

Timed CTL

Lecture #17 of Advanced Model Checking

Joost-Pieter Katoen

Lehrstuhl 2: Software Modeling & Verification

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

January 18, 2011

Timelock, time-divergence and Zenoness

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

$$ExecTime(s_0 \xrightarrow{d_0} s_1 \xrightarrow{d_1} \dots) = \sum_{i=0} d_i = \infty$$

- *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 $J \subseteq \mathbb{R}_{\geq 0}$ is an interval whose bounds are naturals

$\diamond^J \Phi$ asserts that a Φ -state is reached at time instant $t \in J$

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

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

Some abbreviations

“Always” is obtained in the following way:

$$\exists \square^J \Phi = \neg \forall \diamond^J \neg \Phi \quad \text{and} \quad \forall \square^J \Phi = \neg \exists \diamond^J \neg \Phi$$

$\exists \square^J \Phi$ asserts that for some path during the interval J , Φ holds

$\forall \square^J \Phi$ requires this to hold for all paths

Standard \square and \diamond -operator are obtained as follows:

$$\diamond \Phi = \diamond^{[0, \infty)} \Phi \quad \text{and} \quad \square \Phi = \square^{[0, \infty)} \Phi$$

Timed properties in TCTL

Semantics of TCTL

For state $s = \langle \ell, \eta \rangle$ in $TS(TA)$ the satisfaction relation \models is defined by:

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

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

$$s \models g \quad \text{iff} \quad \eta \models g$$

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

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

$$s \models \exists \varphi \quad \text{iff} \quad \pi \models \varphi \text{ for some } \pi \in \text{Paths}_{\text{div}}(s)$$

$$s \models \forall \varphi \quad \text{iff} \quad \pi \models \varphi \text{ for all } \pi \in \text{Paths}_{\text{div}}(s)$$

path quantification over time-divergent paths only

The \Rightarrow relation

For infinite path fragments in $TS(TA)$ performing ∞ many actions let:

$$s_0 \xRightarrow{d_0} s_1 \xRightarrow{d_1} s_2 \xRightarrow{d_2} \dots \quad \text{with } d_0, d_1, d_2 \dots \geq 0$$

denote the equivalence class containing all infinite path fragments induced by execution fragments of the form:

$$s_0 \underbrace{\xrightarrow{d_0^1} \dots \xrightarrow{d_0^{k_0}}}_{\substack{\text{time passage of} \\ d_0 \text{ time-units}}} s_0 + d_0 \xrightarrow{\alpha_1} s_1 \underbrace{\xrightarrow{d_1^1} \dots \xrightarrow{d_1^{k_1}}}_{\substack{\text{time passage of} \\ d_1 \text{ time-units}}} s_1 + d_1 \xrightarrow{\alpha_2} s_2 \underbrace{\xrightarrow{d_2^1} \dots \xrightarrow{d_2^{k_2}}}_{\substack{\text{time passage of} \\ d_2 \text{ time-units}}} s_2 + d_2 \xrightarrow{\alpha_3} \dots$$

where $k_i \in \mathbb{N}$, $d_i \in \mathbb{R}_{\geq 0}$ and $\alpha_i \in Act$ such that $\sum_{j=1}^{k_i} d_i^j = d_i$.

For $\pi \in s_0 \xRightarrow{d_0} s_1 \xRightarrow{d_1} \dots$ we have $ExecTime(\pi) = \sum_{i \geq 0} d_i$

Semantics of TCTL

For time-divergent path $\pi \in s_0 \xRightarrow{d_0} s_1 \xRightarrow{d_1} \dots$, we have:

$$\pi \models \Diamond^J \Psi \quad \text{iff} \quad \exists i \geq 0. s_i + d \models \Psi \text{ for some } d \in [0, d_i] \text{ with}$$

$$\sum_{k=0}^{i-1} d_k + d \in J \quad \text{and}$$

where for $s_i = \langle \ell_i, \eta_i \rangle$ and $d \geq 0$ we have $s_i + d = \langle \ell_i, \eta_i + d \rangle$

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* $Sat(\Phi)$ is defined by:

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

- TA satisfies TCTL-formula Φ iff Φ holds in all initial states of TA :

$$TA \models \Phi \quad \text{if and only if} \quad \forall \ell_0 \in Loc_0. \langle \ell_0, \eta_0 \rangle \models \Phi$$

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

Example

Timed CTL versus CTL

- Due to ignoring time-convergent paths in TCTL semantics possibly:

$$\underbrace{TS(TA) \models_{\text{TCTL}} \forall \varphi}_{\text{TCTL semantics}} \quad \text{but} \quad \underbrace{TS(TA) \not\models_{\text{CTL}} \forall \varphi}_{\text{CTL semantics}}$$

