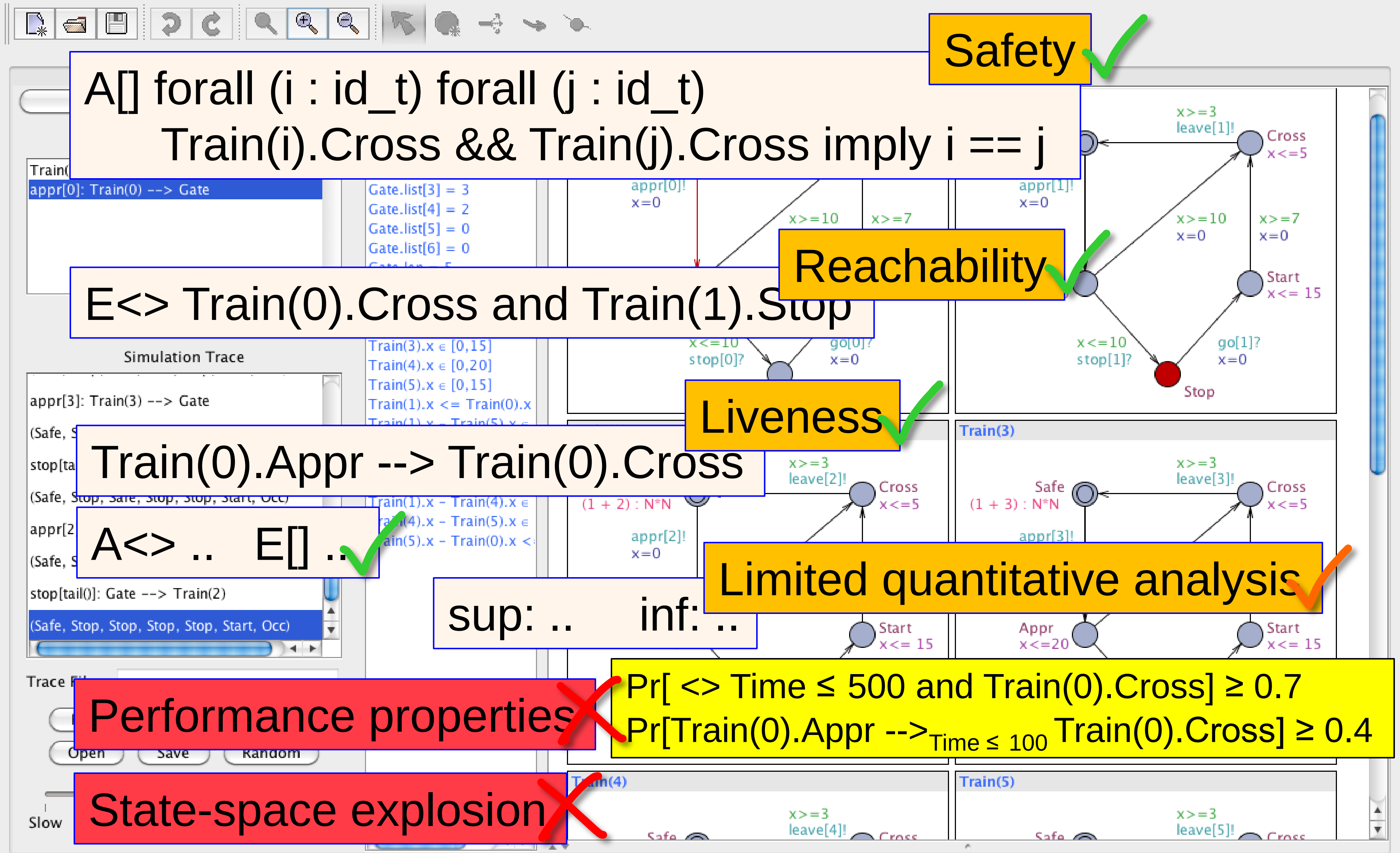
Statistical Model Checking in UPPAAL

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

UPPAAL



Safety ✓

A[] forall (i : id_t) forall (j : id_t)
Train(i).Cross && Train(j).Cross imply i == j

Reachability ✓

E<> Train(0).Cross and Train(1).Stop

Liveness ✓

Train(0).Appr --> Train(0).Cross

A<> .. E[] .. ✓

sup: .. inf: ..

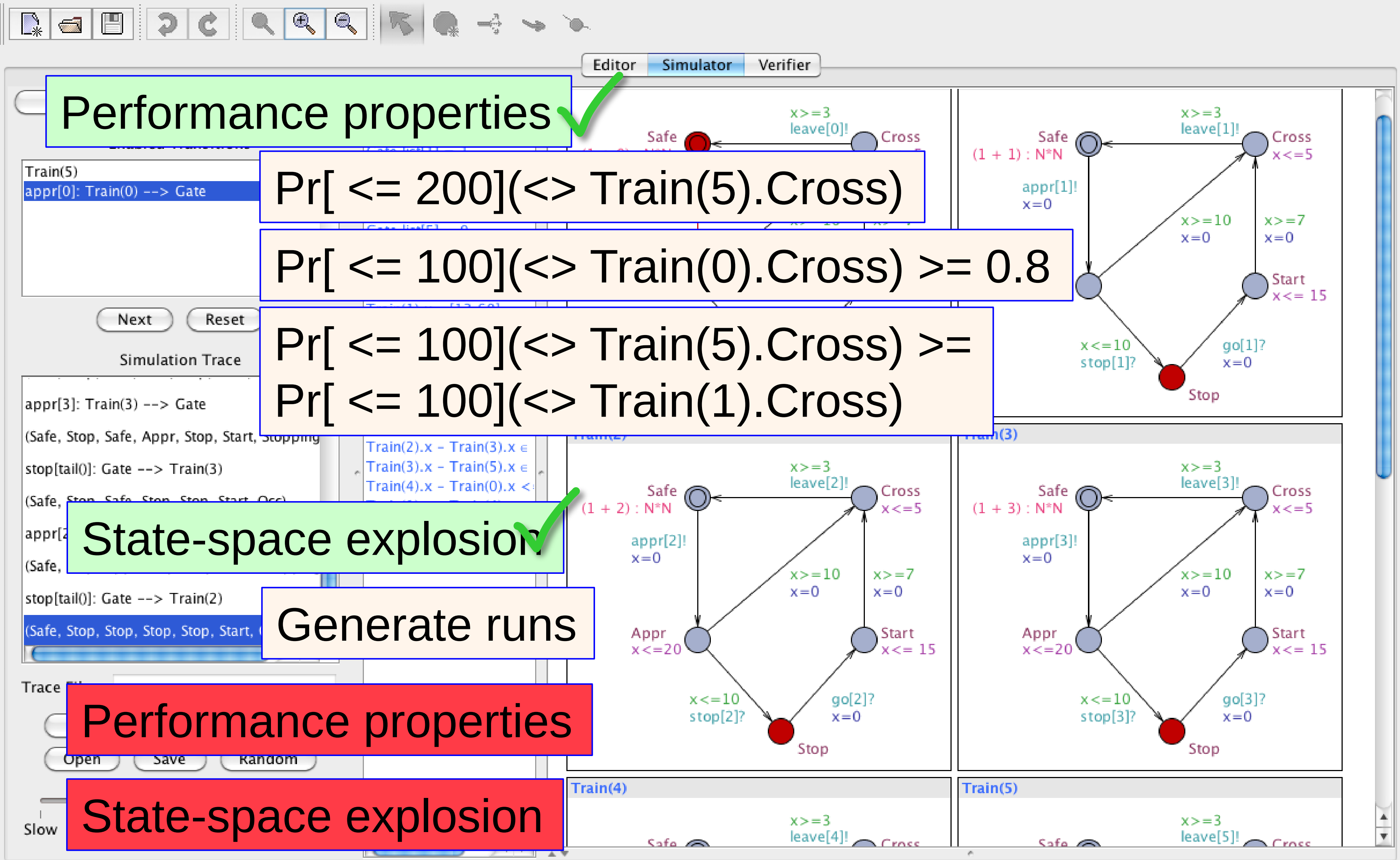
Limited quantitative analysis ✓

Performance properties ✗

Pr[<> Time ≤ 500 and Train(0).Cross] ≥ 0.7
Pr[Train(0).Appr $-->_{Time \leq 100}$ Train(0).Cross] ≥ 0.4

State-space explosion ✗

UPPAAL SMC



Performance properties ✓

Pr[<= 200](<> Train(5).Cross)

Pr[<= 100](<> Train(0).Cross) >= 0.8

Pr[<= 100](<> Train(5).Cross) >= Pr[<= 100](<> Train(1).Cross)

State-space explosion ✓

Generate runs

Performance properties

State-space explosion

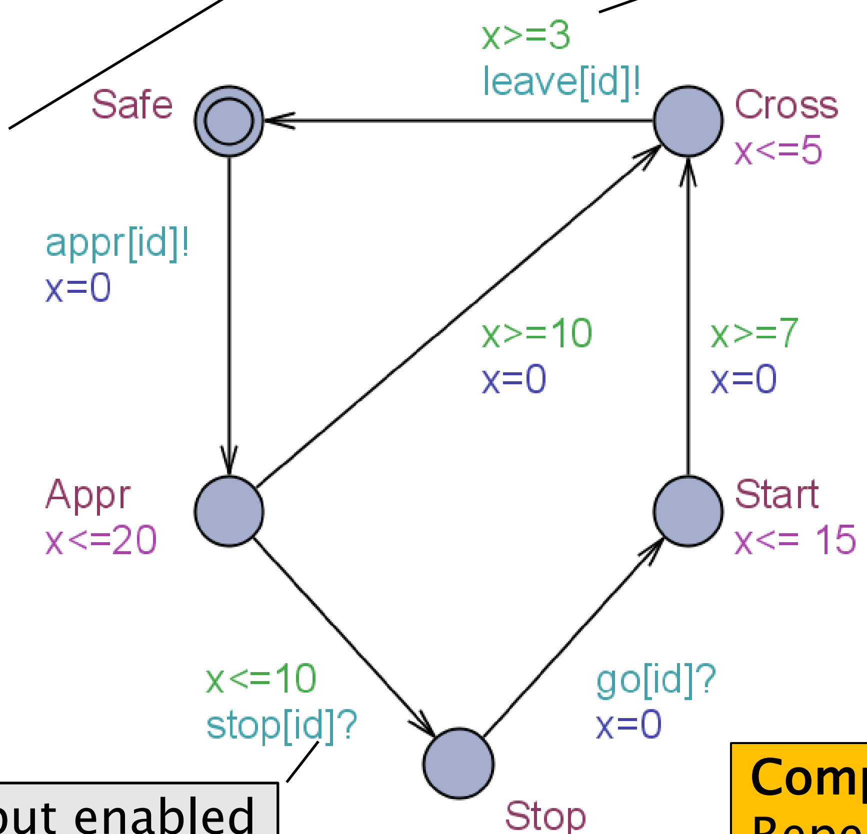
Overview

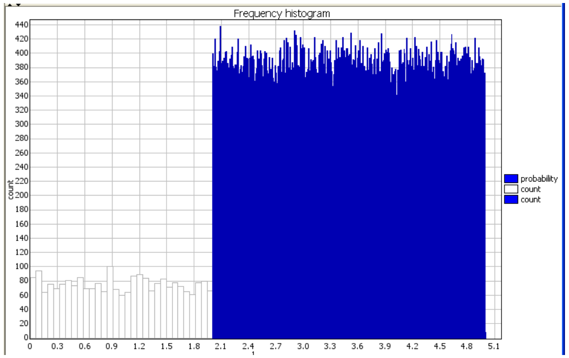
- Stochastic Semantics of Networks of Timed Automata
- Statistical Model Checking in UPPAAL
  - Estimation
  - Sequential Hypothesis Testing
  - Sequential Probability Comparison
  - Parameterized Probability Comparison
- SMC of Hybrid Automata
- Case Studies & Demo

