Real-Time Systems

Lecture 11: Networks of Timed Automata

2013-06-11

Dr. Bernd Westphal

Albert-Ludwigs-Universität Freiburg, Germany

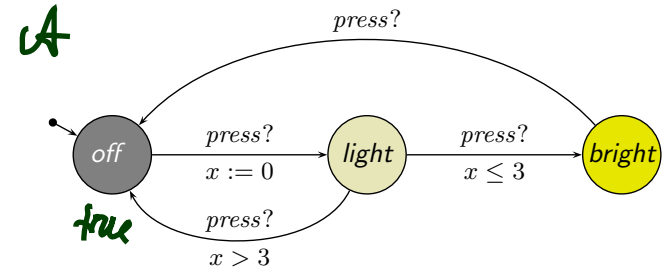# *Contents & Goals*

Last Lecture:

- Timed automata syntax

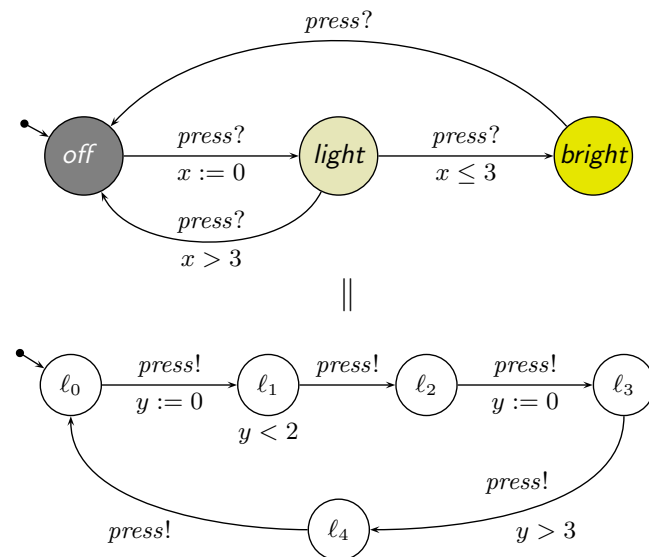- TA operational semantics

This Lecture:

- **Educational Objectives:** Capabilities for following tasks/questions.

  - what's the (syntactical) parallel composition of TA?

- **Content:**

  - 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
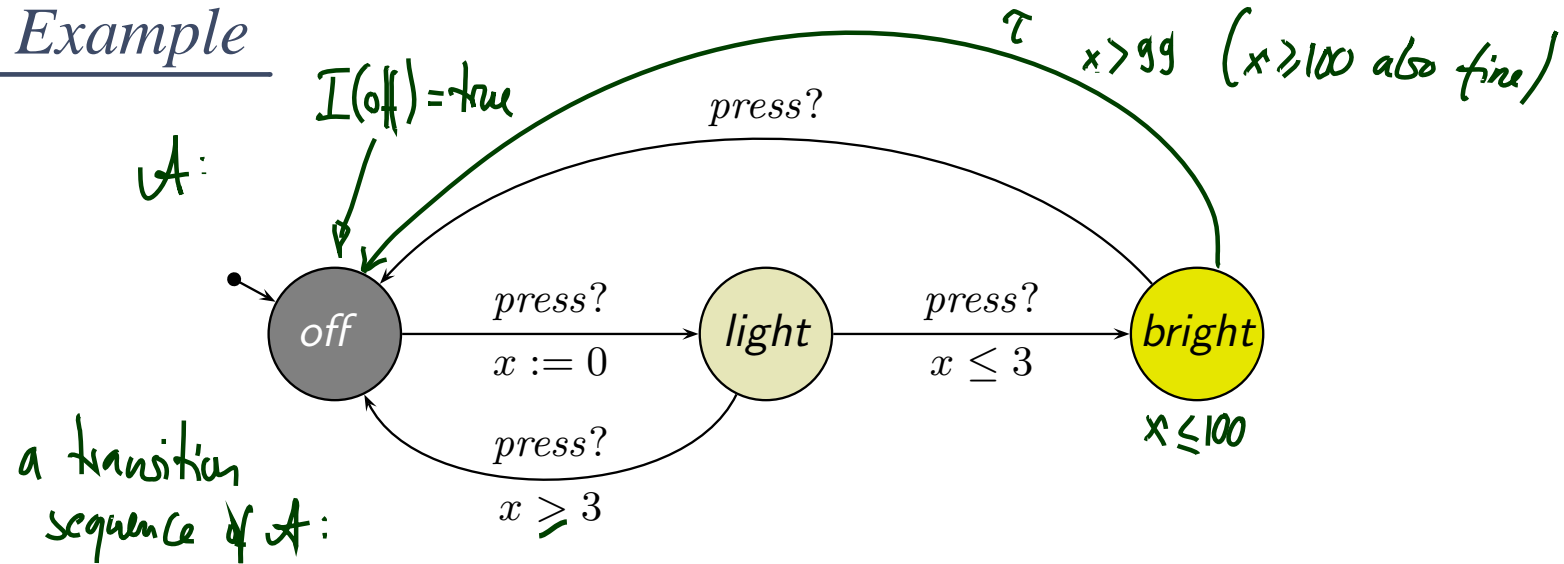- Uppaal Demo, part 1
- Extended timed automata

