

Stochastic Timed Automata

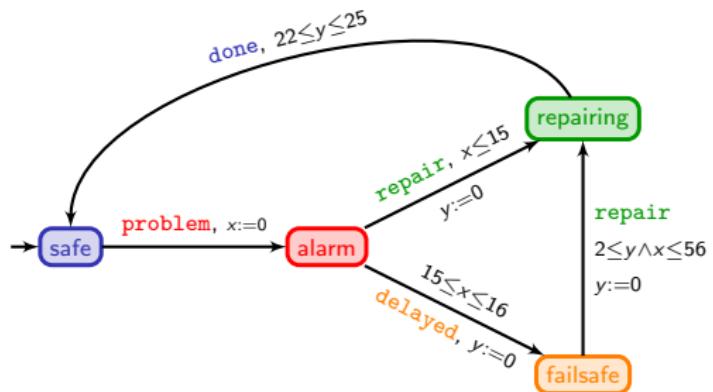
Patricia Bouyer-Decitre

LSV, CNRS & ENS Cachan, France

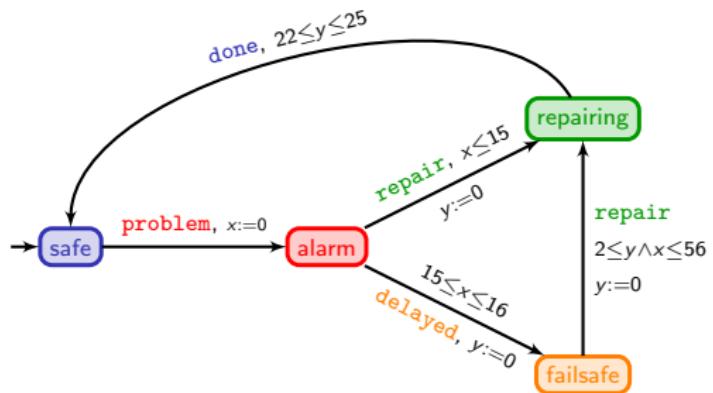
Based on joint works with Nathalie Bertrand, Thomas Brihaye,
Pierre Carlier, Quentin Menet, Christel Baier, ...



The model of timed automata



The model of timed automata



	safe	$\xrightarrow{23}$	safe	$\xrightarrow{\text{problem}}$	alarm	$\xrightarrow{15.6}$	alarm	$\xrightarrow{\text{delayed}}$	failsafe	
x	0		23		0		15.6		15.6	\dots
y	0		23		23		38.6		0	
	failsafe	$\xrightarrow{2.3}$	failsafe	$\xrightarrow{\text{repair}}$	repairing	$\xrightarrow{22.1}$	repairing	$\xrightarrow{\text{done}}$	safe	
\dots	15.6		17.9		17.9		40		40	
	0		2.3		0		22.1		22.1	

An example: The task graph scheduling problem

Compute $D \times (C \times (A+B)) + (A+B) + (C \times D)$ using two processors:

P_1 (fast):



time	
+	2 picoseconds
×	3 picoseconds

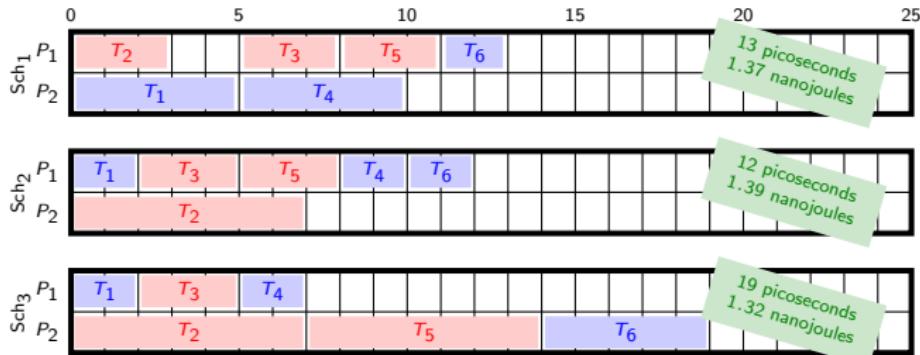
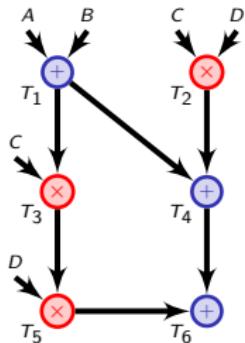
energy	
idle	10 Watt
in use	90 Watts

