A Probabilistic Semantics for Timed Automata

Christel Baier[1], Nathalie Bertrand[2], Patricia Bouyer[3]
Thomas Brihaye[4], Marcus Größer[1]

[1]Technische Universität Dresden – Germany

[2]IRISA/INRIA Rennes – France

[3]LSV – CNRS & ENS Cachan – France

[4]Université de Mons-Hainaut – Belgium

Motivation(s)

- Timed automata, an idealized mathematical model for real-time systems

Motivation(s)

- Timed automata, an idealized mathematical model for real-time systems
  - assumes infinite precision of clocks
  - assumes instantaneous actions
  - *etc...*

Motivation(s)

- Timed automata, an idealized mathematical model for real-time systems
  - assumes infinite precision of clocks
  - assumes instantaneous actions
  - *etc...*

➜ notion of strong robustness defined in [DDR04]

Motivation(s)

- Timed automata, an idealized mathematical model for real-time systems
  - assumes infinite precision of clocks
  - assumes instantaneous actions
  - *etc...*

➜ notion of strong robustness defined in [DDR04]

- In a model, only few traces may violate the correctness property: they may hence not be relevant...

Motivation(s)

- Timed automata, an idealized mathematical model for real-time systems
  - assumes infinite precision of clocks
  - assumes instantaneous actions
  - *etc...*

➜ notion of strong robustness defined in [DDR04]

- In a model, only few traces may violate the correctness property: they may hence not be relevant...

➜ topological notion of tube acceptance in [GHJ97]

Motivation(s)

- Timed automata, an idealized mathematical model for real-time systems
  - assumes infinite precision of clocks
  - assumes instantaneous actions
  - *etc...*

➜ notion of strong robustness defined in [DDR04]

- In a model, only few traces may violate the correctness property: they may hence not be relevant...

➜ topological notion of tube acceptance in [GHJ97]

➜ notion of fair correctness in [VV06] based on probabilities
(for untimed systems) + topological characterization

Motivation(s)

- Timed automata, an idealized mathematical model for real-time systems
  - assumes infinite precision of clocks
  - assumes instantaneous actions
  - *etc...*

➜ notion of strong robustness defined in [DDR04]

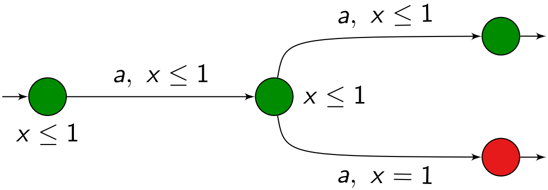
- In a model, only few traces may violate the correctness property: they may hence not be relevant...

➜ topological notion of tube acceptance in [GHJ97]

➜ notion of fair correctness in [VV06] based on probabilities
(for untimed systems) + topological characterization

Aim: Use probabilities to "relax" the semantics of timed automata

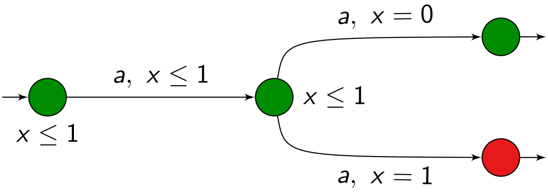
Initial example



Intuition: from the initial state,

this automaton *almost-surely* satisfies "$\mathbf{G}$ green"

A maybe less intuitive example



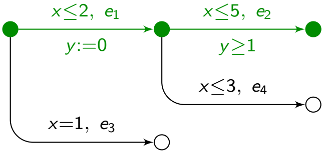
Does it *almost-surely* satisfy "$\mathbf{F}$ red"?

Our proposition

- $\pi(s \xrightarrow{e_1} \dots \xrightarrow{e_n})$: symbolic path from s firing edges $e_1, \dots, e_n$

Our proposition

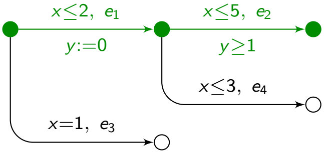
- $\pi(s \xrightarrow{e_1} \dots \xrightarrow{e_n})$: symbolic path from s firing edges $e_1, \dots, e_n$
- Example:



$$\pi(s_0 \xrightarrow{e_1}\xrightarrow{e_2}) = \{s_0 \xrightarrow{\tau_1, e_1} s_1 \xrightarrow{\tau_2, e_2} s_2 \mid \tau_1 \leq 2,\ \tau_1 + \tau_2 \leq 5,\ \tau_2 \geq 1\}$$

Our proposition

- $\pi(s \xrightarrow{e_1} \dots \xrightarrow{e_n})$: symbolic path from s firing edges $e_1, \dots, e_n$
- Example:



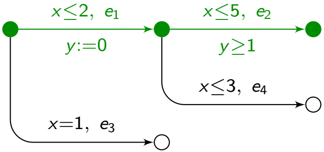
$$\pi(s_0 \xrightarrow{e_1}\xrightarrow{e_2}) = \{s_0 \xrightarrow{\tau_1, e_1} s_1 \xrightarrow{\tau_2, e_2} s_2 \mid \tau_1 \leq 2,\ \tau_1 + \tau_2 \leq 5,\ \tau_2 \geq 1\}$$

- Idea:

From state s_0:

Our proposition

- $\pi(s \xrightarrow{e_1} \ldots \xrightarrow{e_n})$: symbolic path from s firing edges $e_1, \ldots, e_n$
- Example:



$$\pi(s_0 \xrightarrow{e_1}\xrightarrow{e_2}) = \{s_0 \xrightarrow{\tau_1, e_1} s_1 \xrightarrow{\tau_2, e_2} s_2 \mid \tau_1 \leq 2,\ \tau_1 + \tau_2 \leq 5,\ \tau_2 \geq 1\}$$

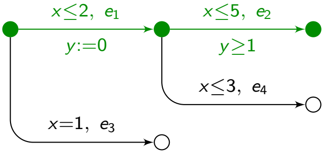
- Idea:

From state s_0:

- randomly choose a delay

Our proposition

- $\pi(s \xrightarrow{e_1} \dots \xrightarrow{e_n})$: symbolic path from s firing edges $e_1, \dots, e_n$
- Example:



$$\pi(s_0 \xrightarrow{e_1}\xrightarrow{e_2}) = \{s_0 \xrightarrow{\tau_1, e_1} s_1 \xrightarrow{\tau_2, e_2} s_2 \mid \tau_1 \leq 2,\ \tau_1 + \tau_2 \leq 5,\ \tau_2 \geq 1\}$$

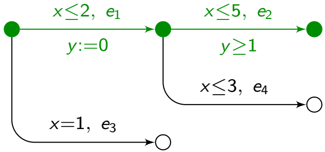
- Idea:

From state s_0:

- randomly choose a delay
- then randomly select an edge

Our proposition

- $\pi(s \xrightarrow{e_1} \ldots \xrightarrow{e_n})$: symbolic path from s firing edges $e_1, \ldots, e_n$
- Example:



$$\pi(s_0 \xrightarrow{e_1}\xrightarrow{e_2}) = \{s_0 \xrightarrow{\tau_1, e_1} s_1 \xrightarrow{\tau_2, e_2} s_2 \mid \tau_1 \leq 2,\ \tau_1 + \tau_2 \leq 5,\ \tau_2 \geq 1\}$$

- Idea:

From state s_0:

- randomly choose a delay
- then randomly select an edge
- then continue

Our proposition

symbolic path: $\pi(s \xrightarrow{e_1} \cdots \xrightarrow{e_n}) = \{s \xrightarrow{\tau_1, e_1} s_1 \cdots \xrightarrow{\tau_n, e_n} s_n\}$

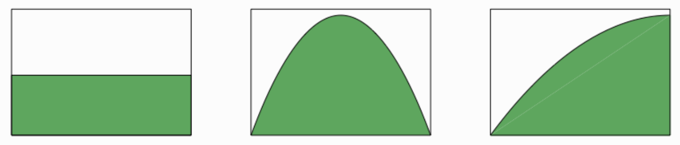
$$\mathbb{P}\Big(\pi(s \xrightarrow{e_1} \cdots \xrightarrow{e_n})\Big) = \int_{t \in I(s, e_1)} p_{s+t}(e_1)\, \mathbb{P}\Big(\pi(s_t \xrightarrow{e_2} \cdots \xrightarrow{e_n})\Big)\, \mathrm{d}\mu_s(t)$$

Our proposition

symbolic path: $\pi(s \xrightarrow{e_1} \cdots \xrightarrow{e_n}) = \{s \xrightarrow{\tau_1, e_1} s_1 \cdots \xrightarrow{\tau_n, e_n} s_n\}$

$$\mathbb{P}\Big(\pi(s \xrightarrow{e_1} \cdots \xrightarrow{e_n})\Big) = \int_{t \in I(s,e_1)} p_{s+t}(e_1)\, \mathbb{P}\Big(\pi(s_t \xrightarrow{e_2} \cdots \xrightarrow{e_n})\Big)\, \mathrm{d}\mu_s(t)$$

- $I(s, e_1) = \{\tau \mid s \xrightarrow{\tau, e_1}\}$ and μ_s distrib. over $I(s) = \bigcup_e I(s, e)$

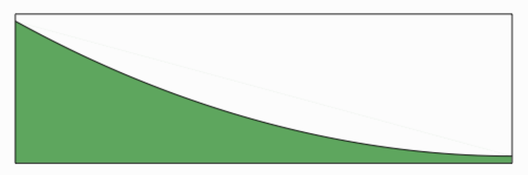


Our proposition

symbolic path: $\pi(s \xrightarrow{e_1} \cdots \xrightarrow{e_n}) = \{s \xrightarrow{\tau_1, e_1} s_1 \cdots \xrightarrow{\tau_n, e_n} s_n\}$

$$\mathbb{P}\Big(\pi(s \xrightarrow{e_1} \cdots \xrightarrow{e_n})\Big) = \int_{t \in I(s,e_1)} p_{s+t}(e_1)\, \mathbb{P}\Big(\pi(s_t \xrightarrow{e_2} \cdots \xrightarrow{e_n})\Big)\, \mathrm{d}\mu_s(t)$$

- $I(s, e_1) = \{\tau \mid s \xrightarrow{\tau, e_1}\}$ and μ_s distrib. over $I(s) = \bigcup_e I(s, e)$



Our proposition

symbolic path: $\pi(s \xrightarrow{e_1} \cdots \xrightarrow{e_n}) = \{s \xrightarrow{\tau_1, e_1} s_1 \cdots \xrightarrow{\tau_n, e_n} s_n\}$

$$\mathbb{P}\Big(\pi(s \xrightarrow{e_1} \cdots \xrightarrow{e_n})\Big) = \int_{t \in I(s,e_1)} p_{s+t}(e_1)\, \mathbb{P}\Big(\pi(s_t \xrightarrow{e_2} \cdots \xrightarrow{e_n})\Big)\, \mathrm{d}\mu_s(t)$$

- $I(s, e_1) = \{\tau \mid s \xrightarrow{\tau, e_1}\}$ and μ_s distrib. over $I(s) = \bigcup_e I(s, e)$
- p_{s+t} distrib. over transitions enabled in $s + t$

Our proposition

symbolic path: $\pi(s \xrightarrow{e_1} \cdots \xrightarrow{e_n}) = \{s \xrightarrow{\tau_1, e_1} s_1 \cdots \xrightarrow{\tau_n, e_n} s_n\}$

$$\mathbb{P}\Big(\pi(s \xrightarrow{e_1} \cdots \xrightarrow{e_n})\Big) = \int_{t \in I(s,e_1)} p_{s+t}(e_1)\, \mathbb{P}\Big(\pi(s_t \xrightarrow{e_2} \cdots \xrightarrow{e_n})\Big)\, \mathrm{d}\mu_s(t)$$

- $I(s, e_1) = \{\tau \mid s \xrightarrow{\tau, e_1}\}$ and μ_s distrib. over $I(s) = \bigcup_e I(s, e)$
- p_{s+t} distrib. over transitions enabled in $s + t$
- $s \xrightarrow{t} s + t \xrightarrow{e_1} s_t$

