Laboratoire Méthodes Formelles

On the (Approximate) Analysis of Stochastic Real-Time Systems

Patricia Bouyer-Decitre

Based on joint works with Nathalie Bertrand, Thomas Brihaye and Pierre Carlier

Thanks to Nathalie Bertrand for her slides

Purpose of this work

- Study stochastic real-time systems, and more generally stochastic continuous-time (or space) processes
- ... with a model-checking approach

We want to design algorithms for verifying properties of (complex) stochastic real-time systems!

➡ Designed algorithms should give guarantees...

[BBBC18] N. Bertrand, P. Bouyer, Th. Brihaye, P. Carlier. When are stochastic transition systems tameable? (J. Log. Algebraic Methods Program, 2018)

Motivations

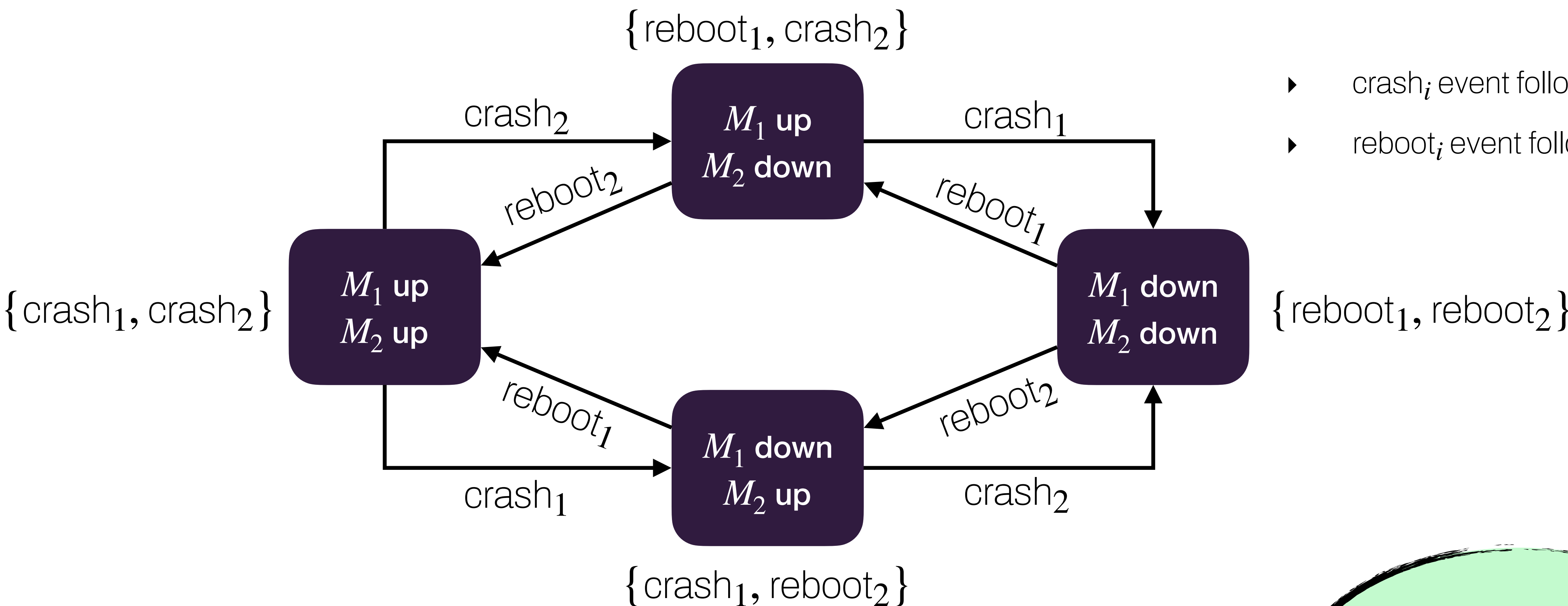
Needs for models with real-time and probabilities

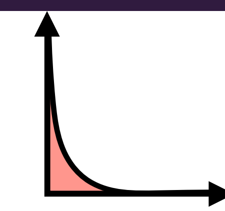
- Clock synchronization protocols
- Root contention protocols
- CSMA : random backoff retransmission time
- Molecular reactions
- ...

