Real-Time Systems

Lecture 11: Networks of Timed Automata

2012-06-26

Dr. Bernd Westphal

Albert-Ludwigs-Universität Freiburg, Germany

Contents & Goals

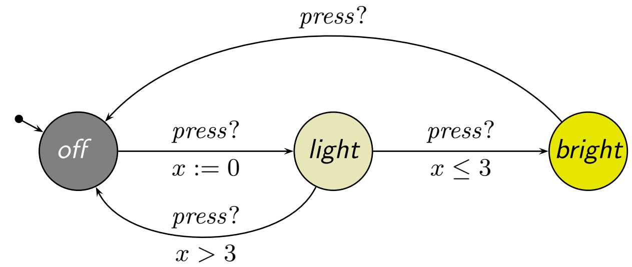
Last Lecture:

- Timed automata syntax
- TA operational semantics

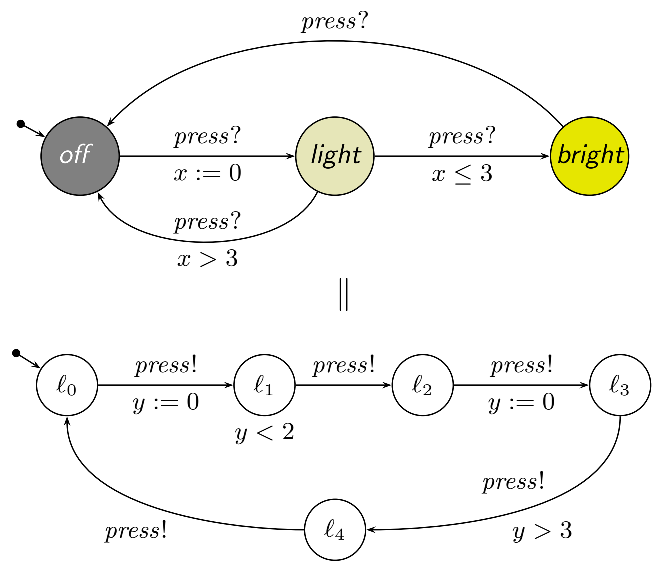
This Lecture:

- **Educational Objectives:** Capabilities for following tasks/questions.
  - what's a transition sequence, computation path, run?
  - what is Zeno behaviour?
  - what's the (syntactical) parallel composition of TA?
- **Content:**
  - transition sequence, computation path, run
  - parallel composition of TA
  - Uppaal demo

Recall: Plan



- **Pure TA** syntax
  - channels, actions
  - (simple) clock constraints
  - Def. TA
- **Pure TA** operational semantics
  - clock valuation, time shift, modification
  - operational semantics
  - discussion
- Transition sequence, computation path, run
- **Network of TA**
  - parallel composition (syntactical)
  - restriction
  - network of TA semantics
- **Uppaal Demo**
- Region abstraction; zones
- **Extended TA**; Logic of Uppaal



Computation Path, Run

Computation Paths

- $\langle \ell, \nu \rangle, t$ is called **time-stamped configuration**

- **time-stamped delay transition**: $\langle \ell, \nu \rangle, t \xrightarrow{t'} \langle \ell, \nu + t' \rangle, t + t'$
iff $t' \in$ Time and $\langle \ell, \nu \rangle \xrightarrow{t'} \langle \ell, \nu + t' \rangle$.

- **time-stamped action transition**: $\langle \ell, \nu \rangle, t \xrightarrow{\alpha} \langle \ell', \nu' \rangle, t$
iff $\alpha \in B_{?!}$ and $\langle \ell, \nu \rangle \xrightarrow{\alpha} \langle \ell', \nu' \rangle$.

