Real-Time Systems

Lecture 13: Location Reachability

(or: The Region Automaton)

2017-12-14

Dr. Bernd Westphal

Albert-Ludwigs-Universität Freiburg, Germany

Content

Introduction

- **Observables and Evolutions**
- **Duration Calculus** (DC)
- Semantical Correctness Proofs
- DC Decidability
- DC Implementables
- **PLC-Automata**

$obs : \mathsf{Time} \to \mathscr{D}(obs)$

- **Timed Automata** (TA), Uppaal ✓
- Networks of Timed Automata ✓
- Region/Zone-Abstraction 21.12.
- TA model-checking
- Extended Timed Automata } 9.1.
- Undecidability Results

$\langle obs_0, \nu_0 \rangle, t_0 \xrightarrow{\lambda_0} \langle obs_1, \nu_1 \rangle, t_1 \ldots$

- **Automatic Verification**...
  ...whether a TA satisfies a DC formula, observer-based
- **Recent Results**:
  - **Timed Sequence Diagrams**, or **Quasi-equal Clocks**, or **Automatic Code Generation**, or ...

Content

- The **Location Reachability Problem**
- ...**is decidable** for TA:
  - **Normalised Constants**
  - **Time Abstract Transition System**
  - **Regions**:
    - Equivalence Classes of Clock Valuations
  - The **Region Automaton**
    - ...is finite
    - ...and effectively constructable.
- The **Constraint Reachability Problem**
  - ...is decidable as well.

The Location Reachability Problem

Given: A timed automaton $\mathcal{A}$ and one of its locations ℓ.

Question: Is ℓ **reachable**?

That is, is there a transition sequence of the form

$$\langle \ell_{ini}, \nu_0 \rangle \xrightarrow{\lambda_1} \langle \ell_1, \nu_1 \rangle \xrightarrow{\lambda_2} \langle \ell_2, \nu_2 \rangle \xrightarrow{\lambda_3} \dots \xrightarrow{\lambda_n} \langle \ell_n, \nu_n \rangle \text{ with } \ell_n = \ell$$

in the labelled transition system $\mathcal{T}(\mathcal{A})$?

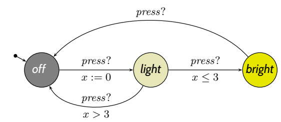
- **Note:** Decidability is not soo obvious, recall that
  - clocks range over real numbers, thus infinitely many configurations,
  - at each configuration, uncountably many transitions $\xrightarrow{t}$ may originate
- **Consequence:** The timed automata as we consider them here cannot encode a 2-counter machine, and they are strictly less expressive than DC.

Decidability of Location Reachability for TA

Decidability of The Location Reachability Problem

Claim: (**Theorem 4.33**)

The location reachability problem is **decidable** for timed automata.



Approach: Constructive proof.

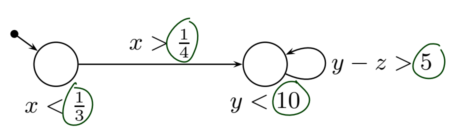
- Observe: clock constraints are **simple**
  – w.l.o.g. assume constants $c \in \mathbb{N}_0$.
- **Def. 4.19**: **time-abstract transition system** $\mathcal{U}(\mathcal{A})$ – abstracts from uncountably many delay transitions, still infinite-state.
- **Lemma 4.20**: location reachability of $\mathcal{A}$ is **preserved** in $\mathcal{U}(\mathcal{A})$.
- **Def. 4.29**: **region automaton** $\mathcal{R}(\mathcal{A})$ – equivalent configurations collapse into regions
- **Lemma 4.32**: location reachability of $\mathcal{U}(\mathcal{A})$ is **preserved** in $\mathcal{R}(\mathcal{A})$.
- **Lemma 4.28**: $\mathcal{R}(\mathcal{A})$ is **finite**.

Without Loss of Generality: Natural Constants

Recall: $\varphi ::= x \sim c \mid x - y \sim c \mid \varphi \wedge \varphi, \quad x, y \in X, \quad c \in \mathbb{Q}_0^+$, and $\sim \in \{<, >, \leq, \geq\}$.

- Let $C(\mathcal{A}) = \{c \in \mathbb{Q}_0^+ \mid c \text{ appears in } \mathcal{A}\}$ – $C(\mathcal{A})$ is **finite**! (Why?)
- Let $t_\mathcal{A}$ be the **least common multiple of the denominators** in $C(\mathcal{A})$.
- Let $t_\mathcal{A} \cdot \mathcal{A}$ be the TA obtained from $\mathcal{A}$ by **multiplying** all constants by $t_\mathcal{A}$.

$\mathcal{A}$:



$C(\mathcal{A}) = \{\frac{1}{4}, \frac{1}{3}, 10, 5\}$

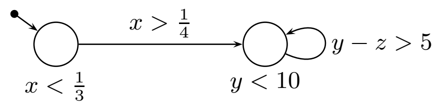
$t_\mathcal{A} = 12$

Without Loss of Generality: Natural Constants

Recall: $\varphi ::= x \sim c \mid x - y \sim c \mid \varphi \wedge \varphi,\ \ x, y \in X,\ \ c \in \mathbb{Q}_0^+$, and $\sim \in \{<, >, \leq, \geq\}$.

- Let $C(\mathcal{A}) = \{c \in \mathbb{Q}_0^+ \mid c$ appears in $\mathcal{A}\}$ – $C(\mathcal{A})$ is **finite**! (Why?)
- Let $t_{\mathcal{A}}$ be the **least common multiple of the denominators** in $C(\mathcal{A})$.
- Let $t_{\mathcal{A}} \cdot \mathcal{A}$ be the TA obtained from $\mathcal{A}$ by **multiplying** all constants by $t_{\mathcal{A}}$.