Numerous models in the literature

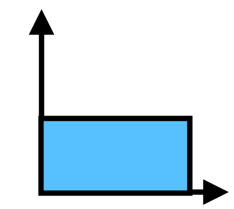
- Continuous-time Markov chains (CTMC)
- Generalized semi-Markov processes (GSMP)
- Stochastic timed automata (STA)
- Stochastic differential equations
- Continuous-space pure jump Markov processes
- ...

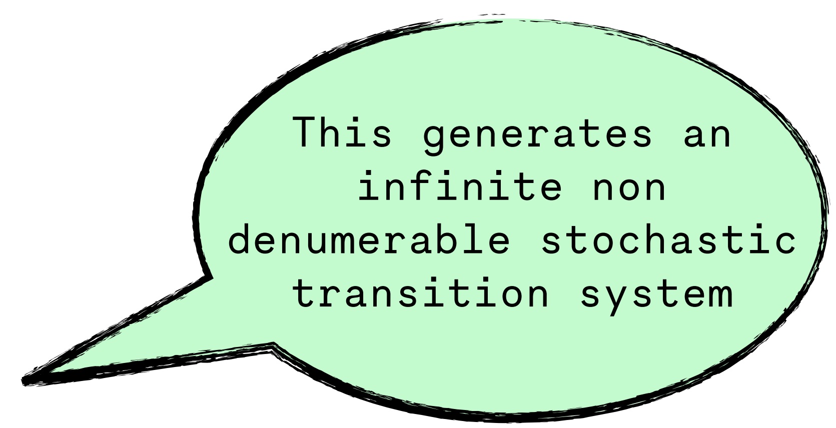
A first GSMP example of a two-machine network





- crash$_i$ event follows exp. distrib.
- reboot$_i$ event follows bounded unif. distrib.





- At state M_1 up M_2 down :
  - Events reboot$_1$ and crash$_2$ are sampled
  - A race condition applies to select the next state

Real-time stochastic systems

Challenges

- Intricate combination of dense time and probabilities
- Uncountable state-space
- Uncountable branching
- Continuous probability distributions

Objectives

- *Qualitative model-checking:* decide if a property holds almost-surely
- *Quantitative model-checking:* compute the probability that a property holds, or an approximation thereof

A focus on discrete-time Markov chains (DTMC)

