Contents & Goals

Last Lecture:
• PLC, PLC automata

This Lecture:
• Educational Objectives:
  • what's notable about TA syntax? What's simple clock constraint?
  • what's a configuration of a TA? When are two in transition relation?
  • what's the difference between guard and invariant? Why have both?
  • what's a computation path? A run? A Zeno behaviour?

Content:
• Timed automata syntax
• TA operationalsemantics
Example

Press $x = 0$ press $x \leq 3$

Press $x > 3$

Problems:
- Deadlock freedom [Behrmann et al., 2004]
- Location reachability ("Is this user able to reach 'bright'?")
- Constraint reachability ("Can the controller's clock go past 5?")

Plan
- Pure TA syntax
- Channels, actions
- (Simple) clock constraints
- Def. TA operation semantics
- Clock valuation, timeshift, modification
- Operation semantics
- Discussion
- Transition sequence, computation path, run
- Network of TA semantics
- Uppaal Demo
- Region abstraction; zones
- Extended TA; Logic of Uppaal

Pure TA Syntax

Channel Names and Actions

To define timed automata formally, we need the following sets of symbols:

- A set $\{a, b \in \text{Chan}\}$ of channel names or channels.
- For each channel $a \in \text{Chan}$, two visible actions: $a?$ and $a!$ denote input and output on the channel ($a?$, $a!$ $\in \text{Chan}$).
- $\tau$ $\in \text{Chan}$ represents an internal action, not visible from outside.

- $\alpha, \beta \in \text{Act} = \{a? | a \in \text{Chan}\} \cup \{a! | a \in \text{Chan}\} \cup \{\tau\}$ is the set of actions.

- An alphabet $B$ is a set of channels, i.e. $B \subseteq \text{Chan}$.
- For each alphabet $B$, we define the corresponding action set:
  - $B?? = \{a? | a \in B\} \cup \{a! | a \in B\} \cup \{\tau\}$. Note: $\text{Chan}?? = \text{Act}$.
\[ \phi = \nu \iff \psi = \psi (v) \land v \leq x \]

If and only if \( \phi = \nu \), we have \( \psi = \psi (v) \land v \leq x \). We have two clock constraints.

\[ \langle L, B, X, I, E, \ell \rangle \]

Let \( (L, B, X, I, E, \ell) \)

Clock Valuations

Clock constraint of the form \( \langle \ell \rangle \).

Initial state \( \ell \) is a finite set of clocks.

The set \( \{ \ell \} \) is a finite set of clocks.

Example

Clock Transition System

Simple Clock Constraint
Example Transition Sequences, Reachability

Discrete transition sequences are any finite or infinite sequence of the form $\langle \ell, \nu \rangle$.

A discrete configuration is a pair $(X, \nu)$.

Example 1

Discrete transition sequences are any finite or infinite sequence of the form $\langle \ell, \nu \rangle$.

A discrete configuration is a pair $(X, \nu)$.

Example 2

Discrete transition sequences are any finite or infinite sequence of the form $\langle \ell, \nu \rangle$.

A discrete configuration is a pair $(X, \nu)$.

Example 3

Discrete transition sequences are any finite or infinite sequence of the form $\langle \ell, \nu \rangle$.

A discrete configuration is a pair $(X, \nu)$.

Example 4

Discrete transition sequences are any finite or infinite sequence of the form $\langle \ell, \nu \rangle$.

A discrete configuration is a pair $(X, \nu)$.

Example 5

Discrete transition sequences are any finite or infinite sequence of the form $\langle \ell, \nu \rangle$.

A discrete configuration is a pair $(X, \nu)$.
Computation Path

A sequncet of time-stamped configurations...
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 a 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 behavior (or unboundedness or progress): $\forall t \in \text{Time} \exists i \in \mathbb{N}_0: t < t_i$

Example: $\ell \leq 2$; $x < 10$, $x := 0$; $x \geq 10$.

References