

Time Divergence, Timelock, and Zenoness

Lecture #16 of Advanced Model Checking

Joost-Pieter Katoen

Lehrstuhl 2: Software Modeling & Verification

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

January 17, 2011

Timed automata

- Timed automaton = finite-state automaton with **clock** variables
- Clocks take non-negative **real** values, i.e., in $\mathbb{R}_{\geq 0}$
- Clocks increase **implicitly**, i.e., clock updates are not allowed
- All clocks increase at the same **pace**, i.e., with rate one
- Clocks may only be inspected and reset to zero
- Boolean conditions on clocks are used as:
 - **guards** of edges: when is an edge enabled?
 - **invariants** of locations: how long is it allowed to stay?

Clock constraints

- A *clock constraint* over set C of clocks is formed according to:

$$g ::= x < c \mid x \leq c \mid x > c \mid x \geq c \mid g \wedge g \quad \text{where } c \in \mathbb{N} \text{ and } x \in C$$

- Let $CC(C)$ denote the set of clock constraints over C
- Clock constraints without any conjunctions are *atomic*
 - let $ACC(C)$ denote the set of atomic clock constraints over C

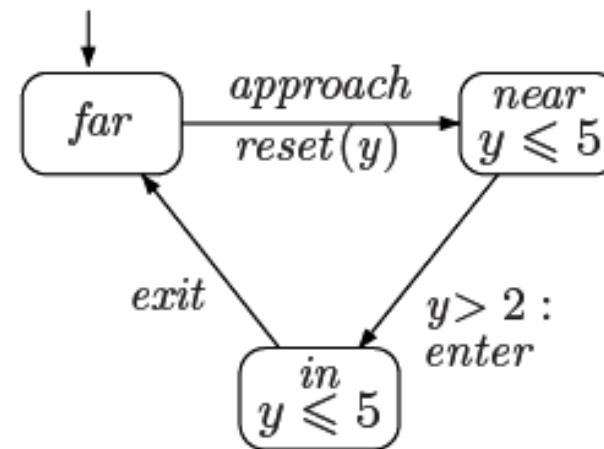
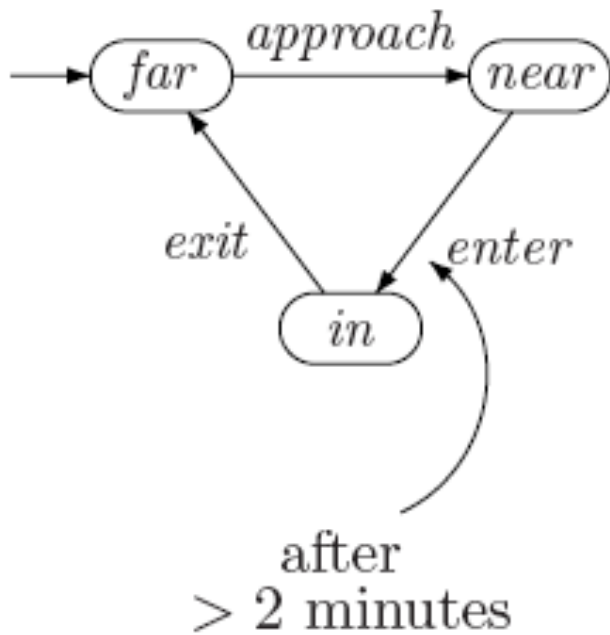
clock difference constraints such as $x - y < c$ can be added at expense of slightly more involved theory

Timed automaton

A *timed automaton* $TA = (Loc, Act, C, \hookrightarrow, Loc_0, Inv, AP, L)$ 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**
- $\hookrightarrow \subseteq Loc \times CC(C) \times Act \times 2^C \times Loc$ is a **transition relation**
- $Inv : Loc \rightarrow CC(C)$ is an **invariant-assignment** function, and
- $L : Loc \rightarrow 2^{AP}$ is a **labeling function**

Timed automata model of train



train is now also assumed to leave crossing within five time units

Clock valuations

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

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

- reset x in (reset y in η) is abbreviated by reset $\{x, y\}$ in η

Timed automaton semantics

For timed automaton $TA = (Loc, Act, C, \hookrightarrow, Loc_0, Inv, AP, L)$:

Transition system $TS(TA) = (S, Act', \rightarrow, I, AP', L')$ where:

- $S = Loc \times Eval(C)$, so states are of the form $s = \langle \ell, \eta \rangle$
- $Act' = Act \cup \mathbb{R}_{\geq 0}$, (discrete) actions and time-passage actions
- $I = \{ \langle \ell_0, \eta_0 \rangle \mid \ell_0 \in Loc_0 \wedge \eta_0(x) = 0 \text{ for all } x \in C \}$
- $AP' = AP \cup ACC(C)$
- $L'(\langle \ell, \eta \rangle) = L(\ell) \cup \{ g \in ACC(C) \mid \eta \models g \}$
- \hookrightarrow is defined on the next slide

Timed automaton semantics

The transition relation \rightarrow is defined by the following two rules:

- **Discrete** transition: $\langle \ell, \eta \rangle \xrightarrow{\alpha} \langle \ell', \eta' \rangle$ if all following conditions hold:
 - there is a transition labeled $(g : \alpha, D)$ from location ℓ to ℓ' such that:
 - g is satisfied by η , i.e., $\eta \models g$
 - $\eta' = \eta$ with all clocks in D reset to 0, i.e., $\eta' = \text{reset } D \text{ in } \eta$
 - η' fulfills the invariant of location ℓ' , i.e., $\eta' \models \text{Inv}(\ell')$
- **Delay** transition: $\langle \ell, \eta \rangle \xrightarrow{d} \langle \ell, \eta + d \rangle$ for $d \in \mathbb{R}_{\geq 0}$ if $\eta + d \models \text{Inv}(\ell)$

Example

Timed paths

Delays may be realized in $TS(TA)$ in uncountably many ways, e.g.:

$\langle off, 0 \rangle$		$\langle off, 1 \rangle$	$\langle on, 0 \rangle$		$\langle on, 2 \rangle$	$\langle off, 2 \rangle$...	
$\langle off, 0 \rangle$	$\langle off, 0.5 \rangle$	$\langle off, 1 \rangle$	$\langle on, 0 \rangle$		$\langle on, 1 \rangle$	$\langle on, 2 \rangle$	$\langle off, 2 \rangle$...
$\langle off, 0 \rangle$	$\langle off, 0.1 \rangle$	$\langle off, 1 \rangle$	$\langle on, 0 \rangle$	$\langle on, 0.53 \rangle$	$\langle on, 1.3 \rangle$	$\langle on, 2 \rangle$	$\langle off, 2 \rangle$...

The effect of $\langle \ell, \eta \rangle \xrightarrow{d_1+d_2} \langle \ell, \eta+d_1+d_2 \rangle$ corresponds to:

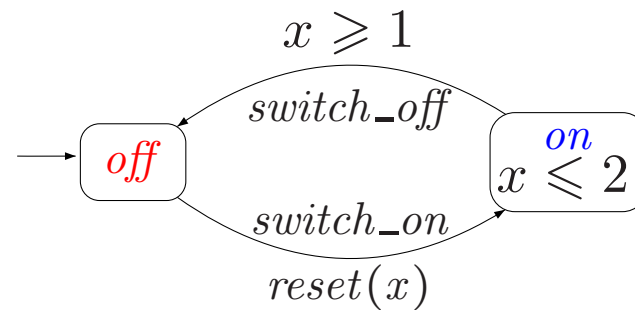
$$\langle \ell, \eta \rangle \xrightarrow{d_1} \langle \ell, \eta+d_1 \rangle \xrightarrow{d_2} \langle \ell, \eta+d_1+d_2 \rangle$$

Thus, uncountably many states of the form $\langle \ell, \eta+t \rangle$ with $0 \leq t \leq d_1+d_2$ are “visited”

Timed paths

- Paths through $TS(TA)$ model possible behaviours of TA
- But, not every path represents a **realistic** behaviour
- Some unrealistic phenomena that may occur:
 - **time convergence**: time converges to some value
 - **timelock**: the passage of time stops
 - **zenoness**: infinitely many actions take place in finite time
- Timelock and zenoness are **modeling flaws** and to be avoided
- Time-convergent paths will be excluded for model checking
 - they are treated similar as **unfair** paths in transition systems