- A sequence of time-stamped configurations

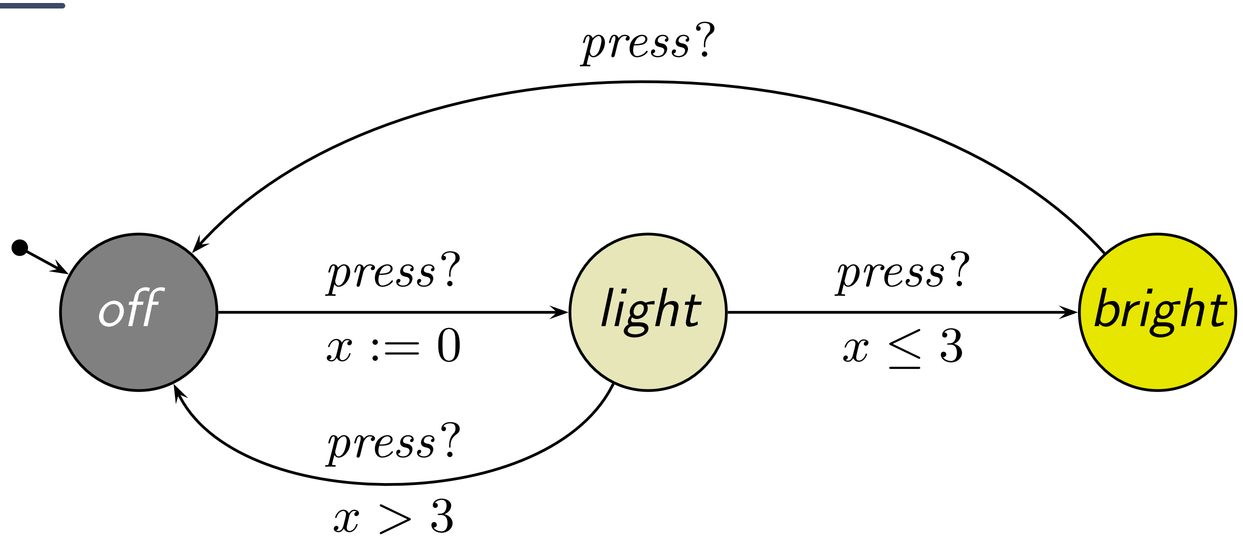
$$\xi = \langle \ell_0, \nu_0 \rangle, t_0 \xrightarrow{\lambda_1} \langle \ell_1, \nu_1 \rangle, t_1 \xrightarrow{\lambda_2} \langle \ell_2, \nu_2 \rangle, t_2 \xrightarrow{\lambda_3} \dots$$

is called **computation path** (or path) of $\mathcal{A}$ **starting in** $\langle \ell_0, \nu_0 \rangle, t_0$
if and only if it is either infinite or maximally finite.

- A **computation path** (or path) is a computation path starting at $\langle \ell_0, \nu_0 \rangle, 0$
where $\langle \ell_0, \nu_0 \rangle \in C_{ini}$.

Example

$\mathcal{A}$:



transition sequence of $\mathcal{A}$!

$$\langle \textit{off}, x = 0\rangle, 0 \xrightarrow{2.5} \langle \textit{off}, x = 2.5\rangle, 2.5 \xrightarrow{1.7} \langle \textit{off}, x = 4.2\rangle, 4.2$$

$$\xrightarrow{press?} \langle \textit{light}, x = 0\rangle, 4.2 \xrightarrow{2.1} \langle \textit{light}, x = 2.1\rangle, 6.3$$

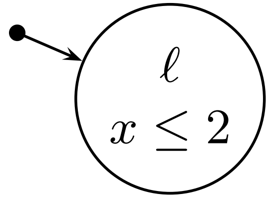
$$\xrightarrow{press?} \langle \textit{bright}, x = 2.1\rangle, 16.3 \xrightarrow{10} \langle \textit{bright}, x = 12.1\rangle, 16.3$$

$$\xrightarrow{press?} \langle \textit{off}, x = 12.1\rangle, 16.3$$

$$\xrightarrow{press?} \langle \textit{light}, x = 0\rangle, 16.3 \xrightarrow{0} \langle \textit{light}, x = 0\rangle, 16.3$$

$$\xrightarrow{press?} \langle \textit{bright}, x = 0\rangle, 16.3 \xrightarrow{press?} \langle \textit{off}, x = 0\rangle, 16.3$$

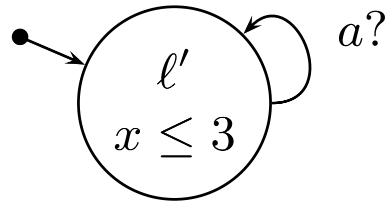
Timelocks and Zeno Behaviour



$$\langle \ell, x=0 \rangle, 0 \xrightarrow{1.0} \langle \ell, x=1.0 \rangle, 1 \xrightarrow{1.0} \langle \ell, x=2.0 \rangle, 2 \xrightarrow{0} \langle \ell, x=2.0 \rangle, 2 \xrightarrow{0} \langle \ell, x=2.0 \rangle, 2$$

Timelock:

"it is not possible to delay $t > 0$ any more"

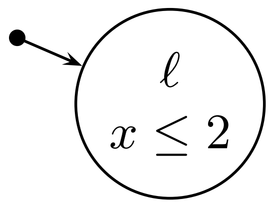


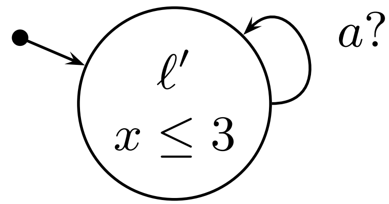
$$\langle \ell', x=0 \rangle, 0 \xrightarrow{3} \langle \ell', x=3 \rangle, 3$$

$$\langle \ell', x=3 \rangle, 3 \xrightarrow{a?} \langle \ell', x=3 \rangle, 3$$

$$\langle \ell', x=3 \rangle, 3 \xrightarrow{0} \langle \ell', x=3 \rangle, 3$$

Timelocks and Zeno Behaviour





- **Timelock**:

$$\langle \ell, x = 0 \rangle, 0 \xrightarrow{2} \langle \ell, x = 2 \rangle, 2$$

$$\langle \ell', x = 0 \rangle, 0 \xrightarrow{3} \langle \ell', x = 3 \rangle, 3 \xrightarrow{a?} \langle \ell', x = 3 \rangle, 3 \xrightarrow{a?} \dots$$

