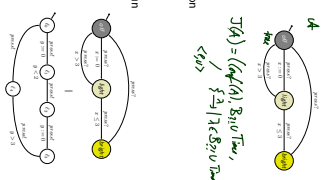


Contents & Goals

- 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

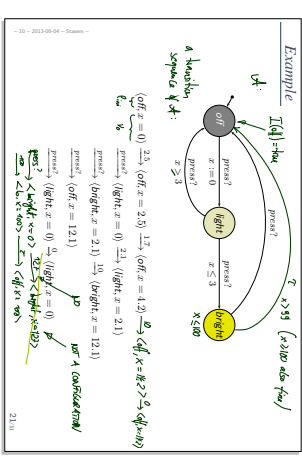
- Pure TA syntax
- channels, actions
- (simple) clock constraints
- Def: TA
- Pure TA operational semantics
- clock valuation, time shift, modification
- operational semantics
- discussion
- transition sequence, computation path, run
- network of TA
- parallel composition (syntactical)
- restriction
- network of TA semantics
- Uppaal Demo, part 1
- Extended timed automata



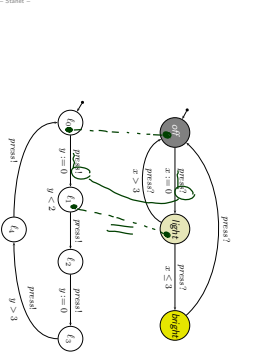
Network of TA



Recall: Pure Timed Automaton



Recall: Light Controller and User



Definition 4.12.

The parallel composition $\mathcal{A}_1 \parallel \mathcal{A}_2$ of two timed automata

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

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

$$\mathcal{A} = (L_1 \times L_2, B_1 \cup B_2, X_1 \cup X_2, I, E, (f_{m_1, 1}, f_{m_2, 2}))$$

where

- $I((t_1, t_2)) := I_1(t_1) \wedge I_2(t_2)$, and
- E consists of **handshake** and **asynchronous communication**.

(→ next slide)

- The complementation function

is defined pointwise as

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

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

$\mathcal{A}_1 \parallel \mathcal{A}_2 = (L_1 \times L_2, B_1 \cup B_2, X_1 \cup X_2, I, E, (f_{m_1, 1}, f_{m_2, 2}))$ with

- **Handshake:**

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

$$(t_1, \alpha, \varphi_1, X_1, t_1') \in E_1, \quad \text{and} \quad (t_2, \alpha, \varphi_2, X_2, t_2') \in E_2,$$

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



- **Asynchrony:**

If $(t_1, \alpha, \varphi_1, X_1, t_1') \in E_1$, then for all $t_2 \in L_2$,

$$((t_1, t_2), \alpha, \varphi_1, X_1, (t_1', t_2)) \in E.$$

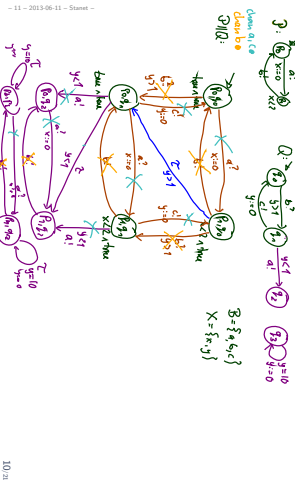
If $(t_2, \alpha', \varphi_2, X_2, t_2') \in E_2$ then for all $t_1 \in L_1$,

$$((t_1, t_2), \alpha', \varphi_2, X_2, (t_1, t_2')) \in E.$$

Example

$L \parallel \mathcal{A} = (L_1 \times L_2, B_1 \cup B_2, X_1 \cup X_2, I, E, (f_{m_1, 1}, f_{m_2, 2}))$

- If $\alpha \in B_1 \cup B_2$, $\tau \in \{a, \alpha, \varphi_1, X_1, t_1', (t_1, t_2') \in E_1, (t_2, \alpha, \varphi_2, X_2, t_2') \in E_2, (t_1, t_2') = (t_1, t_2)$, then $((t_1, t_2), \tau, \varphi_1 \wedge \varphi_2, X_1 \cup X_2, (t_1, t_2')) \in E$.
- If $(t_1, \alpha, \varphi_1, X_1, t_1') \in E_1$, then for all $t_2 \in L_2, ((t_1, t_2), \alpha, \varphi_1, X_1, (t_1, t_2')) \in E$, and **conv**.



Restriction

Definition 4.13.

A local channel b is introduced by the restriction operator which, for a timed automaton $\mathcal{A} = (L, B, X, I, E, f_{m, 1})$ yields

$$\text{chan } b \bullet \mathcal{A} := (L, B \setminus \{b\}, X, I, E', f_{m, 1})$$

where

- $(t, \alpha, \varphi, X, t') \in E'$
- if and only if $(t, \alpha, \varphi, X, t') \in E$ and $\alpha \notin \{b!, b?\}$.

