Verification – Lecture 24
Timed Automata

Bernd Finkbeiner – Sven Schewe
Rayna Dimitrova – Lars Kuhtz – Anne Proetzsch

Wintersemester 2007/2008

REVIEW

Timed automaton

A *timed automaton* is a tuple

$$TA \;=\; \big(Loc, Act, C, \rightsquigarrow, Loc_0, inv, AP, L\big) \quad \text{where:}$$

- Loc is a finite set of locations.
- $Loc_0 \subseteq Loc$ is a set of initial locations
- C is a finite set of clocks
- $L : Loc \to 2^{AP}$ is a labeling function for the locations
- $\rightsquigarrow \subseteq Loc \times CC(C) \times Act \times 2^{C} \times Loc$ is a transition relation, and
- $inv : Loc \to CC(C)$ is an invariant-assignment function

Clock constraints

- *Clock constraints* over set C of clocks are defined by:

$$g ::= \text{true} \;\big|\; x < c \;\big|\; x - y < c \;\big|\; x \leqslant c \;\big|\; x - y \leqslant c \;\big|\; \neg g \;\big|\; g \wedge g$$

  - where $c \in \mathbb{N}$ and clocks $x, y \in C$
  - rational constants would do; neither reals nor addition of clocks!
  - let *CC*(C) denote the set of clock constraints over C
  - shorthands: $x \geqslant c$ denotes $\neg(x < c)$ and $x \in [c_1, c_2)$ or $c_1 \leqslant x < c_2$ denotes $\neg(x < c_1) \wedge (x < c_2)$

- *Atomic clock constraints* do not contain true, $\neg$ and $\wedge$
  - let *ACC*(C) denote the set of atomic clock constraints over C

- Simplification: In the following, we assume constraints are *diagonal-free*, i.e., do neither contain $x - y \leqslant c$ nor $x - y < c$.

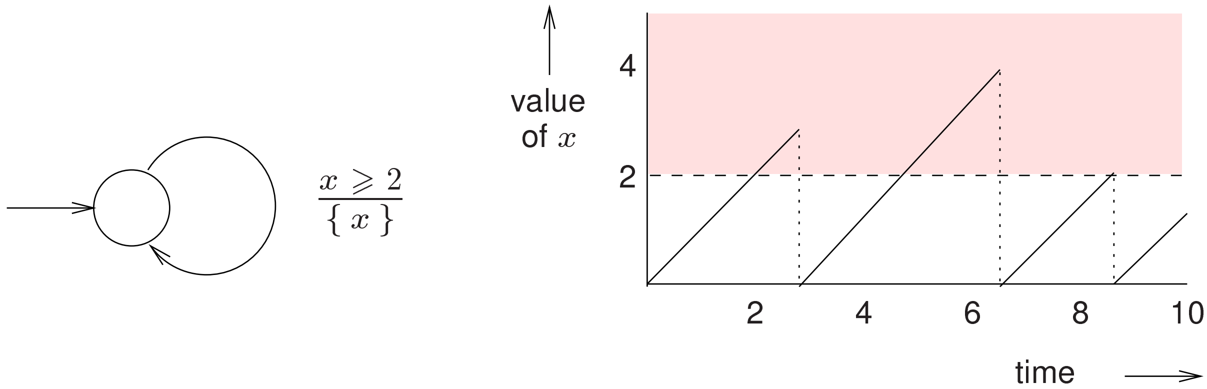
Intuitive interpretation

- Edge $\ell \xrightarrow{g:\alpha, C'} \ell'$ means:
  - action α is enabled once guard g holds
  - when moving from location ℓ to ℓ', any clock in C' will be reset to zero

- *inv*(ℓ) constrains the amount of time that may be spent in location ℓ
  - the location ℓ must be left before the invariant *inv*(ℓ) is violated