- CTL semantics considers **all** paths, timed CTL only **time-divergent** paths

- For $\Phi = \forall \square (on \longrightarrow \forall \diamond off)$ and the light switch

$$TS(Switch) \models_{\text{TCTL}} \Phi \quad \text{whereas} \quad TS(TA) \not\models_{\text{CTL}} \Phi$$

- there are time-convergent paths on which location *on* is never left

Characterizing timelock

- TCTL semantics is also well-defined for TA with timelock
- A state contains a timelock whenever no time-divergent paths emanate from it
- A state is *timelock-free* if and only if it satisfies $\exists \Box \text{true}$
 - some time-divergent path satisfies $\Box \text{true}$, i.e., there is ≥ 1 time-divergent path
 - note: for fair CTL, the states in which a fair path starts also satisfy $\exists \Box \text{true}$
- TA is timelock-free iff $\forall s \in \text{Reach}(TS(TA)): s \models \exists \Box \text{true}$
- Timelocks can thus be checked by a timed CTL formula

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 $RTS(TA)$
 - dependence on Φ is ignored for the moment . . .
- Transform TCTL formula Φ into an “equivalent” CTL-formula $\hat{\Phi}$
- Then: $TA \models_{\text{TCTL}} \Phi$ iff $\underbrace{RTS(TA)}_{\text{finite transition system}} \models_{\text{CTL}} \hat{\Phi}$

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:

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

- where $TA \oplus z$ is TA (over C) extended with $z \notin C$
- E.g., $\exists \Box^{\leq 2} \Phi$ yields $\exists \Box((z \leq 2) \rightarrow \Phi)$

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 :

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

- (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

⇒ 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

⇒ 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: $TA \models \Phi$

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

determine the equivalence classes under \cong ;

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

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

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

how does clock equivalence look like?

First observation

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

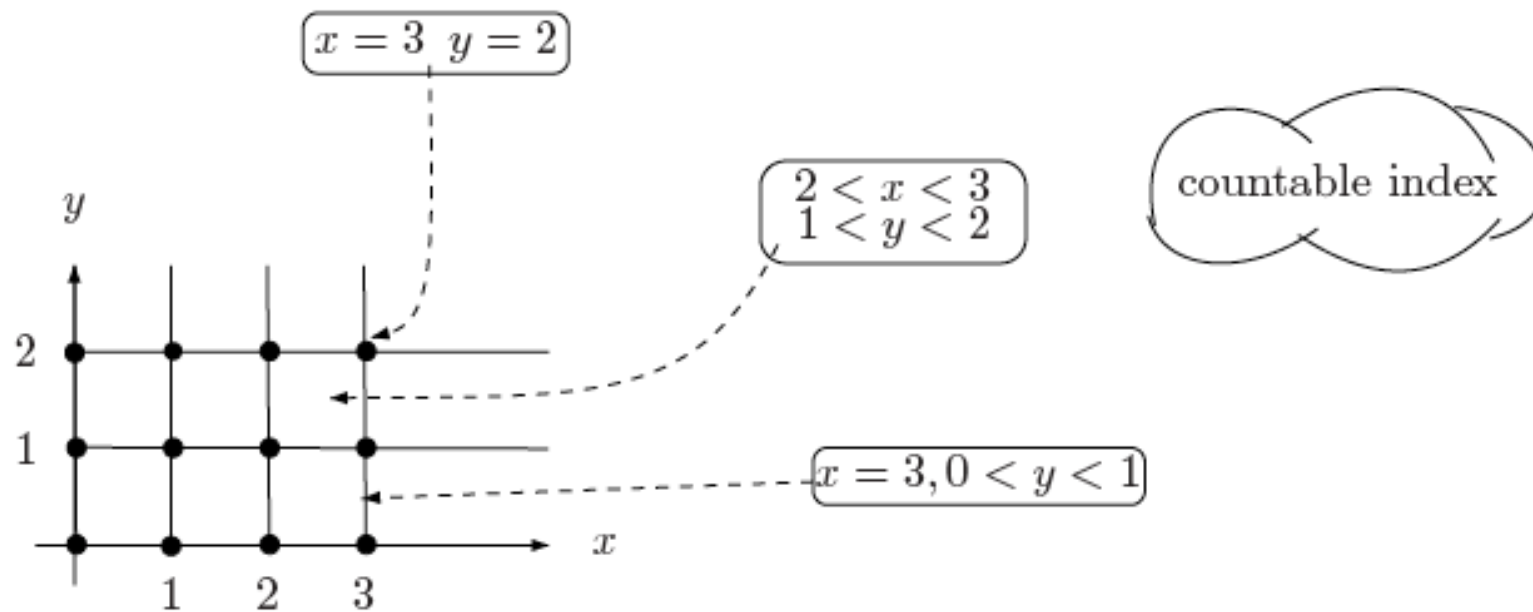
$\Rightarrow \eta \models g$ only depends on $\lfloor \eta(x) \rfloor$, and whether $\text{frac}(\eta(x)) = 0$