- Abbreviation:

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

Networks of Timed Automata

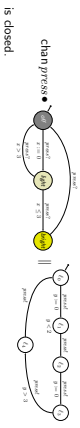
- A timed automaton \mathcal{N} is called **network of timed automata** if and only if it is obtained as

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

- A network $N = \text{chan } h_1 \dots h_m \bullet (A_1 \parallel \dots \parallel A_n)$ is called **closed** if and only if
$$\{h_1, \dots, h_m\} = \bigcup_{i=1}^n B_i.$$

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:



Lemma 4.16. Let $\mathcal{A} = (L_1, B_1, X_1, I_1, B_1, C_{int,1})$ with $i = 1, \dots, n$ be a set of timed automata with disjoint clocks. Then the operational semantics of the network
$$\text{chan } h_1 \dots h_m \bullet (A_1 \parallel \dots \parallel A_n)$$
 yields the labelled transition system $(\text{Conf}(N), \text{Time} \cup B_{int}, \{\Delta\} \mid \lambda \in \text{Time} \cup B_{int}\}, C_{int})$ with

- $X = \bigcup_{i=1}^n X_i,$
- $B = \bigcup_{i=1}^n B_i \setminus \{h_1, \dots, h_m\},$
- $\text{Conf}(N) = \{(\vec{t}, \nu) \mid \ell \in L_1 \times \dots \times L_n \wedge \nu : X \rightarrow \text{Time} \wedge \nu \models \bigwedge_{i=1}^n I_i(k_i), C_{int} = \{((k_{int,1}, \nu_{int,1}), \nu_{int,1}) \cap \text{Conf}(N) \mid \text{where } \nu_{int,i}(x) = 0 \text{ for all } x \in X_i\}$
- and three types of transition relations (\leftarrow next slides).

For each $\lambda \in \text{Time} \cup B_i$, the transition relation $\Delta \subseteq \text{Conf}(N) \times \text{Conf}(N)$ has one of the following three types:

(i) **Local transition:**

- if there is $i \in \{1, \dots, n\}$ such that
 - $(\ell_i, \alpha, \varphi, X_i, \ell'_i) \in E_i, \alpha \in B_i,$
 - $\nu \models \varphi,$
 - $\vec{t} = \vec{t} \uparrow_i := \vec{t}_i,$
 - $\nu' = \nu \uparrow_i \cup \nu'' := 0_i,$ and
 - $\nu' \models I_i(\ell'_i).$
- (i -th automaton has corresp. edge (guard is satisfied) (only i -th location changes) (A_i 's clocks are reset) (destination invariant holds))
- Handwritten note:* lokal wandlung

(ii) **Synchronisation transition:**

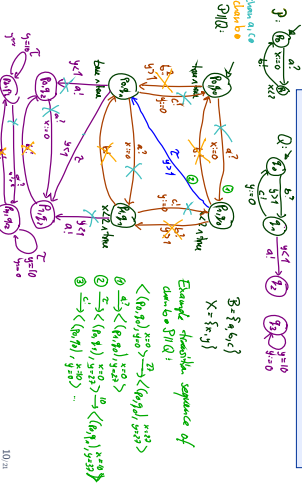
- if there are $i, j \in \{1, \dots, n\}, i \neq j$, and $b \in B_i \cap B_j$, such that
 - $(\ell_i, b_i, \varphi_i, X_i, \ell'_i) \in E_i$ and $(\ell_j, b_j, \varphi_j, X_j, \ell'_j) \in E_j,$
 - $\nu \models \varphi_i \wedge \varphi_j,$
 - $\vec{t} = \vec{t} \uparrow_i := \vec{t}_i,$
 - $\nu' = \nu \uparrow_i \cup \nu'' := 0_i,$ and
 - $\nu' \models I_i(\ell'_i) \wedge I_j(\ell'_j).$
- (i, j -th automata sync on b)

(iii) **Delay transition:**

- if for all $i \in \{0, 1\},$
 - $\nu + t \models \bigwedge_{k=1}^n I_i(k_i).$
- $(\vec{t}, \nu) \xrightarrow{t} (\vec{t}, \nu + t)$

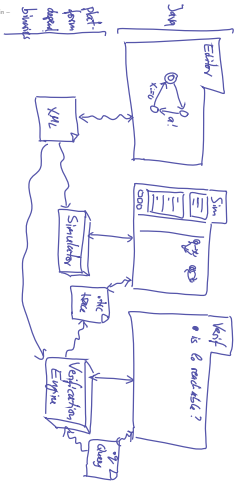
Example

$L[A] = (A_1 \times A_2, B_1 \cup B_2, X \cup X', L, E, (a_{init}, a_{end}))$
 • If $a \in B_1 \cup B_2$ s.t. $(a, a, a), Y_1(a) \in B_1, (a, a, a), Y_2(a) \in B_2, (a, a, a) = (a, a)$,
 then $(a, a, a) \in \text{Pr}[Q]$
 • If $(a, a, a), Y_1(a) \in B_1, (a, a, a), Y_2(a) \in B_2, (a, a, a) \in E$, and conv



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

Uppaal Architecture



References

[Behrmann et al., 2004] Behrmann, G., David, A., and Larsen, K. G. (2004). A tutorial on uppal. 2004-11-17. Technical report, Aalborg University, Denmark.
 [Larsen et al., 1997] Larsen, K. G., Petersson, 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.