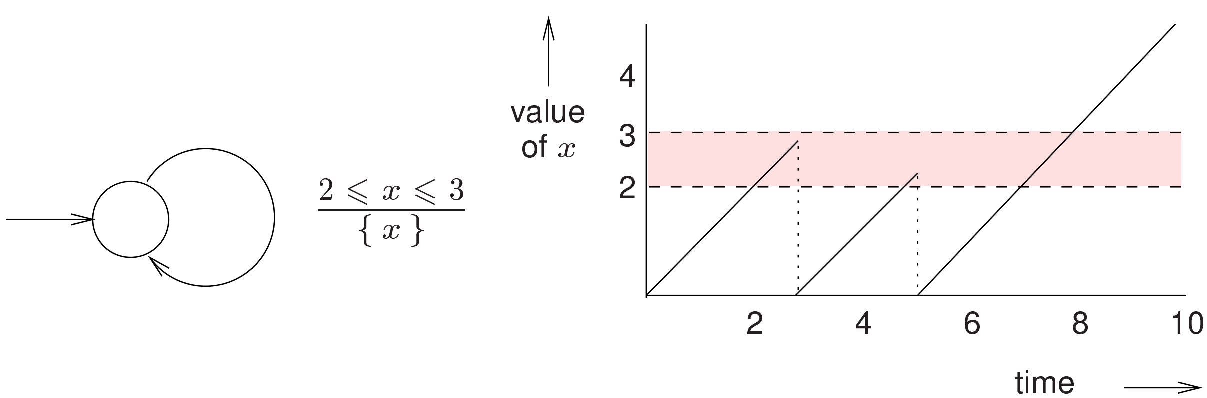
Guards versus location invariants

The effect of a lowerbound guard:



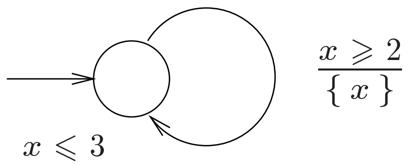
Guards versus location invariants

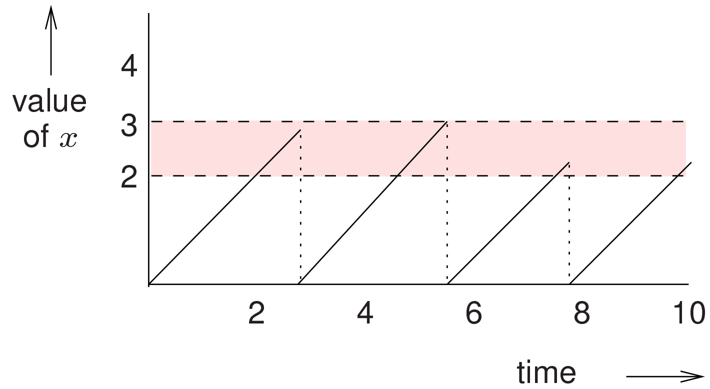
The effect of a lowerbound and upperbound guard:



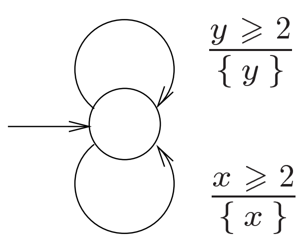
Guards versus location invariants

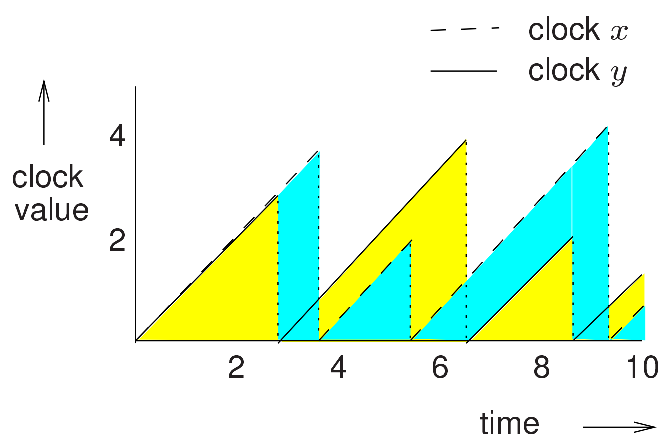
The effect of a guard and an invariant:





Arbitrary clock differences





This is impossible to model in a discrete-time setting

Composing timed automata

Let $\mathit{TA}_i = (\mathit{Loc}_i, \mathit{Act}_i, C_i, \rightsquigarrow_i, \mathit{Loc}_{0,i}, \mathit{inv}_i, \mathit{AP}, L_i)$ and H an action-set

$$\mathit{TA}_1 \,||_H\, \mathit{TA}_2 = (\mathit{Loc}, \mathit{Act}_1 \cup \mathit{Act}_2, C, \rightsquigarrow, \mathit{Loc}_0, \mathit{inv}, \mathit{AP}, L) \quad \text{where:}$$

- $\mathit{Loc} = \mathit{Loc}_1 \times \mathit{Loc}_2$ and $\mathit{Loc}_0 = \mathit{Loc}_{0,1} \times \mathit{Loc}_{0,2}$ and $C = C_1 \cup C_2$
- $\mathit{inv}(\langle \ell_1, \ell_2 \rangle) = \mathit{inv}_1(\ell_1) \,\wedge\, \mathit{inv}_2(\ell_2)$ and $L(\langle \ell_1, \ell_2 \rangle) = L_1(\ell_1) \cup L_2(\ell_2)$
- $\rightsquigarrow$ is defined by the inference rules: for $\alpha \in H$ $\dfrac{\ell_1 \overset{g_1:\alpha,D_1}{\rightsquigarrow_1} \ell_1' \;\wedge\; \ell_2 \overset{g_2:\alpha,D_2}{\rightsquigarrow_2} \ell_2'}{\langle \ell_1, \ell_2 \rangle \overset{g_1 \wedge g_2:\alpha,D_1 \cup D_2}{\rightsquigarrow} \langle \ell_1', \ell_2' \rangle}$

for $\alpha \notin H$: $\dfrac{\ell_1 \overset{g:\alpha,D}{\rightsquigarrow_1} \ell_1'}{\langle \ell_1, \ell_2 \rangle \overset{g:\alpha,D}{\rightsquigarrow} \langle \ell_1', \ell_2 \rangle}$ and $\dfrac{\ell_2 \overset{g:\alpha,D}{\rightsquigarrow_2} \ell_2'}{\langle \ell_1, \ell_2 \rangle \overset{g:\alpha,D}{\rightsquigarrow} \langle \ell_1, \ell_2' \rangle}$

Clock valuations

- A *clock valuation* v for set C of clocks is a function $v : C \longrightarrow \mathbb{R}_{\geqslant 0}$
  - assigning to each clock $x \in C$ its current value $v(x)$
- Clock valuation $v{+}d$ for $d \in \mathbb{R}_{\geqslant 0}$ is defined by:
  - $(v{+}d)(x) = v(x) + d$ for all clocks $x \in C$
- Clock valuation reset x in v for clock x is defined by:

$$(\text{reset } x \text{ in } v)(y) = \begin{cases} v(y) & \text{if } y \neq x \\ 0 & \text{if } y = x. \end{cases}$$

  - reset x in (reset y in v) is abbreviated by reset x, y in v

Timed automaton semantics

For timed automaton $\mathit{TA} = \big(\mathit{Loc}, \mathit{Act}, C, \rightsquigarrow, \mathit{Loc}_0, \mathit{inv}, \mathit{AP}, L\big)$:
state graph $S(\mathit{TA}) = (Q, Q_0, E, L')$ over AP' where:

- $Q = \mathit{Loc} \times \mathit{val}(C)$, state $s = \langle \ell, v \rangle$ for location ℓ and clock valuation v
- $Q_0 = \{ \langle \ell_0, v_0 \rangle \mid \ell_0 \in \mathit{Loc}_0 \ \wedge \ v_0(x) = 0 \text{ for all } x \in C \}$
- $\mathit{AP}' = \mathit{AP} \cup \mathit{ACC}(C)$
- $L'(\langle \ell, v \rangle) = L(\ell) \cup \{ g \in \mathit{ACC}(C) \mid v \models g \}$
- E is the edge set defined on the next slide

Timed automaton semantics

The edge set E consist of the following two types of transitions:

- Discrete transition: $\langle \ell, v \rangle \xrightarrow{\alpha} \langle \ell', v' \rangle$ if all following conditions hold:
  - there is an edge labeled $(g : \alpha, D)$ from location ℓ to ℓ' such that:
  - g is satisfied by v, i.e., $v \models g$
  - $v' = v$ with all clocks in D reset to 0, i.e., $v' =$ reset D in v
  - v' fulfills the invariant of location ℓ', i.e., $v' \models \mathit{inv}(\ell')$
- Delay transition: $\langle \ell, v \rangle \xrightarrow{d} \langle \ell, v{+}d \rangle$ for positive real d
  - if for any $0 \leqslant d' \leqslant d$ the invariant of ℓ holds for $v{+}d'$, i.e. $v{+}d' \models \mathit{inv}(\ell)$

Time divergence

- Let for any $t < d$, for fixed $d \in \mathbb{R}_{>0}$, clock valuation $\eta{+}t \models$ *inv*(ℓ)
- A possible execution fragment starting from the location ℓ is:

$$\langle \ell, \eta \rangle \xrightarrow{d_1} \langle \ell, \eta{+}d_1 \rangle \xrightarrow{d_2} \langle \ell, \eta{+}d_1{+}d_2 \rangle \xrightarrow{d_3} \langle \ell, \eta{+}d_1{+}d_2{+}d_3 \rangle \xrightarrow{d_4} \ldots$$

  – where $d_i > 0$ and the infinite sequence $d_1 + d_2 + \ldots$ *converges* towards d
  – such path fragments are called *time-convergent*
  ⇒ time advances only up to a certain value

- Time-convergent execution fragments are unrealistic and *ignored*
  – much like unfair paths (as we will see later on)

Time divergence

- Infinite path fragment π is *time-divergent* if $ExecTime(\pi) = \infty$
- The function $ExecTime : Act \cup \mathbb{R}_{>0} \rightarrow \mathbb{R}_{\geqslant 0}$ is defined as:

$$ExecTime(\tau) = \begin{cases} 0 & \text{if } \tau \in Act \\ d & \text{if } \tau = d \in \mathbb{R}_{>0} \end{cases}$$

- For infinite execution fragment $\rho = s_0 \xrightarrow{\tau_1} s_1 \xrightarrow{\tau_2} s_2 \ldots$ in $S(TA)$ let:

$$ExecTime(\rho) = \sum_{i=0}^{\infty} ExecTime(\tau_i)$$

  – for path fragment π in $S(TA)$ induced by ρ: $ExecTime(\pi) = ExecTime(\rho)$

- For state s in $S(TA)$: $Paths_{div}(s) = \{ \pi \in Paths(s) \mid \pi \text{ is time-divergent} \}$

Example: light switch

The path π in $S(Switch)$ in which on- and of-periods of one minute alternate:

$$\pi = \langle \mathit{off}, 0\rangle \langle \mathit{off}, 1\rangle \langle on, 0\rangle \langle on, 1\rangle \langle \mathit{off}, 1\rangle \langle \mathit{off}, 2\rangle \langle on, 0\rangle \langle on, 1\rangle \langle \mathit{off}, 1\rangle \ldots$$

is *time-divergent* as $ExecTime(\pi) = 1 + 1 + 1 + \ldots = \infty$.

The path:

$$\pi' = \langle \mathit{off}, 0\rangle \langle \mathit{off}, 1/2\rangle \langle \mathit{off}, 3/4\rangle \langle \mathit{off}, 7/8\rangle \langle \mathit{off}, 15/16\rangle \ldots$$

is *time-convergent*, since $ExecTime(\pi') = \sum_{i \geqslant 1} \left(\frac{1}{2}\right)^i = 1 < \infty$

Timelock

- State $s \in S(TA)$ contains a *timelock* if $Paths_{div}(s) = \varnothing$
  - there is no behavior in s where time can progress *ad infinitum*
  - clearly: any terminal state contains a timelock (but also non-terminal states may do)
  - terminal location does not necessarily yield a state with timelock (e.g. inv = true)
- *TA* is *timelock-free* if no state in *Reach*(*S*(*TA*)) contains a timelock
- Timelocks are considered as *modeling flaws* that should be avoided