- **Zeno** behaviour:

$$\langle \ell, x = 0 \rangle, 0 \xrightarrow{1/2} \langle \ell, x = 1/2 \rangle, \frac{1}{2} \xrightarrow{1/4} \langle \ell, x = 3/4 \rangle, \frac{3}{4} \dots$$

$$\xrightarrow{1/2^n} \langle \ell, x = (2^n - 1)/2^n \rangle, \frac{2^n - 1}{2^n} \dots$$

Real-Time Sequence

Definition 4.9. An infinite sequence

$$t_0, t_1, t_2, \ldots$$

of values $t_i \in \text{Time}$ for $i \in \mathbb{N}_0$ is called **real-time sequence** if and only if it has the following properties:

- **Monotonicity**:

$$\forall\, i \in \mathbb{N}_0 : t_i \leq t_{i+1}$$

- **Non-Zeno behaviour** (or **unboundedness** or **progress**):

$$\forall\, t \in \text{Time}\ \exists\, i \in \mathbb{N}_0 : t < t_i$$

Run

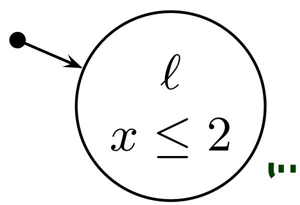
Definition 4.10. A **run** of $\mathcal{A}$ **starting** in the time-stamped configuration $\langle \ell_0, \nu_0 \rangle, t_0$ is an infinite computation path of $\mathcal{A}$

$$\xi = \langle \ell_0, \nu_0 \rangle, t_0 \xrightarrow{\lambda_1} \langle \ell_1, \nu_1 \rangle, t_1 \xrightarrow{\lambda_2} \langle \ell_2, \nu_2 \rangle, t_2 \xrightarrow{\lambda_3} \dots$$

where $(t_i)_{i \in \mathbb{N}_0}$ is a real-time sequence.

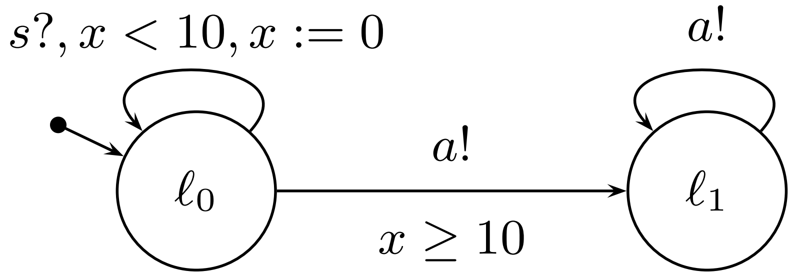
If $\langle \ell_0, \nu_0 \rangle \in C_{ini}$ and $t_0 = 0$, then we call ξ a **run** of $\mathcal{A}$.

Example:



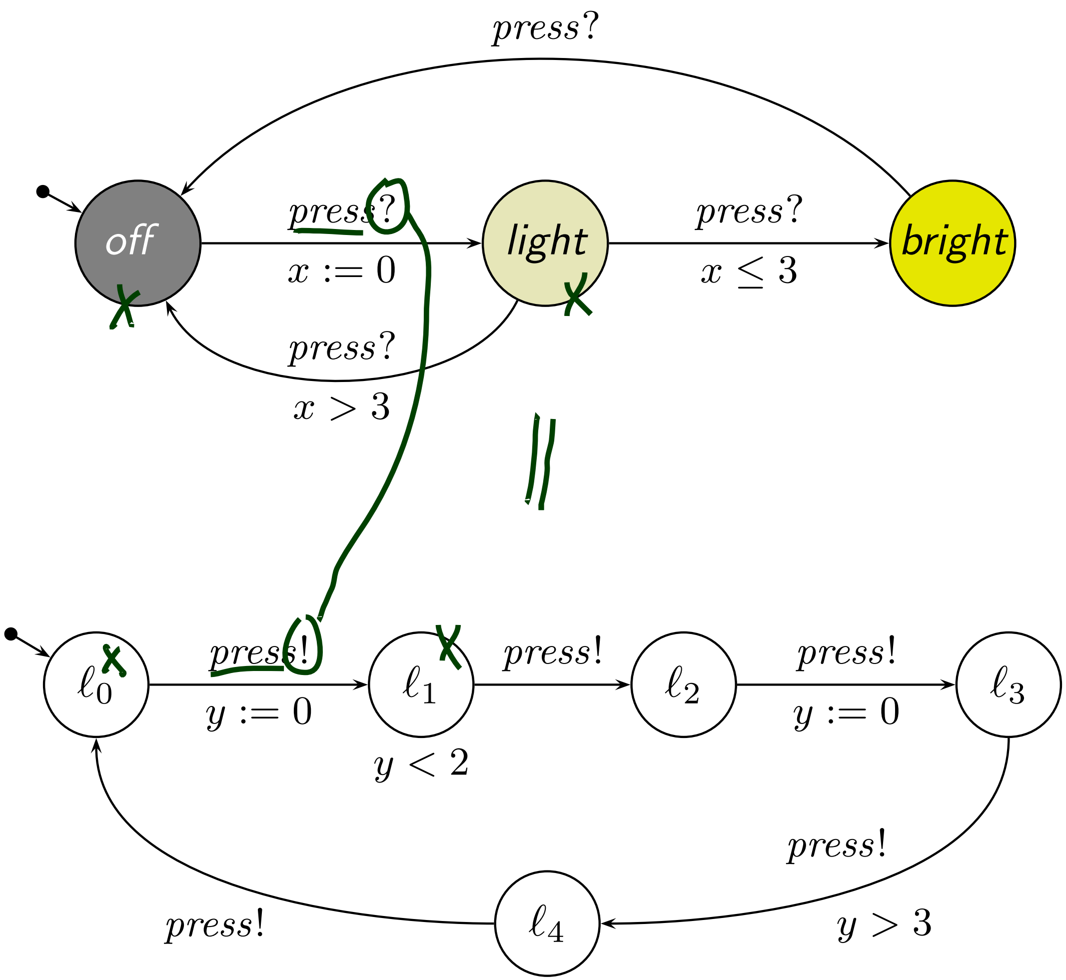
... does not have a run.
(Timestamps cannot grow past 2.0.)