$$\mathcal{A}$$

$$press?$$

off $\xrightarrow[x := 0]{press?}$ light $\xrightarrow[x \leq 3]{press?}$ bright

true

$press?$
$x > 3$

$$\mathcal{J}(A) = \big(\text{Conf}(A), \, B_{?!} \cup \text{Time}, $$
$$\big/ \quad \{ \xrightarrow{\lambda} \mid \lambda \in B_{?!} \cup \text{Time} \}, \, C_{\text{ini}} \big\}$$
$$\langle \ell, \nu \rangle$$

off $\xrightarrow[x := 0]{press?}$ light $\xrightarrow[x \leq 3]{press?}$ bright

$press?$

$press?$
$x > 3$

$\parallel$

$\ell_0 \xrightarrow[y := 0]{press!} \ell_1 \xrightarrow{press!} \ell_2 \xrightarrow[y := 0]{press!} \ell_3$

$y < 2$

$press!$ $\qquad$ $press!$
$y > 3$

ℓ_4

$press!$

Network of TA

Recall: Pure Timed Automaton

Example

$A:$

$I(off) = true$

τ $x > 99$ $(x \geq 100$ also fine$)$

press?

off — press? $x := 0$ → light — press? $x \leq 3$ → bright
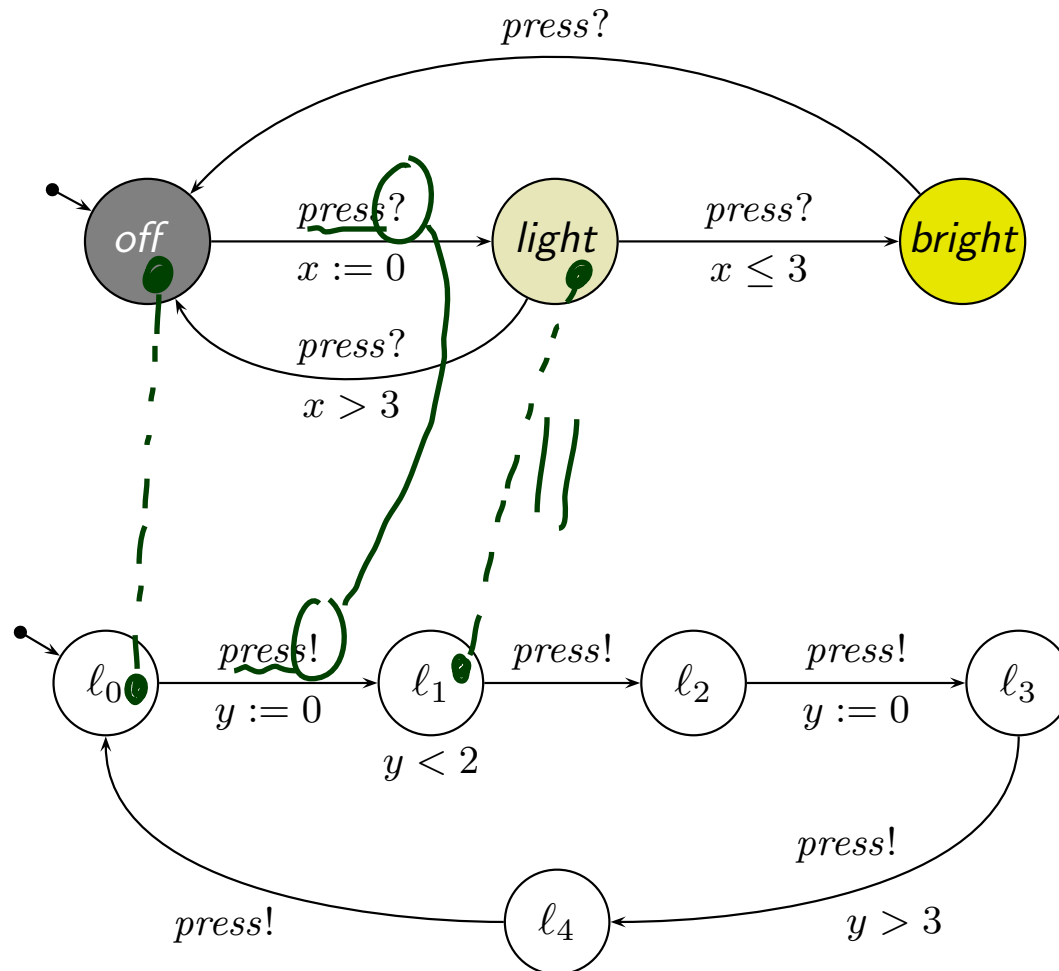
press? $x \geq 3$