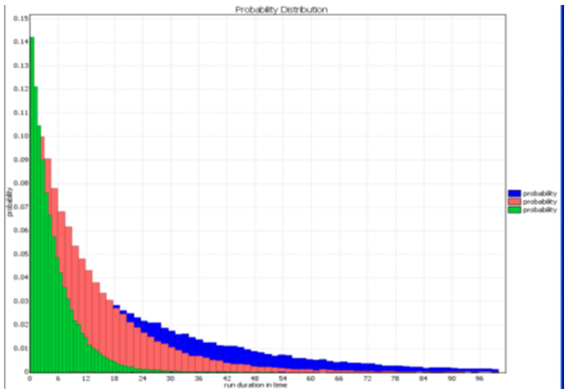
Stochastic Semantics of TA

Exponential Distribution

Uniform Distribution



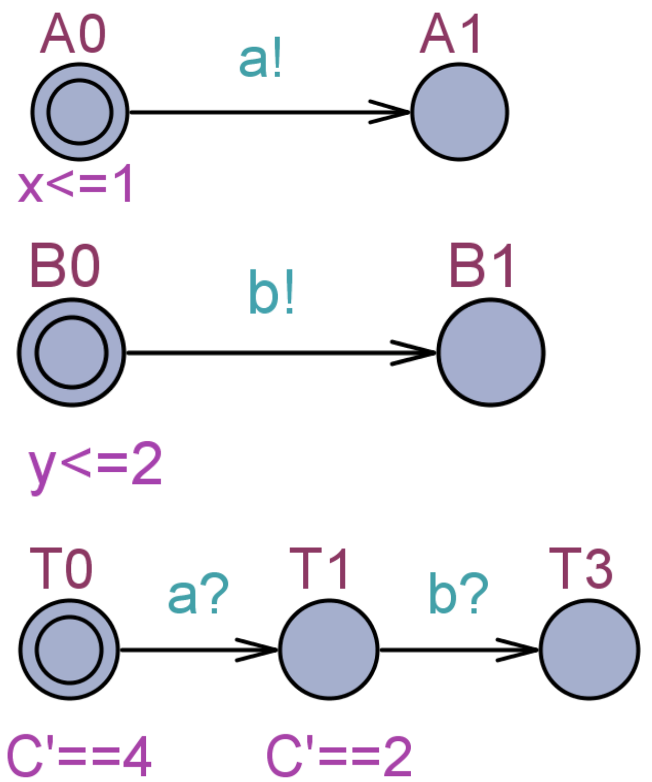


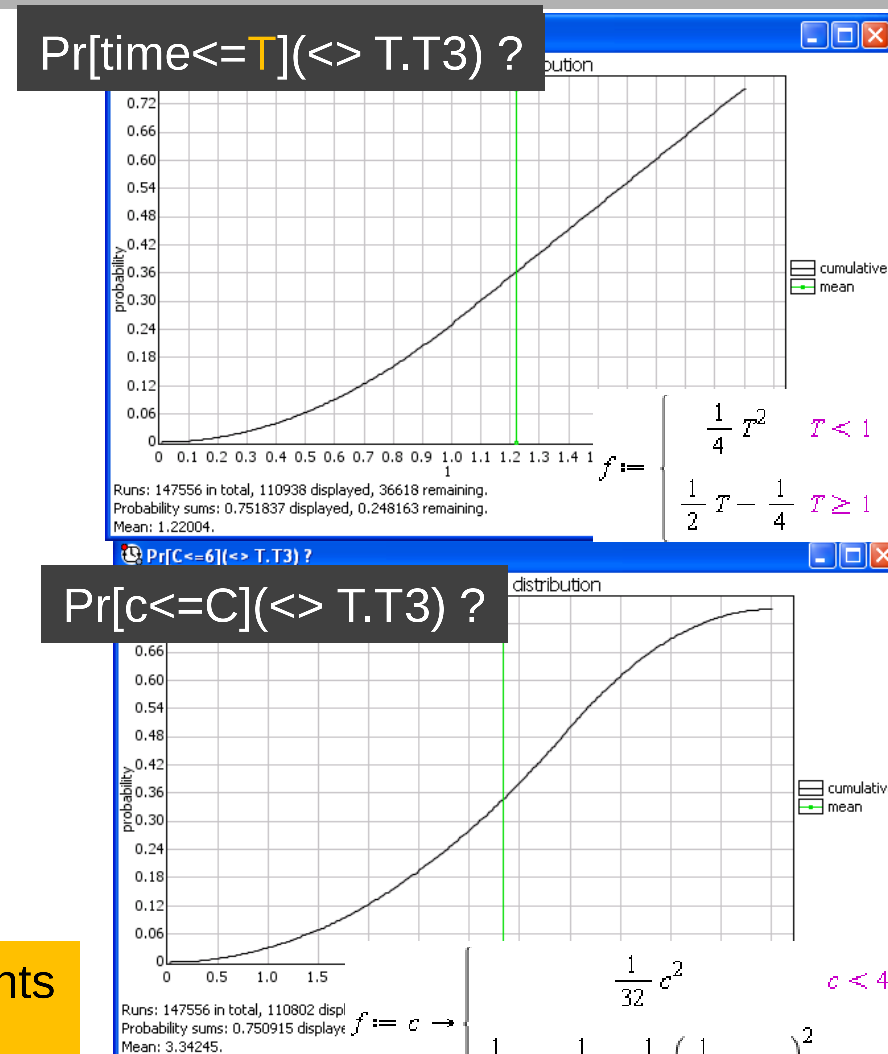


Input enabled

Composition =
Repeated races between components

Stochastic Semantics of Timed Automata





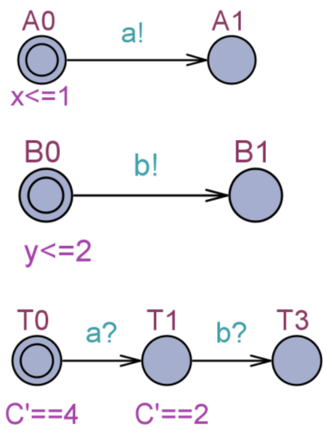
$$f := \begin{cases} \frac{1}{4}T^2 & T < 1 \\ \frac{1}{2}T - \frac{1}{4} & T \geq 1 \end{cases}$$

$$f := c \to \begin{cases} \frac{1}{32}c^2 & c < 4 \\ \frac{1}{4}c - \frac{1}{2} - \frac{1}{4}\left(\frac{1}{2}c - 2\right)^2 & c \geq 4 \end{cases}$$

Composition = Race between components for outputting

Stochastic Semantics of Timed Automata

$\mathcal{A}$



Assumptions:

Component TAs are:

- Input enabled
- Deterministic
- Disjoint set of output actions

π (**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**.

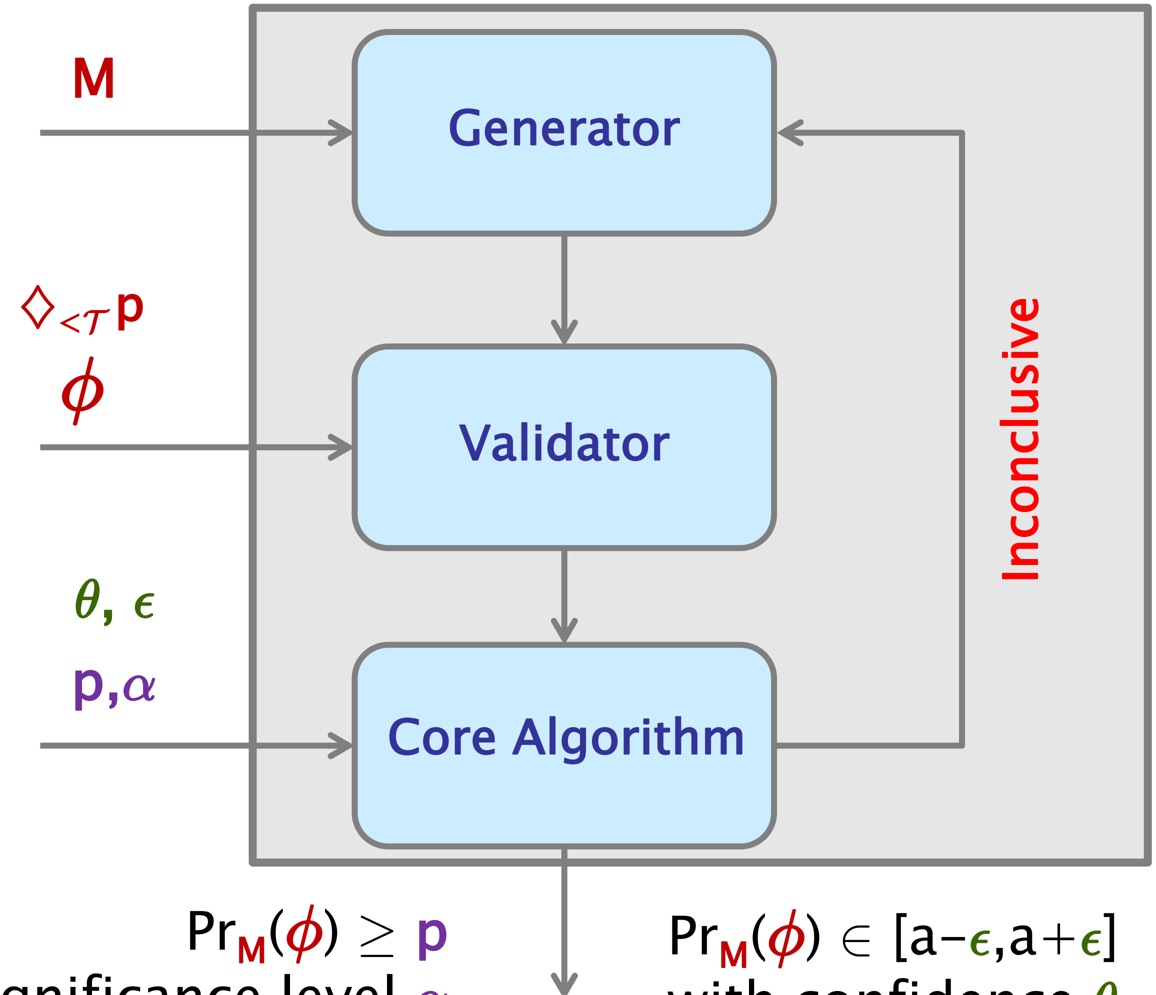
$$\mathbb{P}_{\mathcal{A}}\big(\pi(\mathbf{s},\, a_1 a_2 \ldots a_n)\big) =$$

$$\int_{t\geq 0} \mu_{s_c}(t) \cdot \Big(\prod_{j\neq c} \int_{\tau > t} \mu_{s_j}(\tau)\, d\tau\Big) \cdot \gamma_{s_c{}^t}(a_1) \cdot \mathbb{P}_{\mathcal{A}}\big(\pi(\mathbf{s}^t)^{a_1},\, a_2 \ldots a_n)\big)\, dt$$

where $c = c(a_1)$, and as base case we take $P_{\mathcal{A}}(\pi(\mathbf{s}), \varepsilon) = 1$.