Example



Network of TA

Recall: Light Controller and User



Parallel Composition

Definition 4.12.
The **parallel composition** $\mathcal{A}_1 \parallel \mathcal{A}_2$ of two timed automata

$$\mathcal{A}_i = (L_i, B_i, X_i, I_i, E_i, \ell_{ini,i}), \quad i = 1, 2,$$

with **disjoint** sets of clocks X_1 and X_2 yields the timed automaton

$$\mathcal{A} = (L_1 \times L_2, B_1 \cup B_2, X_1 \cup X_2, I, E, (\ell_{ini,1}, \ell_{ini,2}))$$

where

- $I(\ell_1, \ell_2) := I(\ell_1) \wedge I(\ell_2)$, and
- E consists of **handshake** and **asynchronous communication**. ($\rightarrow$ **next slide**)

Helper: Action Complementation

- The **complementation function**

$$\overline{\cdot} : Act \to Act$$

  is defined pointwise as

  - $\overline{a!} = a?$
  - $\overline{a?} = a!$
  - $\overline{\tau} = \tau$

- **Note**: $\overline{\overline{\alpha}} = \alpha$ for all $\alpha \in Act$.

Parallel Composition: Handshake and Asynchrony

$\mathcal{A}_1 \parallel \mathcal{A}_2 = (L_1 \times L_2, B_1 \cup B_2, X_1 \cup X_2, I, E, (\ell_{ini,1}, \ell_{ini,2}))$ with

- **Handshake**:

  If there is $a \in B_1 \cup B_2$ such that

  $$(\ell_1, \alpha, \varphi_1, Y_1, \ell_1') \in E_1, \text{ and } (\ell_2, \bar{\alpha}, \varphi_2, Y_2, \ell_2') \in E_2,$$

  and $\{a!, a?\} = \{\alpha, \bar{\alpha}\}$, then

  $$((\ell_1, \ell_2), \tau, \varphi_1 \wedge \varphi_2, Y_1 \cup Y_2, (\ell_1', \ell_2')) \in E.$$

- **Asynchrony**:

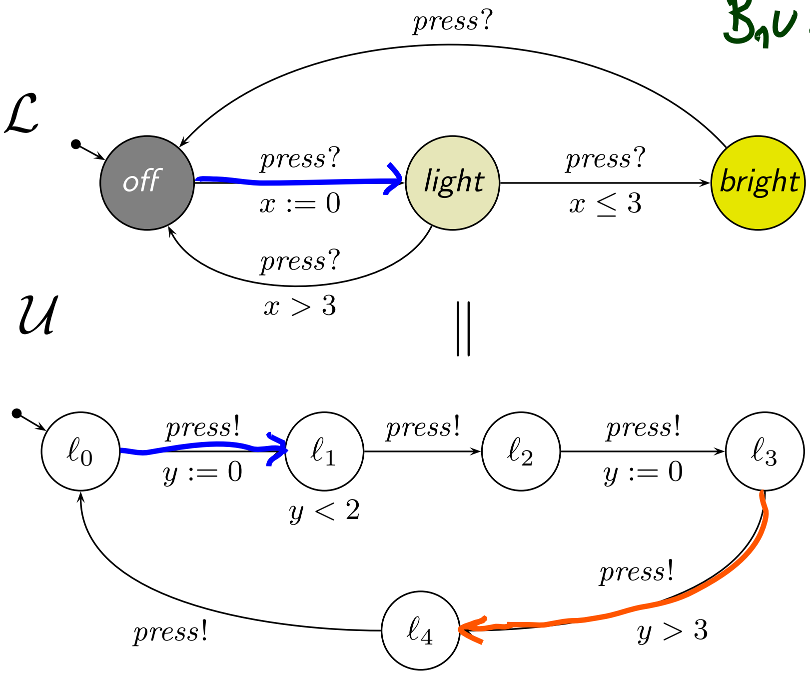
  If $(\ell_1, \alpha, \varphi_1, Y_1, \ell_1') \in E_1$ then for all $\ell_2 \in L_2$,