$x \leq 100$

a transition sequence of A:

$\langle off, x = 0 \rangle \xrightarrow{2.5} \langle off, x = 2.5 \rangle \xrightarrow{1.7} \langle off, x = 4.2 \rangle \xrightarrow{10} \langle off, x = 14.2 \rangle \xrightarrow{0} \langle off, x = 14.2 \rangle$

ℓ_{ini} ν_0

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

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

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

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

NOT A CONFIGURATION

$\xrightarrow{press?} \langle bright, x = 0 \rangle \xrightarrow{127} \langle bright, x = 127 \rangle$

$\xrightarrow{100} \langle b, x = 100 \rangle \xrightarrow{\tau} \langle off, x = 100 \rangle$

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
- $I(\ell_1, \ell_2) := I_1(\ell_1) \wedge I_2(\ell_2)$, and

- E consists of **handshake** and **asynchronous communication**.
  ($\rightarrow$ **next slide**)

- The **complementation function**

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

  is defined pointwise as

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

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

$\mathcal{A}_1 \parallel \mathcal{A}_2 = (L_1 \times L_2, B_1 \cup B_2, X_1 \cup X_2, I, E, (\ell_{ini,1}, \ell_{ini,2}))$ 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 $\ell_2 \in L_2$,

  $$((\ell_1, \ell_2), \alpha, \varphi_1, Y_1, (\ell'_1, \ell_2)) \in E.$$
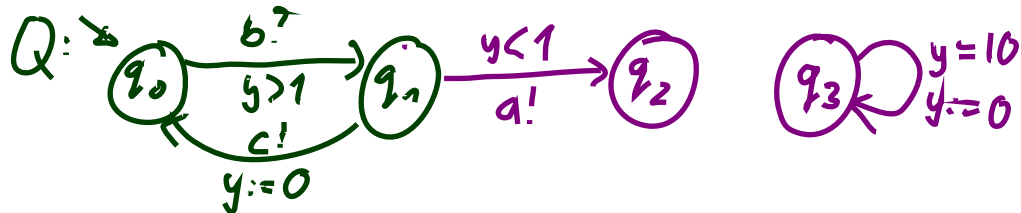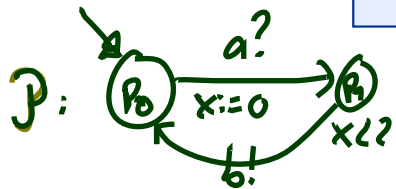
  If $(\ell_2, \alpha, \varphi_2, Y_2, \ell'_2) \in E_2$ then for all $\ell_1 \in L_1$,