Statistical Model Checking

[FORMATS11, LPAR12, RV12]

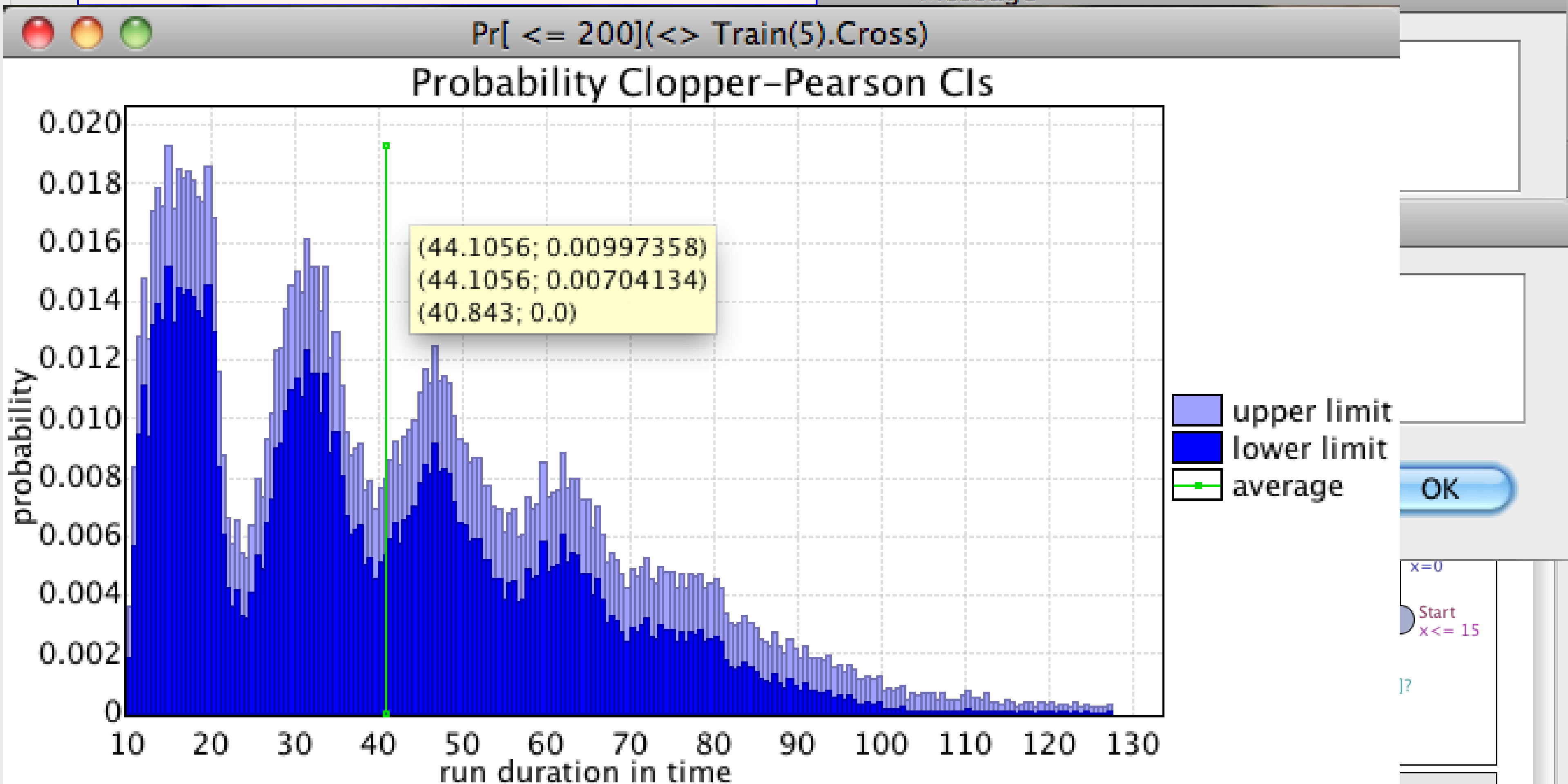


$Pr_M(\phi) \geq p$ at significance level α

$Pr_M(\phi) \in [a-\epsilon, a+\epsilon]$ with confidence θ

Queries in UPPAAL SMC

Pr[<= 200](<> Train(5).Cross)

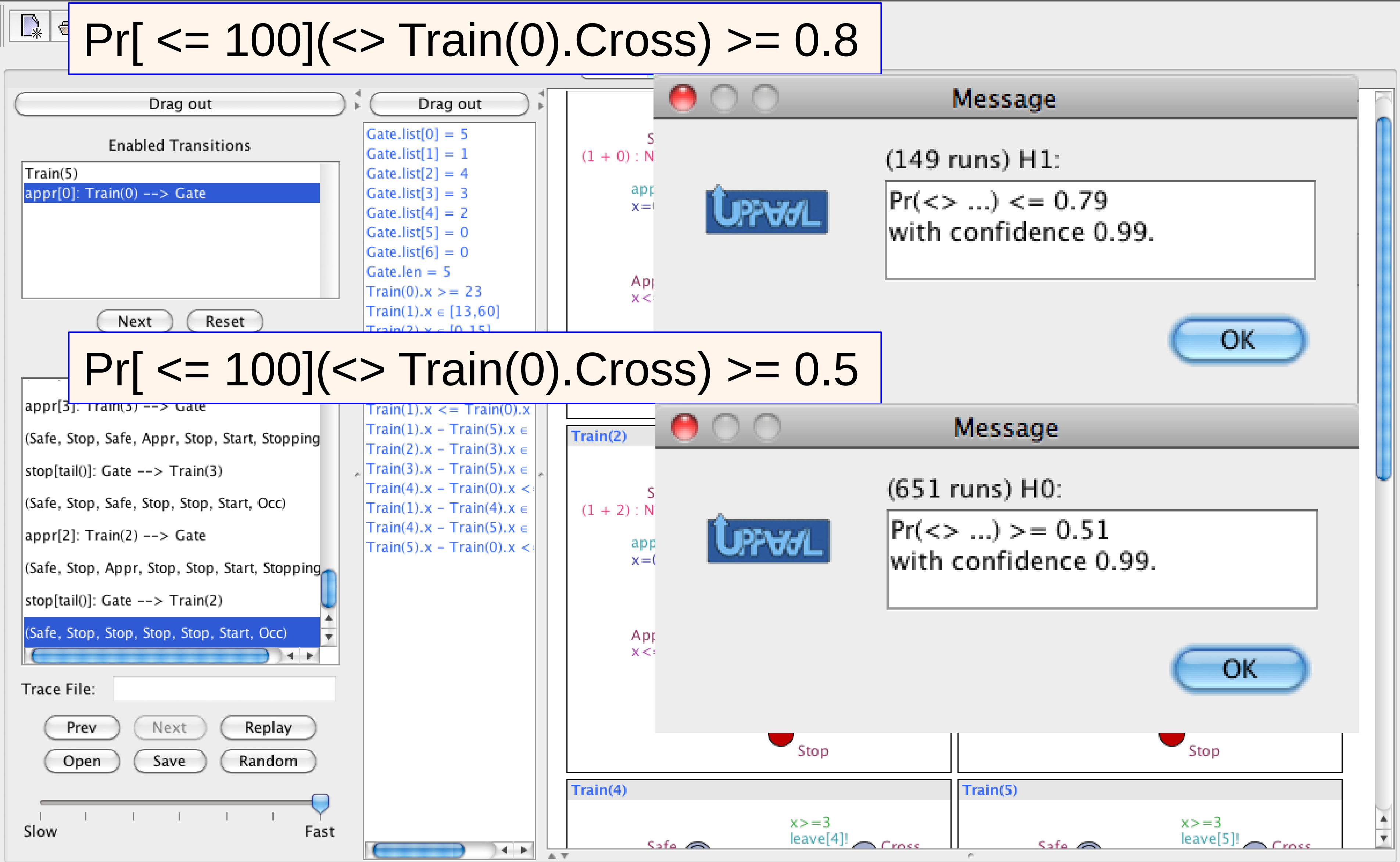


Parameters: α=0.01, ε=0.01, bucket width=0.587972, bucket count=200.
Runs: 26492 in total, 26492 displayed, 0 remaining.
Probability sums: 1 displayed, 0 remaining.
Average: 40.843.