  $$((\ell_1, \ell_2), \alpha, \varphi_1, Y_1, (\ell_1', \ell_2)) \in E.$$

  If $(\ell_2, \alpha, \varphi_2, Y_2, \ell_2') \in E_2$ then for all $\ell_1 \in L_1$,

  $$((\ell_1, \ell_2), \alpha, \varphi_2, Y_2, (\ell_1, \ell_2')) \in E.$$

Example



$L_1 \times L_2 = \{(off, \ell_0), \dots, (off, \ell_4), (li, \ell_0), \dots, (li, \ell_4), (bs, \ell_0), \dots, (bs, \ell_4)\}$

$B_1 \cup B_2 = \{press\}$ $\quad X_1 \cup X_2 = \{x, y\}$ $\quad (\ell_{ini,1}, \ell_{ini,2}) = (off, \ell_0)$

$I(off, \ell_0) = true \wedge true$

$I(li, \ell_1) = true \wedge y < 2$

$\vdots$

Examples for handshake:

- $((off, \ell_0), \tau, true \wedge true, \{x, y\}, (li, \ell_1))$
- $((li, \ell_0), \tau, x \leq 3 \wedge true, \{y\}, (bs, \ell_1))$
- $((li, \ell_0), \tau, x > 3 \wedge true, \{y\}, (off, \ell_1))$

Examples for asynchrony:

- $((li, \ell_3), press!, y > 3, \emptyset, (li, \ell_4))$
- $((li, \ell_3), press?, x \leq 3, \emptyset, (bs, \ell_3))$
- $((li, \ell_3), press?, x > 3, \emptyset, (off, \ell_3))$

$\mathcal{L} \,\|\, \mathcal{U} = (L_1 \times L_2, B_1 \cup B_2, X_1 \cup X_2, I, E, (\ell_{ini,1}, \ell_{ini,2}))$

- If $a \in B_1 \cup B_2$ s.t. $(\ell_1, \alpha, \varphi_1, Y_1, \ell_1') \in E_1$ and $(\ell_2, \bar{\alpha}, \varphi_2, Y_2, \ell_2') \in E_2$ and $\{a!, a?\} = \{\alpha, \bar{\alpha}\}$ then $((\ell_1, \ell_2), \tau, \varphi_1 \wedge \varphi_2, Y_1 \cup Y_2, (\ell_1', \ell_2')) \in E$
- If $(\ell_1, \alpha, \varphi_1, Y_1, \ell_1') \in E_1$ then for all $\ell_2 \in L_2$, $((\ell_1, \ell_2), \alpha, \varphi_1, Y_1, (\ell_1', \ell_2)) \in E$, and conversely

Restriction

Definition 4.13.
A **local channel** b is introduced by the **restriction operator** which, for a timed automaton $\mathcal{A} = (L, B, X, I, E, \ell_{ini})$ yields

$$\text{chan}\, b \bullet \mathcal{A} := (L, B \setminus \{b\}, X, I, E', \ell_{ini})$$

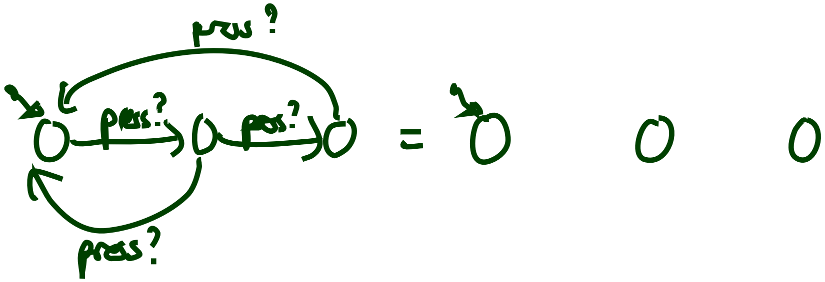
where

- $(\ell, \alpha, \varphi, Y, \ell') \in E'$
  if and only if $(\ell, \alpha, \varphi, Y, \ell') \in E$ and $\alpha \notin \{b!, b?\}$.