P_2 (slow):



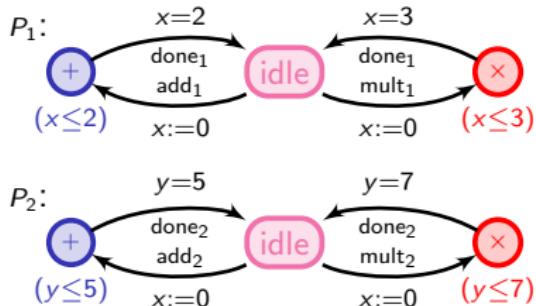
time	
+	5 picoseconds
×	7 picoseconds

energy	
idle	20 Watts
in use	30 Watts

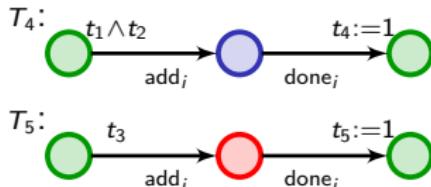


Modelling the task graph scheduling problem

- Processors

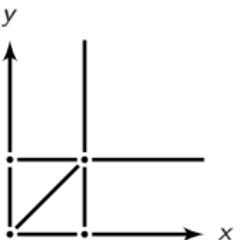
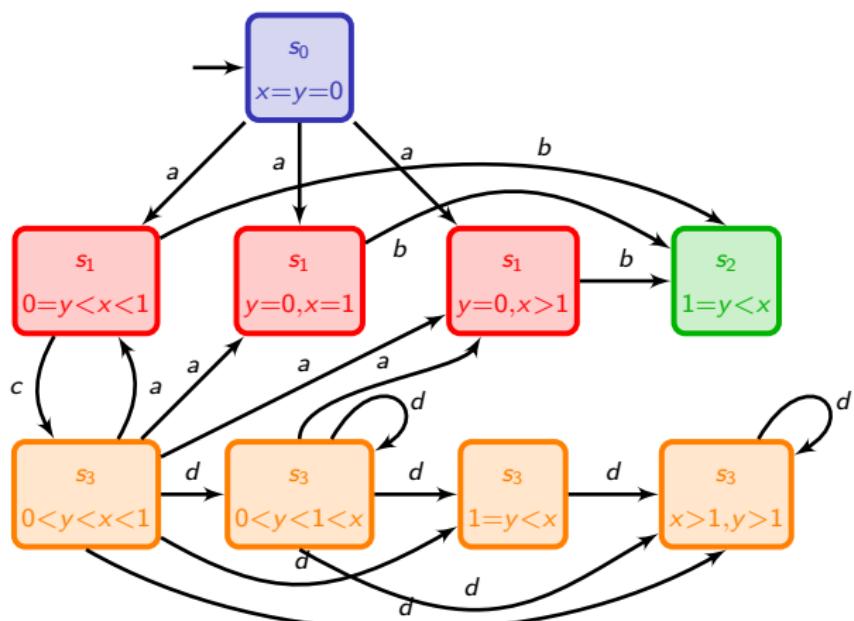
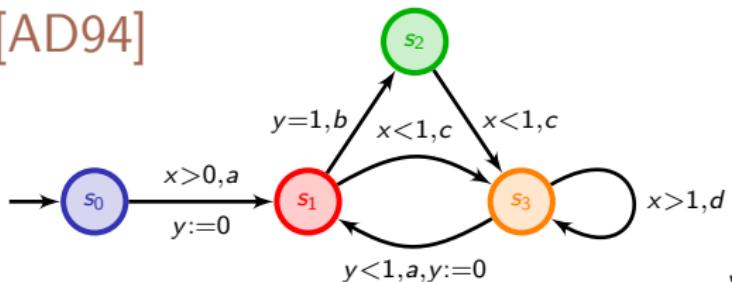


- Tasks



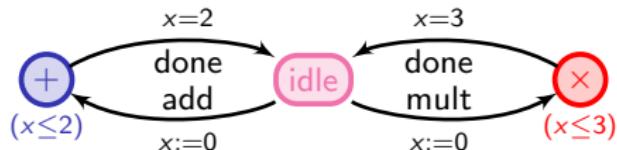
A schedule is a path in the product automaton

An example [AD94]



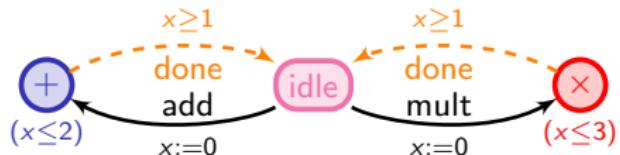
How to model uncertainty over delays?