Zenoness

- A *TA* that performs infinitely many actions in finite time is *Zeno*
- Path π in *S*(*TA*) is *Zeno* if:
  - it is time-convergent, and
  - infinitely many actions $\alpha \in$ *Act* are executed along π
- *TA* is *non-Zeno* if there does not exist an initial Zeno path in *S*(*TA*)
  - any π in *S*(*TA*) is time-divergent or
  - is time-convergent with nearly all (i.e., all except for finitely many) transitions being delay transitions
- Zeno paths are considered as *modeling flaws* that should be avoided

A sufficient criterion for Non-Zenoness

Let *TA* with set C of clocks such that for every control cycle:

$$\ell_0 \overset{g_1:\alpha_1,C_1}{\rightsquigarrow} \ell_1 \overset{g_2:\alpha_2,C_2}{\rightsquigarrow} \ldots \overset{g_n:\alpha_n,C_n}{\rightsquigarrow} \ell_n$$

there exists a clock $x \in C$ such that:

1. $x \in C_i$ for some $0 < i \leqslant n$, and
2. there exists a constant $c \in \mathbb{N}_{>0}$ such that for all clock evaluations η:

$$\eta(x) < c \text{ implies } (\eta \not\models g_j \text{ or } \eta \not\models \textit{inv}(\ell_j)), \text{ for some } 0 < j \leqslant n$$

Then: *TA* is *non-Zeno*

Timelock, time-divergence and Zenoness

- A timed automaton is only considered an adequate model of a time-critical system if it is:

  non-Zeno and timelock-free

- Time-convergent paths will be explicitly excluded from the analysis.

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 φ 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

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

Some abbreviations

- $\Diamond^J \Phi \;=\; \text{true}\, \mathsf{U}^J\, \Phi$
- $\exists\Box^J\,\Phi \;=\; \neg\forall\Diamond^J\,\neg\Phi$ and $\forall\Box^J\,\Phi \;=\; \neg\exists\Diamond^J\,\neg\Phi$
- $\Diamond\,\Phi = \Diamond^{[0,\infty)}\,\Phi$ and $\Box\,\Phi = \Box^{[0,\infty)}\,\Phi$

Semantics of TCTL

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

$s \models \text{true}$		
$s \models a$	iff	$a \in L(\ell)$
$s \models g$	iff	$\eta \models g$
$s \models \neg\,\Phi$	iff	not $s \models \Phi$
$s \models \Phi \wedge \Psi$	iff	$(s \models \Phi)$ and $(s \models \Psi)$
$s \models \exists\varphi$	iff	$\pi \models \varphi$ for some $\pi \in$ *Paths*$_{div}(s)$
$s \models \forall\varphi$	iff	$\pi \models \varphi$ for all $\pi \in$ *Paths*$_{div}(s)$

path quantification over time-divergent paths only

The $\Longrightarrow$ relation

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

$$s_0 \overset{d_0}{\Longrightarrow} s_1 \overset{d_1}{\Longrightarrow} s_2 \overset{d_2}{\Longrightarrow} \ldots \quad \text{with } d_0, d_1, d_2 \ldots \geqslant 0$$

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

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

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

Notation: $s_i{+}d = \langle \ell_i, \eta_i{+}d \rangle$ where $s_i = \langle \ell_i, \eta_i \rangle$.

Semantics of TCTL

For time-divergent path $\pi \in s_0 \overset{d_0}{\Longrightarrow} s_1 \overset{d_1}{\Longrightarrow} \ldots$:

$\pi \models \Phi \, \mathsf{U}^J \, \Psi$

iff

$\exists\, i \geqslant 0.\, s_i{+}d \models \Psi$ for some $d \in [0, d_i]$ with $\sum_{k=0}^{i-1} d_k + d \in J$

and

$\forall j \leqslant i.\, s_j{+}d' \models \Phi \vee \Psi$ for every $d' \in [0, d_j]$ with $\sum_{j=0}^{j-1} d_k + d' \leqslant \sum_{k=0}^{i-1} d_k + d$

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 $x \in C$