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

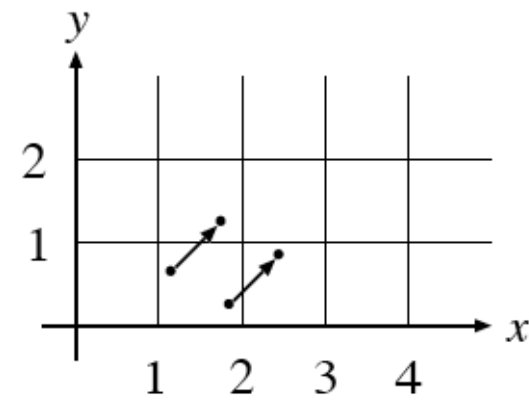
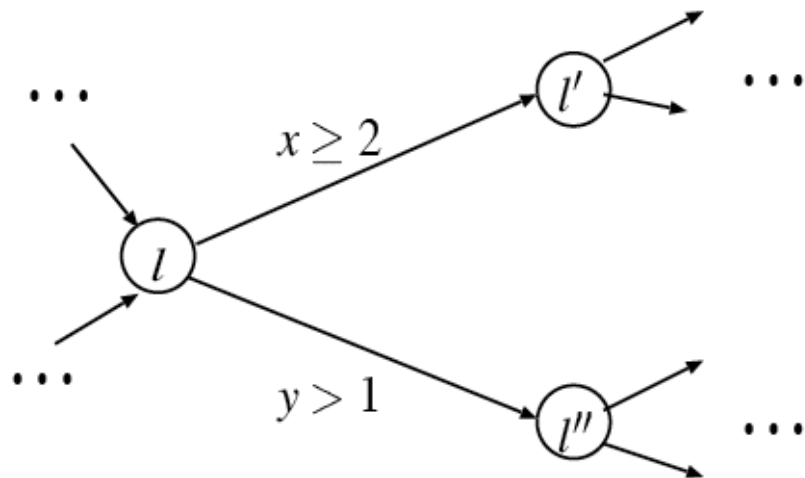
$$\lfloor \eta(x) \rfloor = \lfloor \eta'(x) \rfloor \quad \text{and} \quad \text{frac}(\eta(x)) = 0 \text{ iff } \text{frac}(\eta'(x)) = 0$$