- Using timed games



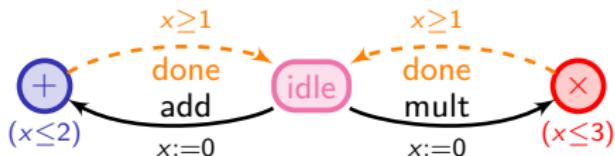
How to model uncertainty over delays?

- Using timed games

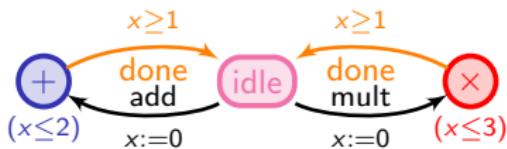


How to model uncertainty over delays?

- Using timed games

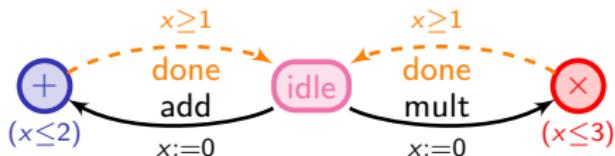


- Using stochastic delays

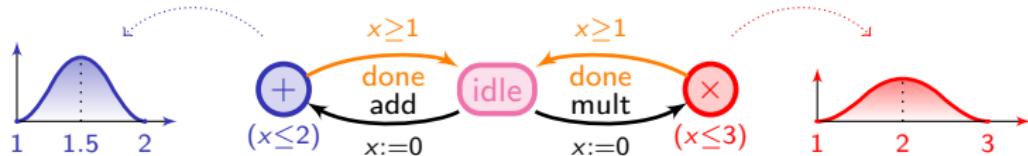


How to model uncertainty over delays?

- Using timed games



- Using stochastic delays



Existing models?

Models based on timed automata

54 s.

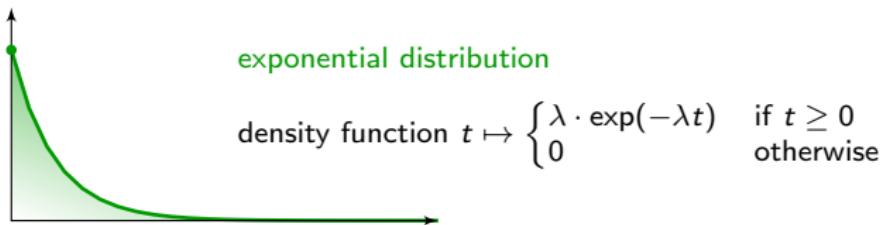
- Probabilistic timed automata [KNSS99]
~ only discrete probabilities over edges
- Continuous probabilistic timed automata [KNSS00]
~ resets of clocks are randomized, but only few results

[KNSS99] Kwiatkowska, Norman, Segala, Sproston. Automatic verification of real-time systems with discrete probability distributions (*ARTS'99*).

[KNSS00] Kwiatkowska, Norman, Segala, Sproston. Verifying quantitative properties of continuous probabilistic timed automata (*CONCUR'00*).

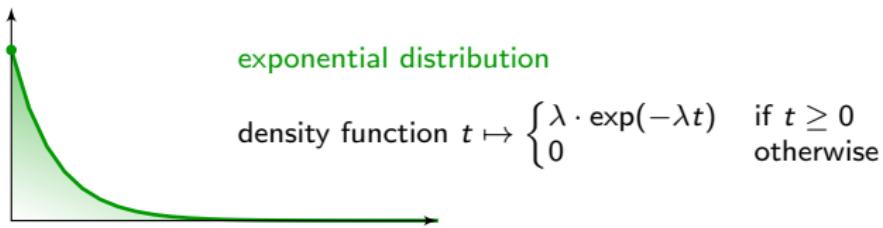
How can we attach probabilities to delays?