Our proposition

$$\mathbb{P}\Big(\pi\big(s \xrightarrow{e_1} \cdots \xrightarrow{e_n}\big)\Big) = \int_{t \in I(s,e_1)} p_{s+t}(e_1)\, \mathbb{P}\Big(\pi\big(s_t \xrightarrow{e_2} \cdots \xrightarrow{e_n}\big)\Big)\, \mathrm{d}\mu_s(t)$$

Our proposition

$$\mathbb{P}\Big(\pi\big(s \xrightarrow{e_1} \cdots \xrightarrow{e_n}\big)\Big) = \int_{t \in I(s,e_1)} p_{s+t}(e_1)\, \mathbb{P}\Big(\pi\big(s_t \xrightarrow{e_2} \cdots \xrightarrow{e_n}\big)\Big)\, \mathrm{d}\mu_s(t)$$

- Can be viewed as an *n*-dimensional integral

Our proposition

$$\mathbb{P}\Big(\pi\big(s \xrightarrow{e_1} \cdots \xrightarrow{e_n}\big)\Big) = \int_{t \in I(s,e_1)} p_{s+t}(e_1)\, \mathbb{P}\Big(\pi\big(s_t \xrightarrow{e_2} \cdots \xrightarrow{e_n}\big)\Big)\, \mathrm{d}\mu_s(t)$$

- Can be viewed as an n-dimensional integral
- Easy extension to constrained symbolic paths
  $$\pi_{\mathcal{C}}\big(s \xrightarrow{e_1} \cdots \xrightarrow{e_n}\big) = \{s \xrightarrow{\tau_1, e_1} s_1 \cdots \xrightarrow{\tau_n, e_n} s_n \mid (\tau_1, \cdots, \tau_n) \models \mathcal{C}\}$$

Our proposition

$$\mathbb{P}\Big(\pi(s \xrightarrow{e_1} \cdots \xrightarrow{e_n})\Big) = \int_{t \in I(s,e_1)} p_{s+t}(e_1)\, \mathbb{P}\Big(\pi(s_t \xrightarrow{e_2} \cdots \xrightarrow{e_n})\Big)\, \mathrm{d}\mu_s(t)$$

- Can be viewed as an n-dimensional integral
- Easy extension to constrained symbolic paths
  $$\pi_{\mathcal{C}}(s \xrightarrow{e_1} \cdots \xrightarrow{e_n}) = \{s \xrightarrow{\tau_1, e_1} s_1 \cdots \xrightarrow{\tau_n, e_n} s_n \mid (\tau_1, \cdots, \tau_n) \models \mathcal{C}\}$$
- Definition over sets of infinite runs:

Our proposition

$$\mathbb{P}\Big(\pi(s \xrightarrow{e_1} \cdots \xrightarrow{e_n})\Big) = \int_{t \in I(s,e_1)} p_{s+t}(e_1)\,\mathbb{P}\Big(\pi(s_t \xrightarrow{e_2} \cdots \xrightarrow{e_n})\Big)\,\mathrm{d}\mu_s(t)$$

- Can be viewed as an n-dimensional integral
- Easy extension to constrained symbolic paths
  $\pi_{\mathcal{C}}(s \xrightarrow{e_1} \cdots \xrightarrow{e_n}) = \{s \xrightarrow{\tau_1, e_1} s_1 \cdots \xrightarrow{\tau_n, e_n} s_n \mid (\tau_1, \cdots, \tau_n) \models \mathcal{C}\}$
- Definition over sets of infinite runs:
  - $\mathsf{Cyl}(\pi_{\mathcal{C}}(s \xrightarrow{e_1} \cdots \xrightarrow{e_n})) = \{\varrho \cdot \varrho' \mid \varrho \in \pi_{\mathcal{C}}(s \xrightarrow{e_1} \cdots \xrightarrow{e_n})\}$

Our proposition

$$\mathbb{P}\Big(\pi\big(s \xrightarrow{e_1} \cdots \xrightarrow{e_n}\big)\Big) = \int_{t \in I(s,e_1)} p_{s+t}(e_1)\,\mathbb{P}\Big(\pi\big(s_t \xrightarrow{e_2} \cdots \xrightarrow{e_n}\big)\Big)\,\mathrm{d}\mu_s(t)$$

- Can be viewed as an *n-dimensional integral*
- Easy extension to constrained symbolic paths
  $$\pi_{\mathcal{C}}\big(s \xrightarrow{e_1} \cdots \xrightarrow{e_n}\big) = \{s \xrightarrow{\tau_1, e_1} s_1 \cdots \xrightarrow{\tau_n, e_n} s_n \mid (\tau_1, \cdots, \tau_n) \models \mathcal{C}\}$$
- Definition over sets of infinite runs:
  - $\mathrm{Cyl}(\pi_{\mathcal{C}}(s \xrightarrow{e_1} \cdots \xrightarrow{e_n})) = \{\varrho \cdot \varrho' \mid \varrho \in \pi_{\mathcal{C}}(s \xrightarrow{e_1} \cdots \xrightarrow{e_n})\}$
  - $\mathbb{P}\Big(\mathrm{Cyl}(\pi_{\mathcal{C}}(s \xrightarrow{e_1} \cdots \xrightarrow{e_n}))\Big) = \mathbb{P}\Big(\pi_{\mathcal{C}}(s \xrightarrow{e_1} \cdots \xrightarrow{e_n})\Big)$

Our proposition

$$\mathbb{P}\Big(\pi\big(s \xrightarrow{e_1} \cdots \xrightarrow{e_n}\big)\Big) = \int_{t \in I(s,e_1)} p_{s+t}(e_1)\,\mathbb{P}\Big(\pi\big(s_t \xrightarrow{e_2} \cdots \xrightarrow{e_n}\big)\Big)\,\mathrm{d}\mu_s(t)$$

