Cyber-Physical Systems
(automata-based modelling)

Renato Neves

Universidade do Minho

HASLab
HIGH-ASSURANCE
SOFTWARE LABORATORY

Architecture and Calculi Course Unit

Table of Contents

Recall the need for time-critical systems

Specifying an airbag saying that in a car crash the airbag eventually inflates maybe not enough, but:

in a car crash the airbag eventually inflates within 20ms

> *Correctness in time-critical systems not only depends on the logical result of the computation, but also on the time at which the results are produced*
>
> [Baier & Katoen, 2008]

What about this case?

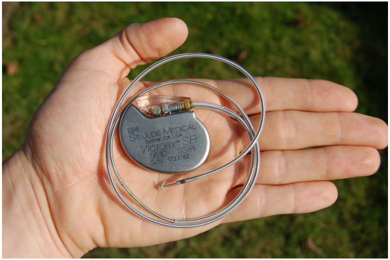
A thermostat reaches the target temperature within 5 min

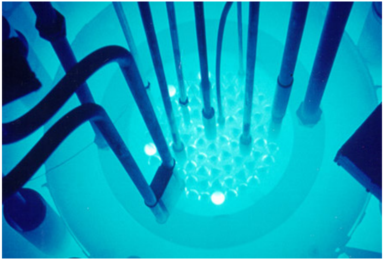
Two physical processes involved: time and temperature

> We shift from time-critical systems to systems that closely interact with physical processes other than time.
>
> [Lee & Seshia, 2017]

Cyber-Physical Systems

Distributed devices that closely interact with their physical environment

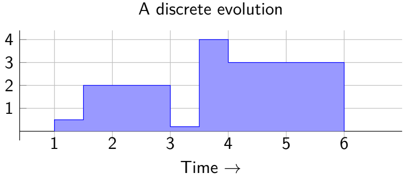


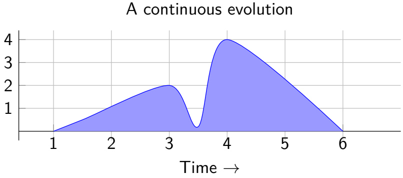


The challenge underlying cyber-physical systems

Cyber-Physical systems

intertwine discrete with continuous behaviour.





Discrete evolution is treated by classical models of computation

Continuous evolution is treated by differential equations

How to combine both formalisms?

Table of Contents

A cheatsheet on differential equations

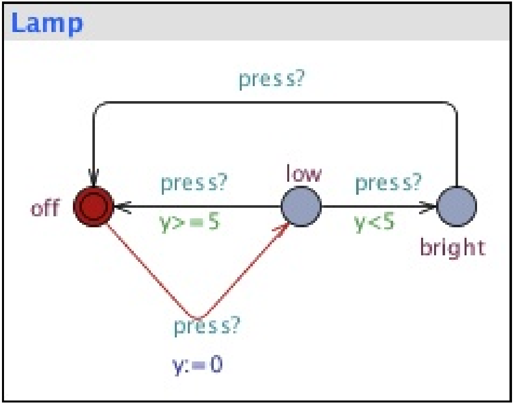
$\dot{x} = 1$: x 'grows' with velocity 1; represents the passage of time.

$\dot{p} = v, \dot{v} = a$: position (p) varies according to velocity; velocity (v) varies according to acceleration (a).

$\dot{x} = x$: what about this case?

Recall timed automata

A Lamp



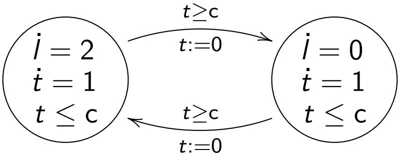
A formalism for cyber-physical systems

Hybrid Automata

Classical automata enriched with machinery to specify continuous evolutions and discrete resets [Henzinger' 96].