$\mathcal{A}$:



$C(\mathcal{A}) = \{\frac{1}{3}, \frac{1}{4}, 5, 10\}$

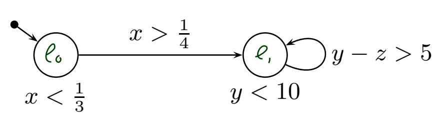
$t_{\mathcal{A}} = 12$

Without Loss of Generality: Natural Constants

Recall: $\varphi ::= x \sim c \mid x - y \sim c \mid \varphi \wedge \varphi,\ \ x, y \in X,\ \ c \in \mathbb{Q}_0^+$, and $\sim \in \{<, >, \leq, \geq\}$.

- Let $C(\mathcal{A}) = \{c \in \mathbb{Q}_0^+ \mid c$ appears in $\mathcal{A}\}$ – $C(\mathcal{A})$ is **finite**! (Why?)
- Let $t_{\mathcal{A}}$ be the **least common multiple of the denominators** in $C(\mathcal{A})$.
- Let $t_{\mathcal{A}} \cdot \mathcal{A}$ be the TA obtained from $\mathcal{A}$ by **multiplying** all constants by $t_{\mathcal{A}}$.

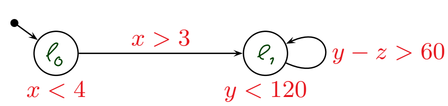
$\mathcal{A}$:



$C(\mathcal{A}) = \{\frac{1}{3}, \frac{1}{4}, 5, 10\}$

$t_{\mathcal{A}} = 12$

$t_{\mathcal{A}} \cdot \mathcal{A}$:



$c_x = 4$
$c_y = 120$
$c_z = 60$

Without Loss of Generality: Natural Constants

Recall: $\varphi ::= x \sim c \mid x - y \sim c \mid \varphi \wedge \varphi,\ \ x, y \in X,\ \ c \in \mathbb{Q}_0^+$, and $\sim \in \{<, >, \leq, \geq\}$.

- Let $C(\mathcal{A}) = \{c \in \mathbb{Q}_0^+ \mid c \text{ appears in } \mathcal{A}\}$ – $C(\mathcal{A})$ is **finite**! (Why?)
- Let $t_\mathcal{A}$ be the **least common multiple of the denominators** in $C(\mathcal{A})$.
- Let $t_\mathcal{A} \cdot \mathcal{A}$ be the TA obtained from $\mathcal{A}$ by **multiplying** all constants by $t_\mathcal{A}$.
- **Then**:
  - $C(t_\mathcal{A} \cdot \mathcal{A}) \subset \mathbb{N}_0$.
  - A location ℓ is reachable in $t_\mathcal{A} \cdot \mathcal{A}$ if and only if ℓ is reachable in $\mathcal{A}$.
- **That is**: we can, **without loss of generality**, in the following consider only timed automata $\mathcal{A}$ with $C(\mathcal{A}) \subset \mathbb{N}_0$.

Definition. Let x be a clock of timed automaton $\mathcal{A}$ (with $C(\mathcal{A}) \subset \mathbb{N}_0$).
We denote by $c_x \in \mathbb{N}_0$ the **largest time constant** c that appears together with x in a constraint of $\mathcal{A}$.

– 13 – 2017-12-14 – Snatc –

Decidability of The Location Reachability Problem

Claim: (**Theorem 4.33**)

The location reachability problem is **decidable** for timed automata.

Approach: Constructive proof.

✔ Observe: clock constraints are **simple** – w.l.o.g. assume constants $c \in \mathbb{N}_0$.

✘ **Def. 4.19**: **time-abstract transition system** $\mathcal{U}(\mathcal{A})$ – abstracts from uncountably many delay transitions, still infinite-state.

✘ **Lemma 4.20**: location reachability of $\mathcal{A}$ is **preserved** in $\mathcal{U}(\mathcal{A})$.

✘ **Def. 4.29**: **region automaton** $\mathcal{R}(\mathcal{A})$ – equivalent configurations collapse into regions

✘ **Lemma 4.32**: location reachability of $\mathcal{U}(\mathcal{A})$ is **preserved** in $\mathcal{R}(\mathcal{A})$.

✘ **Lemma 4.28**: $\mathcal{R}(\mathcal{A})$ is **finite**.

– 13 – 2017-12-14 – Stats –

Helper: Relational Composition

Recall: $\mathcal{T}(\mathcal{A}) = (\mathit{Conf}(\mathcal{A}), \mathsf{Time} \cup B_{?!}, \{\xrightarrow{\lambda} \mid \lambda \in \mathsf{Time} \cup B_{?!}\}, C_{ini})$

- Note: The $\xrightarrow{\lambda}$ are binary relations on configurations.

$r_1 \subseteq A \times B$
$r_2 \subseteq B \times C$
$r_1 \circ r_2 \subseteq A \times C$

Definition. Let $\mathcal{A}$ be a TA. For all $\langle \ell_1, \nu_1 \rangle, \langle \ell_2, \nu_2 \rangle \in \mathit{Conf}(\mathcal{A})$,

$$\langle \ell_1, \nu_1 \rangle \xrightarrow{\lambda_1} \circ \xrightarrow{\lambda_2} \langle \ell_2, \nu_2 \rangle$$

if and only if there **exists some** $\langle \ell', \nu' \rangle \in \mathit{Conf}(\mathcal{A})$ such that

$$\langle \ell_1, \nu_1 \rangle \xrightarrow{\lambda_1} \langle \ell', \nu' \rangle \text{ and } \langle \ell', \nu' \rangle \xrightarrow{\lambda_2} \langle \ell_2, \nu_2 \rangle.$$

Remark. The following property of **time additivity** holds.