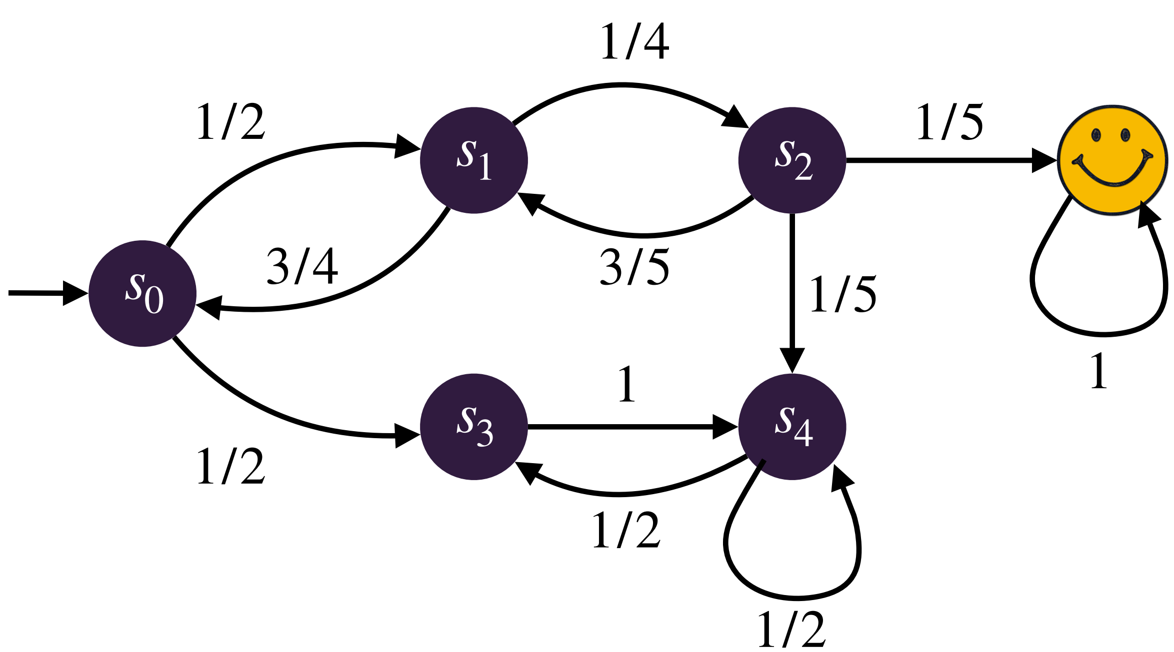
—

Decisiveness

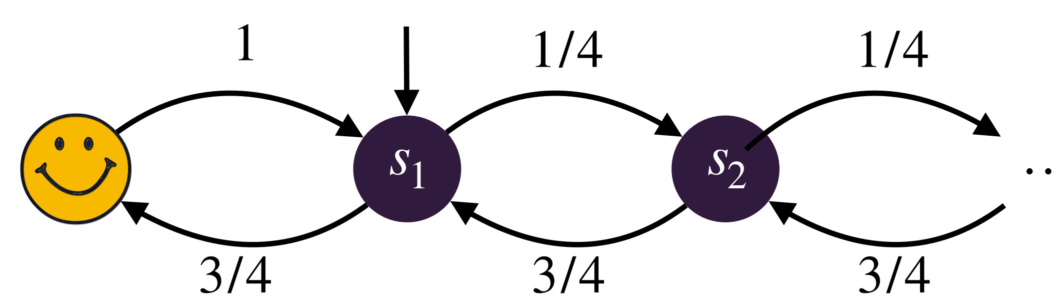
Discrete-time Markov chains

Discrete-time Markov chain (DTMC)

$$\mathcal{M} = (S, s_0, \delta) \text{ with } S \text{ denumerable, } s_0 \in S \text{ and } \delta : S \to \mathrm{Dist}(S)$$



Finite Markov chain



Denumerable Markov chain

Quantitative model-checking

- <u>Aim:</u> compute the probability of property $\mathbf{F}$ ☺
  [Note: very useful even for ω-regular properties, where analysis amounts to computing the probability of reaching good BSCCs]

- For state s, let x_s be such that:

$$x_s = \begin{cases} 1 & \text{if } s = \text{☺} \\ 0 & \text{if } s \not\models \exists\mathbf{F}\text{☺} \\ \sum_t \mathbb{P}(s \to t) \cdot x_t & \text{otherwise} \end{cases}$$

- The least fixpoint characterizes $\mathbb{P}_s(\mathbf{F}\text{☺})$

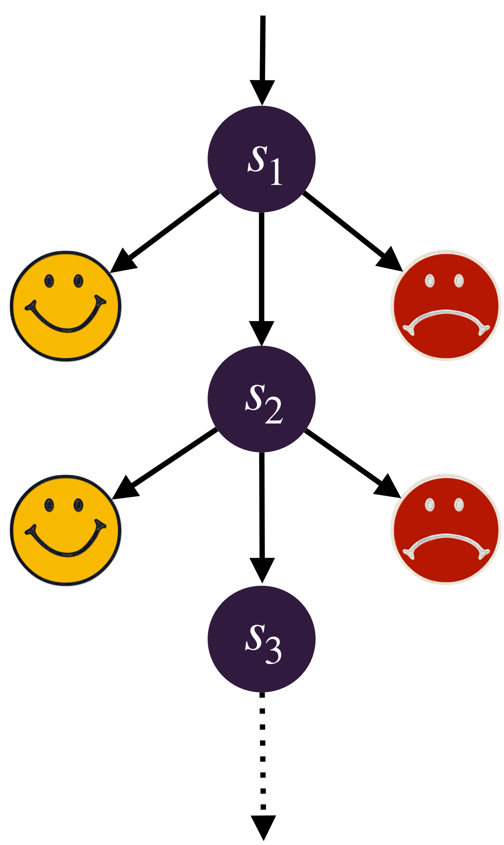
- For finite DTMCs, it amounts to solving a system of linear equations
  - For not-too-big DTMCs, this can be computed

- What can we do for infinite DTMCs?
  - Exact solutions do not exist in general
  - Ad-hoc approximate solutions are developed

$$\lim_{n\to\infty} \mathbb{P}_{\leq n}(\mathbf{F}\text{☺}) = \mathbb{P}(\mathbf{F}\text{☺})$$