- Can be viewed as an n-dimensional integral
- Easy extension to constrained symbolic paths
  $$\pi_{\mathcal{C}}\big(s \xrightarrow{e_1} \cdots \xrightarrow{e_n}\big) = \{s \xrightarrow{\tau_1, e_1} s_1 \cdots \xrightarrow{\tau_n, e_n} s_n \mid (\tau_1, \cdots, \tau_n) \models \mathcal{C}\}$$
- Definition over sets of infinite runs:
  - $\mathrm{Cyl}(\pi_{\mathcal{C}}(s \xrightarrow{e_1} \cdots \xrightarrow{e_n})) = \{\varrho \cdot \varrho' \mid \varrho \in \pi_{\mathcal{C}}(s \xrightarrow{e_1} \cdots \xrightarrow{e_n})\}$
  - $\mathbb{P}\Big(\mathrm{Cyl}(\pi_{\mathcal{C}}(s \xrightarrow{e_1} \cdots \xrightarrow{e_n}))\Big) = \mathbb{P}\Big(\pi_{\mathcal{C}}(s \xrightarrow{e_1} \cdots \xrightarrow{e_n})\Big)$
  - unique extension of $\mathbb{P}$ to the generated σ-algebra

Our proposition

$$\mathbb{P}\Big(\pi(s \xrightarrow{e_1} \cdots \xrightarrow{e_n})\Big) = \int_{t \in I(s,e_1)} p_{s+t}(e_1)\, \mathbb{P}\Big(\pi(s_t \xrightarrow{e_2} \cdots \xrightarrow{e_n})\Big)\, \mathrm{d}\mu_s(t)$$

- Can be viewed as an *n*-dimensional integral
- Easy extension to constrained symbolic paths
  $$\pi_{\mathcal{C}}(s \xrightarrow{e_1} \cdots \xrightarrow{e_n}) = \{s \xrightarrow{\tau_1, e_1} s_1 \cdots \xrightarrow{\tau_n, e_n} s_n \mid (\tau_1, \cdots, \tau_n) \models \mathcal{C}\}$$
- Definition over sets of infinite runs:
  - $\mathsf{Cyl}(\pi_{\mathcal{C}}(s \xrightarrow{e_1} \cdots \xrightarrow{e_n})) = \{\varrho \cdot \varrho' \mid \varrho \in \pi_{\mathcal{C}}(s \xrightarrow{e_1} \cdots \xrightarrow{e_n})\}$
  - $\mathbb{P}\Big(\mathsf{Cyl}(\pi_{\mathcal{C}}(s \xrightarrow{e_1} \cdots \xrightarrow{e_n}))\Big) = \mathbb{P}\Big(\pi_{\mathcal{C}}(s \xrightarrow{e_1} \cdots \xrightarrow{e_n})\Big)$
  - unique extension of $\mathbb{P}$ to the generated σ-algebra
- Property: $\mathbb{P}$ is a probability measure over sets of infinite runs

Our proposition

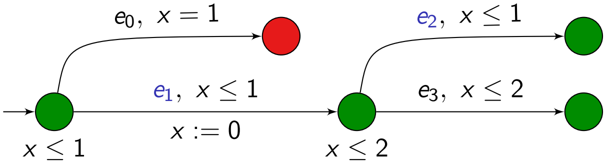
$$\mathbb{P}\Big(\pi(s \xrightarrow{e_1} \cdots \xrightarrow{e_n})\Big) = \int_{t \in I(s,e_1)} p_{s+t}(e_1)\, \mathbb{P}\Big(\pi(s_t \xrightarrow{e_2} \cdots \xrightarrow{e_n})\Big)\, \mathrm{d}\mu_s(t)$$

- Can be viewed as an n-dimensional integral
- Easy extension to constrained symbolic paths

  $$\pi_{\mathcal{C}}(s \xrightarrow{e_1} \cdots \xrightarrow{e_n}) = \{s \xrightarrow{\tau_1, e_1} s_1 \cdots \xrightarrow{\tau_n, e_n} s_n \mid (\tau_1, \cdots, \tau_n) \models \mathcal{C}\}$$

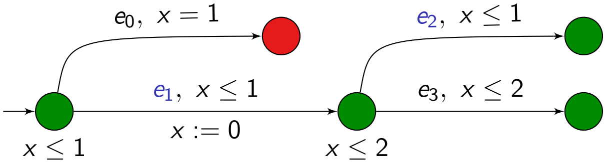
- Definition over sets of infinite runs:
  - $\mathsf{Cyl}(\pi_{\mathcal{C}}(s \xrightarrow{e_1} \cdots \xrightarrow{e_n})) = \{\varrho \cdot \varrho' \mid \varrho \in \pi_{\mathcal{C}}(s \xrightarrow{e_1} \cdots \xrightarrow{e_n})\}$
  - $\mathbb{P}\Big(\mathsf{Cyl}(\pi_{\mathcal{C}}(s \xrightarrow{e_1} \cdots \xrightarrow{e_n}))\Big) = \mathbb{P}\Big(\pi_{\mathcal{C}}(s \xrightarrow{e_1} \cdots \xrightarrow{e_n})\Big)$
  - unique extension of $\mathbb{P}$ to the generated σ-algebra
- Property: $\mathbb{P}$ is a probability measure over sets of infinite runs
- Example:
  - $\mathsf{Zeno}(s) = \bigcup_{M \in \mathbb{N}} \bigcap_{n \in \mathbb{N}} \bigcup_{(e_1, \cdots, e_n) \in E^n} \mathsf{Cyl}(\pi_{\Sigma_i \tau_i \leq M}(s \xrightarrow{e_1} \cdots \xrightarrow{e_n}))$

An example of computation (with uniform distributions)



The probability of the symbolic path $\pi(s_0 \xrightarrow{e_1}\xrightarrow{e_2})$ is $\frac{1}{4}$.

