

Real-Time Systems

Lecture 13: Location Reachability (or: The Region Automaton)

2014-07-15

Dr. Bernd Westphal

Albert-Ludwigs-Universität Freiburg, Germany

– 13 – 2014-07-15 – main –

Contents & Goals

Last Lecture:

- Networks of Timed Automata
- Uppaal Demo

This Lecture:

- **Educational Objectives:** Capabilities for following tasks/questions.
 - What are decidable problems of TA?
 - How can we show this? What are the essential premises of decidability?
 - What is a region? What is the region automaton of this TA?
 - What's the time abstract system of a TA? Why did we consider this?
 - What can you say about the complexity of Region-automaton based reachability analysis?
- **Content:**
 - Timed Transition System of network of timed automata
 - Location Reachability Problem
 - Constructive, region-based decidability proof

– 13 – 2014-07-15 – Prelim –

The Location Reachability Problem

The Location Reachability Problem

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

Question: Is ℓ **reachable**?

That is, is there a transition sequence of the form

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

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.

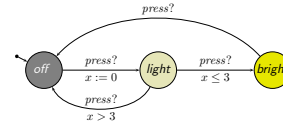
Decidability of The Location Reachability Problem

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.
- **Lem. 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
- **Lem. 4.32:** location reachability of $\mathcal{U}(\mathcal{A})$ is **preserved** in $\mathcal{R}(\mathcal{A})$.
- **Lem. 4.28:** $\mathcal{R}(\mathcal{A})$ is **finite**.



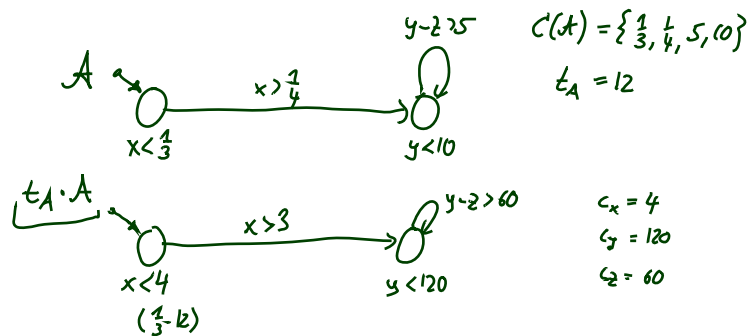
- 13 - 2014-07-15 - Sdec -

5/33

Without Loss of Generality: Natural Constants

Recall: Simple clock constraints are $\varphi ::= x \sim c \mid x - y \sim c \mid \varphi \wedge \varphi$ with $x, y \in X$, $c \in \mathbb{Q}_0^+$, 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 $t_{\mathcal{A}}$.



- 13 - 2014-07-15 - Sdec -

6/33

Without Loss of Generality: Natural Constants

Recall: Simple clock constraints are $\varphi ::= x \sim c \mid x - y \sim c \mid \varphi \wedge \varphi$
with $x, y \in X$, $c \in \mathbb{Q}_0^+$, 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} .

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.
- ✗ **Lem. 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
- ✗ **Lem. 4.32:** location reachability of $\mathcal{U}(\mathcal{A})$ is **preserved** in $\mathcal{R}(\mathcal{A})$.
- ✗ **Lem. 4.28:** $\mathcal{R}(\mathcal{A})$ is **finite**.

Helper: Relational Composition

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

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

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

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

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

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

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

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

where

$$\xRightarrow{\alpha} \subseteq \text{Conf}(\mathcal{A}) \times \text{Conf}(\mathcal{A})$$

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

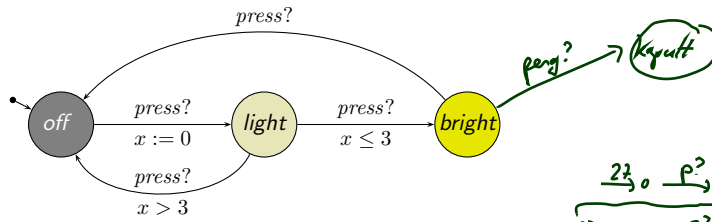
$$\langle \ell, \nu \rangle \xRightarrow{\alpha} \langle \ell', \nu' \rangle$$