- The example of continuous-time Markov chains



How can we attach probabilities to delays?

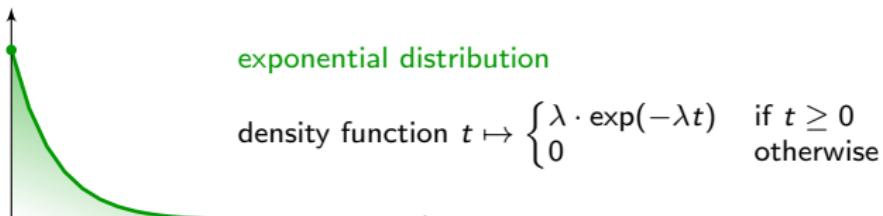
- The example of continuous-time Markov chains



~> this is ok if delays are in $[0, +\infty)$

How can we attach probabilities to delays?

- The example of continuous-time Markov chains

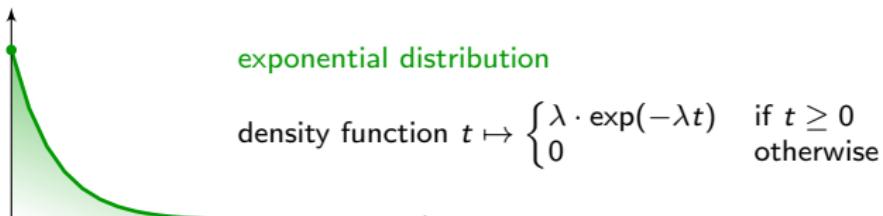


~ this is ok if delays are in $[0, +\infty)$

- But what if bounded intervals?

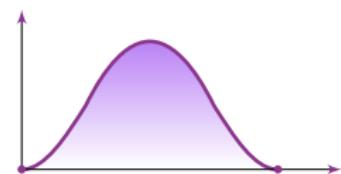
How can we attach probabilities to delays?

- The example of continuous-time Markov chains



~ this is ok if delays are in $[0, +\infty)$

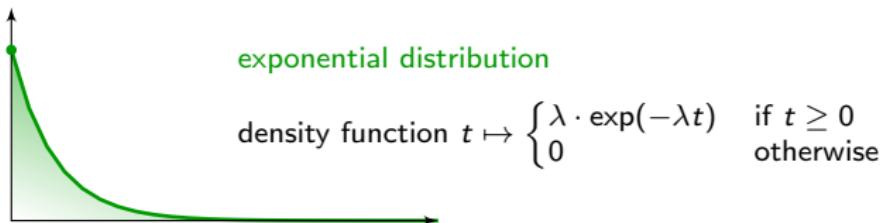
- But what if bounded intervals?



truncated normal distribution

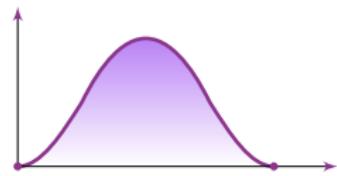
How can we attach probabilities to delays?

- The example of continuous-time Markov chains

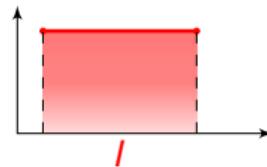


~ this is ok if delays are in $[0, +\infty)$

- But what if bounded intervals?