$$\forall\, t_1, t_2 \in \mathsf{Time} : \xrightarrow{t_1} \circ \xrightarrow{t_2} \; = \; \xrightarrow{t_1 + t_2}$$

Time-abstract Transition System

Definition 4.19. [*Time-abstract transition system*]
Let $\mathcal{A}$ be a timed automaton.
The **time-abstract transition system** $\mathcal{U}(\mathcal{A})$ is obtained from $\mathcal{T}(\mathcal{A})$ (Def. 4.4) by taking

$$\mathcal{U}(\mathcal{A}) = (\mathit{Conf}(\mathcal{A}), B_{?!}, \{\stackrel{\alpha}{\Longrightarrow} \mid \alpha \in B_{?!}\}, C_{ini})$$

where

$$\stackrel{\alpha}{\Longrightarrow} \subseteq \mathit{Conf}(\mathcal{A}) \times \mathit{Conf}(\mathcal{A})$$

is defined as follows: Let $\langle \ell, \nu \rangle, \langle \ell', \nu' \rangle \in \mathit{Conf}(\mathcal{A})$ be configurations of $\mathcal{A}$ and $\alpha \in B_{?!}$ an action. Then

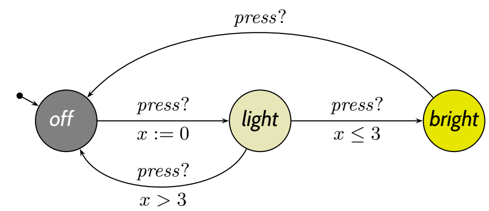
$$\langle \ell, \nu \rangle \stackrel{\alpha}{\Longrightarrow} \langle \ell', \nu' \rangle$$

if and only if there exists $t \in \mathsf{Time}$ such that

$$\langle \ell, \nu \rangle \xrightarrow{t} \circ \xrightarrow{\alpha} \langle \ell', \nu' \rangle.$$

Example

$\langle \ell,\nu\rangle \xRightarrow{\alpha} \langle \ell',\nu'\rangle$ iff $\exists\, t \in \mathsf{Time} \bullet \langle \ell,\nu\rangle \xrightarrow{t} \circ \xrightarrow{\alpha} \langle \ell',\nu'\rangle$



- $\langle \textit{light}, x=0\rangle \xRightarrow{press?} \langle \textit{off}, x=27\rangle$ YES, with $t=27$ we have $\langle l,0\rangle \overbrace{\xrightarrow{27} \langle l,27\rangle \xrightarrow{press?}}^{\xrightarrow{27}\circ\xrightarrow{press?}} \langle o,27\rangle$
- $\langle \textit{off}, x=4\rangle \xRightarrow{press?} \langle \textit{light}, x=0\rangle$ YES, any $t\in\mathbb{R}_0^+$ works
- $\langle \textit{off}, x=4\rangle \xRightarrow{press?} \langle \textit{light}, x=1\rangle$ NO, $\langle o,4\rangle \xrightarrow{t} \circ \xrightarrow{press?} \langle l,t'\rangle$ implies $t'=0$
- $\langle \textit{off}, x=0\rangle \xRightarrow{press?} \langle \cancel{\textit{light}}\ \textit{off}, x=5\rangle$ NO, no α s.t. $\langle o,5\rangle \xrightarrow{\alpha} \langle o,5\rangle$
- $\langle \textit{off}, x=0\rangle \xRightarrow{press?} \langle \textit{bright}, x=5\rangle$ NO, needs two actions
- $\langle \textit{light}, x=1\rangle \xRightarrow{press?} \langle \textit{bright}, x=1\rangle$ YES, with $t=0$

Location Reachability is preserved in $\mathcal{U}(\mathcal{A})$

Lemma 4.20. For all locations ℓ of a given timed automaton $\mathcal{A}$ the following holds:

ℓ is ($\xrightarrow{\lambda}$-)reachable in $\mathcal{T}(\mathcal{A})$ if and only if ℓ is ($\xRightarrow{\alpha}$-)reachable in $\mathcal{U}(\mathcal{A})$.

Proof:

$\xrightarrow{t_{o_1}} \circ \cdots \circ \xrightarrow{t_{o_{n_0}}} \;=\; \xrightarrow{t_{o_1}+\cdots+t_{o_{n_0}}} \qquad \xRightarrow{\alpha_1}$

- "$\Longleftarrow$": easy
- "$\Longrightarrow$": ℓ is reachable in $\mathcal{T}(\mathcal{A})$

  iff $\langle \ell_0,\nu_0\rangle \xrightarrow{t_{0_1}} \langle \ell_{0_1},\nu_{0_1}\rangle \xrightarrow{t_{0_2}} \langle \ell_{0_2},\nu_{0_2}\rangle \xrightarrow{t_{0_3}} \cdots \xrightarrow{t_{0_{n_0}}} \langle \ell_{0_{n_0}},\nu_{0_{n_0}}\rangle \xrightarrow{\alpha_1} \langle \ell_1,\nu_1\rangle$

  $\xrightarrow{t_{1_1}} \langle \ell_{1_1},\nu_{1_1}\rangle \xrightarrow{t_{1_2}} \cdots \xrightarrow{\alpha_2} \langle \ell_2,\nu_2\rangle$

  $\vdots$

  $\xrightarrow{t_{m_1}} \langle \ell_{m_1},\nu_{m_1}\rangle \xrightarrow{t_{m_2}} \cdots \xrightarrow{\alpha_{m+1}} \langle \ell,\nu_{m+1}\rangle$

Location Reachability is preserved in $\mathcal{U}(\mathcal{A})$

Lemma 4.20. For all locations ℓ of a given timed automaton $\mathcal{A}$ the following holds:

ℓ is ($\xrightarrow{\lambda}$-)reachable in $\mathcal{T}(\mathcal{A})$ if and only if ℓ is ($\xRightarrow{\alpha}$-)reachable in $\mathcal{U}(\mathcal{A})$.