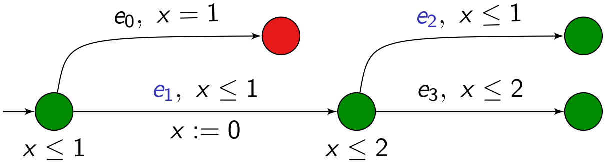
An example of computation (with uniform distributions)



The probability of the symbolic path $\pi(s_0 \xrightarrow{e_1}\xrightarrow{e_2})$ is $\frac{1}{4}$.

$$\mathbb{P}\Big(\pi(s_0 \xrightarrow{e_1}\xrightarrow{e_2})\Big) = \int_0^1 \mathbb{P}\Big(\pi(s_1 \xrightarrow{e_2})\Big)\mathrm{d}\mu_{s_0}(t) + \int_1^1 \frac{\mathbb{P}\Big(\pi(s_1 \xrightarrow{e_2})\Big)}{2}\mathrm{d}\mu_{s_0}(t)$$

An example of computation (with uniform distributions)

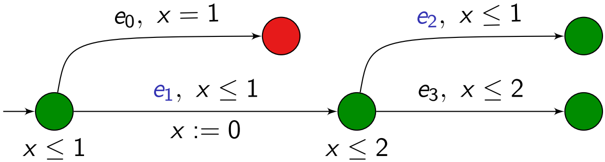


The probability of the symbolic path $\pi(s_0 \xrightarrow{e_1}\xrightarrow{e_2})$ is $\frac{1}{4}$.

$$\mathbb{P}\Big(\pi(s_0 \xrightarrow{e_1}\xrightarrow{e_2})\Big) = \int_0^1 \mathbb{P}\Big(\pi(s_1 \xrightarrow{e_2})\Big)\mathrm{d}\mu_{s_0}(t) + \int_1^1 \frac{\mathbb{P}\Big(\pi(s_1 \xrightarrow{e_2})\Big)}{2}\mathrm{d}\mu_{s_0}(t)$$

$$= \int_0^1 \int_0^1 \left(\frac{\mathbb{P}\Big(\pi(s_2)\Big)}{2}\mathrm{d}\mu_{s_1}(u)\right) \mathrm{d}\mu_{s_0}(t)$$

An example of computation (with uniform distributions)



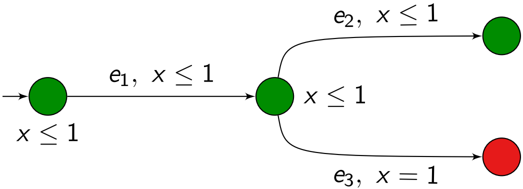
The probability of the symbolic path $\pi(s_0 \xrightarrow{e_1}\xrightarrow{e_2})$ is $\frac{1}{4}$.

$$\mathbb{P}\Big(\pi\big(s_0 \xrightarrow{e_1}\xrightarrow{e_2}\big)\Big) = \int_0^1 \mathbb{P}\Big(\pi\big(s_1 \xrightarrow{e_2}\big)\Big)\mathrm{d}\mu_{s_0}(t) + \int_1^1 \frac{\mathbb{P}\Big(\pi\big(s_1 \xrightarrow{e_2}\big)\Big)}{2}\mathrm{d}\mu_{s_0}(t)$$

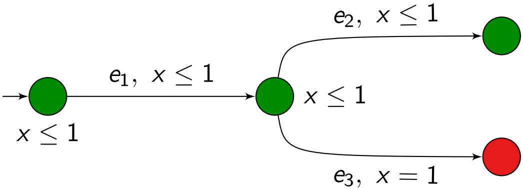
$$= \int_0^1 \int_0^1 \left(\frac{\mathbb{P}\big(\pi(s_2)\big)}{2}\mathrm{d}\mu_{s_1}(u)\right)\mathrm{d}\mu_{s_0}(t)$$

$$= \int_0^1 \int_0^1 \left(\frac{1}{2}\frac{\mathrm{d}u}{2}\right)\mathrm{d}t \quad = \frac{1}{4}$$

Back to the first example

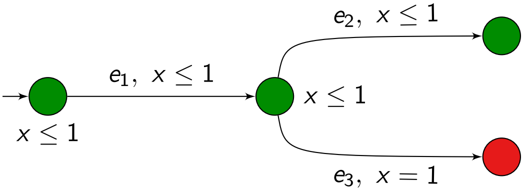


Back to the first example



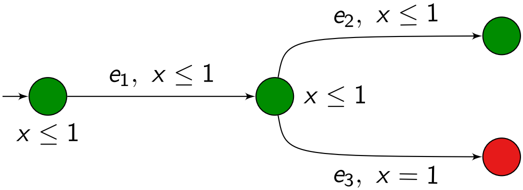
- $\mathbb{P}\Big(\pi(s_0 \xrightarrow{e_1}\xrightarrow{e_2})\Big) = 1$

Back to the first example



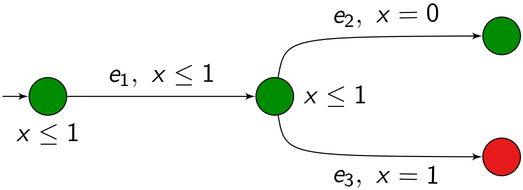
- $\mathbb{P}\Big(\pi\big(s_0 \xrightarrow{e_1}\xrightarrow{e_2}\big)\Big) = 1$
- $\mathbb{P}\Big(\pi\big(s_0 \xrightarrow{e_1}\xrightarrow{e_3}\big)\Big) = 0$

Back to the first example

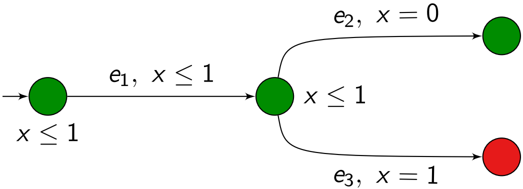


- $\mathbb{P}\Big(\pi\big(s_0 \xrightarrow{e_1}\xrightarrow{e_2}\big)\Big) = 1$
- $\mathbb{P}\Big(\pi\big(s_0 \xrightarrow{e_1}\xrightarrow{e_3}\big)\Big) = 0$
- $\mathbb{P}\Big(\mathbf{G}\ \text{green}\Big) = 1$

Back to the second example

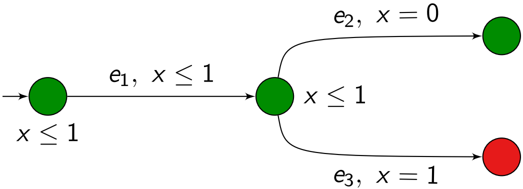


Back to the second example



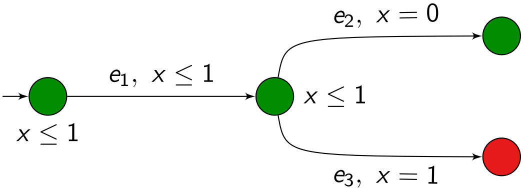
- $\mathbb{P}\Big(\pi\big(s_0 \xrightarrow{e_1}\xrightarrow{e_2}\big)\Big) = 0$