Example

Water Level Regulator



Recall the definition of timed automata

Definition

$$\langle L, L_0, Act, C, Tr, Inv \rangle$$

where

- L is a set of locations, and $L_0 \subseteq L$ the set of initial locations
- Act is a set of actions and C a set of clocks
- $Tr \subseteq L \times \mathcal{C}(C) \times Act \times \mathcal{P}(C) \times L$ is the transition relation

  $$\ell_1 \xrightarrow{g,a,U} \ell_2$$

  denotes a transition from location ℓ_1 to ℓ_2, labelled by a, enabled if guard g is valid, which, when performed, resets the set U of clocks
- $Inv : L \longrightarrow \mathcal{C}(C)$ is the assigment of invariants to locations

The definition of hybrid automata

Definition

$$\langle L, L_0, Act, X, Tr, Inv, Dyn \rangle$$

where

- L is a set of locations, and $L_0 \subseteq L$ the set of initial locations
- Act is a set of actions and X is a set of variables $\{x_1, \ldots, x_n\}$
- $Tr \subseteq L \times \mathcal{C}(X) \times Act \times Cmd(X) \times L$ is the transition relation

  $$\ell_1 \xrightarrow{g,a,c} \ell_2$$

  denotes a transition from location ℓ_1 to ℓ_2, labelled by a, enabled if guard g is valid, which, when performed, applies command c
- $Inv : L \longrightarrow \mathcal{C}(C)$ is the assigment of invariants to locations
- $Dyn : L \longrightarrow DiffEq(X)$ is a function that associates to every location a system of differential equations

The ...

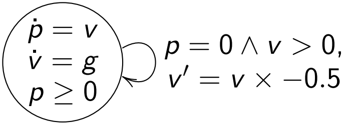
Example

$$\dot{p} = v$$
$$\dot{v} = g$$
$$p \geq 0$$

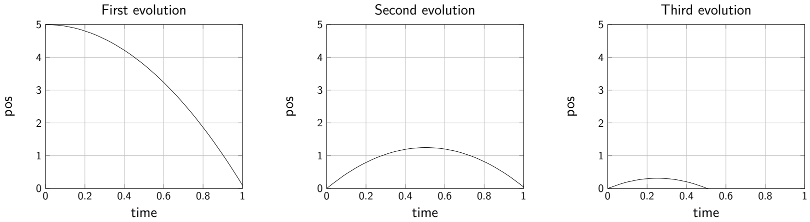
$$p = 0 \wedge v > 0,$$
$$v' = v \times -0.5$$

The ...

Example



$\dot{p} = v$
$\dot{v} = g$
$p \geq 0$
$p = 0 \wedge v > 0,$
$v' = v \times -0.5$



First evolution
Second evolution
Third evolution
pos
time
0
1
2
3
4
5
0.2
0.4
0.6
0.8

Exercise

We wish to model a cruise controller whose goal is to reach and mantain the velocity of $10m/s$. However, we need to comply with the following restrictions:

1. the controller can only accelerate at $2m/s^2$ or break at $-2m/s^2$
2. the controller cannot change twice its execution mode in less than one second.

Parallel composition of hybrid automata

Similarly to timed automata,

- action labels serve as channel identifiers,
- communication is achieved by forced handshaking over a subset of common actions.

Parallel composition of hybrid automata

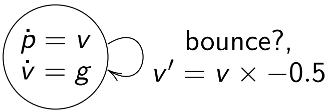
Let $H \subseteq Act_1 \cap Act_2$. The parallel composition of ha_1 and ha_2 synchronizing on H is the hybrid automata

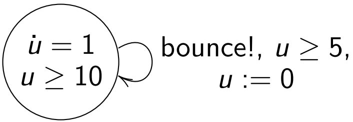
$$ha_1 \parallel_H ha_2 := \langle L_1 \times L_2, L_{0,1} \times L_{0,2}, Act_{\parallel_H}, X_1 + X_2, Tr_{\parallel_H}, Inv_{\parallel_H}, Dyn_{\parallel_H} \rangle$$