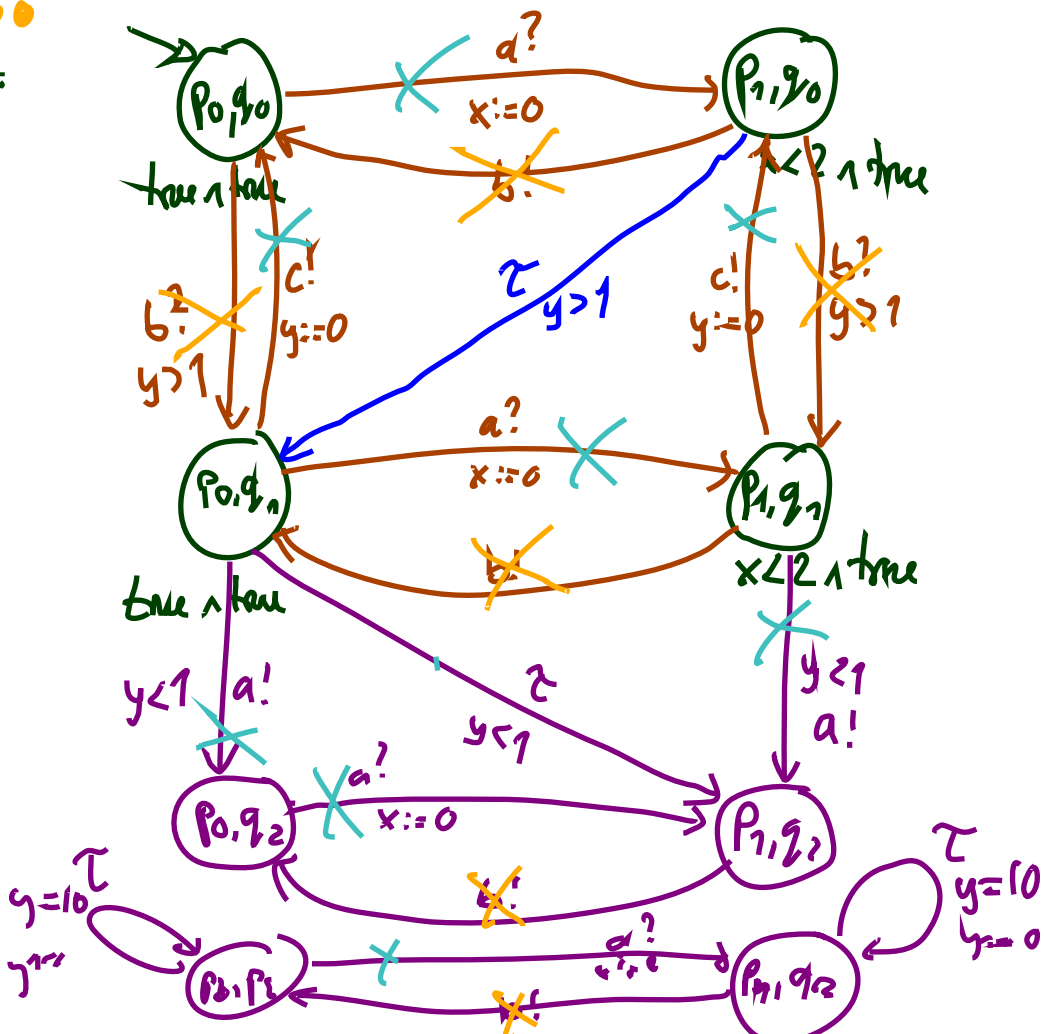 $$((\ell_1, \ell_2), \alpha, \varphi_2, Y_2, (\ell_1, \ell'_2)) \in E.$$

Example

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

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

$P:$

$Q:$

chan a, c
chan b

$P \| Q:$

$B = \{a, b, c\}$

$X = \{x, y\}$

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**:

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

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

$$\text{chan } b_1 \ldots b_m \bullet (\mathcal{A}_1 \parallel \ldots \parallel \mathcal{A}_n)$$
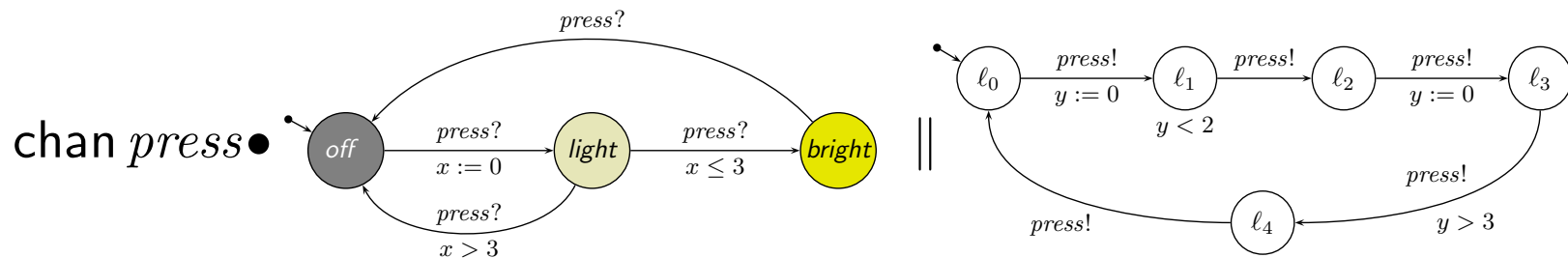
Closed Networks

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

- 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:

is closed.

$$\text{chan } b_1, \dots, b_k \bullet A_1 \parallel \dots \parallel A_n$$

$$\text{chan } b \bullet B$$

$$A_1 \parallel A_2$$

Def. 4.12

Lemma 4.16

A

Def. 4.4

$$J(N) = \left(\text{Conf}(N), \dots, \{ \xrightarrow{a} \mid \dots \}, C_{\text{init}} \right) = J(A) = \left(\text{Conf}(A), \dots, \{ \xrightarrow{a} \mid \dots \}, C_{\text{init}} \right)$$

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

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

yields the labelled transition system

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

with
- $X = \bigcup_{i=1}^{n} X_i,$
- $B = \bigcup_{i=1}^{n} B_i \setminus \{b_1, \ldots, b_m\},$
- $Conf(\mathcal{N}) = \{\langle \vec{\ell}, \nu \rangle \mid$
  $\vec{\ell} \in L_1 \times \cdots \times L_n \wedge \nu : X \to \text{Time} \wedge \nu \models \bigwedge_{k=1}^{n} I_k(\ell_k)\},$
- $C_{ini} = \{\langle (\ell_{ini,1}, \ldots, \ell_{ini,n}), \nu_{ini} \rangle\} \cap Conf(\mathcal{N})$
  where $\nu_{ini}(x) = 0$ for all $x \in X,$

- and three types of transition relations ($\to$ **next slides**).

For each $\lambda \in \text{Time} \cup B_{!?}$ the transition relation $\xrightarrow{\lambda} \subseteq Conf(\mathcal{N}) \times Conf(\mathcal{N})$ 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 $i \in \{1, \ldots, n\}$ 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)

tuple
modification

(ii) **Synchronisation transition**:

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

if there are $i, j \in \{1, \ldots, n\}$, $i \neq j$, 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
- $\nu' \models I_i(\ell_i') \wedge I_j(\ell_j')$.

(iii) **Delay transition**:

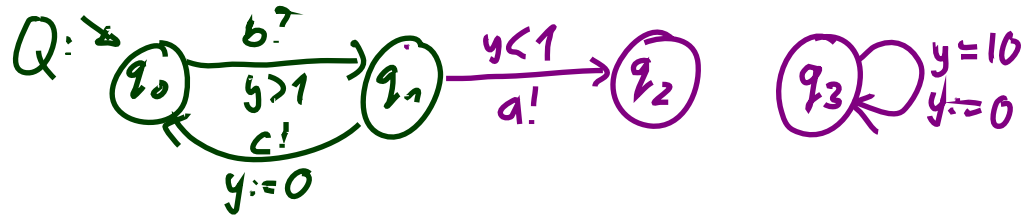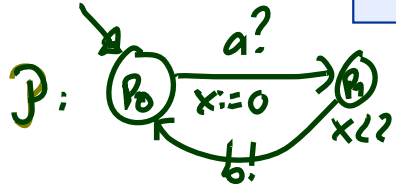$$\langle \vec{\ell}, \nu \rangle \xrightarrow{t} \langle \vec{\ell}, \nu + t \rangle$$