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

Example



A problem

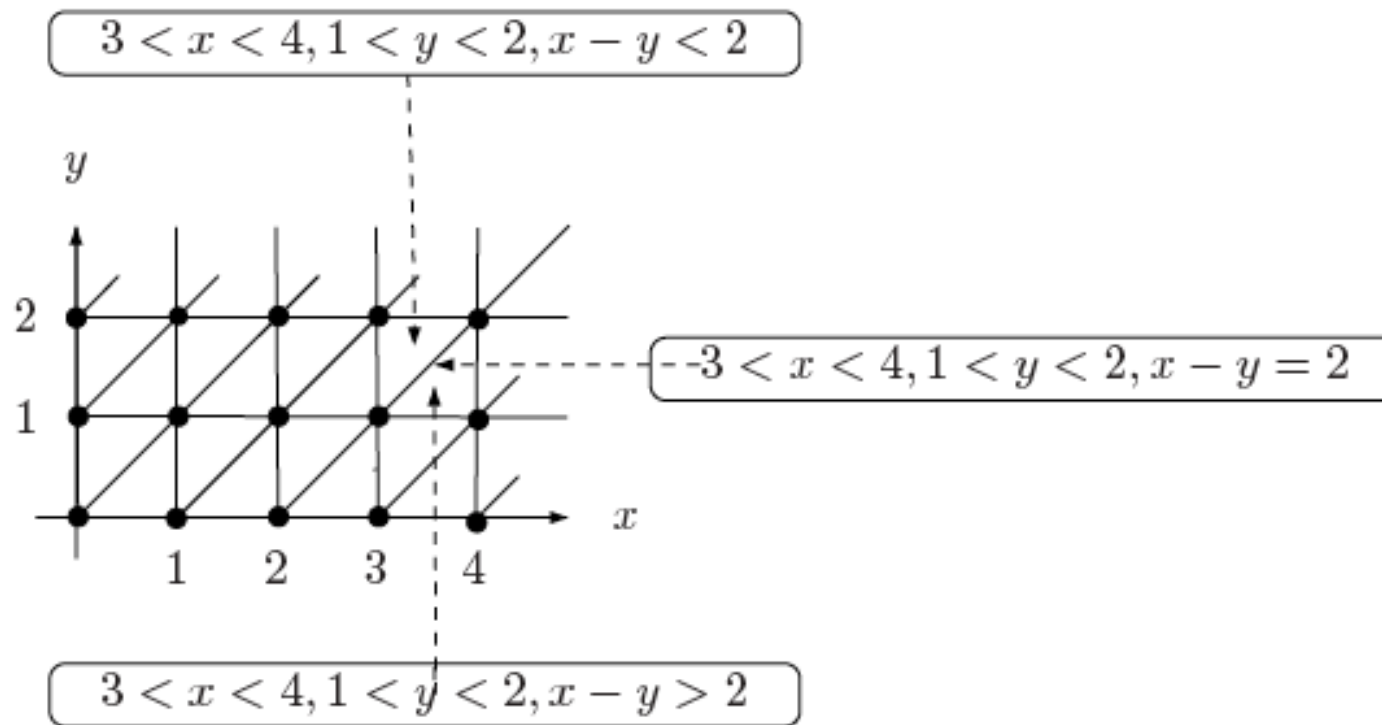


Second observation

- Consider location ℓ with $Inv(\ell) = \text{true}$ and only outgoing transitions:
 - one guarded with $x \geq 2$ (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 $x \geq y$
 - e.g., if $\text{frac}(\eta(x)) \geq \text{frac}(\eta(y))$, action α is enabled first
- Suggestion for **strengthening** of initial proposal for all $x, y \in C$ by:

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

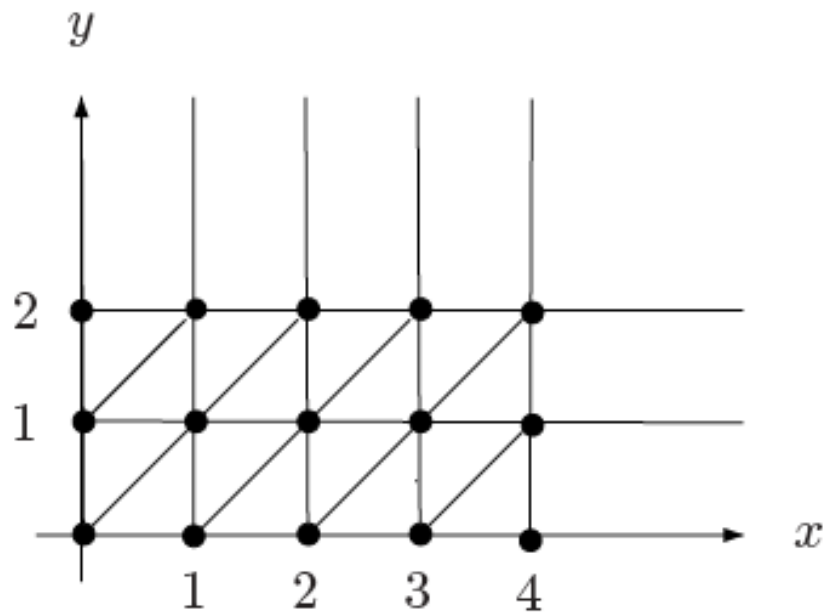
Example



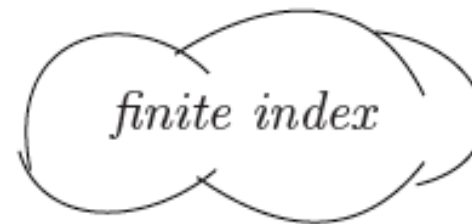
Final observation

- 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)

Example revisited



$$c_y = 2 \quad c_x = 4$$



Clock equivalence

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

- for all $x \in C$: $\eta(x) > c_x$ and $\eta'(x) > c_x$, or
- for any $x, y \in C$ with $\eta(x) \leq c_x$ and $\eta'(x) \leq c_x$, and $\eta(y) \leq c_y$ and $\eta'(y) \leq c_y$ it holds:
 - $\lfloor \eta(x) \rfloor = \lfloor \eta'(x) \rfloor$ and $\text{frac}(\eta(x)) = 0$ iff $\text{frac}(\eta'(x)) = 0$, and
 - $\text{frac}(\eta(x)) \leq \text{frac}(\eta(y))$ iff $\text{frac}(\eta'(x)) \leq \text{frac}(\eta'(y))$.

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

Regions

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

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

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

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

Example