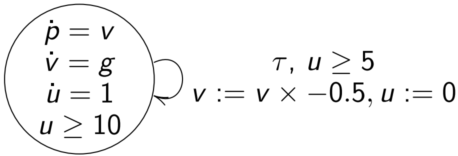
where

- $Act_{\parallel_H} = ((Act_1 \cup Act_2) - H) \cup \{\tau\}$
- $Inv_{\parallel_H}\langle \ell_1, \ell_2 \rangle = Inv_1(\ell_1) \wedge Inv_2(\ell_2)$
- $Tr_{\parallel_H}$ is given by:
  - $\langle \ell_1, \ell_2 \rangle \xrightarrow{g,a,c} \langle \ell_1', \ell_2 \rangle$ if $a \notin H \wedge \ell_1 \xrightarrow{g,a,c} \ell_1'$
  - $\langle \ell_1, \ell_2 \rangle \xrightarrow{g,a,c} \langle \ell_1, \ell_2' \rangle$ if $a \notin H \wedge \ell_2 \xrightarrow{g,a,c} \ell_2'$
  - $\langle \ell_1, \ell_2 \rangle \xrightarrow{g,\tau,c_1 + c_2} \langle \ell_1', \ell_2' \rangle$ if $a \in H \wedge \ell_1 \xrightarrow{g_1,a,c_1} \ell_1' \wedge \ell_2 \xrightarrow{g_2,a,c_2} \ell_2'$ with $g = g_1 \wedge g_2$
- $Dyn_{\parallel_H}\langle \ell_1, \ell_2 \rangle = Dyn_1(\ell_1) \wedge Dyn_2(\ell_2)$

The bouncing ball revisited







Exercise

Recall the previous exercise in which we modelled a cruise controller.

One requirement was that the execution modes could not change twice in less than one second.

Consider now the case in which the cruise controller waits for an external signal to switch between execution modes.

Additionally, consider a system that gives such a signal every half a second.

Calculate the parallel composition of this system and the modified cruise controller.

Table of Contents

Timed Labelled Transition Systems

Syntax	Semantics
How to write	*How to execute*
Hybrid Automaton	TLTS (Timed LTS)

Timed Labelled Transition Systems

Syntax	Semantics
How to write	*How to execute*
Hybrid Automaton	TLTS (Timed LTS)

Timed LTS

Introduce delay transitions to capture the passage of time within a LTS:

$s \xrightarrow{a} s'$ for $a \in Act$, are ordinary transitions due to action occurrence

$s \xrightarrow{d} s'$ for $d \in \mathcal{R}^+$, are delay transitions

subject to a number of constraints, eg,

Dealing with time in system models

Timed LTS

- time additivity

$$(s \xrightarrow{d} s' \wedge 0 \leq d' \leq d) \;\Rightarrow\; s \xrightarrow{d'} s'' \xrightarrow{d-d'} s' \text{ for some state } s''$$

- delay transitions are deterministic

$$(s \xrightarrow{d} s' \wedge s \xrightarrow{d} s'') \;\Rightarrow\; s' = s''$$

Semantics of Hybrid Automata

Semantics of HA:

Every HA *ha* defines a TLTS

$$\mathcal{H}(ta)$$

whose states are pairs

$$\langle \text{location}, \text{variable valuation} \rangle$$

with infinitely, even uncountably many states

Variable valuations

Definition

A valuation valuation η for a set of variables X is a function

$$\eta : X \longrightarrow \mathcal{R}$$

assigning to each variable $x \in X$ its current value $\eta\, x$.

Satisfaction of variable constraints

$$\begin{aligned}
\eta \models x \,\square\, n &\iff \eta\, x \,\square\, n \\
\eta \models x - y \,\square\, n &\iff (\eta\, x - \eta\, y) \,\square\, n \\
\eta \models g_1 \wedge g_2 &\iff \eta \models g_1 \wedge \eta \models g_2
\end{aligned}$$

Some syntatic sugar

Solution

For every system of differential equations $Dyn(I)$ we assume the existence of a solution $sol(Dyn(I)) : \mathcal{R}^X \times \mathcal{R}_0^+ \to \mathcal{R}^X$ for this system.

Resets

The result $\eta[c]$ of applying a command c to a valuation η is given by

$$\eta[c]\, x \;=\; c[\eta(x_1)/x_1 \ldots \eta(x_n)/x_n]_x$$

Example

Assume that $\eta(x_1) = 1$ and $\eta(x_2) = 2$. Then,

$$(x_1 := x_1{+}x_2, x_2 := 0)_{x_1}[\eta(x_1)/x_1, \eta(x_2)/x_2]_{x_1} = (x_1 := 1{+}2, x_2 := 0)_{x_1} = 3$$

