Statistical Model Checking in UPPAAL

Alexandre David, Kim G. Larsen, Marius Mikucionis
Axel Legay, Wang Zheng, Peter Bulychev,
Jonas van Vliet, Danny Poulsen,
Dehui Du, Guangyuan Li

CAV 11, PDMC 11, FORMATS 11,
QAPL12, LPAR12, iWIGPL12,
RV12, FORMATS12, HBS12,
ISOLA12, SCIENCE China,
NFM13, RV13, AVOCS13
\[ \forall (i : \text{id}_t) \forall (j : \text{id}_t) \text{Train}(i).\text{Cross} \land \text{Train}(j).\text{Cross} \implies i = j \]

Safety

\[ \exists (\text{Train}(0).\text{Cross}) \land \exists (\text{Train}(1).\text{Stop}) \]

Reachability

\[ \text{Train}(0).\text{Appr} \to \text{Train}(0).\text{Cross} \]

Liveness

\[ \text{Pr}[\text{Time} \leq 500 \land \text{Train}(0).\text{Cross}] \geq 0.7 \]

Performance properties

\[ \text{Pr}[\text{Train}(0).\text{Appr} \to \text{Time} \leq 100 \land \text{Train}(0).\text{Cross}] \geq 0.4 \]

Limited quantitative analysis

State-space explosion

VTSA Summer School, 2013.
Kim Larsen [2]
Performance properties

\[ \Pr[\leq 200](\langle\rangle \text{Train}(5).\text{Cross}) \]

\[ \Pr[\leq 100](\langle\rangle \text{Train}(0).\text{Cross}) \geq 0.8 \]

\[ \Pr[\leq 100](\langle\rangle \text{Train}(5).\text{Cross}) \geq \Pr[\leq 100](\langle\rangle \text{Train}(1).\text{Cross}) \]

State-space explosion

Generate runs

Performance properties

State-space explosion

VTSA Summer School, 2013.
Kim Larsen [3]
Overview

- Stochastic Semantics of Networks of Timed Automata
- Statistical Model Checking in UPPAAL
  - Estimation
  - Sequential Hypothesis Testing
  - Sequential Probability Comparison
  - Parameterized Probability Comparison
- SMC of Hybrid Automata
- Case Studies & Demo
Stochastic Semantics of TA

Exponential Distribution

Safe
- apr[id]!
- x=0

Appr
- x<=20

x<=10
- stop[id]?
- x=0

Input enabled

Uniform Distribution

Cross
- x<=5
- x=7
- x=0

Composition = Repeated races between components

Stochastic Semantics of Timed Automata

Composition = Race between components for outputting

VTSA Summer School, 2013.
Stochastic Semantics of Timed Automata

Assumptions:
Component TAs are:
• Input enabled
• Deterministic
• Disjoint set of output actions

\[ \pi(s, a_1 a_2 \ldots a_n) : \]
the set of maximal runs from \( s \) with a prefix \( t_1 a_1 t_2 a_2 \ldots t_n a_k \)
for some \( t_1, \ldots, t_n \in R \).

\[
P_{\mathcal{A}}(\pi(s, a_1 a_2 \ldots a_n)) = \\
\int_{t \geq 0} \mu_{s_c}(t) \cdot \left( \prod_{j \neq c} \int_{\tau > t} \mu_{s_j}(\tau) d\tau \right) \cdot \gamma_{s_c t}(a_1) \cdot P_{\mathcal{A}}(\pi(s^t a_1, a_2 \ldots a_n)) \, dt
\]
where \( c = c(a_1) \), and as base case we take \( P_{\mathcal{A}}(\pi(s), \varepsilon) = 1 \).
Statistical Model Checking

\[ \Pr_M(\phi) \geq p \text{ at significance level } \alpha \]

\[ \Pr_M(\phi) \in [a-\epsilon, a+\epsilon] \text{ with confidence } \theta \]
Queries in UPPAAL SMC

\[ \Pr[ \leq 200](\nleftrightarrow \text{Train}(5)\cdot \text{Cross}) \]

VTSA Summer School 2013
Kim Larsen [9]
Queries in UPPAAL SMC

\[ \Pr[ \leq 100](\langle \rangle \text{Train}(0).\text{Cross}) \geq 0.8 \]

\[ \Pr[ \leq 100](\langle \rangle \text{Train}(0).\text{Cross}) \geq 0.5 \]
Queries in UPPAAL SMC

\[
\Pr[\leq 100](\langle\rangle \text{Train(5).Cross}) \geq \Pr[\leq 100](\langle\rangle \text{Train(1).Cross})
\]

∀\(T \leq 100\)

\[
\Pr[\leq T](\langle\rangle \text{Train(5).Cross}) \geq \Pr[\leq T](\langle\rangle \text{Train(1).Cross})
\]

VTSA Summer School, 2013.

Kim Larsen [11]
Analysis Tool: Plot Composer

VTSA Summer School, 2013.

Kim Larsen [12]
Stochastic Hybrid Systems

simulate 1 \[\leq 20\]{\text{Ball1.p, Ball2.p}}

\[\Pr[\leq 20](\langle \rangle (\text{time} \geq 12 \land \text{Ball.p} > 4))\]
Stochastic Hybrid Systems

simulate 1 \( [\leq 100]\{\text{Temp}(0).T, \text{Temp}(1).T\} \)

simulate 10 \( [\leq 100]\{\text{Temp}(0).T, \text{Temp}(1).T\} \)

\( \Pr[\leq 100]( <> \text{Temp}(1).T \leq 5 \text{ and time} > 30 ) \geq 0.2 \)

\( \Pr[\leq 100]( <> \text{Temp}(0).T \geq 10 ) \)

VTSA Summer School, 2013.