Proof:

- "$\Longleftarrow$": easy
- "$\Longrightarrow$": ℓ is reachable in $\mathcal{T}(\mathcal{A})$

$n_0 \in \mathbb{N}_0$, *i.e. sequence may be empty*

$$\text{iff} \quad \langle \ell_0, \nu_0 \rangle \underbrace{\xrightarrow{t_{0_1}} \langle \ell_{0_1}, \nu_{0_1} \rangle \xrightarrow{t_{0_2}} \langle \ell_{0_2}, \nu_{0_2} \rangle \xrightarrow{t_{0_3}} \dots \xrightarrow{t_{0_{n_0}}}}_{t_1 := \sum_{i=1}^{n_0} t_{0_i}} \langle \ell_{0_{n_0}}, \nu_{0_{n_0}} \rangle \xrightarrow{\alpha_1} \langle \ell_1, \nu_1 \rangle$$

$$\xrightarrow{t_{1_1}} \langle \ell_{1_1}, \nu_{1_1} \rangle \xrightarrow{t_{1_2}} \dots \qquad \xrightarrow{\alpha_2} \langle \ell_2, \nu_2 \rangle$$

$$\vdots$$

$$\xrightarrow{t_{m_1}} \langle \ell_{m_1}, \nu_{m_1} \rangle \xrightarrow{t_{m_2}} \dots \qquad \xrightarrow{\alpha_{m+1}} \langle \ell, \nu_{m+1} \rangle$$

by $\xrightarrow{t_2} \circ \xrightarrow{\alpha_2}$

$$\text{implies } \langle \ell_0, \nu_0 \rangle \xRightarrow{\alpha_1} \langle \ell_1, \nu_1 \rangle \xRightarrow{\alpha_2} \dots \xRightarrow{\alpha_{m+1}} \langle \ell, \nu_{m+1} \rangle$$

Decidability of The Location Reachability Problem

Claim: (**Theorem 4.33**)

The location reachability problem is **decidable** for timed automata.

Approach: Constructive proof.

✔ Observe: clock constraints are **simple** – w.l.o.g. assume constants $c \in \mathbb{N}_0$.

✔ **Def. 4.19**: **time-abstract transition system** $\mathcal{U}(\mathcal{A})$ – abstracts from uncountably many delay transitions, still infinite-state.

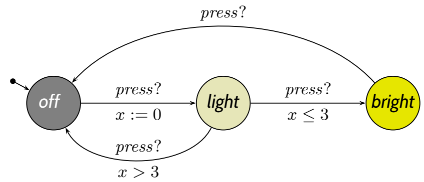
✔ **Lemma 4.20**: location reachability of $\mathcal{A}$ is **preserved** in $\mathcal{U}(\mathcal{A})$.

✘ **Def. 4.29**: **region automaton** $\mathcal{R}(\mathcal{A})$ – equivalent configurations collapse into regions

✘ **Lemma 4.32**: location reachability of $\mathcal{U}(\mathcal{A})$ is **preserved** in $\mathcal{R}(\mathcal{A})$.

✘ **Lemma 4.28**: $\mathcal{R}(\mathcal{A})$ is **finite**.

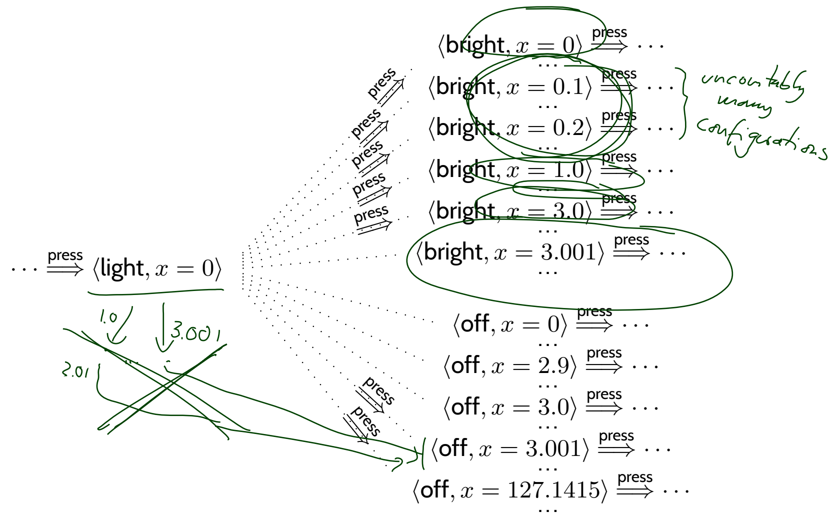
Indistinguishable Configurations



$\varphi ::= x \sim c \mid x - y \sim c \mid \varphi \wedge \varphi$

$x \geq 0$
$x > 0$
$x < 1$
$x \leq 1$

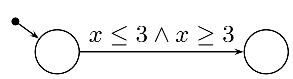
$\mathcal{U}(\mathcal{A})$:

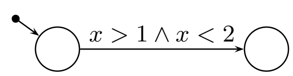


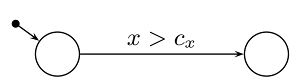
Distinguishing Clock Valuations: One Clock

- Assume $\mathcal{A}$ with only a single clock, i.e. $X = \{x\}$ (**recall**: $C(\mathcal{A}) \subset \mathbb{N}$).
  - $\mathcal{A}$ **could detect**, for a given ν, whether $\nu(x) \in \{0, \dots, c_x\}$.
  - $\mathcal{A}$ **cannot distinguish** ν_1 and ν_2 if $\nu_i(x) \in (k, k+1)$, $i = 1, 2$, and $k \in \{0, \dots, c_x - 1\}$.

    open interval
  - $\mathcal{A}$ **cannot distinguish** ν_1 and ν_2 if $\nu_i(x) > c_x$, $i = 1, 2$.







