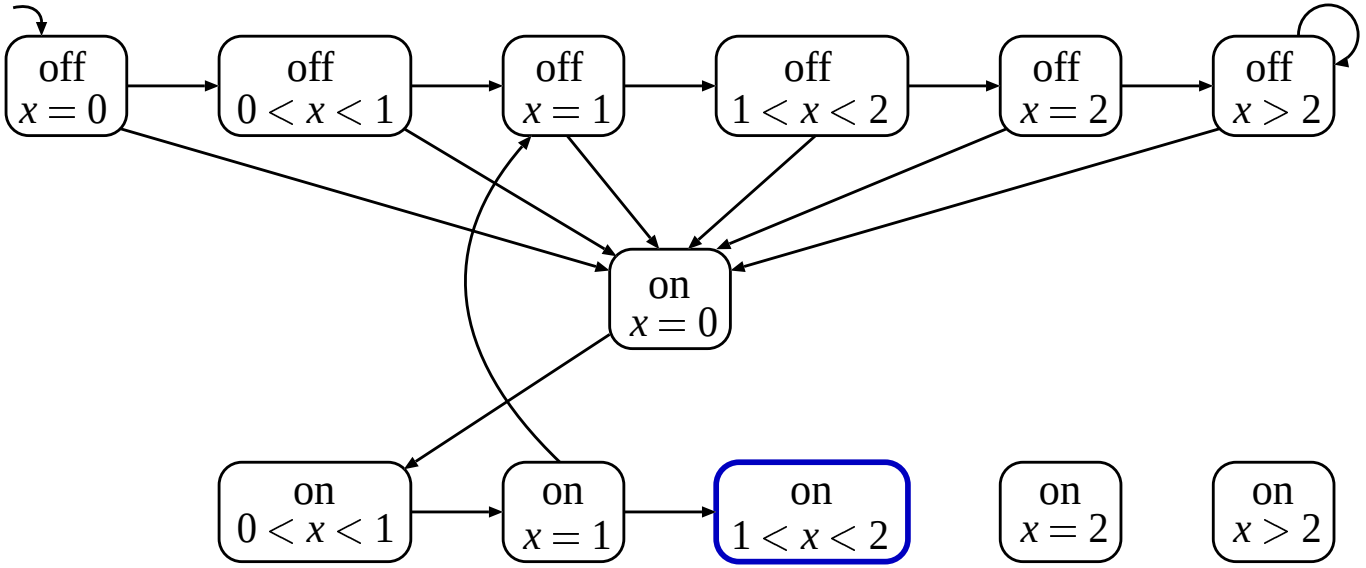
Proof

Timelock freedom

For non-Zeno *TA*:

TA is timelock-free iff no reachable state in $RG(TA)$ is terminal

Example


off
$x = 0$
off
$0 < x < 1$
off
$x = 1$
off
$1 < x < 2$
off
$x = 2$
off
$x > 2$
on
$x = 0$
on
$0 < x < 1$
on
$x = 1$
on
$1 < x < 2$
on
$x = 2$
on
$x > 2$

Overview TCTL model checking

Input: timed automaton *TA* and TCTL formula Φ (both over *AP* and C)
Output: $TA \models \Phi$

$\widehat{\Phi} :=$ eliminate the timing parameters from Φ;

determine the equivalence classes under $\cong$;

construct the region transition system $TS = RG(TA)$;

apply the CTL model-checking algorithm to check $TS \models \widehat{\Phi}$;

$TA \models \Phi$ if and only if $TS \models \widehat{\Phi}$

Other verification problems

1. The TCTL model-checking problem is PSPACE-complete
2. The model-checking problem for timed LTL (and TCTL*) is undecidable
3. The satisfaction problem for TCTL is undecidable

all facts without proof