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

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

Example

$$\langle l, v \rangle \xrightarrow{\alpha} \langle l', v' \rangle \text{ iff } \exists t \in \text{Time} \bullet \langle l, v \rangle \xrightarrow{t} \circ \xrightarrow{\alpha} \langle l', v' \rangle$$



- $\langle \text{light}, x=0 \rangle \xrightarrow{\text{press?}} \langle \text{off}, x=27 \rangle$ YES, with $t=27$ we have $\langle \text{light}, x=0 \rangle \xrightarrow{27} \langle \text{li}, 27 \rangle \xrightarrow{p?} \langle \text{off}, 27 \rangle$
- $\langle \text{off}, x=4 \rangle \xrightarrow{p?} \langle \text{li}, x=0 \rangle$ YES, any $t \in \mathbb{R}_0^+$ works
- $\langle \text{off}, x=4 \rangle \xrightarrow{?} \langle \text{li}, x=1 \rangle$ NO, $\langle \text{off}, x=4 \rangle \xrightarrow{t} \circ \xrightarrow{\alpha} \langle \text{li}, x=t \rangle$ implies $\alpha = p?$ and $t' = 0+4$
- $\langle \text{off}, x=0 \rangle \xrightarrow{?} \langle \text{off}, x=5 \rangle$ NO, no α with $\langle \text{off}, x=5 \rangle \xrightarrow{\alpha} \langle \text{off}, x=5 \rangle$
- $\langle \text{off}, x=0 \rangle \xrightarrow{?} \langle \text{bright}, x=0 \rangle$ NO, needs two actions
- $\langle \text{li}, x=1 \rangle \xrightarrow{?} \langle \text{bright}, x=1 \rangle$ YES, $t=0, \alpha = p?$
- $\langle \text{Kagult}, x=13 \rangle \xrightarrow{?} \langle l, x=t \rangle$ NO, no edge

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

Proof:

\Leftarrow : easy

\Rightarrow : l reachable in $\mathcal{T}(\mathcal{A})$

\Leftrightarrow there is $\langle l_0, v_0 \rangle \xrightarrow{t_0} \langle l_0, v_0 \rangle \xrightarrow{t_1} \dots \xrightarrow{t_n} \langle l_0, v_n \rangle \xrightarrow{\alpha_1} \langle l_1, v_1 \rangle$

$$t_n := \sum_{i=1}^{n_0} t_{0_i}$$

\vdots
 $\langle l_1, v_1 \rangle \xrightarrow{t_{11}} \dots \xrightarrow{\alpha_2} \langle l_2, v_2 \rangle$
 \vdots
 $\langle l_m, v_m \rangle \xrightarrow{t_{m1}} \dots \xrightarrow{t_{m2}} \langle l_m, v_m \rangle \xrightarrow{\alpha_{m+1}} \langle l_{m+1}, v_{m+1} \rangle, l_{m+1} = l$

$\Rightarrow \langle l_0, v_0 \rangle \xrightarrow{\alpha_1} \langle l_1, v_1 \rangle \xrightarrow{\alpha_2} \dots \xrightarrow{\alpha_{m+1}} \langle l_{m+1}, v_{m+1} \rangle, l_{m+1} = l$

by $t_1 \xrightarrow{\alpha_1}$

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

Decidability of The Location Reachability Problem

Claim: (Theorem 4.33)

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

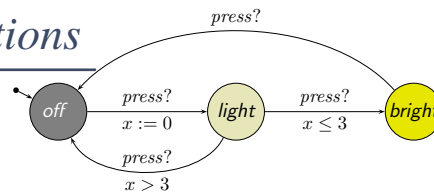
Approach: Constructive proof.

- ✓ 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.
- ✓ **Lem. 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
- ✗ **Lem. 4.32:** location reachability of $\mathcal{U}(\mathcal{A})$ is **preserved** in $\mathcal{R}(\mathcal{A})$.
- ✗ **Lem. 4.28:** $\mathcal{R}(\mathcal{A})$ is **finite**.