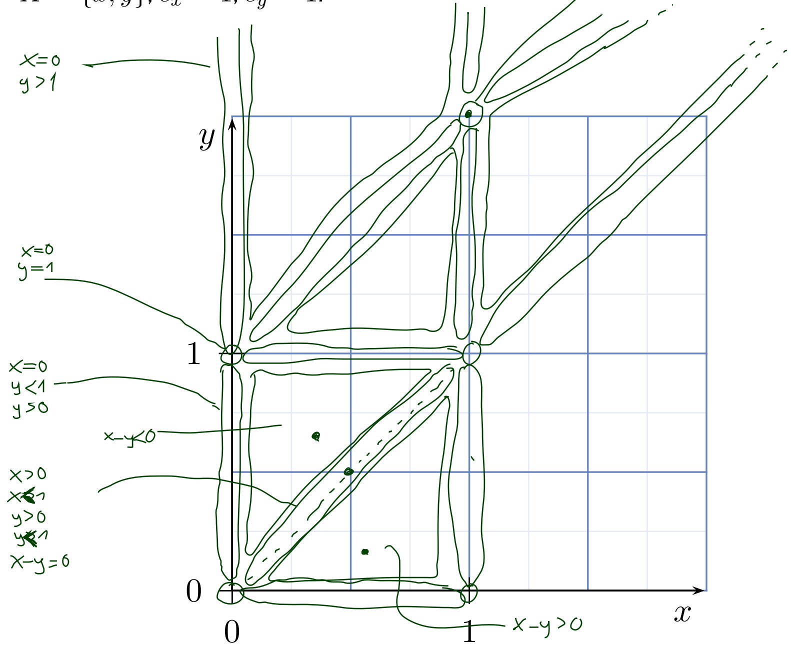
- If $c_x \geq 1$, there are $(2c_x + 2)$ **equivalence classes**:

$$\{\{0\}, (0,1), \{1\}, (1,2), \dots, \{c_x\}, (c_x, \infty)\}$$

  If $\nu_1(x)$ and $\nu_2(x)$ are in the **same** equivalence class, then ν_1 and ν_2 are **indistiguishable** by $\mathcal{A}$.

$\varphi ::= x \sim c$
$\quad x - y \sim c$
$\quad \varphi \wedge \varphi$

- $X = \{x, y\}$, $c_x = 1$, $c_y = 1$.



Helper: Floor and Fraction

- **Recall**:

  Each $q \in \mathbb{R}_0^+$ can be split into
  - **floor** $\lfloor q \rfloor \in \mathbb{N}_0$ and
  - **fraction** $frac(q) \in [0, 1)$ open interval

  such that
$$q = \lfloor q \rfloor + frac(q).$$

$\lfloor 3.14 \rfloor = 3$
$frac(3.14) = 0.14$

An Equivalence-Relation on Valuations

Definition. Let X be a set of clocks, $c_x \in \mathbb{N}_0$ for each clock $x \in X$, and ν_1, ν_2 clock valuations of X.

We set $\nu_1 \cong \nu_2$ if and only if the following **four** conditions are satisfied:

(1) For all $x \in X$, $\lfloor \nu_1(x) \rfloor = \lfloor \nu_2(x) \rfloor$ or **both** $\nu_1(x) > c_x$ and $\nu_2(x) > c_x$.

(2) For all $x \in X$ with $\nu_1(x) \leq c_x$,

$$\mathit{frac}(\nu_1(x)) = 0 \text{ if and only if } \mathit{frac}(\nu_2(x)) = 0.$$

(3) For all $x, y \in X$,

$$\lfloor \nu_1(x) - \nu_1(y) \rfloor = \lfloor \nu_2(x) - \nu_2(y) \rfloor$$

$$\text{or both } |\nu_1(x) - \nu_1(y)| > c \text{ and } |\nu_2(x) - \nu_2(y)| > c.$$

(4) For all $x, y \in X$ with $-c \leq \nu_1(x) - \nu_1(y) \leq c$,

$$\mathit{frac}(\nu_1(x) - \nu_1(y)) = 0 \text{ if and only if } \mathit{frac}(\nu_2(x) - \nu_2(y)) = 0.$$

Where $c = \max\{c_x, c_y\}$.

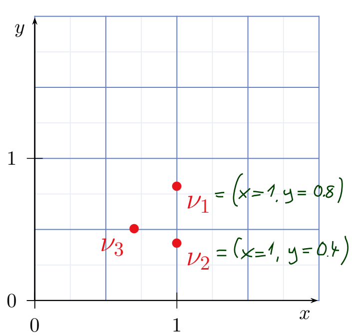
Example: Regions

(1) $\forall x \in X \bullet \lfloor \nu_1(x) \rfloor = \lfloor \nu_2(x) \rfloor \vee (\nu_1(x) > c_x \wedge \nu_2(x) > c_x)$

(2) $\forall x \in X \bullet \nu_1(x) \leq c_x \implies (\mathit{frac}(\nu_1(x)) = 0 \iff \mathit{frac}(\nu_2(x)) = 0)$

(3) $\forall x, y \in X \bullet \lfloor \nu_1(x) - \nu_1(y) \rfloor = \lfloor \nu_2(x) - \nu_2(y) \rfloor$
$\vee (|\nu_1(x) - \nu_1(y)| > c \wedge |\nu_2(x) - \nu_2(y)| > c)$

(4) $\forall x, y \in X \bullet -c \leq \nu_1(x) - \nu_1(y) \leq c$
$\implies (\mathit{frac}(\nu_1(x) - \nu_1(y)) = 0 \iff \mathit{frac}(\nu_2(x) - \nu_2(y)) = 0)$