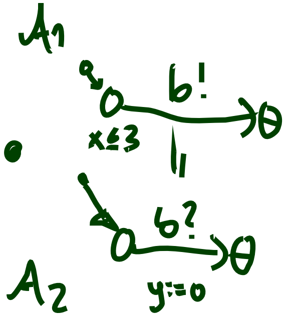
- **Abbreviation**:

$$\text{chan}\, b_1 \dots b_m \bullet \mathcal{A} := \text{chan}\, b_1 \bullet \dots \text{chan}\, b_m \bullet \mathcal{A}$$

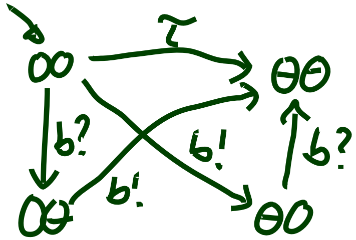
chan press •



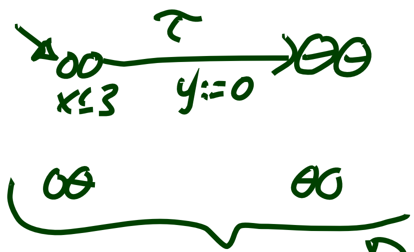
chan b •



= chan b •



=



$B_1 = \{b, c\}$

$B_2 = \{b\}$

a network of TA

by

chan c • $A_1 \| A_2$

a network

(a closed network)

Networks of Timed Automata

- A timed automaton $\mathcal{N}$ is called **network of timed automata** if and only if it is obtained as

$$\mathsf{chan}\, b_1 \ldots b_m \bullet (\mathcal{A}_1 \parallel \ldots \parallel \mathcal{A}_n)$$

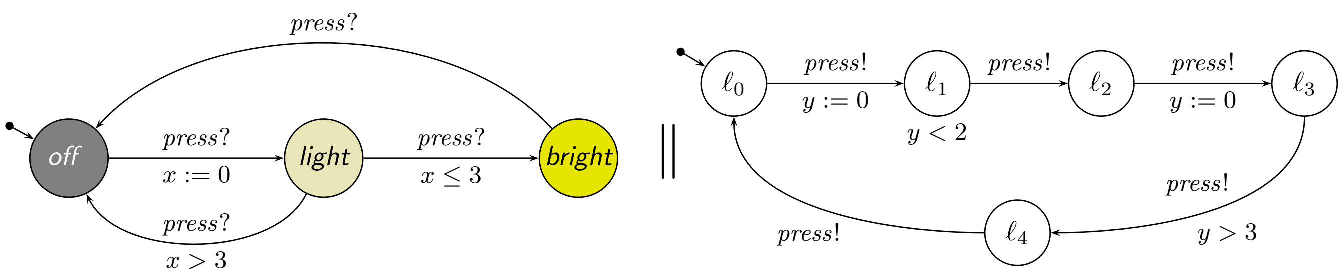
Closed Networks

- A network
$$\mathcal{N} = \mathsf{chan}\, b_1 \dots b_m \bullet (\mathcal{A}_1 \parallel \dots \parallel \mathcal{A}_n)$$
is called **closed** if and only if
$$\{b_1, \dots, b_m\} = \bigcup_{i=1}^{n} B_i.$$

- Then, by Lemma 4.16 (later), **local transitions** don't occur (since $B = \emptyset$). Transitions are thus either internal actions τ or delay transitions.

Example:

chan $press \bullet$



is closed.

Operational Semantics of Networks

Lemma 4.16. Let $\mathcal{A}_i = (L_i, B_i, X_i, I_i, E_i, \ell_{ini,i})$ with $i = 1, \dots, n$ be a set of timed automata with disjoint clocks. Then the operational semantics of the network