Time divergence



The timed path:

$$\langle \textit{off}, 0 \rangle \xrightarrow{2^{-1}} \langle \textit{off}, 1 - 2^{-1} \rangle \xrightarrow{2^{-2}} \langle \textit{off}, 1 - 2^{-2} \rangle \xrightarrow{2^{-3}} \langle \textit{off}, 1 - 2^{-3} \rangle \dots$$

visits infinitely many states in the interval $[\frac{1}{2}, 1]$

Time divergence

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

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

- where $d_i > 0$ and the infinite sequence $d_1 + d_2 + \dots$ *converges* towards d
 - such path fragments are called *time-convergent*
- \Rightarrow 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 $ExecTime : Act \cup \mathbb{R}_{>0} \rightarrow \mathbb{R}_{\geq 0}$ 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 $\rho = s_0 \xrightarrow{\tau_1} s_1 \xrightarrow{\tau_2} s_2 \dots$ in $TS(TA)$ let:

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

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

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

Example: light switch

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

$$\pi = \langle off, 0 \rangle \langle off, 1 \rangle \langle on, 0 \rangle \langle on, 1 \rangle \langle off, 1 \rangle \langle off, 2 \rangle \langle on, 0 \rangle \langle on, 1 \rangle \langle off, 2 \rangle \dots$$

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

The path:

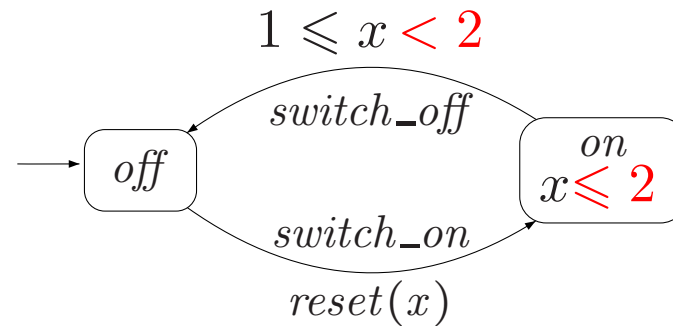
$$\pi' = \langle off, 0 \rangle \langle off, 1/2 \rangle \langle off, 3/4 \rangle \langle off, 7/8 \rangle \langle off, 15/16 \rangle \dots$$

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

Timelock

- State $s \in TS(TA)$ contains a *timelock* if $Paths_{div}(s) = \emptyset$
 - there is no behavior in s where time can progress *ad infinitum*
 - 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 $Reach(TS(TA))$ contains a timelock
- Timelocks are considered as *modeling flaws* that should be avoided
 - like deadlocks, we need mechanisms to check their presence

A **non** timelock-free timed automaton

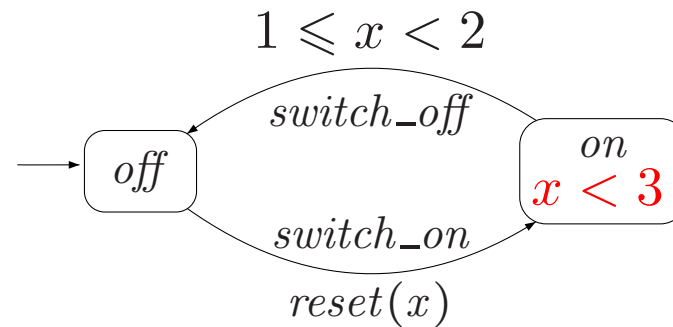


State $\langle on, 2 \rangle$ is reachable in transition system $TS(TA)$, e.g., via:

$$\langle off, 0 \rangle \xrightarrow{switch_on} \langle on, 0 \rangle \xrightarrow{2} \langle on, 2 \rangle$$

As $\langle on, 2 \rangle$ is a terminal state, $Paths_{div}(\langle on, 2 \rangle) = \emptyset$

Another **non** timelock-free timed automaton



State $\langle on, 2 \rangle$ is not terminal, , e.g., the time-convergent path in:

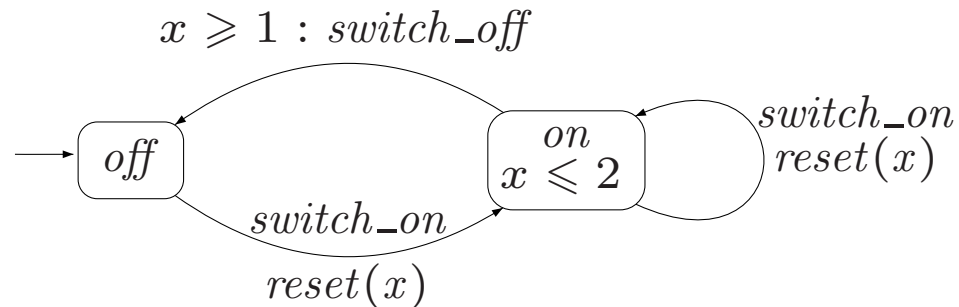
$\langle on, 2 \rangle \langle on, 2.9 \rangle \langle on, 2.99 \rangle \langle on, 2.999 \rangle \langle on, 2.9999 \rangle \dots$

emanates from it. But, $Paths_{div}(\langle on, 2 \rangle) = \emptyset$

Zenoness

- A TA that performs infinitely many actions in finite time is *zeno*
- Path π in $TS(TA)$ 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 a zeno path in $TS(TA)$
 - any π in $TS(TA)$ 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
 - like timelocks (and deadlocks), we need mechanisms to check zenoness
 - this, however, turns out to be difficult \Rightarrow resort to *sufficient* conditions

Zeno paths of a (yet another) light switch



The paths induced by the following execution fragments:

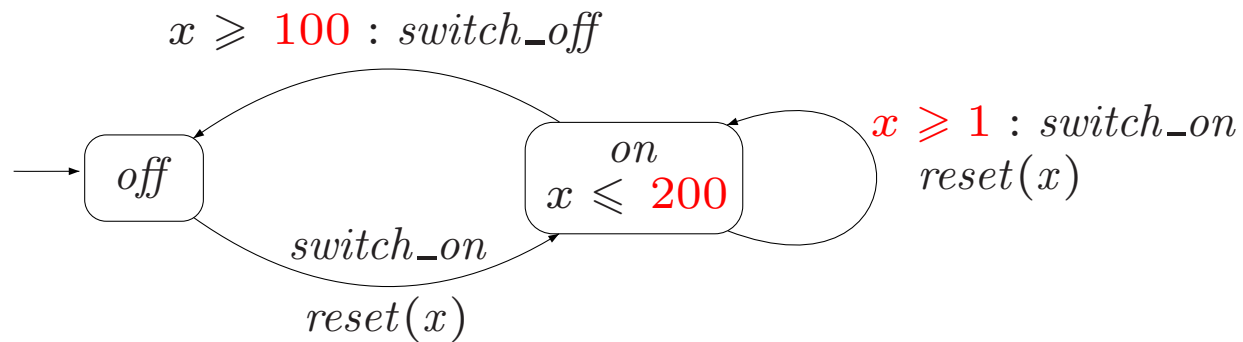
$$\langle \textit{off}, 0 \rangle \xrightarrow{\textit{sw_on}} \langle \textit{on}, 0 \rangle \xrightarrow{\textit{sw_on}} \langle \textit{on}, 0 \rangle \xrightarrow{\textit{sw_on}} \langle \textit{on}, 0 \rangle \xrightarrow{\textit{sw_on}} \dots$$

$$\langle \textit{off}, 0 \rangle \xrightarrow{\textit{sw_on}} \langle \textit{on}, 0 \rangle \xrightarrow{0.5} \langle \textit{on}, 0.5 \rangle \xrightarrow{\textit{sw_on}} \langle \textit{on}, 0 \rangle \xrightarrow{0.25} \langle \textit{on}, 0.25 \rangle \xrightarrow{\textit{sw_on}} \dots$$

are **zeno** paths during which the user presses the on button faster and faster

avoid by imposing a minimal delay, e.g., $\frac{1}{100}$, between successive on's

A non-zero variant



Strong zenoness

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

$$l_0 \xrightarrow{g_1:\alpha_1,C_1} l_1 \xrightarrow{g_2:\alpha_2,C_2} \dots \xrightarrow{g_n:\alpha_n,C_n} l_n = l_0$$

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

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

$$\eta(x) < c \quad \text{implies} \quad (\exists 0 < j \leq n. \eta \not\models g_j \quad \text{or} \quad \eta \not\models \text{Inv}(l_j))$$

Then: TA is **non-zeno**

Proof

Example

Timelock, time-divergence and zenoness

- A timed automaton is adequately modeling a time-critical system whenever it is:

 non-zeno and **timelock-free**
- Time-divergent paths will be explicitly excluded for analysis purposes