$\nu_1 \cong \nu_2$ **because**

- $\lfloor \nu_1(x) \rfloor = \lfloor 1 \rfloor = 1 = \lfloor 1 \rfloor = \lfloor \nu_2(x) \rfloor$
  $\lfloor \nu_1(y) \rfloor = \lfloor 0.8 \rfloor = 0 = \lfloor 0.4 \rfloor = \lfloor \nu_2(y) \rfloor$
- $\mathit{frac}(\nu_1(x)) = 0 = \mathit{frac}(\nu_2(x))$
  $\mathit{frac}(\nu_1(y)) = \mathit{frac}(0.8) = 0.8 \neq 0$
  $\mathit{frac}(\nu_2(y)) = \mathit{frac}(0.4) = 0.4 \neq 0$
- $\lfloor \nu_1(x) - \nu_1(y) \rfloor = \lfloor 1 - 0.8 \rfloor = 0$
  $= \lfloor 1 - 0.4 \rfloor = \lfloor \nu_2(x) - \nu_2(y) \rfloor$
- ...

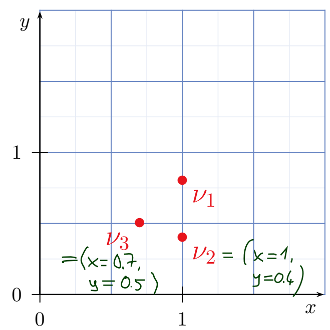
Example: Regions

(1) $\forall x \in X \bullet \lfloor \nu_1(x) \rfloor = \lfloor \nu_2(x) \rfloor \vee (\nu_1(x) > c_x \wedge \nu_2(x) > c_x)$

(2) $\forall x \in X \bullet \nu_1(x) \leq c_x \implies (\mathit{frac}(\nu_1(x)) = 0 \iff \mathit{frac}(\nu_2(x)) = 0)$

(3) $\forall x, y \in X \bullet \lfloor \nu_1(x) - \nu_1(y) \rfloor = \lfloor \nu_2(x) - \nu_2(y) \rfloor$
$\vee (|\nu_1(x) - \nu_1(y)| > c \wedge |\nu_2(x) - \nu_2(y)| > c)$

(4) $\forall x, y \in X \bullet -c \leq \nu_1(x) - \nu_1(y) \leq c$
$\implies (\mathit{frac}(\nu_1(x) - \nu_1(y)) = 0 \iff \mathit{frac}(\nu_2(x) - \nu_2(y)) = 0)$



$\nu_1 \cong \nu_2$ **because**

- $\lfloor \nu_1(x) \rfloor = \lfloor 1 \rfloor = 1 = \lfloor 1 \rfloor = \lfloor \nu_2(x) \rfloor$
  $\lfloor \nu_1(y) \rfloor = \lfloor 0.8 \rfloor = 0 = \lfloor 0.4 \rfloor = \lfloor \nu_2(y) \rfloor$
- $\mathit{frac}(\nu_1(x)) = 0 = \mathit{frac}(\nu_2(x))$
  $\mathit{frac}(\nu_1(y)) = \mathit{frac}(0.8) = 0.8 \neq 0$
  $\mathit{frac}(\nu_2(y)) = \mathit{frac}(0.4) = 0.4 \neq 0$
- $\lfloor \nu_1(x) - \nu_1(y) \rfloor = \lfloor 1 - 0.8 \rfloor = 0$
  $= \lfloor 1 - 0.4 \rfloor = \lfloor \nu_2(x) - \nu_2(y) \rfloor$
- …

$\nu_2 \not\cong \nu_3$ **because**

- $\lfloor \nu_2(x) \rfloor = \lfloor 1 \rfloor = 1$
  $\lfloor \nu_3(x) \rfloor = \lfloor 0.7 \rfloor = 0$

Regions

Proposition. $\cong$ is an equivalence relation.

Definition 4.27.
For a given valuation ν we denote by $[\nu]$ the equivalence class of ν.

We call the equivalence classes of $\cong$ regions.

The Region Automaton

Definition 4.29. [*Region Automaton*] The **region automaton** $\mathcal{R}(\mathcal{A})$ of the timed automaton $\mathcal{A}$ is the labelled transition system

$$\mathcal{R}(\mathcal{A}) = (\ Conf(\mathcal{R}(\mathcal{A})),\ B_{?!},\ \{\xrightarrow{\alpha}_{R(\mathcal{A})} \mid \alpha \in B_{?!}\},\ C_{ini}\ )$$

where

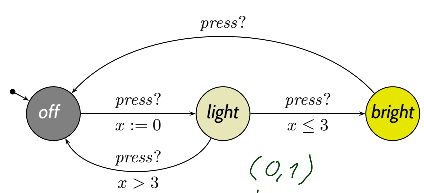
- $Conf(\mathcal{R}(\mathcal{A})) = \{\langle \ell, [\nu] \rangle \mid \ell \in L, \nu : X \to \mathsf{Time}, \nu \models I(\ell)\}$,
- for each $\alpha \in B_{?!}$,

$$\langle \ell, [\nu] \rangle \xrightarrow{\alpha}_{R(\mathcal{A})} \langle \ell', [\nu'] \rangle \text{ if and only if } \langle \ell, \nu \rangle \xRightarrow{\alpha} \langle \ell', \nu' \rangle$$

  in $\mathcal{U}(\mathcal{A})$, and
- $C_{ini} = \{\langle \ell_{ini}, [\nu_{ini}] \rangle\} \cap Conf(\mathcal{R}(\mathcal{A}))$ with $\nu_{ini}(X) = \{0\}$.