Kim Larsen [16]
Stochastic Hybrid Systems

- A Bouncing Ball

**UPPAAL SMC**

Uniform distributions (bounded delay)  
Exponential distributions (unbounded delay)  
Syntax for discrete probabilistic choice  
Distribution on next state by use of random  
Hybrid flow by use of ODEs  
+ usual stuff (structured variables, user-defined types, user-defined functions, ...)

**Networks**

Repeated races between components for outputting

VTSA Summer School, 2013.

Kim Larsen [17]
Schedulability & Performance Analysis
Task Scheduling

Task scheduling involves the process of determining the order in which tasks are executed on a CPU. This is often done according to a given priority, such as Fixed Priority, Earliest Deadline, etc.

For a task $T_i$, the following parameters are defined:
- $P(i)$: period or earliest/latest arrival
- $C(i)$: execution time
- $D(i)$: deadline

Tasks are ordered according to some priority criteria. For example, in Fixed Priority scheduling, a task with a lower priority number is given precedence over a task with a higher priority number.

In the diagram, $T_2$ is running, and the tasks $\{T_4, T_1, T_3\}$ are ready and ordered according to a given priority.
Modeling Task

Scheduler

VTSA Summer School, 2013.

Kim Larsen [20]
Modeling Scheduler

Scheduler

VTSA Summer School, 2013.
Kim Larsen [21]
Modeling Queue

// Put an element at the end of the queue
void enqueue(id_t element)
{
    int tmp=0;
    list[len++] = element;
    if (len>0)
    {
        int i=len-1;
        while (i>1 && P[list[i]]>P[list[i-1]])
        {
            tmp = list[i-1];
            list[i-1] = list[i];
            list[i] = tmp;
            i--;
        }
    }
}

// Remove the front element of the queue
void dequeue()
{
    .......
}
Schedulability Analysis

simulate 1 [<=400]
{ Task0.Ready + 2*Task0.Running + 3*Task0.Blocked,
  Task1.Ready + 2*Task1.Running + 3*Task1.Blocked + 4,
  Task2.Ready + 2*Task2.Running + 3*Task2.Blocked + 8,

A[] not (Task0.Error or Task1.Error or Task2.Error or Task3.Error)
Schedulability Analysis

simulate 10000 [<=400]
{ Task0.Ready + 2*Task0.Running + 3*Task0.Blocked - Task0.Error, 
  Task1.Ready + 2*Task1.Running + 3*Task1.Blocked + 4, 
  Task2.Ready + 2*Task2.Running + 3*Task2.Blocked + 8 - Task2.Error, 
}:
1 : (Task0.Error or Task1.Error or Task2.Error or Task3.Error)

VTSA Summer School, 2013.
Kim Larsen [24]
Performance Analysis

sup : Task2.r, Task3.r
Performance Analysis

D = 400

D = 100

D = 200

D = 100

VTSA Summer School, 2013.
Kim Larsen [26]
Herschel–Planck Scientific Mission at ESA

Attitude and Orbit Control Software
TERMA A/S Steen Ulrik Palm, Jan Storbank Pedersen, Poul Hougaard
Modeling in UPPAAL

UPPAAL 4.1 Framework
ISoLA 2010

VTSA Summer School, 2013.
Kim Larsen [28]
Symbolic MC vs. Statistical MC

Symbolic analysis:
- Preemptive scheduler requires *stop-watches*.
- Exact reachability of stop-watch automata is *undecidable*.
- UPJAAAL provides *over-approximation* for stop-watches.
- $\Rightarrow$ symbolic analysis may give spurious errors, but still suitable for *proving safety/schedulability*.

Statistical analysis:
- can show *presence of errors* but not absence.
- $\Rightarrow$ suitable for *disproving schedulability*.

<table>
<thead>
<tr>
<th>$f = \text{BCET/WCET}$</th>
<th>0-71%</th>
<th>72-86%</th>
<th>87-89%</th>
<th>90-100%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Symbolic MC:</td>
<td>maybe</td>
<td>maybe</td>
<td>n/a</td>
<td>Safe</td>
</tr>
<tr>
<td>Statistical MC:</td>
<td>Unsafe</td>
<td>maybe</td>
<td>maybe</td>
<td>maybe</td>
</tr>
</tbody>
</table>
SMC Simulation to Find Error

Herschel deadline violation with $f = 50\%$:

simulate 10000 [$\leq 300$] {
}

: 1 : error
Other Case Studies

FIREWIRE

BLUETOOTH

10 node LMAC

Schedulability Analysis for Mix Cr Sys

Smart Grid Demand / Response

Energy Aware Buildings

Genetic Oscillator (HBS)

Passenger Seating in Aircraft

Battery Scheduling (SENSATION) Erik Wogensen
Formal & Informal Methods

- Model Checking vs Stat MC, Simulation
- Qualitative vs Quantitative (metrics)
- State Space Expl vs Confidence Expl
- Correctness (overap) vs Counterex (underap)
- Worst Case vs Expected Case
- Synthesis on abstract models vs Performance eval on refined models

VTSA Summer School, 2013. Kim Larsen [32]
SMC Queries – Examples

- $\Pr[\leq 100](<> \text{ goal})$
- $\Pr[#\leq 10][]\text{ safe}$
- $\Pr[x\leq 200](<> \text{ goal}) \geq 0.3$
- $\text{E}[\leq 100; 1000](\text{min: expr})$
- $\text{simulate} 10 [\leq 100] \{ e_1, e_2, x_1 \}$
- $\text{simulate} 100 [\leq 10] \{ e \} : 2 : \text{goal}$

Exercise 28 (Jobshop scheduling part 2)