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

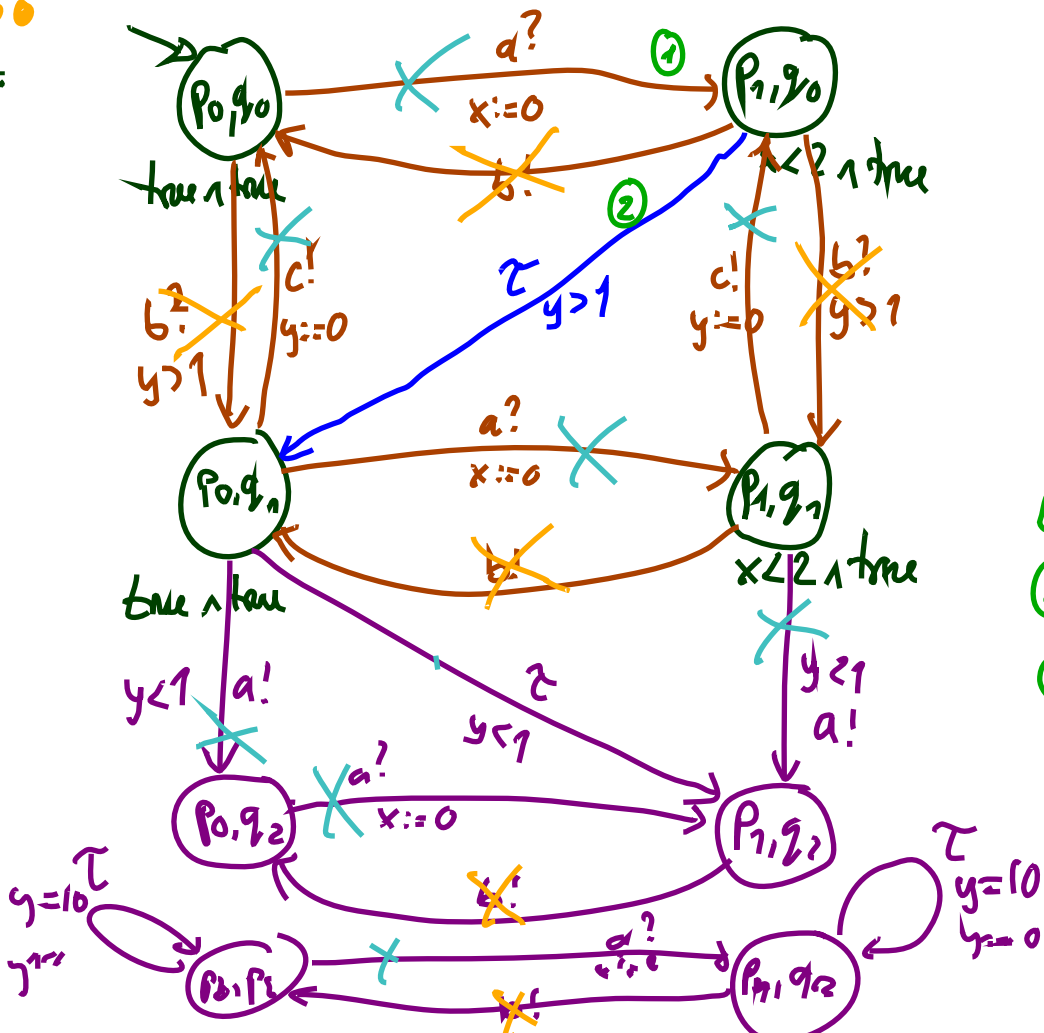- $\nu + t' \models \bigwedge_{k=1}^{n} I_k(\ell_k)$.

Example

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

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

chan a, c●
chan b●

$B = \{a, b, c\}$

$X = \{x, y\}$

Example transition sequence of chan b● $\mathcal{P} \| Q$:

$$\langle (P_0, q_0), \begin{smallmatrix} x=0 \\ y=0 \end{smallmatrix} \rangle \xrightarrow{27} \langle (P_0, q_0), \begin{smallmatrix} x=27 \\ y=27 \end{smallmatrix} \rangle$$

① $\xrightarrow{a?} \langle (P_1, q_0), \begin{smallmatrix} x=0 \\ y=27 \end{smallmatrix} \rangle$