Proposition. The transition relation of $\mathcal{R}(\mathcal{A})$ is **well-defined**, that is, independent of the choice of the representative ν of a region $[\nu]$.

Example: Region Automaton



$\mathcal{R}(\mathcal{A})$:

~~$\mathcal{U}(\mathcal{A})$~~):

press $\to$ R(A)

(0,1)

$\cdots \xRightarrow{\text{press}} \langle \text{light}, [x = 0] \rangle$

$\langle \text{bright}, [x = 0] \rangle \xRightarrow{\text{press}} \cdots$

$\langle \text{bright}, [x = 0.1] \rangle \xRightarrow{\text{press}} \cdots$

$\langle \text{bright}, [x = 1.0] \rangle \xRightarrow{\text{press}} \cdots$

$\cdots$

$\langle \text{bright}, [x = 3.0] \rangle \xRightarrow{\text{press}} \cdots$

$\langle \text{bright}, [x = 3.001] \rangle \xRightarrow{\text{press}} \cdots$

$(3, \infty)$

$\langle \text{off}, [x = 0] \rangle \xRightarrow{\text{press}} \cdots$

$\cdots$

$\langle \text{off}, [x = 2.9] \rangle \xRightarrow{\text{press}} \cdots$

$\langle \text{off}, [x = 3.0] \rangle \xRightarrow{\text{press}} \cdots$

$\langle \text{off}, [x = 3.001] \rangle \xRightarrow{\text{press}} \cdots$

press $\to$ R(A)

Remark

Remark 4.30. A configuration $\langle \ell, [\nu] \rangle$ is reachable in $\mathcal{R}(\mathcal{A})$ if and only if all $\langle \ell, \nu' \rangle$ with $\nu' \in [\nu]$ are reachable.

In other words: it is possible to **enter** the configuration $\langle \ell, \nu' \rangle$ with an **action transition** (possibly some delay before).

The clock values reachable by staying / letting time pass in ℓ are **not explicitly** represented by the regions of $\mathcal{R}(\mathcal{A})$.

– 13 – 2017-12-14 – Sregaut –

Decidability of The Location Reachability Problem

Claim: (**Theorem 4.33**)

The location reachability problem is **decidable** for timed automata.

Approach: Constructive proof.

- ✔ Observe: clock constraints are **simple** – w.l.o.g. assume constants $c \in \mathbb{N}_0$.
- ✔ **Def. 4.19**: **time-abstract transition system** $\mathcal{U}(\mathcal{A})$ – abstracts from uncountably many delay transitions, still infinite-state.
- ✔ **Lemma 4.20**: location reachability of $\mathcal{A}$ is **preserved** in $\mathcal{U}(\mathcal{A})$.
- ✔ **Def. 4.29**: **region automaton** $\mathcal{R}(\mathcal{A})$ – equivalent configurations collapse into regions
- ✘ **Lemma 4.32**: location reachability of $\mathcal{U}(\mathcal{A})$ is **preserved** in $\mathcal{R}(\mathcal{A})$.
- ✘ **Lemma 4.28**: $\mathcal{R}(\mathcal{A})$ is **finite**.

– 13 – 2017-12-14 – Sraprop –

Region Automaton Properties

Lemma 4.32. [*Correctness*]
For all locations ℓ of a given timed automaton $\mathcal{A}$ the following holds:

ℓ is reachable in $\mathcal{U}(\mathcal{A})$ if and only if ℓ is reachable in $\mathcal{R}(\mathcal{A})$.

For the **Proof**:

$c \xRightarrow{\alpha} c' \implies \exists d' \bullet d \xrightarrow{\alpha}_{\mathcal{R}(\mathcal{A})} d'$

Definition 4.21. [*Bisimulation*] An equivalence relation $\sim$ on valuations is a **(strong) bisimulation** if and only if, whenever

$$\nu_1 \sim \nu_2 \text{ and } \langle \ell, \nu_1 \rangle \xRightarrow{\alpha} \langle \ell', \nu_1' \rangle$$

then there exists ν_2' with $\nu_1' \sim \nu_2'$ and $\langle \ell, \nu_2 \rangle \xRightarrow{\alpha} \langle \ell', \nu_2' \rangle$.

Lemma 4.26. [*Bisimulation*] $\cong$ is a **strong bisimulation**.

Decidability of The Location Reachability Problem

Claim: (**Theorem 4.33**)

The location reachability problem is **decidable** for timed automata.

Approach: Constructive proof.

- ✔ Observe: clock constraints are **simple**
  – w.l.o.g. assume constants $c \in \mathbb{N}_0$.
- ✔ **Def. 4.19**: **time-abstract transition system** $\mathcal{U}(\mathcal{A})$ – abstracts from uncountably many delay transitions, still infinite-state.
- ✔ **Lemma 4.20**: location reachability of $\mathcal{A}$ is **preserved** in $\mathcal{U}(\mathcal{A})$.
- ✔ **Def. 4.29**: **region automaton** $\mathcal{R}(\mathcal{A})$ – equivalent configurations collapse into regions
- ✔ **Lemma 4.32**: location reachability of $\mathcal{U}(\mathcal{A})$ is **preserved** in $\mathcal{R}(\mathcal{A})$.
- ✘ **Lemma 4.28**: $\mathcal{R}(\mathcal{A})$ is **finite**.

The Number of Regions

Lemma 4.28. Let X be a set of clocks, $c_x \in \mathbb{N}_0$ the maximal constant for each $x \in X$, and $c = \max\{c_x \mid x \in X\}$. Then

$$\underbrace{(2c+2)^{|X|} \cdot (4c+3)^{\frac{1}{2}|X|\cdot(|X|-1)}}_{=: D}$$

is an **upper bound** on the **number of regions**.