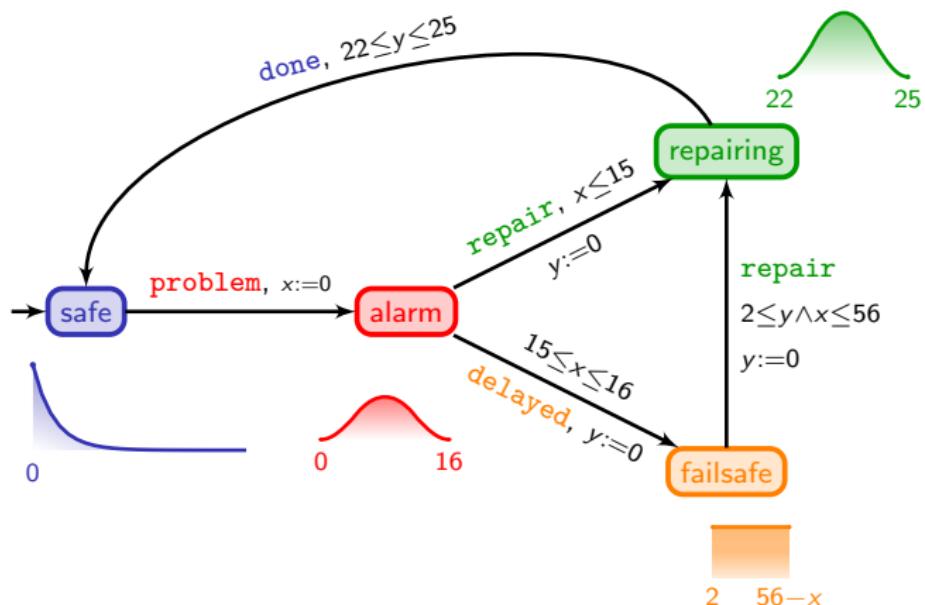


truncated normal distribution



uniform distribution
density function $t \mapsto \begin{cases} \frac{1}{|I|} & \text{if } t \geq 0 \\ 0 & \text{otherwise} \end{cases}$

How does a STA look like?



Some remarks

- This defines a purely stochastic process

Some remarks

- This defines a purely stochastic process
- Continuous-time Markov chains = STA with a single “useless” clock which is reset on all transitions. The distributions on delays are exponential distributions with a rate per location

Some remarks

- This defines a purely stochastic process
- Continuous-time Markov chains = STA with a single “useless” clock which is reset on all transitions. The distributions on delays are exponential distributions with a rate per location
- Finite-state generalized semi-Markov processes (residual-lifetime semantics) are STAs (if no fixed-delay events)

Some remarks

3:15

- This defines a purely stochastic process
- Continuous-time Markov chains = STA with a single “useless” clock which is reset on all transitions. The distributions on delays are exponential distributions with a rate per location
- Finite-state generalized semi-Markov processes (residual-lifetime semantics) are STAs (if no fixed-delay events)
- Allows to express richer timing constraints

Almost-sure model-checking

We are interested in (automatic) model-checking algorithms!

- Qualitative model-checking: decide whether

$$\mathbb{P}(\{\varrho \in \text{Runs}(s) \mid \varrho \models \varphi\}) = 1$$

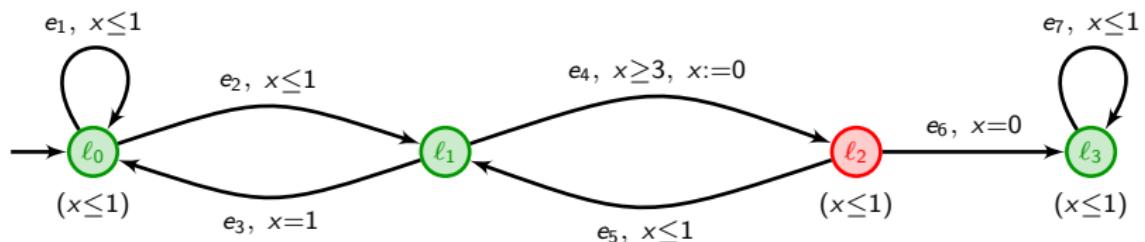
We write $s \approx \varphi$ whenever it is the case.

This is the almost-sure model-checking problem.