Back to the second example



- $\mathbb{P}\Big(\pi\big(s_0 \xrightarrow{e_1}\xrightarrow{e_2}\big)\Big) = 0$
- $\mathbb{P}\Big(\pi\big(s_0 \xrightarrow{e_1}\xrightarrow{e_3}\big)\Big) = 1$

Back to the second example



- $\mathbb{P}\Big(\pi\big(s_0 \xrightarrow{e_1}\xrightarrow{e_2}\big)\Big) = 0$
- $\mathbb{P}\Big(\pi\big(s_0 \xrightarrow{e_1}\xrightarrow{e_3}\big)\Big) = 1$
- $\mathbb{P}\Big(\mathbf{F}\ \text{red}\Big) = 1$

Almost-sure model-checking

If φ is an LTL formula,

$$s \mathrel{|\!\!\approx} \varphi \quad \stackrel{\text{def}}{\Leftrightarrow} \quad \mathbb{P}\Big(\{\varrho \in \mathsf{Runs}(s) \mid \varrho \models \varphi\}\Big) = 1$$

Almost-sure model-checking

If φ is an LTL formula,

$$s \mathrel{|\!\!\approx} \varphi \quad \overset{\text{def}}{\Leftrightarrow} \quad \mathbb{P}\Big(\{\varrho \in \mathsf{Runs}(s) \mid \varrho \models \varphi\}\Big) = 1$$

(This definition extends naturally to CTL* specifications...)

Almost-sure model-checking

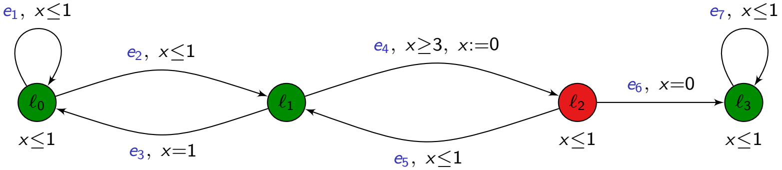
If φ is an LTL formula,

$$s \mathrel{|\!\!\approx} \varphi \quad \stackrel{\text{def}}{\Leftrightarrow} \quad \mathbb{P}\Big(\{\varrho \in \mathsf{Runs}(s) \mid \varrho \models \varphi\}\Big) = 1$$

(This definition extends naturally to $\mathsf{CTL}^\star$ specifications...)

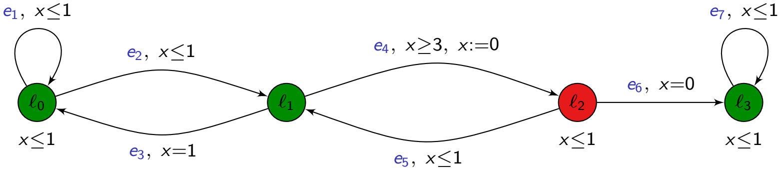
We want to decide the almost-sure model-checking...
(This is a qualitative question)

An example



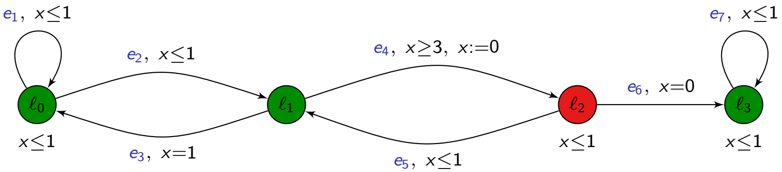
e_1, $x \leq 1$
e_2, $x \leq 1$
e_4, $x \geq 3$, $x := 0$
e_6, $x = 0$
e_7, $x \leq 1$
ℓ_0
ℓ_1
ℓ_2
ℓ_3
$x \leq 1$
e_3, $x = 1$
e_5, $x \leq 1$
$x \leq 1$
$x \leq 1$

An example



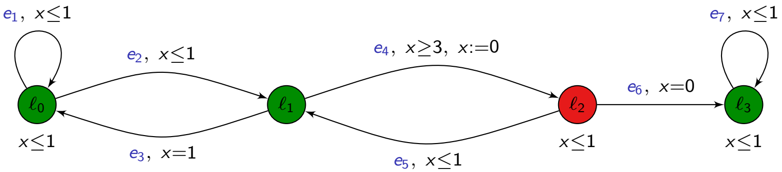
$\mathcal{A} \not\models$ **G**(green $\Rightarrow$ **F** red)

An example



$\mathcal{A} \not\models \mathbf{G}(\text{green} \Rightarrow \mathbf{F}\ \text{red})$ but $\mathcal{A} \mathrel{|}\!\approx \mathbf{G}(\text{green} \Rightarrow \mathbf{F}\ \text{red})$

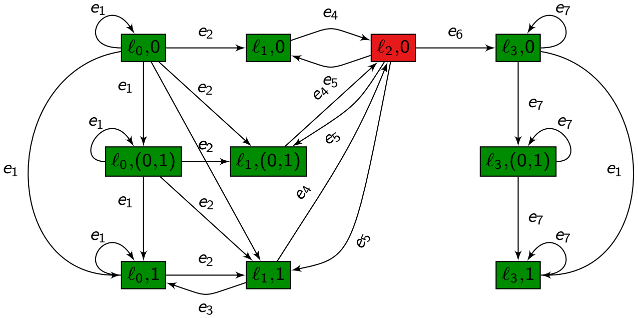
An example



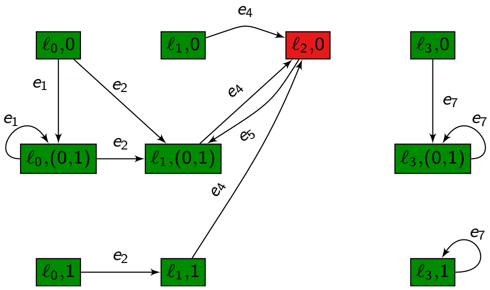
$$\mathcal{A} \not\models \mathbf{G}(\text{green} \Rightarrow \mathbf{F}\ \text{red}) \qquad \text{but} \qquad \mathcal{A} \models\!\!\!\approx \mathbf{G}(\text{green} \Rightarrow \mathbf{F}\ \text{red})$$