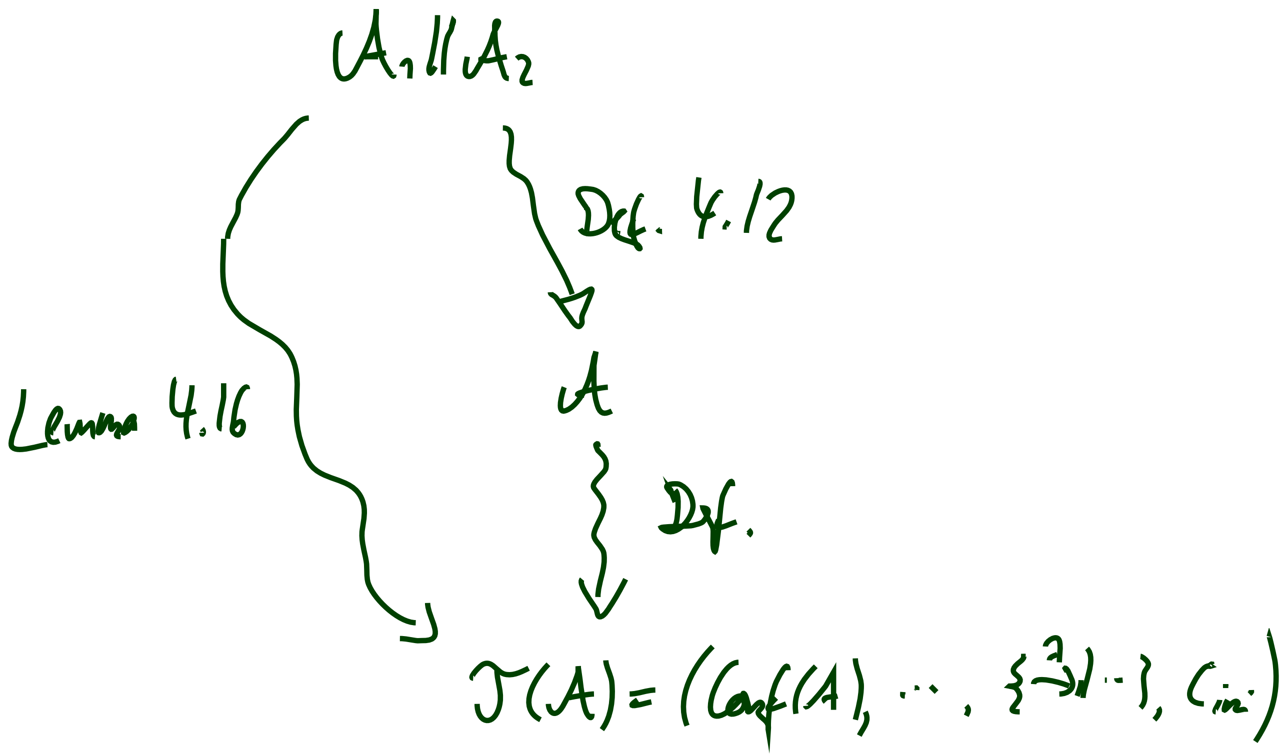
$$\mathsf{chan}\, b_1 \dots b_m \bullet (\mathcal{A}_1 \parallel \dots \parallel \mathcal{A}_n)$$

yields the labelled transition system

$$(\mathit{Conf}(\mathcal{N}), \mathsf{Time} \cup B_{?!}, \{\xrightarrow{\lambda} \mid \lambda \in \mathsf{Time} \cup B_{?!}\}, C_{ini})$$

with

- $X = \bigcup_{i=1}^{n} X_i$,
- $B = \bigcup_{i=1}^{n} B_i \setminus \{b_1, \dots, b_m\}$,
- $\mathit{Conf}(\mathcal{N}) = \{\langle \vec{\ell}, \nu \rangle \mid$
  $\vec{\ell} \in L_1 \times \dots \times L_n \wedge \nu : X \to \mathsf{Time} \wedge \nu \models \bigwedge_{k=1}^{n} I_k(\ell_k)\}$,
- $C_{ini} = \{\langle (\ell_{ini,1}, \dots, \ell_{ini,n}), \nu_{ini} \rangle\} \cap \mathit{Conf}(\mathcal{N})$
  where $\nu_{ini}(x) = 0$ for all $x \in X$,
- and three types of transition relations ($\to$ **next slides**).


A1 || A2
Def. 4.12
A
Def.
Lemma 4.16
J(A) = (Conf(A), ..., {→...}, Cini)

Operational Semantics of Networks: Local Transitions

For each $\lambda \in \mathsf{Time} \cup B_{!?}$ the transition relation $\xrightarrow{\lambda} \subseteq \mathit{Conf}(\mathcal{N}) \times \mathit{Conf}(\mathcal{N})$ has one of the following three types:

(i) **Local transition**:

$$\langle \vec{\ell}, \nu \rangle \xrightarrow{\alpha} \langle \vec{\ell'}, \nu' \rangle$$

if there is $i \in \{1, \dots, n\}$ such that

- $(\ell_i, \alpha, \varphi, Y, \ell'_i) \in E_i$, $\alpha \in B_{!?}$, (i-th automaton has corresp. edge)
- $\nu \models \varphi$, (guard is satisfied)
- $\vec{\ell'} = \vec{\ell}[\ell_i := \ell'_i]$, (only i-th location changes)
- $\nu' = \nu[Y := 0]$, and ($\mathcal{A}_i$'s clocks are reset)
- $\nu' \models I_i(\ell'_i)$. (destination invariant holds)

(ii) **Synchronisation transition**:

$$\langle \vec{\ell}, \nu \rangle \xrightarrow{\tau} \langle \vec{\ell'}, \nu' \rangle$$

if there are $i, j \in \{1, \dots, n\}$, $i \neq j$, and $b \in B_i \cap B_j$, such that

- $(\ell_i, b!, \varphi_i, Y_i, \ell'_i) \in E_i$ and $(\ell_j, b?, \varphi_j, Y_j, \ell'_j) \in E_j$,
- $\nu \models \varphi_i \wedge \varphi_j$,
- $\vec{\ell'} = \vec{\ell}[\ell_i := \ell'_i][\ell_j := \ell'_j]$,
- $\nu' = \nu[Y_i \cup Y_j := 0]$, and
- $\nu' \models I_i(\ell'_i) \wedge I_j(\ell'_j)$.