Queries in UPPAAL SMC

Pr[<= 100](<> Train(0).Cross) >= 0.8

Pr[<= 100](<> Train(0).Cross) >= 0.5



Queries in UPPAAL SMC

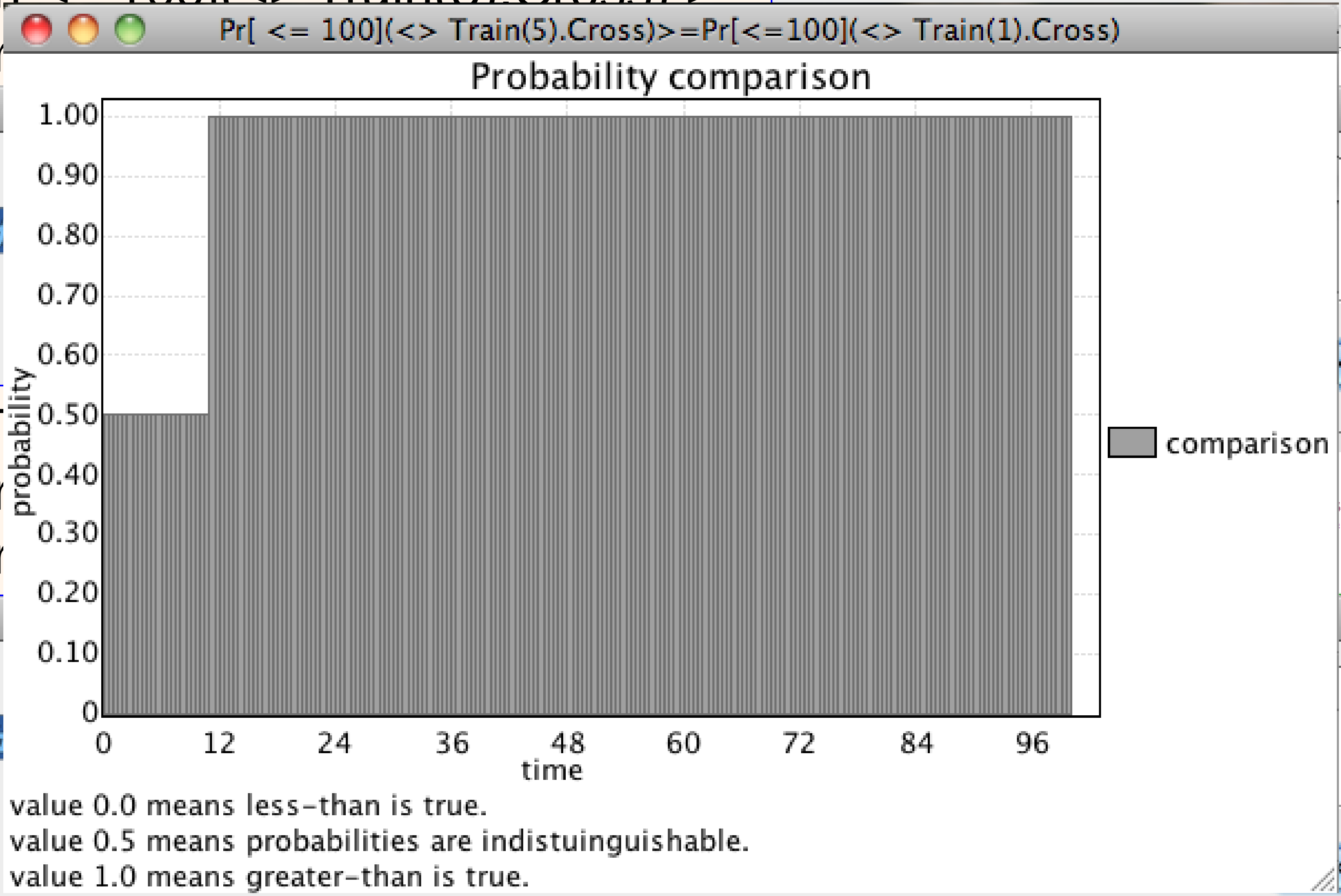
Pr[<= 100](<> Train(5).Cross) >=

Pr

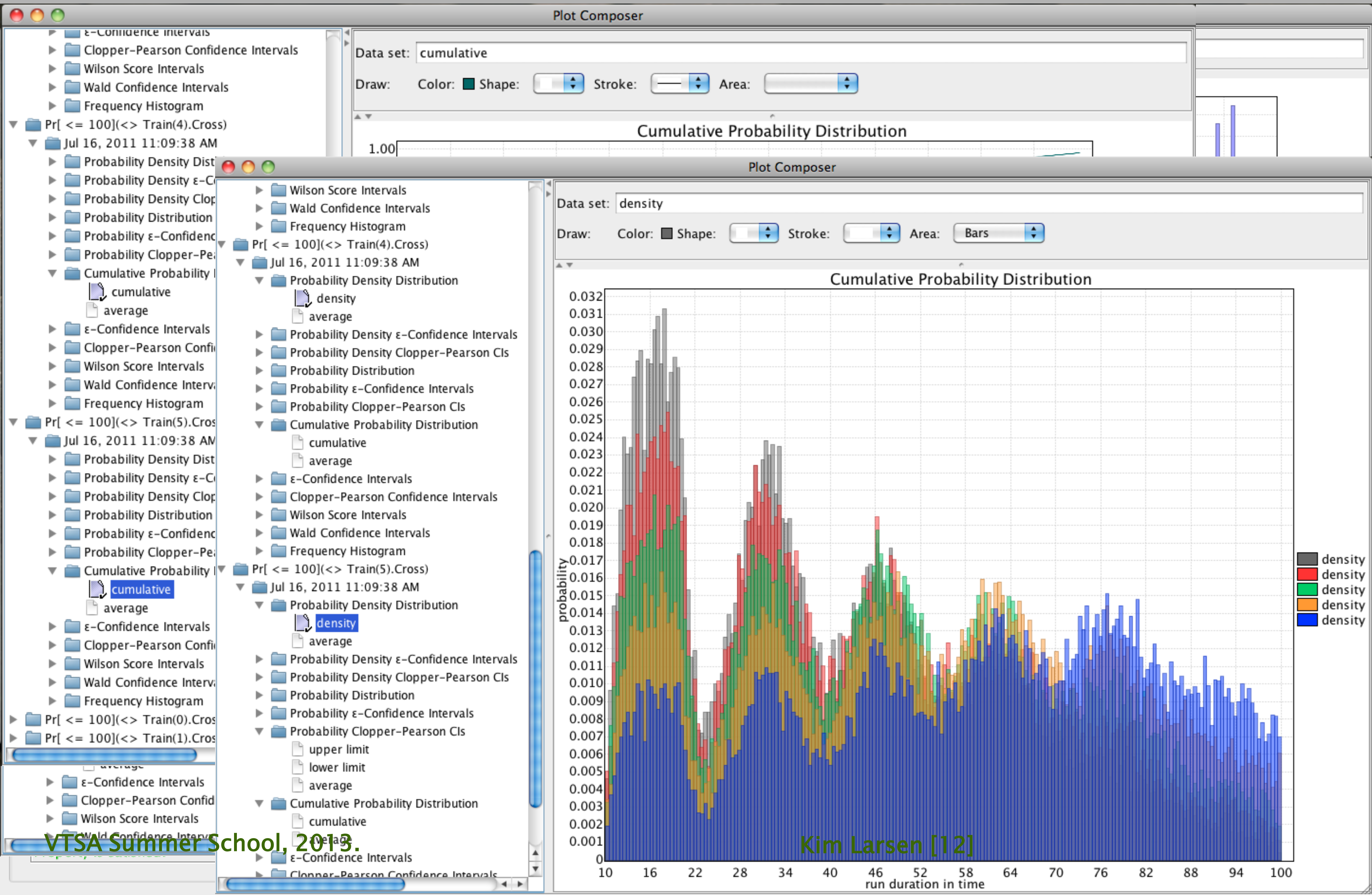
∀

Pr

Pr

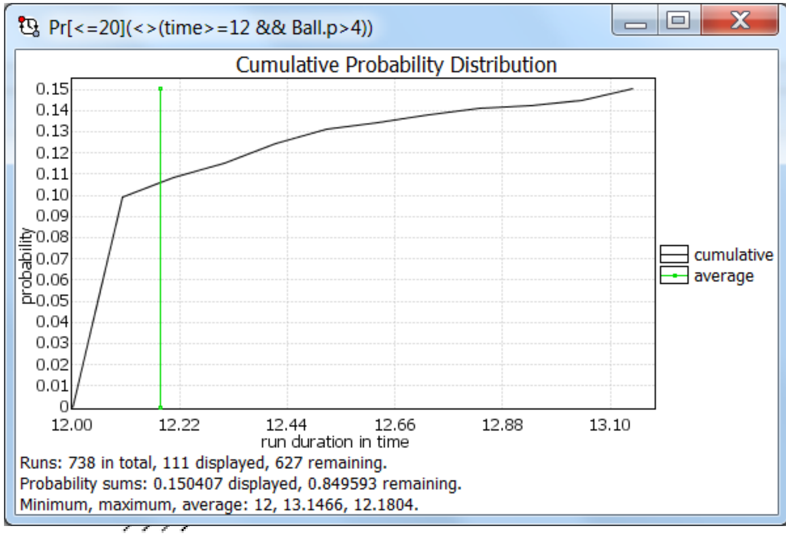


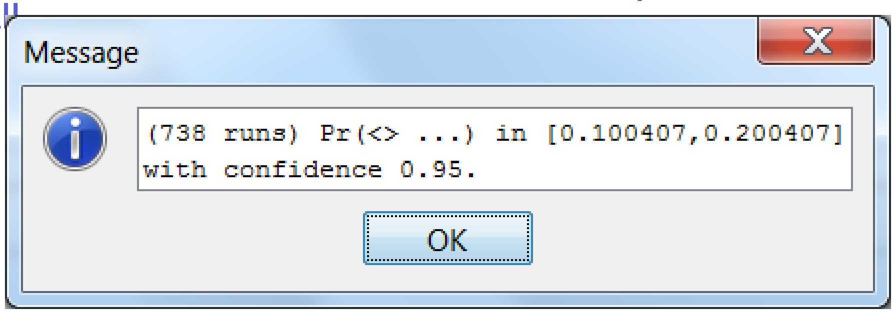
Analysis Tool: Plot Composer

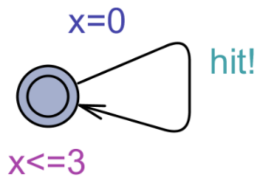


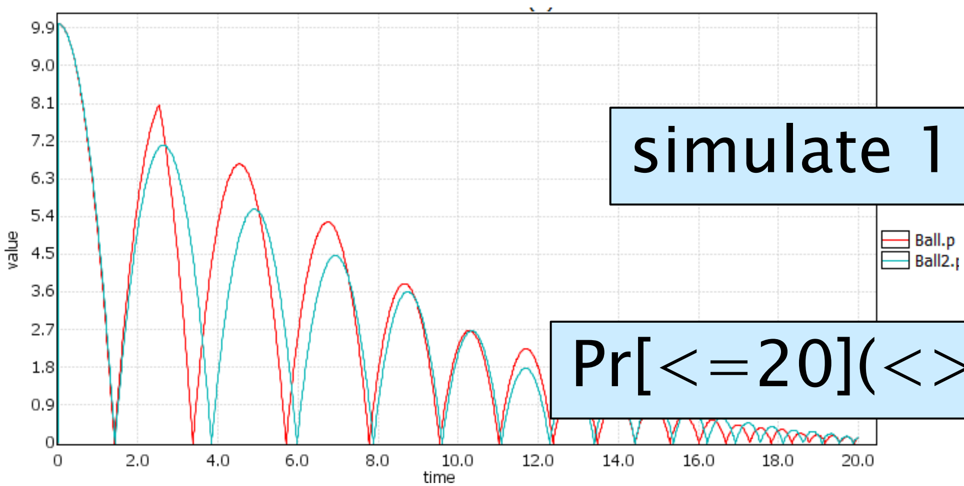
Demo

Stochastic Hybrid Systems









simulate 1 [<=20]{Ball1.p, Ball2.p}

Pr[<=20](<>(time>=12 && Ball.p>4))