- Quantitative model-checking: compute (or approximate) the value

$$\mathbb{P}(\{\varrho \in \text{Runs}(s) \mid \varrho \models \varphi\})$$

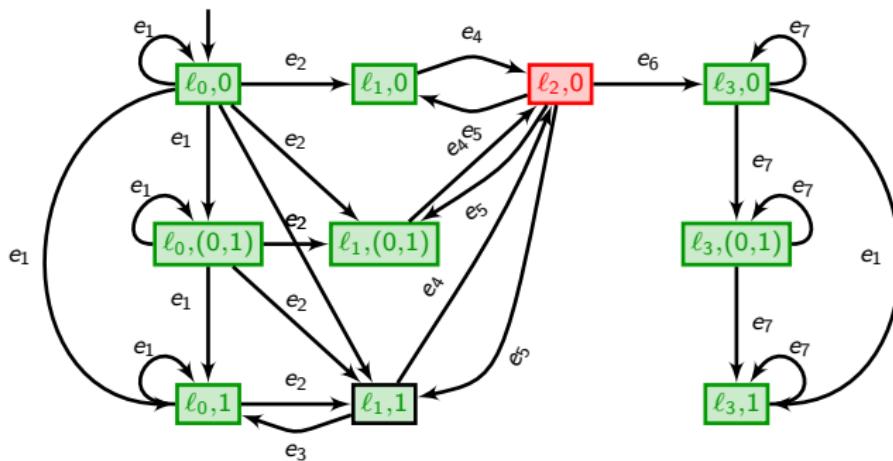
An example



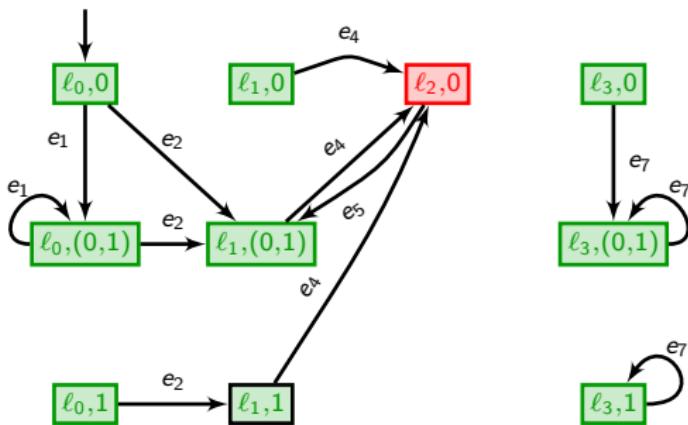
$$\mathcal{A} \not\models \mathbf{G}(\text{green} \Rightarrow \mathbf{F} \text{ red}) \quad \text{but} \quad \mathbb{P}(\mathcal{A} \models \mathbf{G}(\text{green} \Rightarrow \mathbf{F} \text{ red})) = 1$$

Indeed, almost surely, paths are of the form $e_1^* e_2 (e_4 e_5)^\omega$

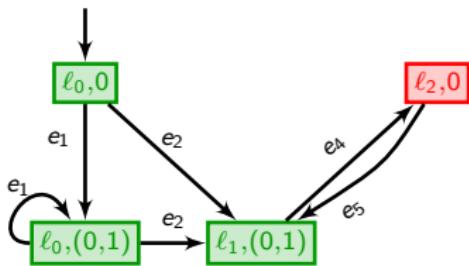
The classical region automaton



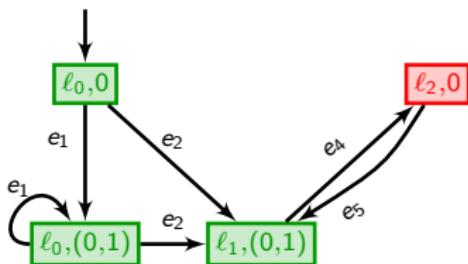
The pruned region automaton



The pruned region automaton



The pruned region automaton

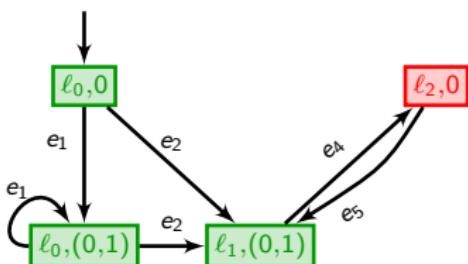


... viewed as a finite Markov chain $MC(\mathcal{A})$

It holds as well that:

$$\mathbb{P}(MC(\mathcal{A}) \models \mathbf{G}(\text{green} \Rightarrow \mathbf{F} \text{ red})) = 1$$

The pruned region automaton



... viewed as a finite Markov chain $MC(\mathcal{A})$

15:40

It holds as well that:

$$\mathbb{P}(MC(\mathcal{A}) \models \mathbf{G}(\text{green} \Rightarrow \mathbf{F} \text{ red})) = 1$$

When is that the case that

$$\mathbb{P}(\mathcal{A} \models \varphi) = 1 \quad \text{iff} \quad \mathbb{P}(MC(\mathcal{A}) \models \varphi) = 1 ?$$