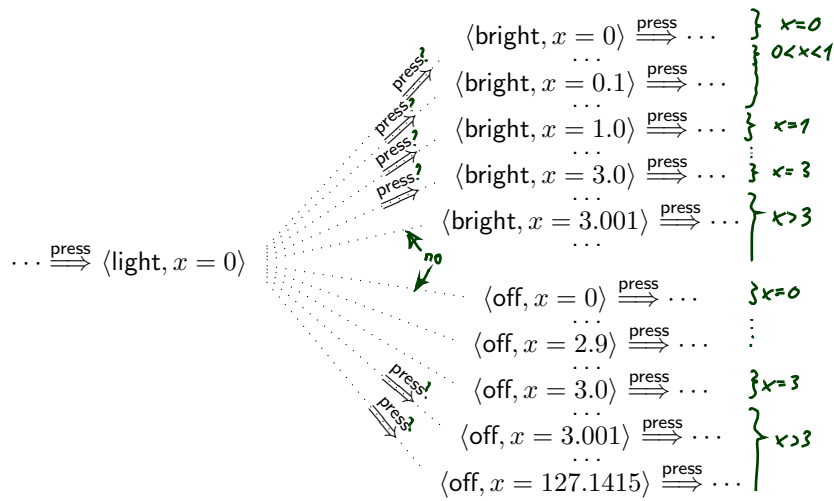
- 13 - 2014-07-15 - Sdec -

12/33

Indistinguishable Configurations



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



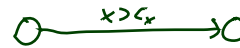
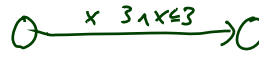
- 13 - 2014-07-15 - Sdec -

13/33

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 $k \in \{0, \dots, c_x - 1\}$.
- \mathcal{A} **cannot distinguish** ν_1 and ν_2 if $\nu_i(x) > c_x$, $i = 1, 2$.



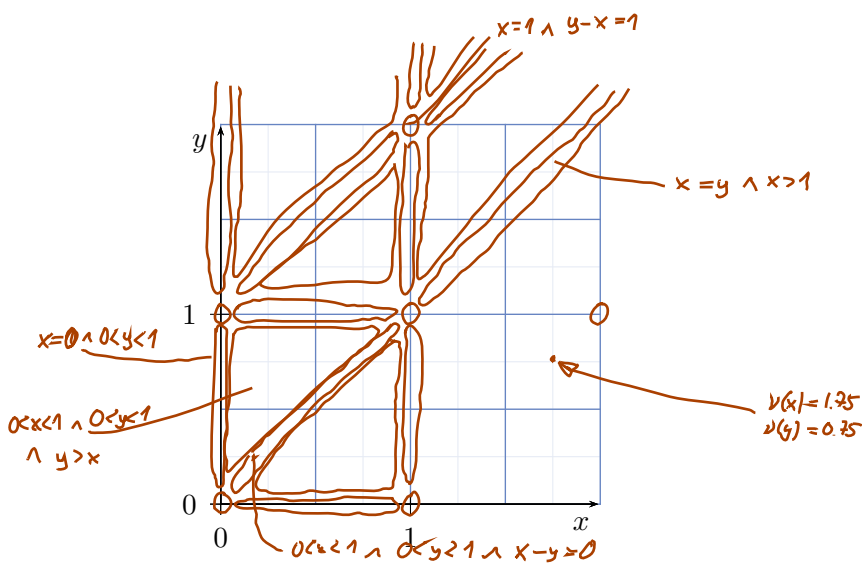
- If $c_x \geq 1$, there are $(2c_x + 2)$ **equivalence classes**:

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

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

Distinguishing Clock Valuations: Two Clocks

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



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** $\text{frac}(q) \in [0, 1)$

such that

$$q = \lfloor q \rfloor + \text{frac}(q).$$

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 $\nu_1 \cong \nu_2$ iff the following **four** conditions are satisfied.