DTMC: Approximate quantitative model-checking

- Aim: compute probability of $\mathbf{F}$ ☺
- ☹ $= \{ s \in S \mid s \not\models \exists \mathbf{F}$ ☺ $\}$



Approximation scheme

Given $\varepsilon > 0$, for every n, compute:

$$\begin{cases} p_n^{\text{yes}} & = & \mathbb{P}(\mathbf{F}_{\leq n} \text{☺}) \\ p_n^{\text{no}} & = & \mathbb{P}(\neg \text{☺}\, \mathbf{U}_{\leq n} \text{☹}) \end{cases}$$

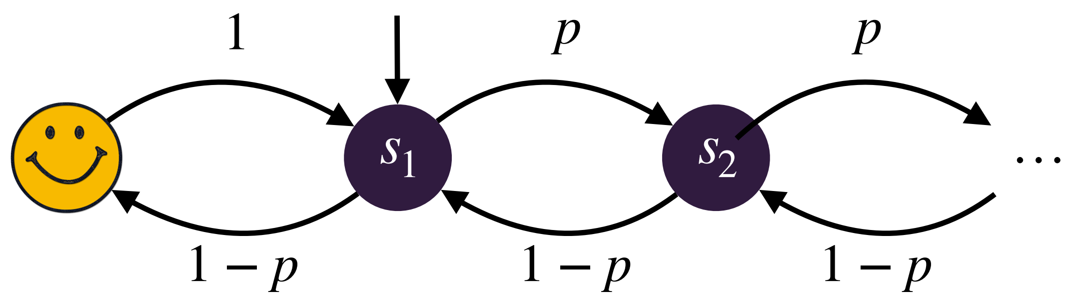
until $p_n^{\text{yes}} + p_n^{\text{no}} \geq 1 - \varepsilon$

$$\begin{array}{ccccc} p_1^{\text{yes}} & \leq \mathbb{P}(\mathbf{F} \text{☺}) \leq & 1 - p_1^{\text{no}} \\ \text{I}\wedge & & \vee\text{I} \\ p_2^{\text{yes}} & \leq \mathbb{P}(\mathbf{F} \text{☺}) \leq & 1 - p_2^{\text{no}} \\ \text{I}\wedge & \vdots & \vee\text{I} \end{array}$$

Does it converge?

[IN97] P. Iyer, M. Narasimha. Probabilistic lossy channel systems (TAPSOFT'97)

Non-converging example
The unbalanced random walk



$$\lim_{n\to\infty} \mathbb{P}_{\leq n}(\mathbf{F}\,☺) = \mathbb{P}(\mathbf{F}\,☺)$$

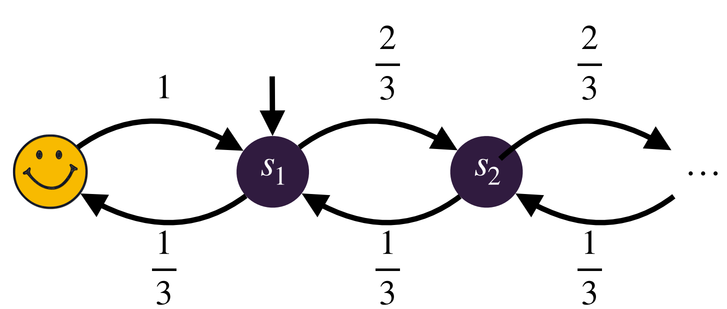
- ☹ $= \varnothing$, hence for all $n \in \mathbb{N}$, $p_n^{\text{no}} = \mathbb{P}(\mathbf{F}_{\leq n}\,☹) = 0$
- If $p > \dfrac{1}{2}$, then
  - $\mathbb{P}(\mathbf{F}\,☺) = 1 - \eta < 1$, hence for all $n \in \mathbb{N}$, $p_n^{\text{yes}} \leq 1 - \eta$
  - The sequences $(p_n^{\text{yes}})_n$ and $(1 - p_n^{\text{no}})_n$ are not adjacent
  - The approximation scheme does not converge

Decisiveness — 1

Decisiveness

A DTMC is *decisive* w.r.t. 🙂 if for all state s, $\mathbb{P}_s(\mathbf{F}\,🙂 \vee \mathbf{F}\,☹) = 1$