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

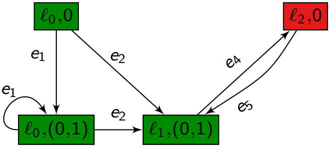
The classical region automaton



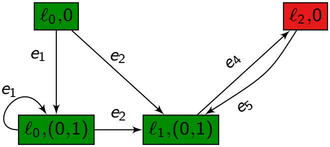
The pruned region automaton



The pruned region automaton

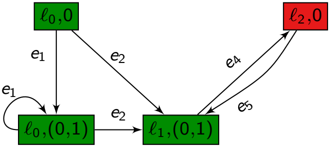


The pruned region automaton



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

The pruned region automaton



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

Theorem

For single-clock timed automata,

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

Result

Theorem

For single-clock timed automata, the almost-sure model-checking

- of LTL is PSPACE-Complete
- of ω-regular properties is NLOGSPACE-Complete

Result

Theorem

For single-clock timed automata, the almost-sure model-checking

- of LTL is PSPACE-Complete
- of ω-regular properties is NLOGSPACE-Complete

- Complexity:

Result

Theorem

For single-clock timed automata, the almost-sure model-checking

- of LTL is PSPACE-Complete
- of ω-regular properties is NLOGSPACE-Complete

- Complexity:
  - size of single-clock region automata = polynomial [LMS04]

Result

Theorem

For single-clock timed automata, the almost-sure model-checking

- of LTL is PSPACE-Complete
- of ω-regular properties is NLOGSPACE-Complete

- Complexity:
  - size of single-clock region automata = polynomial [LMS04]
  - apply result of [CSS03] to the finite Markov chain

Result

Theorem

For single-clock timed automata, the almost-sure model-checking

- of LTL is PSPACE-Complete
- of ω-regular properties is NLOGSPACE-Complete

- Complexity:
  - size of single-clock region automata = polynomial [LMS04]
  - apply result of [CSS03] to the finite Markov chain
- Correctness: the proof is rather involved

Result

Theorem

For single-clock timed automata, the almost-sure model-checking

- of LTL is PSPACE-Complete
- of ω-regular properties is NLOGSPACE-Complete

- Complexity:
  - size of single-clock region automata = polynomial [LMS04]
  - apply result of [CSS03] to the finite Markov chain
- Correctness: the proof is rather involved
  - requires the definition of a topology over the set of paths

Result

Theorem

For single-clock timed automata, the almost-sure model-checking

- of LTL is PSPACE-Complete
- of ω-regular properties is NLOGSPACE-Complete

- Complexity:
  - size of single-clock region automata = polynomial [LMS04]
  - apply result of [CSS03] to the finite Markov chain
- Correctness: the proof is rather involved
  - requires the definition of a topology over the set of paths
  - notions of largeness (for proba 1) and meagerness (for proba 0)

Result

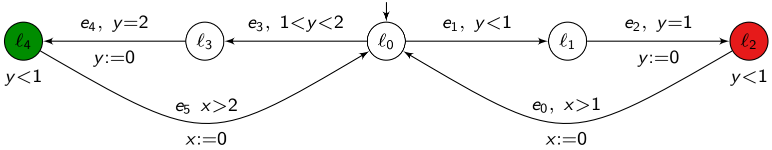
Theorem

For single-clock timed automata, the almost-sure model-checking

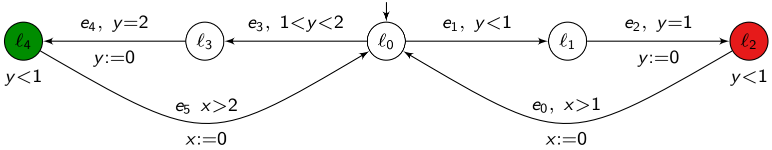
- of LTL is PSPACE-Complete
- of ω-regular properties is NLOGSPACE-Complete

- Complexity:
  - size of single-clock region automata = polynomial [LMS04]
  - apply result of [CSS03] to the finite Markov chain
- Correctness: the proof is rather involved
  - requires the definition of a topology over the set of paths
  - notions of largeness (for proba 1) and meagerness (for proba 0)
  - link between probabilities and topology thanks to the topological games called Banach-Mazur games

An example with two clocks

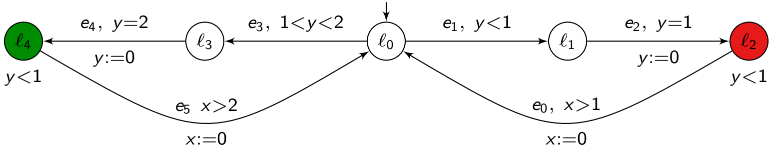


An example with two clocks



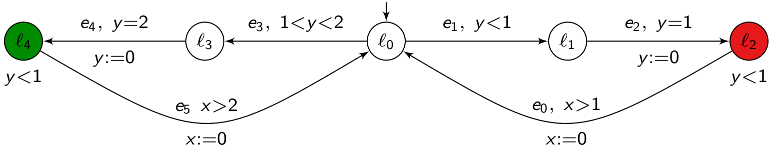
- If the previous algorithm was correct, $\mathcal{A} \not\approx \mathbf{G}\,\mathbf{F}$ red $\wedge$ $\mathbf{G}\,\mathbf{F}$ green

An example with two clocks



- If the previous algorithm was correct, $\mathcal{A} \mathrel{|\!\!\approx} \mathbf{G}\,\mathbf{F}$ red $\wedge$ $\mathbf{G}\,\mathbf{F}$ green
- However, we can prove that $\mathbb{P}\big(\mathbf{G}\,\neg\text{red}\big) > 0$

An example with two clocks



- If the previous algorithm was correct, $\mathcal{A} \approx\!\!\!\mid \mathbf{G}\,\mathbf{F}$ red $\wedge$ $\mathbf{G}\,\mathbf{F}$ green
- However, we can prove that $\mathbb{P}\Big(\mathbf{G}\,\neg\text{red}\Big) > 0$
- There is a *strange* convergence phenomenon: along an execution, if $\delta_i > 0$ is the delay in location ℓ_4, then we have that $\sum_i \delta_i \leq 1$