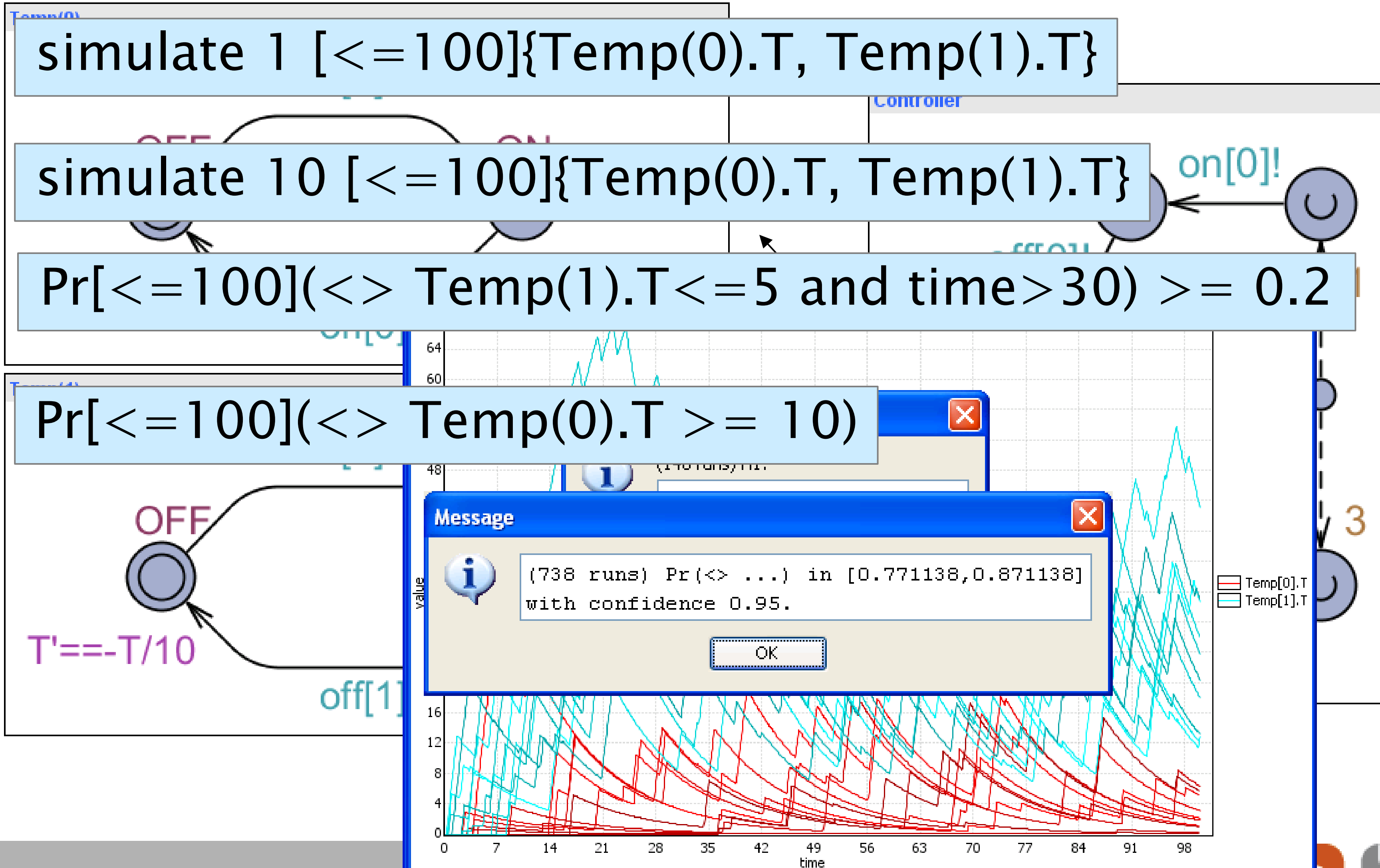
Stochastic Hybrid Systems

simulate 1 [<=100]{Temp(0).T, Temp(1).T}

simulate 10 [<=100]{Temp(0).T, Temp(1).T}

Pr[<=100](<> Temp(1).T<=5 and time>30) >= 0.2

Pr[<=100](<> Temp(0).T >= 10)



Stochastic Hybrid Systems

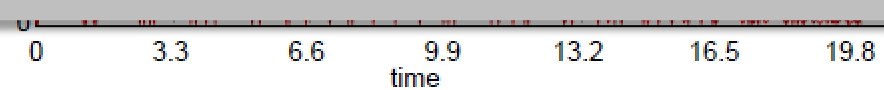
- A Bouncing Ball

UPPAAL SMC

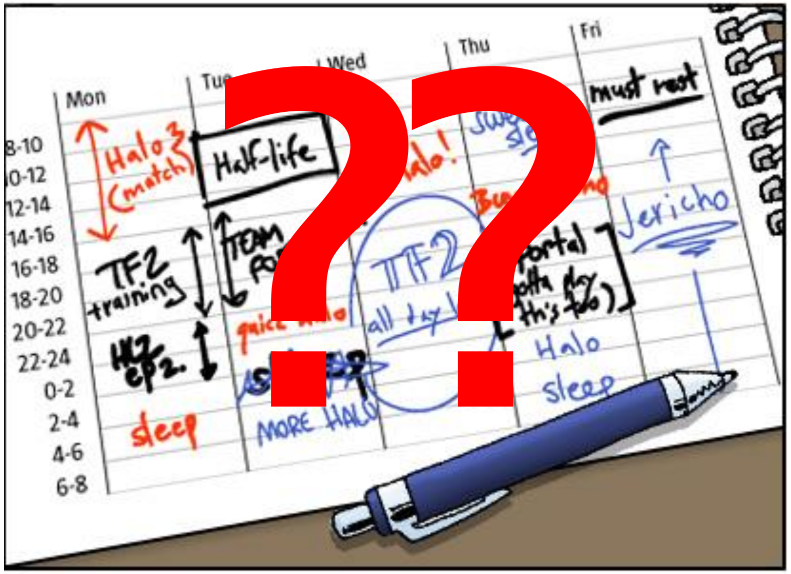
Uniform distributions (bounded delay)
Exponential distributions (unbounded delay)
Syntax for discrete probabilistic choice
Distribution on next state by use of `random`
Hybrid flow by use of ODEs
+ usual stuff (structured variables, user-defined types
  user-defined functions, ….)

Networks

Repeated races between components for outputting

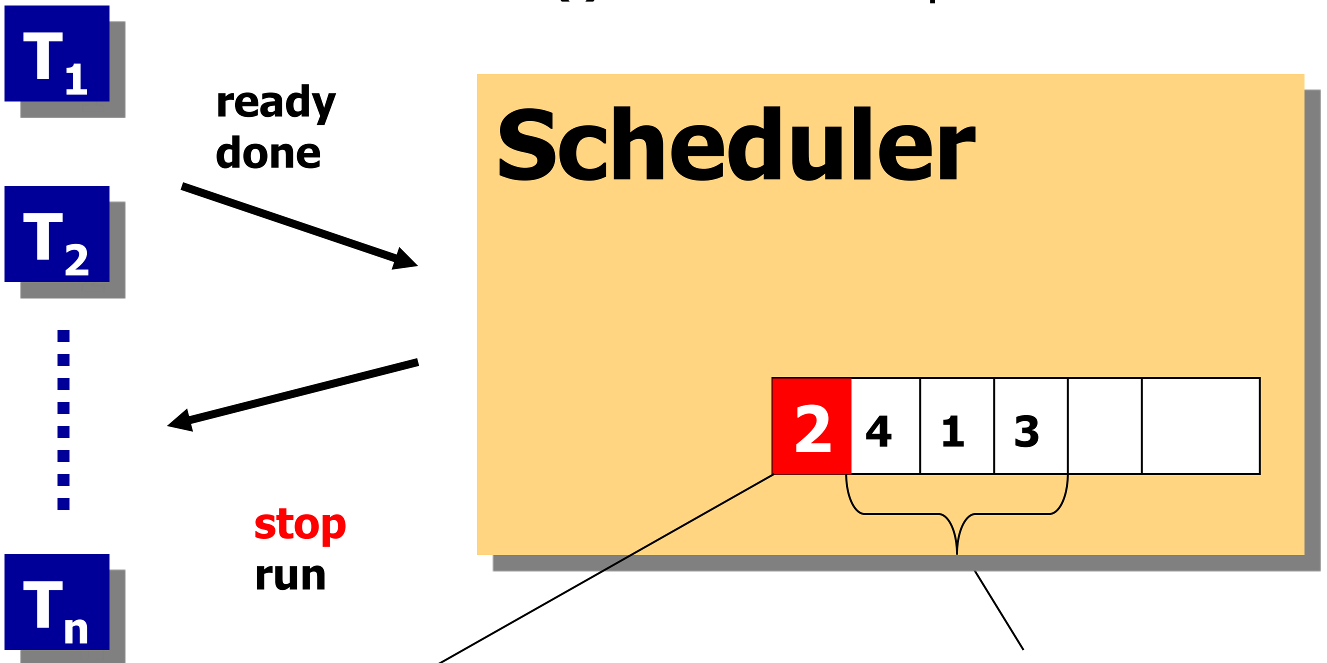


Schedulability
& Performance Analysis



Task Scheduling *utilization of CPU*

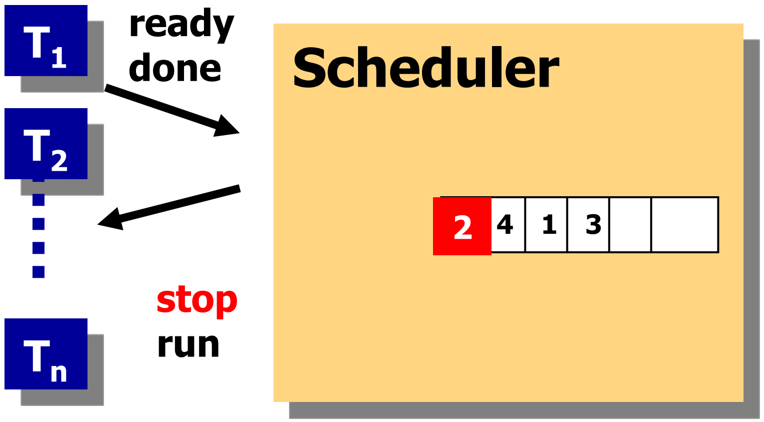
P(i), UNI[E(i), L(i)], .. : period or earliest/latest arrival or .. for T_i
C(i), UNI[BC(i),WC(i)] : execution time for T_i
D(i): deadline for T_i