- Examples of decisive Markov chains: finite Markov chains, probabilistic lossy channel systems, probabilistic VASS, noisy Turing machines, ...
- Counterexample: unbalanced random walk



Not decisive w.r.t. 🙂
since $\mathbb{P}(\mathbf{F}\,🙂 \vee \mathbf{F}\,☹) < 1$

[ABM07] P.A. Abdulla, N. Ben Henda, R. Mayr. Decisive Markov chains (LMCS, 2007)

Decisiveness — 2

Approximation scheme

Given $\varepsilon > 0$:

$$\begin{cases} p_n^{\text{yes}} & = & \mathbb{P}(\mathbf{F}_{\leq n} \smiley) \\ p_n^{\text{no}} & = & \mathbb{P}(\neg \smiley \, \mathbf{U}_{\leq n} \frownie) \end{cases}$$

until $p_n^{\text{yes}} + p_n^{\text{no}} \geq 1 - \varepsilon$

If $\mathscr{M}$ is decisive w.r.t. $\smiley$ then the approximation scheme converges and is correct.

[ABM07] P.A. Abdulla, N. Ben Henda, R. Mayr. Decisive Markov chains (LMCS, 2007)

Beyond reachability
Repeated reachability

- Aim: compute probability of $\mathbf{GF}$ 🙂
- 😟 $= \{ s \in S \mid s \not\models \exists \mathbf{F}$ ☹️ $\}$

Approximation scheme

Given $\varepsilon > 0$ for every n, compute:

$$\begin{cases} q_n^{\text{yes}} = \mathbb{P}(\mathbf{F}_{\leq n} \text{😟}) \\ q_n^{\text{no}} = \mathbb{P}(\mathbf{F}_{\leq n} \text{☹️}) \end{cases}$$

until $q_n^{\text{yes}} + q_n^{\text{no}} \geq 1 - \varepsilon$

If $\mathscr{M}$ is decisive w.r.t. 🙂 and ☹️, then the approximation scheme converges and is correct.

$$\begin{array}{ccccc} q_1^{\text{yes}} & \leq & \mathbb{P}(\mathbf{GF}\,\text{🙂}) & \leq & 1 - q_1^{\text{no}} \\ \text{I}\wedge & & & & \vee\text{I} \\ q_2^{\text{yes}} & \leq & \mathbb{P}(\mathbf{GF}\,\text{🙂}) & \leq & 1 - q_2^{\text{no}} \\ \text{I}\wedge & & \vdots & & \vee\text{I} \end{array}$$

Does it converge?

[ABM07] P.A. Abdulla, N. Ben Henda, R. Mayr. Decisive Markov chains (LMCS, 2007)

Beyond reachability: ω-regular (Muller) properties

Attractor

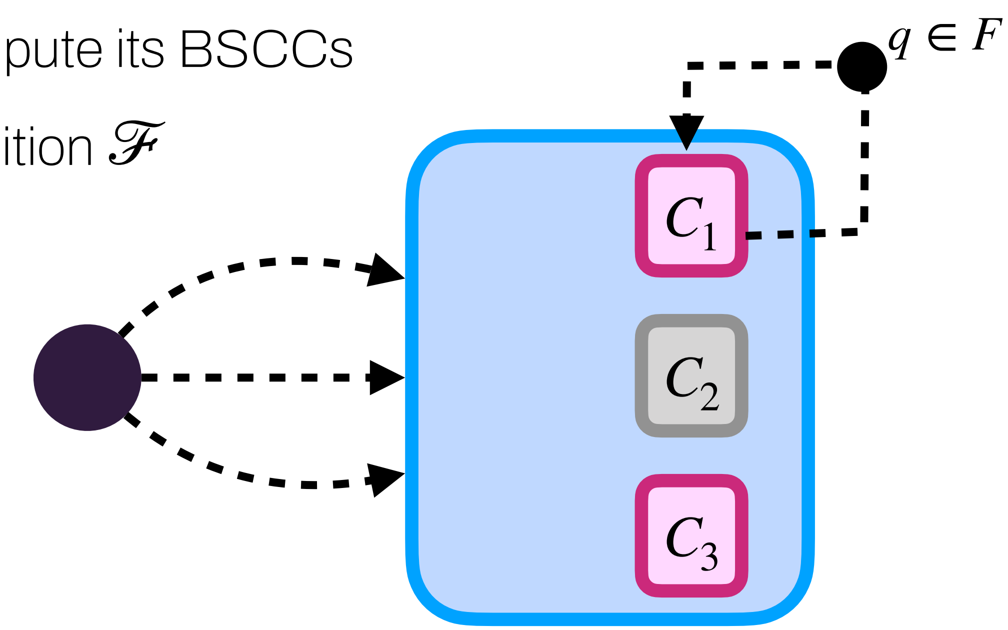
Attr is an attractor if for every state $s \in S$, $\mathbb{P}_s(\mathbf{F}\,\mathsf{Attr}) = 1$