② $\xrightarrow{\tau} \langle (P_0, q_1), \begin{smallmatrix} x=0 \\ y=27 \end{smallmatrix} \rangle \xrightarrow{10} \langle (P_0, q_1), \begin{smallmatrix} x=10 \\ y=37 \end{smallmatrix} \rangle$

③ $\xrightarrow{c!} \langle (P_0, q_0), \begin{smallmatrix} x=10 \\ y=0 \end{smallmatrix} \rangle \dots$

– 11 – 2013-06-11 – Stanet –

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

Uppaal Architecture
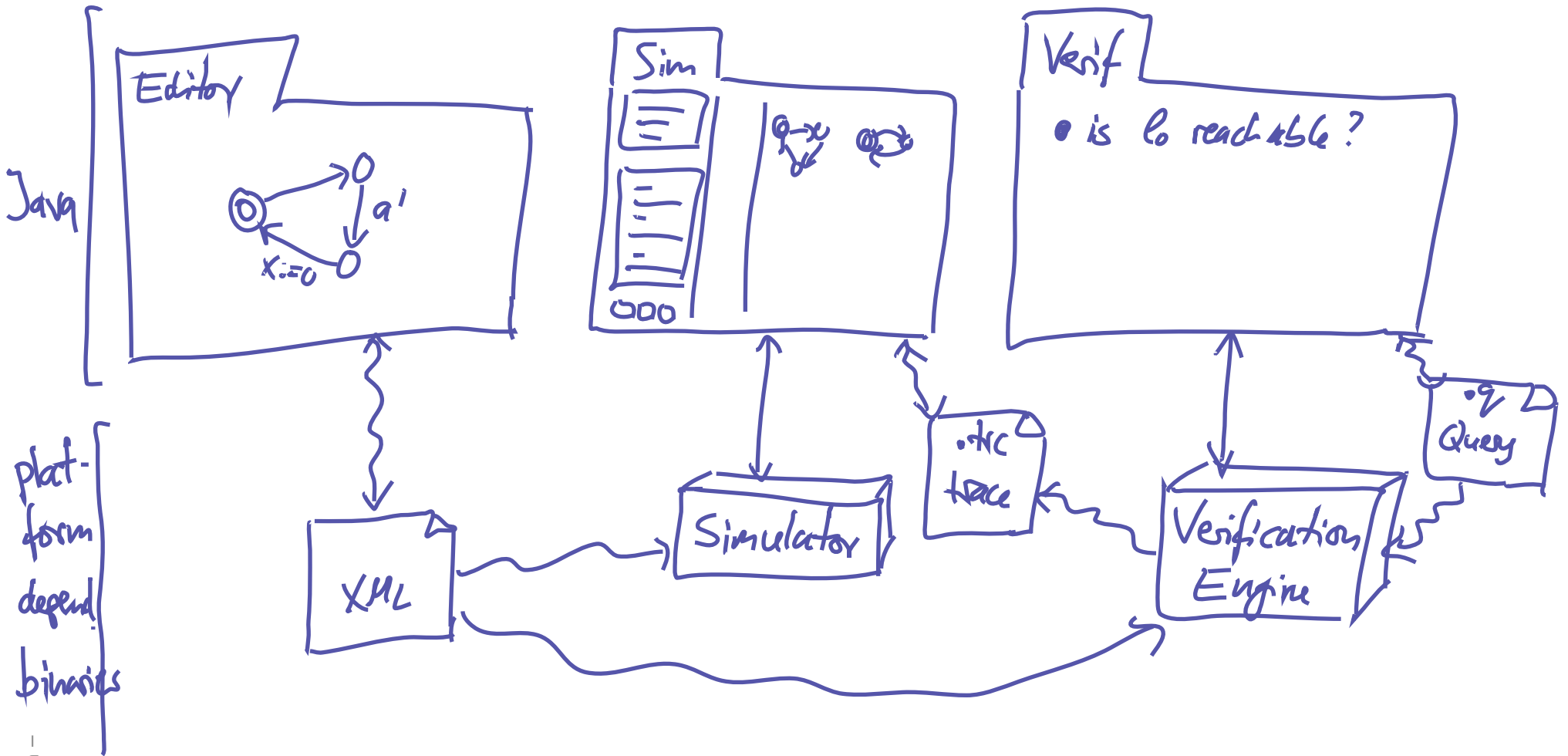
References

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.