A note on Zeno behaviours

- The set of Zeno behaviours is measurable:

$$\mathsf{Zeno}(s) = \bigcup_{M \in \mathbb{N}} \bigcap_{n \in \mathbb{N}} \bigcup_{(e_1, \cdots, e_n) \in E^n} \mathsf{Cyl}(\pi(s \xrightarrow{e_1} \cdots \xrightarrow{e_n}))$$

A note on Zeno behaviours

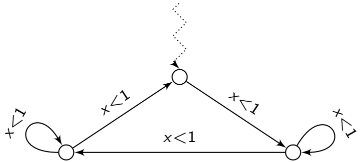
- The set of Zeno behaviours is measurable:
$$\text{Zeno}(s) = \bigcup_{M \in \mathbb{N}} \bigcap_{n \in \mathbb{N}} \bigcup_{(e_1, \cdots, e_n) \in E^n} \text{Cyl}(\pi(s \xrightarrow{e_1} \cdots \xrightarrow{e_n}))$$
- In single-clock timed automata, we can decide in NLOGSPACE whether $\mathbb{P}\Big(\text{Zeno}(s)\Big) = 0$:

A note on Zeno behaviours

- The set of Zeno behaviours is measurable:

$$\mathrm{Zeno}(s) = \bigcup_{M\in\mathbb{N}} \bigcap_{n\in\mathbb{N}} \bigcup_{(e_1,\cdots,e_n)\in E^n} \mathrm{Cyl}(\pi(s \xrightarrow{e_1} \cdots \xrightarrow{e_n}))$$

- In single-clock timed automata, we can decide in NLOGSPACE whether $\mathbb{P}\Big(\mathrm{Zeno}(s)\Big) = 0$:
  - check whether there is a purely Zeno BSCC in $MC(\mathcal{A})$

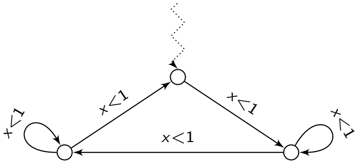


A note on Zeno behaviours

- The set of Zeno behaviours is measurable:

$$\text{Zeno}(s) = \bigcup_{M \in \mathbb{N}} \bigcap_{n \in \mathbb{N}} \bigcup_{(e_1,\cdots,e_n) \in E^n} \text{Cyl}(\pi(s \xrightarrow{e_1} \cdots \xrightarrow{e_n}))$$

- In single-clock timed automata, we can decide in NLOGSPACE whether $\mathbb{P}\big(\text{Zeno}(s)\big) = 0$:
  - check whether there is a purely Zeno BSCC in $MC(\mathcal{A})$



  - an interesting notion of non-Zeno timed automata

Related works

- Other "probabilistic and timed" (automata-)based models

Related works

- Other “probabilistic and timed” (automata-)based models
  - probabilistic timed automata *à la* PRISM [KNSS02]

Related works

- Other “probabilistic and timed” (automata-)based models
  - probabilistic timed automata *à la* PRISM [KNSS02]
  - real-time probabilistic systems [ACD91,ACD92]

Related works

- Other "probabilistic and timed" (automata-)based models
  - probabilistic timed automata *à la* PRISM [KNSS02]
  - real-time probabilistic systems [ACD91,ACD92]
  - dense-time Markov chains [BHHK03]

Related works

- Other “probabilistic and timed” (automata-)based models
  - probabilistic timed automata *à la* PRISM [KNSS02]
  - real-time probabilistic systems [ACD91,ACD92]
  - dense-time Markov chains [BHHK03]

NB: our model generalizes dense-time Markov chains

Related works

- Other “probabilistic and timed” (automata-)based models
  - probabilistic timed automata *à la* PRISM [KNSS02]
  - real-time probabilistic systems [ACD91,ACD92]
  - dense-time Markov chains [BHHK03]

  NB: our model generalizes dense-time Markov chains

- Labelled Markov processes over a continuum [DGJP03,04]

Related works

- Other "probabilistic and timed" (automata-)based models
  - probabilistic timed automata *à la* PRISM [KNSS02]
  - real-time probabilistic systems [ACD91,ACD92]
  - dense-time Markov chains [BHHK03]

  NB: our model generalizes dense-time Markov chains

- Labelled Markov processes over a continuum [DGJP03,04]

- Strong relation with robustness

Related works

- Other "probabilistic and timed" (automata-)based models
  - probabilistic timed automata *à la* PRISM [KNSS02]
  - real-time probabilistic systems [ACD91,ACD92]
  - dense-time Markov chains [BHHK03]

  NB: our model generalizes dense-time Markov chains

- Labelled Markov processes over a continuum [DGJP03,04]

- Strong relation with robustness
  - robust timed automata [GHJ97,HR00]
  - robust model-checking
    [Puri98,DDR04,DDMR04,ALM05,BMR06,BMR08]
    cf Pierre-Alain Reynier's talk tomorrow

Conclusions

- a probabilistic semantics for timed automata which removes "unlikely" (sequences of) events
- qualitative model-checking has a topological interpretation
- algorithm for qualitative LTL model-checking

Conclusions

- a probabilistic semantics for timed automata which removes "unlikely" (sequences of) events
- qualitative model-checking has a topological interpretation
- algorithm for qualitative LTL model-checking
- remark: extends to hybrid systems with finite bisimulation quotient

Conclusions

- a probabilistic semantics for timed automata which removes "unlikely" (sequences of) events
- qualitative model-checking has a topological interpretation
- algorithm for qualitative LTL model-checking
- remark: extends to hybrid systems with finite bisimulation quotient

Ongoing works

- quantitative analysis
- games

Conclusions

- a probabilistic semantics for timed automata which removes "unlikely" (sequences of) events
- qualitative model-checking has a topological interpretation
- algorithm for qualitative LTL model-checking
- remark: extends to hybrid systems with finite bisimulation quotient

Ongoing works

- quantitative analysis
- games

Further works

- efficient zone-based algorithm
- apply to relevant examples
- add non-determinism (*à la* MDP)
- handle several clocks
- timed properties