(1) For all $x \in X$,

$$\lfloor \nu_1(x) \rfloor = \lfloor \nu_2(x) \rfloor \text{ or both } \nu_1(x) > c_x \text{ and } \nu_2(x) > c_x.$$

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

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

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

$$\begin{aligned} \lfloor \nu_1(x) - \nu_1(y) \rfloor &= \lfloor \nu_2(x) - \nu_2(y) \rfloor \\ \text{or both } |\nu_1(x) - \nu_1(y)| > c &\text{ and } |\nu_2(x) - \nu_2(y)| > c. \end{aligned}$$

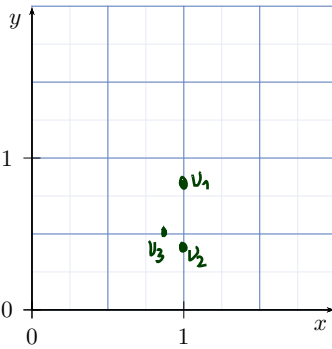
(4) For all $x, y \in X$ with $-c \leq \nu_1(x) - \nu_1(y) \leq c$,

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

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

Example: Regions

- (1) $\forall x \in X : \lfloor \nu_1(x) \rfloor = \lfloor \nu_2(x) \rfloor \vee (\nu_1(x) > c_x \wedge \nu_2(x) > c_x)$
- (2) $\forall x \in X : \nu_1(x) \leq c_x$
 $\implies (\text{frac}(\nu_1(x)) = 0 \iff \text{frac}(\nu_2(x)) = 0)$
- (3) $\forall x, y \in X : \lfloor \nu_1(x) - \nu_1(y) \rfloor = \lfloor \nu_2(x) - \nu_2(y) \rfloor$
 $\vee (|\nu_1(x) - \nu_1(y)| > c \wedge |\nu_2(x) - \nu_2(y)| > c)$
- (4) $\forall x, y \in X : -c \leq \nu_1(x) - \nu_1(y) \leq c \implies$
 $(\text{frac}(\nu_1(x) - \nu_1(y)) = 0 \iff \text{frac}(\nu_2(x) - \nu_2(y)) = 0)$



- (1) $\lfloor \nu_1(x) \rfloor = 1 = \lfloor \nu_2(x) \rfloor$
 $\lfloor \nu_1(y) \rfloor = 0 = \lfloor \nu_2(y) \rfloor$
- (2) $\text{frac}(\nu_1(x)) = 0 = \text{frac}(\nu_2(x))$
 $\text{frac}(\nu_1(y)) \neq 0, \text{frac}(\nu_2(y)) \neq 0$
- (3) $\lfloor \nu_1(x) - \nu_1(y) \rfloor = 0 = \lfloor \nu_2(x) - \nu_2(y) \rfloor$
- (4) " "

$\nu_2, \nu_3:$
 $\nu_2 \not\equiv \nu_3$
 because (1) not satisfied

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 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

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

where

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

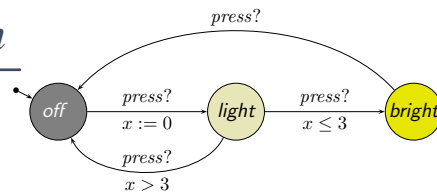
$$\langle \ell, [\nu] \rangle \xrightarrow{\alpha}_{\mathcal{R}(\mathcal{A})} \langle \ell', [\nu'] \rangle \text{ if and only if } \langle \ell, \nu \rangle \xrightarrow{\alpha} \langle \ell', \nu' \rangle$$