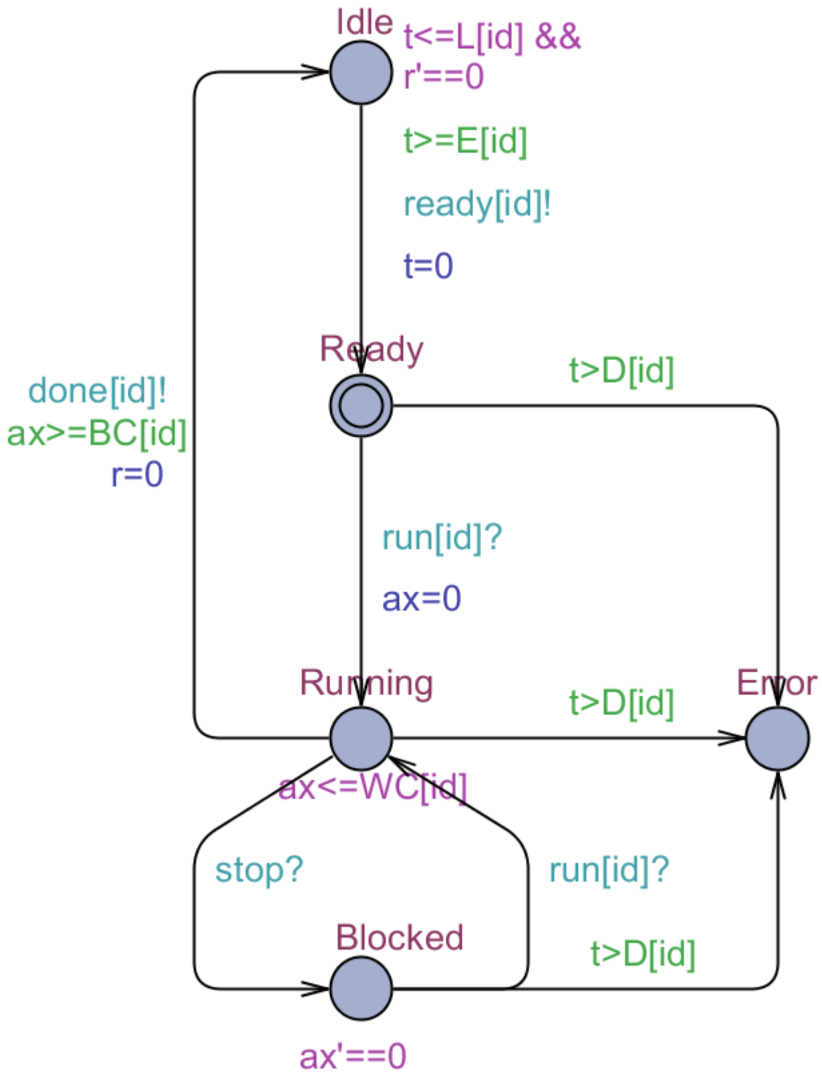


T_2 is running

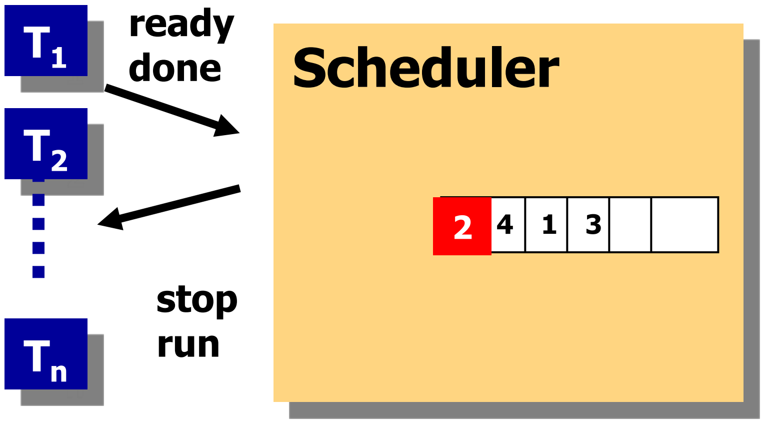
{ T_4 , T_1 , T_3 } ready ordered according to some given priority: (e.g. Fixed Priority, Earliest Deadline,..)

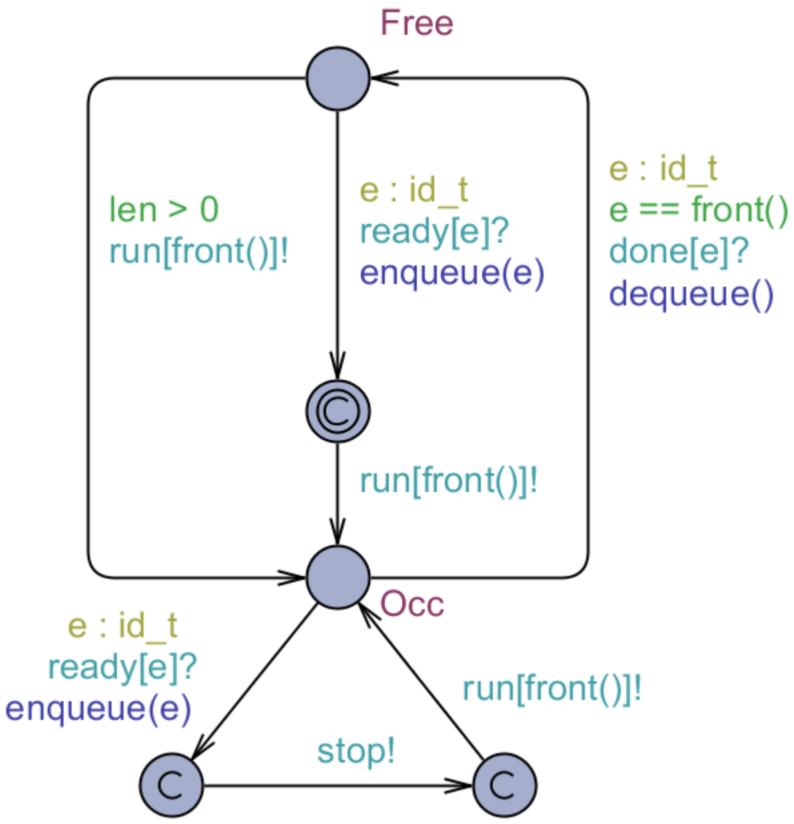
Modeling Task



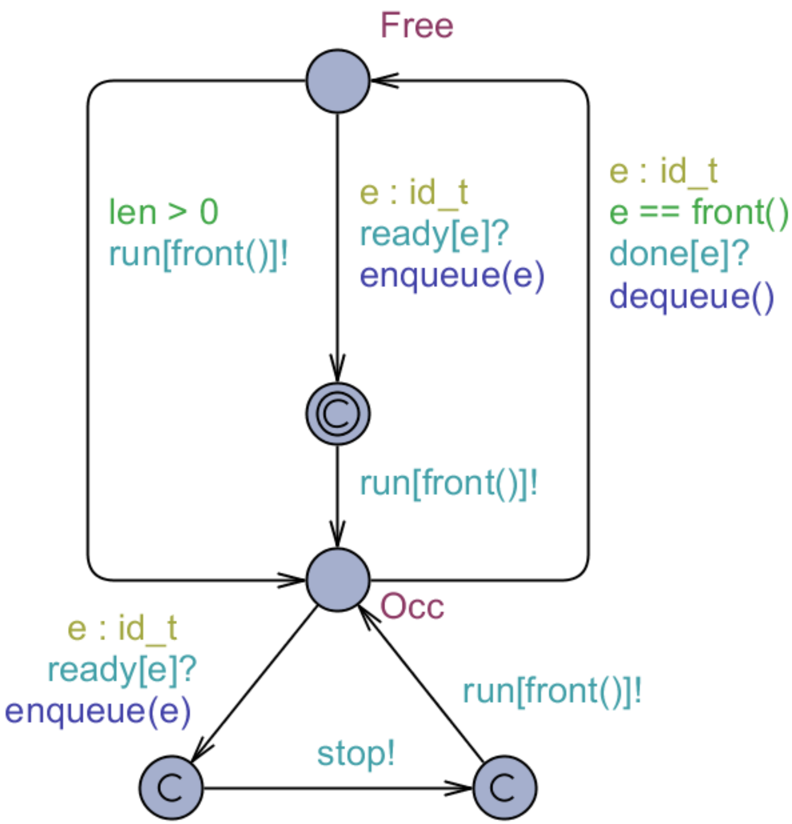


Modeling Scheduler





Modeling Queue



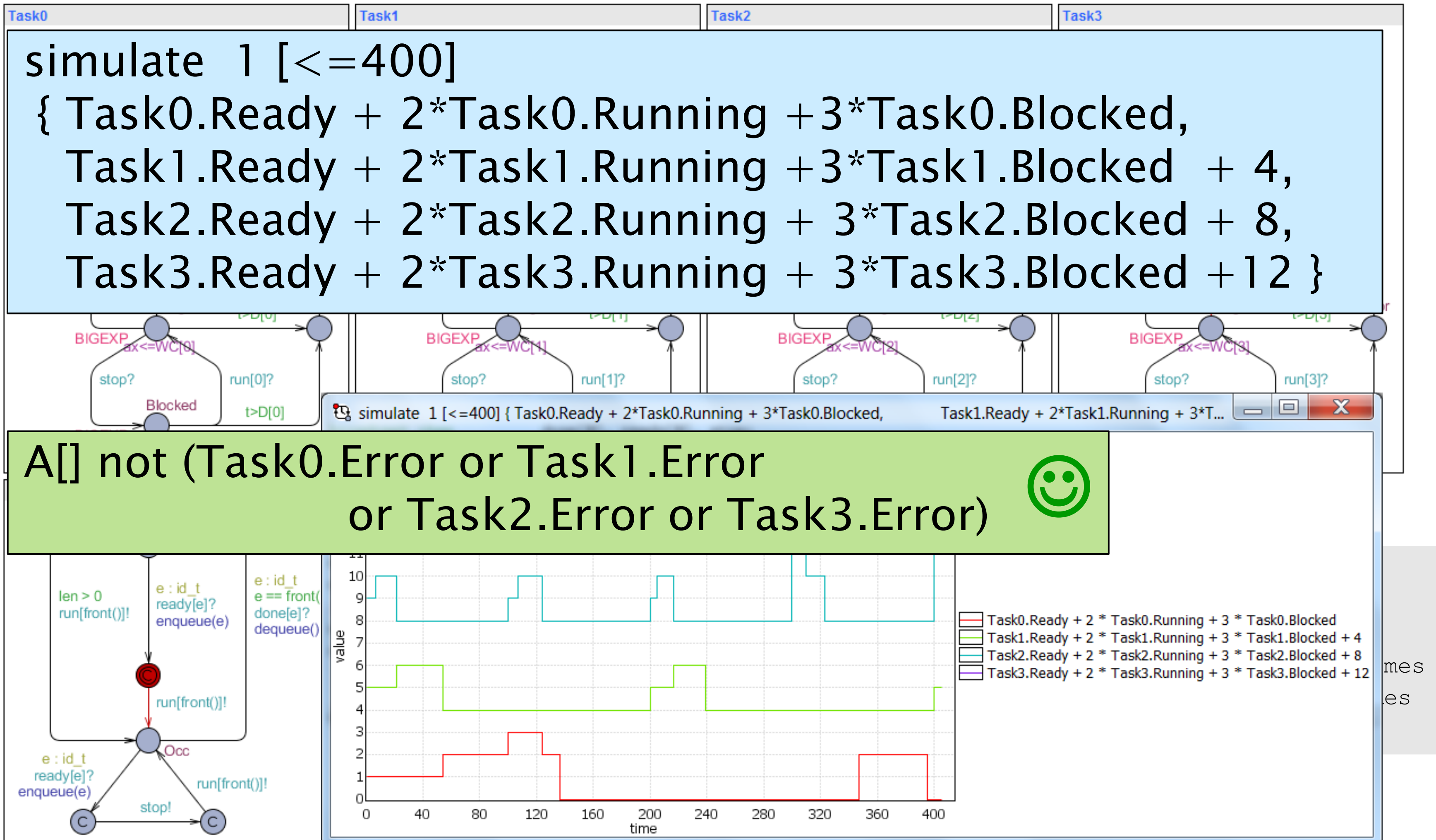

```
// Put an element at the end of the queue
void enqueue(id_t element)
{
int tmp=0;
list[len++] = element;
if (len>0)
{
        int i=len-1;
        while (i>1 && P[list[i]]>P[list[i-1]])
        {
                tmp = list[i-1];
                list[i-1] = list[i];
                list[i] = tmp;
                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,
  Task3.Ready + 2*Task3.Running + 3*Task3.Blocked +12 }
```

