**UPPAAL's Modeling and Specification Language**

Paul Pettersson
BRICS@Aalborg

---

**Timed Automata**

Clocks: \( x, y \)

Guard: boolean combination of \( x \) with integer bounds

State: \((\text{location}, x, y)\) where \( x, y \) are in \( \mathbb{R} \)

Transitions:
- \((n, x = 2.4, y = 3.1415) \rightarrow (m, x = 3.5, y = 4.2415)\)
- \((n, x = 2.4, y = 3.1415) \rightarrow (n, x = 3.5, y = 4.2415)\)

---

**Timed Safety Automata**

Timed Automata + Invariants

Clocks: \( x, y \)

Invariants:
- \( x \leq 5 \)
- \( y \leq 10 \)

Transitions:
- \((n, x = 2.4, y = 3.1415) \rightarrow (n, x = 0, y = 3.1415)\)
- \((n, x = 2.4, y = 3.1415) \rightarrow (m, x = 3.5, y = 4.2415)\)
- \((m, x = 0, y = 3.1415) \rightarrow (m, x = 3.5, y = 4.2415)\)

---

**Timed Automata in UPPAAL**

- Timed (Safety) Automata
- urgent actions
- urgent locations
- committed locations
- data-variables (with bounded domains)
- arrays of data-variables + constants
- guards and assignments over data-variables and arrays...
- templates with local clocks, data-variables, and constants.

---

**Declarations in UPPAAL**

```plaintext
clock x, ..., xn;
int i, ..., in;
chan a1, ..., ao;
const c1 n1, ..., cp np;
```

Examples:
- clock x, y;
- int i, J0: int[0,1] k[5];
- const delay 5, true 1, false 0;

Array \( k \) of five booleans.
Timed Automata in UPPAAL

\[ x = n \]

\[ x \leq 5 \land y > 3 \]

\[ x := 0 \]

\[ x \leq 5 \]

\[ y \leq 10 \]

\[ g_1 \]

\[ g_2 \]

\[ g_3 \]

\[ g_4 \]

\[ \text{invariant} \]

\[ \text{clock guards} \]

\[ \text{data guards} \]

\[ \text{clock assignments} \]

Urgent Channels

\[ \text{urgent chan hurry;} \]

Informal Semantics:
• There will be no delay if transition with urgent action can be taken.

Restrictions:
• No clock guard allowed on transitions with urgent actions.
• Invariants and data-variable guards are allowed.

Urgent Locations

Click “Urgent” in State Editor.

Informal Semantics:
• No delay in urgent location.

Note: the use of urgent locations reduces the number of clocks in a model, and thus the complexity of the analysis.

Committed Locations

Click “Committed” in State Editor.

Informal Semantics:
• No delay in committed location.
• Next transition must involve automata in committed location.

Note: the use of committed locations reduces the number of clocks in a model, and allows for more space and time efficient analysis.

UPPAAL Specification Language

\[ \text{A}[] p \]

\[ \text{E<>} p \]

\[ (\text{AG} p) \]

\[ (\text{EF} p) \]

\[ p ::= a . l \mid g d \mid g c \mid p \land p \mid p \lor p \mid \text{not} \ p \mid p \implies p \mid (p) \]