in $\mathcal{U}(\mathcal{A}),$ and

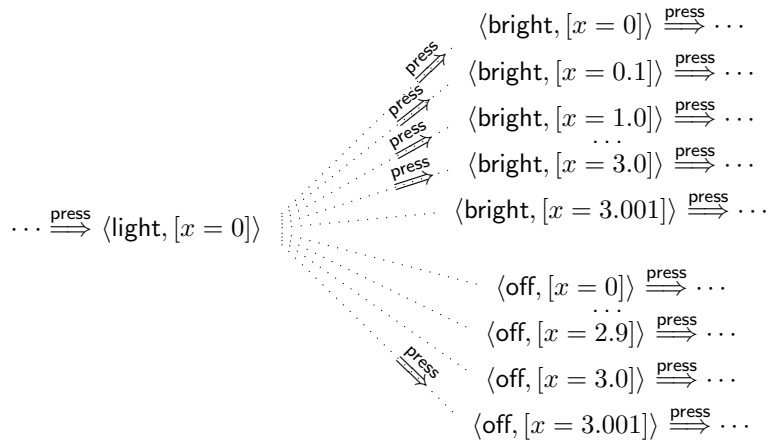
- $C_{ini} = \{\langle \ell_{ini}, [\nu_{ini}] \rangle\} \cap \text{Conf}(\mathcal{R}(\mathcal{A}))$ with $\nu_{ini}(X) = \{0\}.$

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



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



Remark

Remark 4.30. That a configuration $\langle \ell, [\nu] \rangle$ is reachable in $\mathcal{R}(\mathcal{A})$ represents the fact, that all $\langle \ell, \nu \rangle$ are reachable.

IAW: in \mathcal{A} , we can observe ν when

location ℓ has **just been entered**.

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

Decidability of The Location Reachability Problem

Claim: (Theorem 4.33)

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

Approach: Constructive proof.

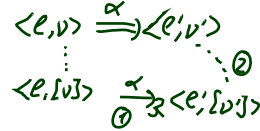
- ✓ Observe: clock constraints are **simple**
— w.l.o.g. assume constants $c \in \mathbb{N}_0$.
- ✓ **Def. 4.19: time-abstract transition system** $\mathcal{U}(\mathcal{A})$ — abstracts from uncountably many delay transitions, still infinite-state.
- ✓ **Lem. 4.20:** location reachability of \mathcal{A}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

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**:



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

$$\nu_1 \sim \nu_2 \text{ and } \langle l, \nu_1 \rangle \xrightarrow{\alpha} \langle l', \nu'_1 \rangle$$

then there exists ν'_2 with $\nu'_1 \sim \nu'_2$ and $\langle l, \nu_2 \rangle \xrightarrow{\alpha} \langle l', \nu'_2 \rangle$.

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.
- ✓ **Lem. 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
- ✓ **Lem. 4.32:** location reachability of $\mathcal{U}(\mathcal{A})$ is **preserved** in $\mathcal{R}(\mathcal{A})$.
- ✗ **Lem. 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

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

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

Proof: [Olderog and Dierks, 2008]

Observations Regarding the Number of Regions

- 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**, not an **exact bound**.

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.
- ✓ **Lem. 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
- ✓ **Lem. 4.32:** location reachability of $\mathcal{U}(\mathcal{A})$ is **preserved** in $\mathcal{R}(\mathcal{A})$.
- ✓ **Lem. 4.28:** $\mathcal{R}(\mathcal{A})$ is **finite**.

– 13 – 2014-07-15 – Sdec –

28/33

Putting It All Together

Let $\mathcal{A} = (L, B, X, I, E, \ell_{ini})$ be a timed automaton, $\ell \in L$ 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 (C_{init} of $\mathcal{R}(\mathcal{A})$ is empty) or whether there exists a sequence

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

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

So we have

Theorem 4.33. [Decidability]

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

– 13 – 2014-07-15 – Sdec –

29/33

The Constraint Reachability Problem

- **Given:** A timed automaton \mathcal{A} , one of its control 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

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

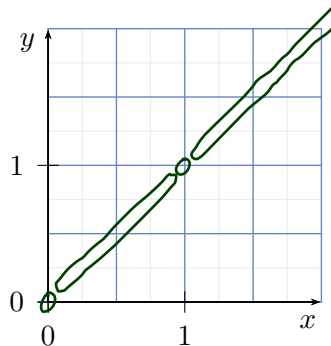
in the labelled transition system $\mathcal{T}(\mathcal{A})$ with $\nu \models \varphi$?

- **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 $delay[\nu] := \{\nu' + t \mid \nu' \cong \nu \text{ and } t \in \text{Time}\}$.



- **Note:** $delay[\nu]$ can be represented as a **finite** union of regions.
For example, with our two-clock example we have

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

References

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