$\mathscr{M}$ admits a finite attractor $\Longrightarrow$ $\mathscr{M}$ is decisive w.r.t. any goal

- From Attr build the graph $Graph(\mathsf{Attr})$ and compute its BSCCs
- Identify BSCCs that are good w.r.t. the Muller condition $\mathscr{F}$

  C is good if there is $F \in \mathscr{F}$ s.t.
  - For all q, $C \to^* q$ implies $q \in F$
  - For all $q \in F$, $C \to^* q$

- Then, $\mathbb{P}(\mathsf{Inf} \in \mathscr{F}) = \displaystyle\sum_{C \text{ good BSCC}} \mathbb{P}(\mathbf{F}C)$



Use approximation scheme to compute $\mathbb{P}(\mathbf{F}C)$

[ABM07] P.A. Abdulla, N. Bertrand, A. Rabinovich, Ph. Schnoebelen. Verification of Probabilistic Systems with Faulty Communication (Inf & Comp, 2005)
[BBBC18] N. Bertrand, P. Bouyer, Th. Brihaye, P. Carlier. When are stochastic transition systems tameable? (J. Log. Algebraic Methods Program, 2018)

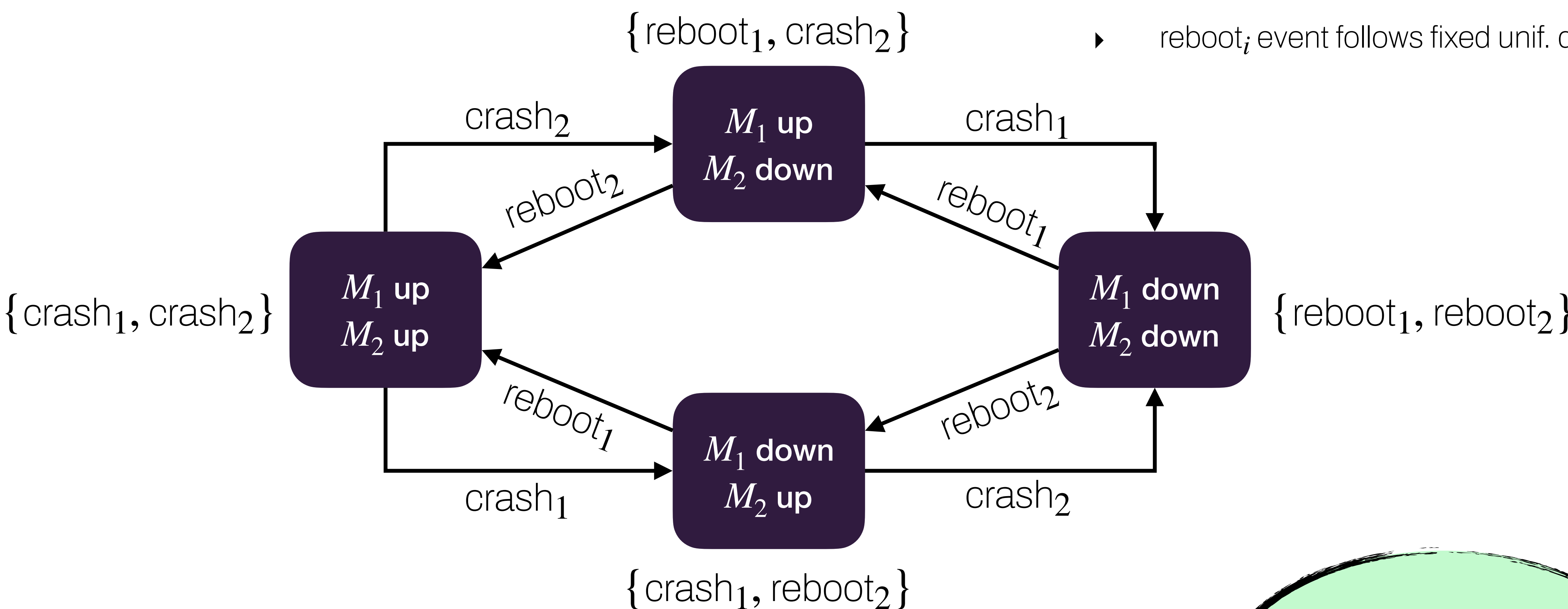
Real-time stochastic systems

—

Decisiveness and abstractions

A first GSMP example of a two-machine network

- crash_i event follows fixed exp. distrib. with param. λ_i
- reboot_i event follows fixed unif. distrib. over $[0,U_i]$



This generates an infinite non denumerable stochastic transition system

- At state M_1 up M_2 down :
  - Events reboot_1 and crash_2 are sampled
  - A race condition applies to select the next state

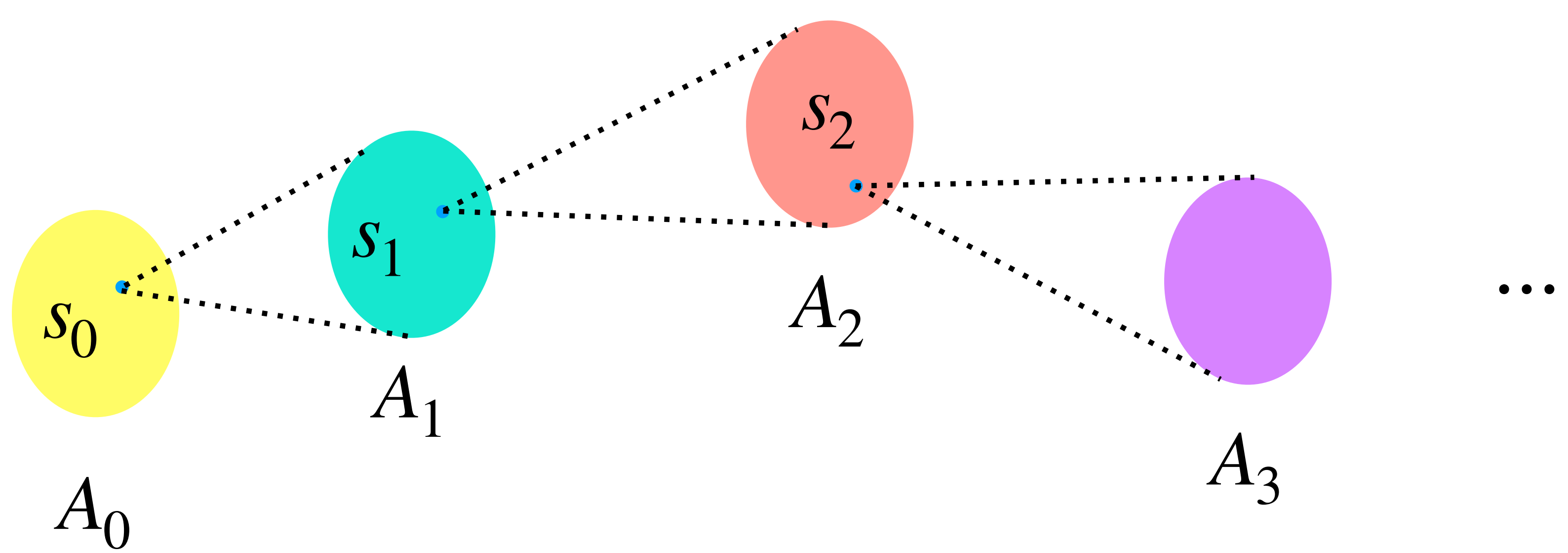
Stochastic transition systems (STS)

Stochastic transition systems (STS)

$\mathcal{T} = (S, \Sigma, \kappa)$ with (S, Σ) a measurable space and $\kappa : S \times \Sigma \to [0,1]$ a Markov kernel such that for all $s \in S$, $\kappa(s, \cdot\,) \in \mathsf{Dist}(S)$