Proof: Olderog and Dierks (2008)

$$Conf(\mathcal{R}(\mathcal{A})) = L \times \underbrace{Val/_{\cong}}_{\text{Regions}}$$

$$|L| \cdot D$$

The Number of Regions

Lemma 4.28. Let X be a set of clocks, $c_x \in \mathbb{N}_0$ the maximal constant for each $x \in X$, and $c = \max\{c_x \mid x \in X\}$. Then

$$(2c+2)^{|X|} \cdot (4c+3)^{\frac{1}{2}|X|\cdot(|X|-1)}$$

is an **upper bound** on the **number of regions**.

Proof: Olderog and Dierks (2008)

- Lemma 4.28 **in particular** tells us that each timed automaton (in our definition) has **finitely many** regions.
- Note: the upper bound is a **worst case** / **upper bound**, not an **exact number**.

Decidability of The Location Reachability Problem

Claim: (**Theorem 4.33**)

The location reachability problem is **decidable** for timed automata.

Approach: Constructive proof.

- ✔ Observe: clock constraints are **simple**
  – w.l.o.g. assume constants $c \in \mathbb{N}_0$.
- ✔ **Def. 4.19**: **time-abstract transition system** $\mathcal{U}(\mathcal{A})$ – abstracts from uncountably many delay transitions, still infinite-state.
- ✔ **Lemma 4.20**: location reachability of $\mathcal{A}$ is **preserved** in $\mathcal{U}(\mathcal{A})$.
- ✔ **Def. 4.29**: **region automaton** $\mathcal{R}(\mathcal{A})$ – equivalent configurations collapse into regions
- ✔ **Lemma 4.32**: location reachability of $\mathcal{U}(\mathcal{A})$ is **preserved** in $\mathcal{R}(\mathcal{A})$.
- ✔ **Lemma 4.28**: $\mathcal{R}(\mathcal{A})$ is **finite**.

Putting It All Together

Let $\mathcal{A} = (L, B, X, I, E, \ell_{ini})$ be a timed automaton and $\ell \in L$ a location.

- $\mathcal{R}(\mathcal{A})$ can be **constructed effectively**.
- There are **finitely many locations** in L (by definition).
- There are **finitely many regions** by Lemma 4.28.
- So $Conf(\mathcal{R}(\mathcal{A}))$ is **finite** (by construction).
- It is **decidable** whether there exists a sequence

$$\langle \ell_{ini}, [\nu_{ini}] \rangle \xrightarrow{\alpha}_{R(\mathcal{A})} \langle \ell_1, [\nu_1] \rangle \xrightarrow{\alpha}_{R(\mathcal{A})} \dots \xrightarrow{\alpha}_{R(\mathcal{A})} \langle \ell_n, [\nu_n] \rangle$$

  such that $\ell_n = \ell$ (reachability in graphs).

Thus we have just shown:

> **Theorem 4.33.** [*Decidability*]
> The location reachability problem for timed automata is **decidable**.

The Constraint Reachability Problem

(dl.light ∧ x = 27)

- **Given:** Timed automaton $\mathcal{A}$, one of its locations ℓ, and a clock constraint φ.
- **Question:** Is a configuration $\langle \ell, \nu \rangle$ **reachable** where $\nu \models \varphi$, i.e. is there a transition sequence of the form

$$\langle \ell_{ini}, \nu_{ini} \rangle \xrightarrow{\lambda_1} \langle \ell_1, \nu_1 \rangle \xrightarrow{\lambda_2} \langle \ell_2, \nu_2 \rangle \xrightarrow{\lambda_3} \dots \xrightarrow{\lambda_n} \langle \ell_n, \nu_n \rangle = \langle \ell, \nu \rangle$$

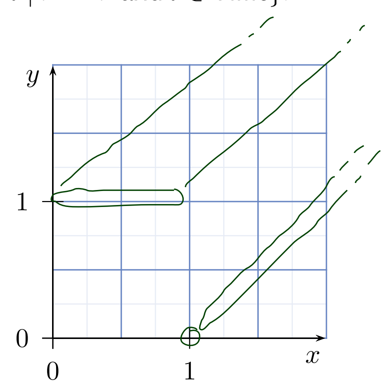
in the labelled transition system $\mathcal{T}(\mathcal{A})$ with $\nu \models \varphi$?

- **Note**: we just observed that $\mathcal{R}(\mathcal{A})$ loses some information about the clock valuations that are possible in / from a region.

> **Theorem 4.34.**
> The constraint reachability problem for timed automata is decidable.

The Delay Operation

- Let $[\nu]$ be a clock region.
- We set $delay[\nu] := \{\nu' + t \mid \nu' \cong \nu \text{ and } t \in \mathsf{Time}\}$.



- **Note**: $delay[\nu]$ can be represented as a **finite** union of regions.

For example, with our two-clock example we have

$$delay[x = y = 0] = [x = y = 0] \cup [0 < x = y < 1] \cup [x = y = 1] \cup [1 < x = y]$$

Tell Them What You've Told Them…

- **Location Reachab~~le~~ility Problem**:
  is location ℓ reachable in $\mathcal{A}$?
- Decidability proof: [AD94]
  - **normalise constants**,
  - construct the **Time Abstract Transition System**
    - "get rid of" **delay transitions**,
    - still **uncountably many configurations**
  - collapse **equivalent** clock valuations into **regions**
    - obtain **finitely many (abstract) configurations**
  - construct the **Region Automaton**
    - it is **finite**, ✓
    - and **preserves location reachability**. from $\mathcal{U}(\mathcal{A})$
- Thus: there are chances to get **automatic verification** for TA.
- Result can easily be lifted to **constraint reachability**.

References

References

Olderog, E.-R. and Dierks, H. (2008). *Real-Time Systems - Formal Specification and Automatic Verification*. Cambridge University Press.