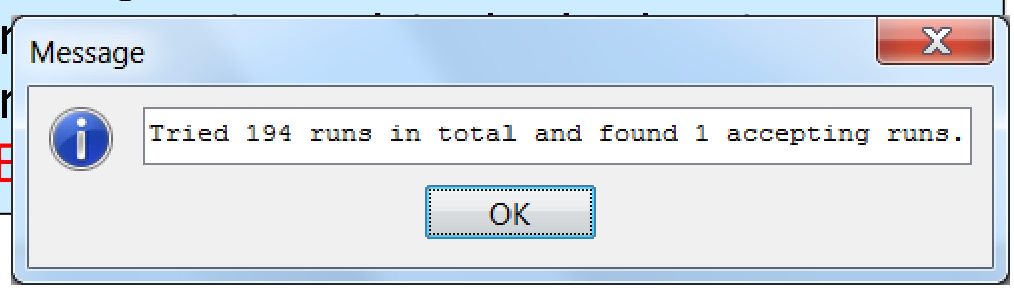
```
A[] not (Task0.Error or Task1.Error
          or Task2.Error or Task3.Error)
```

☺

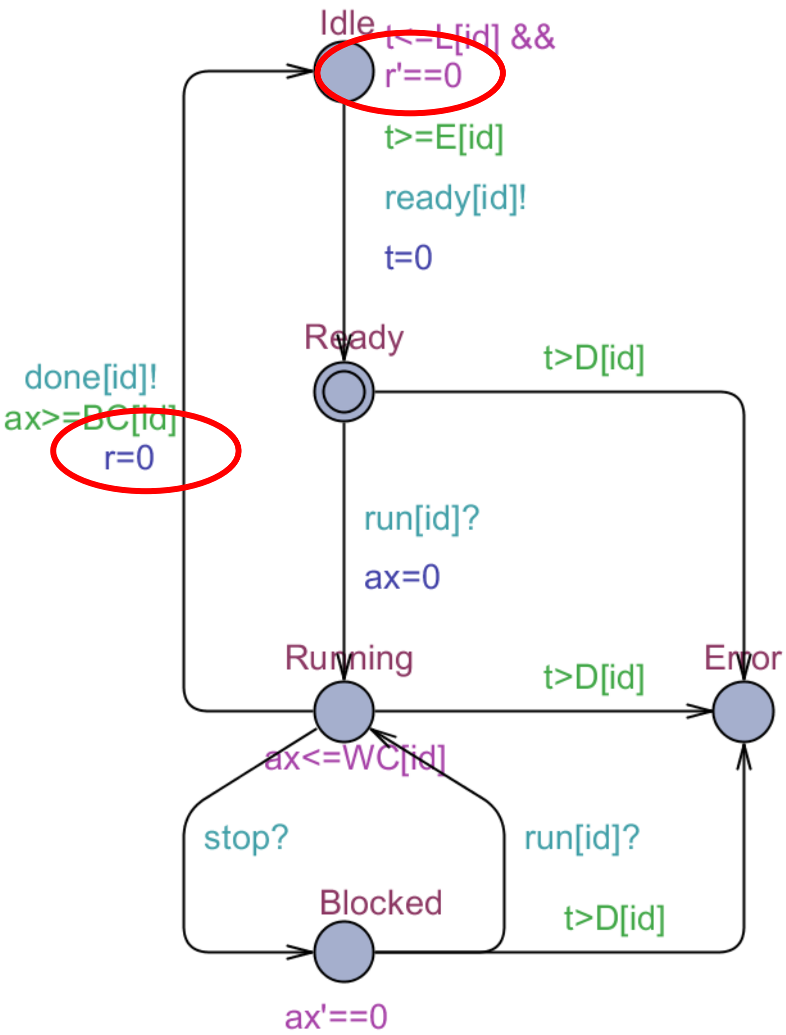


Schedulability Analysis

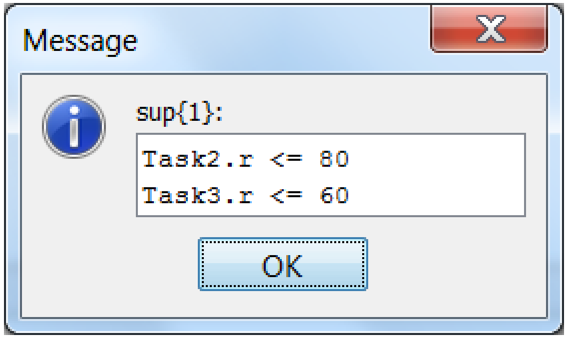
simulate 10000 [<=400]
{ Task0.Ready + 2*Task0.Running +3*Task0.Blocked,
Task1.Ready + 2*Task1.Running +3*Task1.Blocked + 4,
Task2.Ready + 2*Task2.Run
Task3.Ready + 2*Task3.Run
: 1 : (Task0.Error or Task1.E