From ha to $\mathcal{H}(ha)$

Let $ha = \langle L, L_0, Act, X, Tr, Inv, Dyn \rangle$

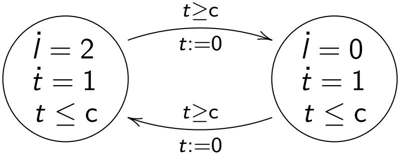
$$\mathcal{T}(ta) \; = \; \langle S, S_0 \subseteq S, N, T \rangle$$

where

- $S = \{\langle l, \eta \rangle \in L \times \mathcal{R}^X \mid \eta \models Inv(l)\}$
- $S_0 = \{\langle \ell_0, \eta \rangle \mid \ell_0 \in L_0 \;\wedge\; \eta\, x = 0 \text{ for all } x \in X\}$
- $N = Act + \mathcal{R}_0^+$ (ie, transitions can be labelled by actions or delays)
- $T \subseteq S \times N \times S$ is given by:

$$\begin{array}{rcl}
\langle l, \eta \rangle \xrightarrow{a} \langle l', \eta' \rangle & \Leftarrow & \exists_{l \xrightarrow{g,a,c} l' \in Tr} \;\; \eta \models g \;\wedge\; \eta' = \eta[c] \;\wedge\; \eta' \models Inv(l') \\
\langle l, \eta \rangle \xrightarrow{d} \langle l, \eta' \rangle & \Leftarrow & \exists_{d \in \mathcal{R}_0^+} \;\; \eta' = sol(Dyn(l))(\eta, d) \\
 & \wedge & \forall_{t \in [0,d]} sol(Dyn(l))(\eta, d) \models Inv(l)
\end{array}$$

Water level regulator revisited



$$S = \{\langle 1, \langle v_1, v_2 \rangle\rangle \mid v_2 \leq c\} \cup \{\langle 2, \langle v_1, v_2 \rangle\rangle \mid v_2 \leq c\}$$

$$\begin{aligned}
\langle 1, \langle v_1, v_2 \rangle\rangle \xrightarrow{d} \langle 1, \langle v_1 + 2, v_2 + 1 \rangle\rangle \quad &\Leftarrow \quad v_2 + 1 \leq c \\
\langle 1, \langle v_1, v_2 \rangle\rangle \xrightarrow{\star} \langle 2, \langle v_1, 0 \rangle\rangle \quad &\Leftarrow \quad v_2 \geq c \wedge v_2 \leq c
\end{aligned}$$

Table of Contents

Bisimulation

Timed bisimulation (between states of timed LTS)

A relation R is a timed simulation iff whenever $\langle l_1, \eta_1\rangle R \langle l_2, \eta_2\rangle$, for any action a and delay d,

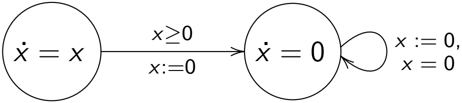
$$\langle l_1, \eta_1\rangle \xrightarrow{a} \langle l_1', \eta_1'\rangle \;\Rightarrow\; \text{there is a transition} \quad \langle l_2, \eta_2\rangle \xrightarrow{a} \langle l_2', \eta_1'\rangle \wedge \langle l_1', \eta_1'\rangle R \langle l_2', \eta_1'\rangle$$

$$\langle l_1, \eta_1\rangle \xrightarrow{d} \langle l_1', \eta_1'\rangle \;\Rightarrow\; \text{there is a transition} \quad \langle l_2, \eta_2\rangle \xrightarrow{d} \langle l_2', \eta_1'\rangle \wedge \langle l_1', \eta_1'\rangle R \langle l_2', \eta_1'\rangle$$

And a timed bisimulation if its converse is also a timed simulation.

Exercise

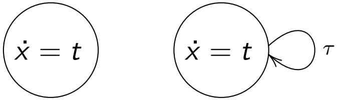
Can you minimize the automaton below into a automaton with a single state?



Limitation of bisimulation

Bisimulation for hybrid automata is often too strict:

- It forces two hybrid automata to always match jumps, *e.g.* the two hybrid automata below are different from the point of view of bisimulation



- jumps must occur at exactly the same time.

There are several variants of hybrid automata (probabilistic, weighted . . .), but to a large extent,

$\boxed{\text{no uniform theory of bisimulation for hybrid automata}}$