Operational Semantics of Networks: Delay

(iii) **Delay transition**:

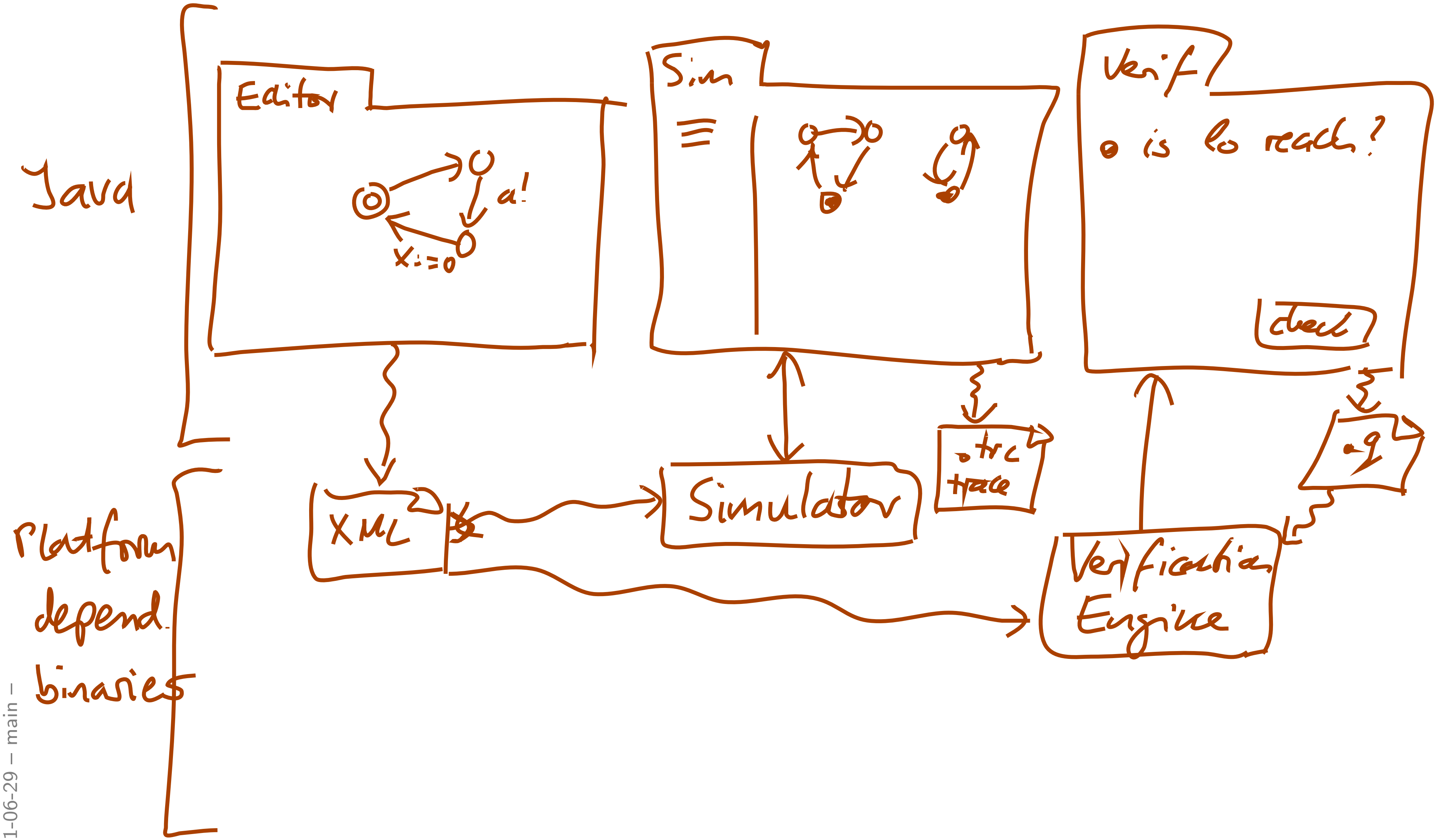
$$\langle \vec{\ell}, \nu \rangle \xrightarrow{t} \langle \vec{\ell}, \nu + t \rangle$$

if for all $t' \in [0, t]$,

- $\nu + t' \models \bigwedge_{k=1}^{n} I_k(\ell_k)$.

Uppaal [Larsen et al., 1997, Behrmann et al., 2004]
Demo, Vol. 1

Uppaal Architecture



References

References

[Behrmann et al., 2004] Behrmann, G., David, A., and Larsen, K. G. (2004). A tutorial on uppaal 2004-11-17. Technical report, Aalborg University, Denmark.

[Larsen et al., 1997] Larsen, K. G., Pettersson, P., and Yi, W. (1997). UPPAAL in a nutshell. *International Journal on Software Tools for Technology Transfer*, 1(1):134–152.

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