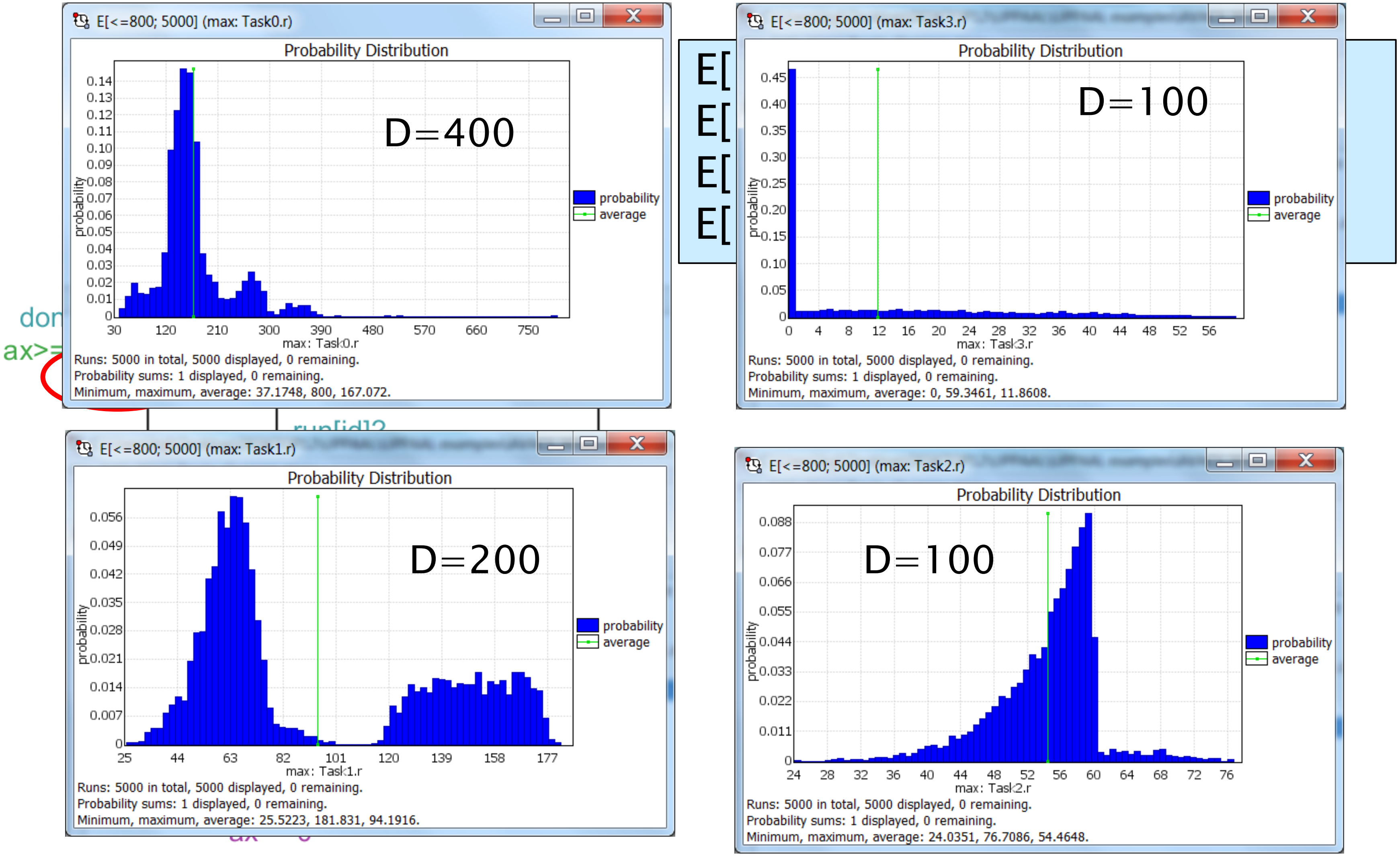
Performance Analysis



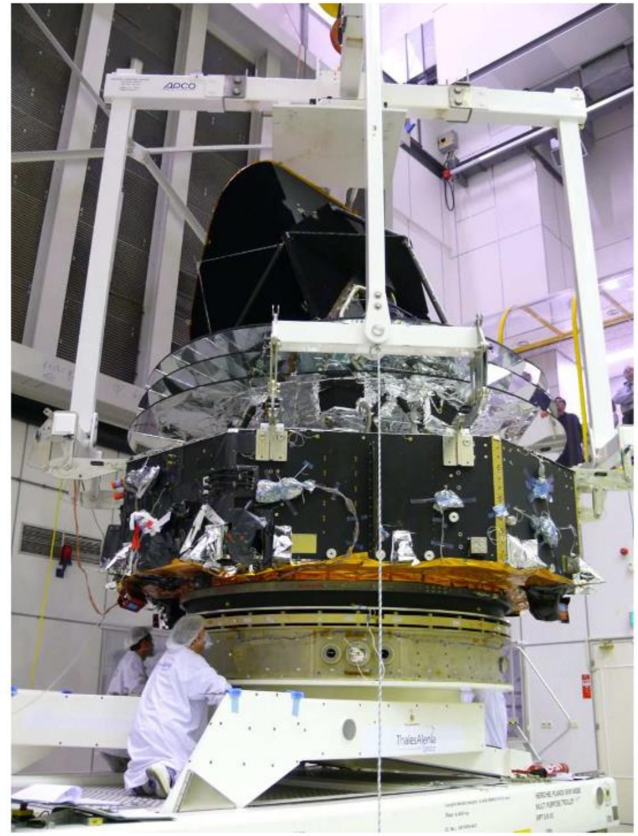
sup : Task2.r, Task3.r



Performance Analysis



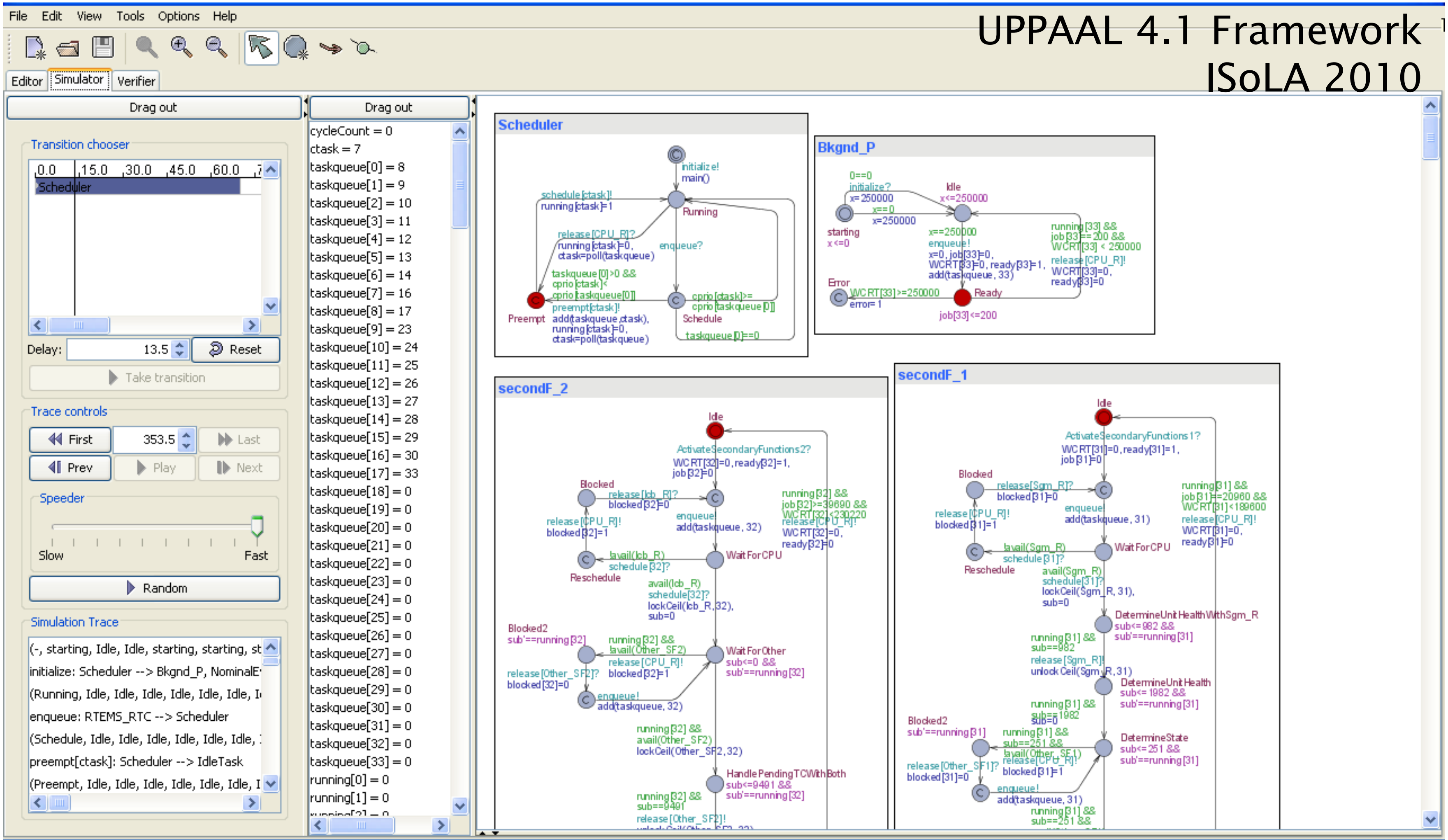
Herschel-Planck Scientific Mission at ESA



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

Modeling in UPPAAL

TERMA®



Symbolic MC vs. Statistical MC

Symbolic analysis:

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

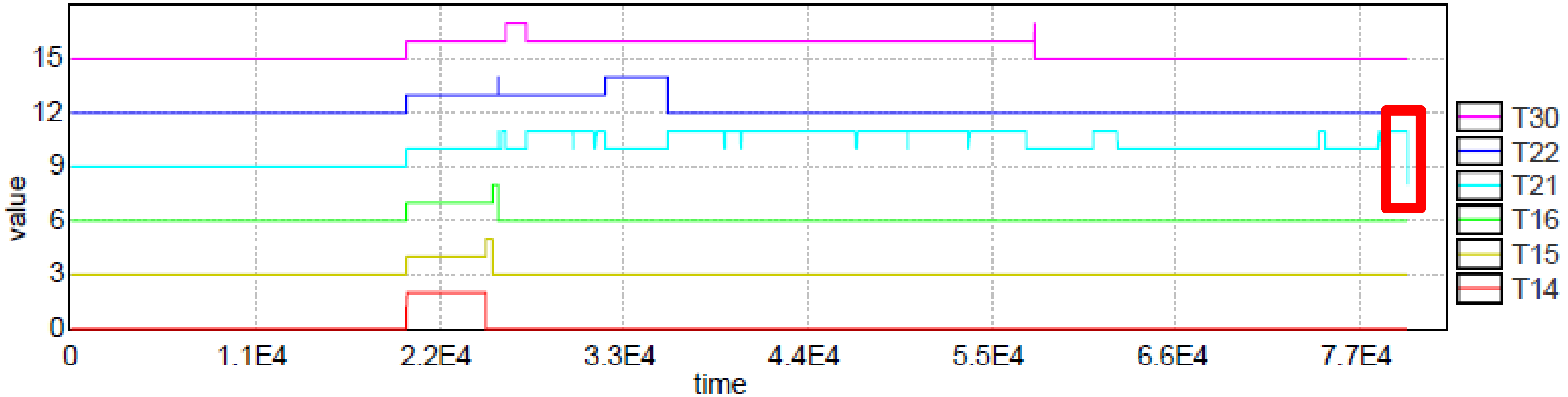
Statistical analysis:

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

$f = \mathrm{BCET}/\mathrm{WCET}$:	0-71%	72-86%	87-89%	90-100%
Symbolic MC:	maybe	maybe	n/a	**Safe**
Statistical MC:	**Unsafe**	maybe	maybe	maybe

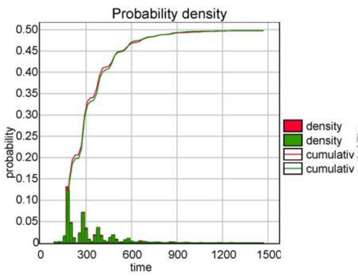
SMC Simulation to Find Error

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

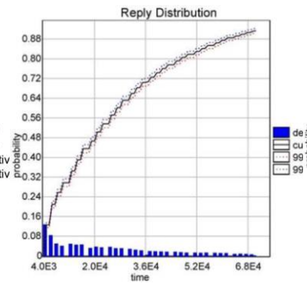


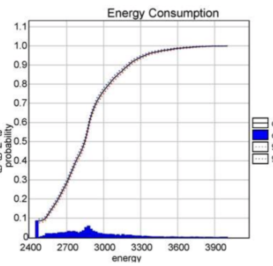

```
simulate 10000 [<=300] {
   (T(1).Ready+T(1).Computing+T(1).Release+runs[1]-2*T(1)
   (T(2).Ready+T(2).Computing+T(2).Release+runs[2]-2*T(1)
   (T(3).Ready+T(3).Computing+T(3).Release+runs[3]-2*T(1)
} : 1 : error
```

Other Case Studies

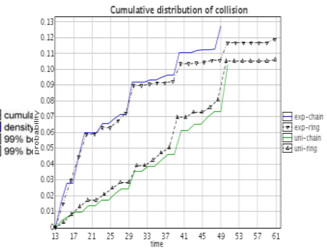


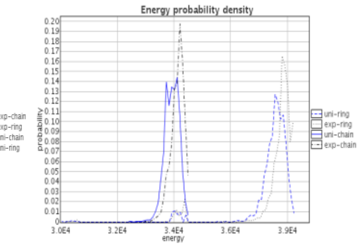
FIREWIRE



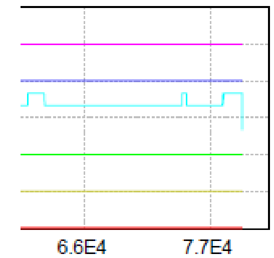


BLUETOOTH

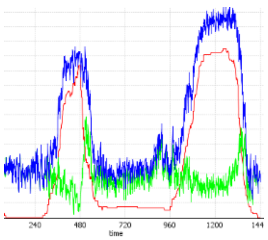




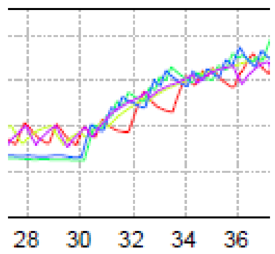
10 node LMAC



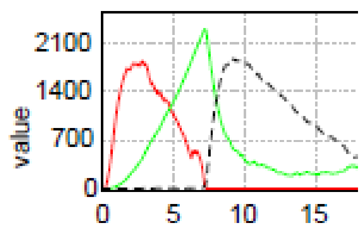
Schedulability Analysis for Mix Cr Sys



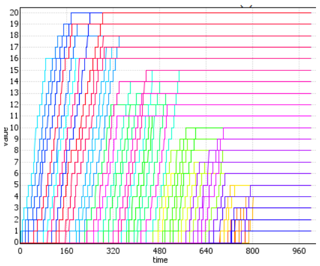
Smart Grid Demand / Response



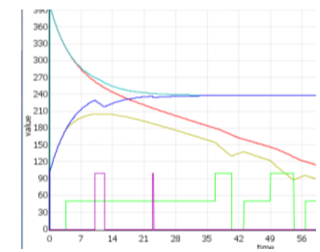
Energy Aware Buildings



Genetic Oscilator (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
- ....

www.uppaal.{org,com}

SMC Queries – Examples

- Pr[<=100](<> goal)
- Pr[#<=10]([] safe)
- Pr[x<=200](<> goal) >= 0.3
- E[<=100; 1000](min: expr)
- simulate 10 [<=100] { e1, e2, x1 }
- simulate 100 [<=10] { e } : 2 : goal

Exercise 28 (Jobshop scheduling part 2)