- This defines a probability measure over infinite paths

$$\mathbb{P}_\mu(A_0, A_1, \ldots, A_n) = \int_{s_0 \in A_0} \int_{s_1 \in A_1} \cdots \int_{s_{n-1} \in A_{n-1}} \kappa(s_0, \mathrm{d}s_1)\kappa(s_1, \mathrm{d}s_2)\ldots\kappa(s_{n-2}, \mathrm{d}s_{n-1})\kappa(s_{n-1}, A_n)\ \mu(\mathrm{d}s_0)$$



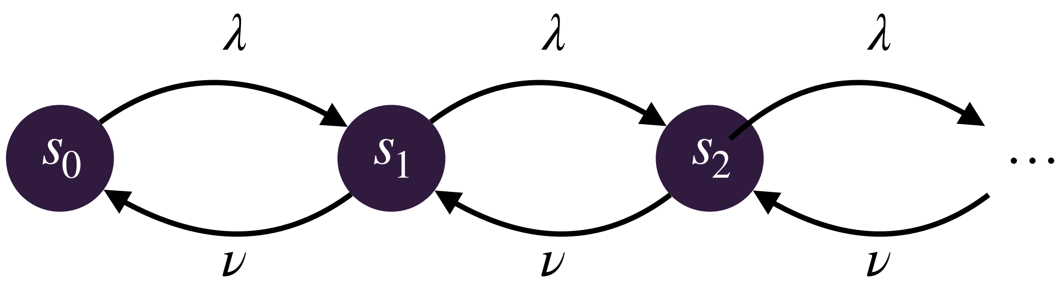
Some examples

$$\Sigma = 2^S$$
$$\kappa(s, \{s'\}) = p(s, s')$$

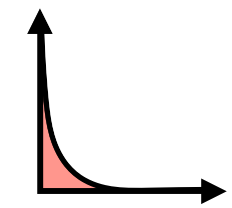
- Examples:
  - Countable Markov chains
  - Continuous-time Markov chains (CTMC)
  - Stochastic timed automata (STA)
  - Generalized semi-Markov processes (GSMP)
  - Stochastic Petri nets (SPN)
  - Etc...

Continuous-time Markov chains (CTMC)

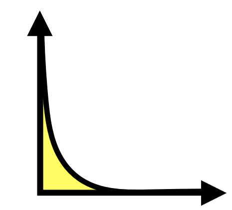
A simple queueing system:



- Arrival time parameter: λ
  (i.e. exponential distrib. with parameter λ)



- Serving time parameter: ν
  (i.e. exponential distrib. with parameter ν)

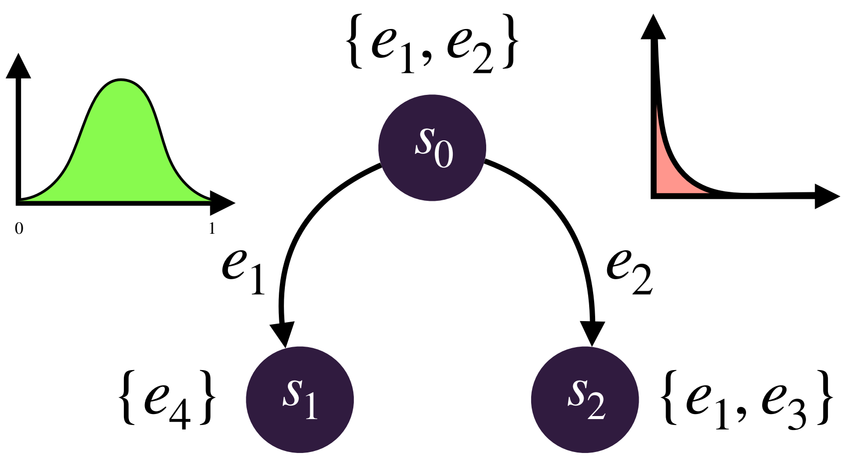


- Semantics from state $\gamma = (s, t)$ where t is the absolute time:
  - Apply a race condition to available events $e \in \mathbf{E}(s)$ (with an exp. distrib. with param. λ_e)

- Kernel at $\gamma = (s, t)$ for $B = \{s'\} \times [t + d_1, t + d_2]$:

$$\kappa(\gamma, B) = \frac{\lambda_e}{\sum_{e' \in \mathbf{E}(s)} \lambda_{e'}} \int_{d_1}^{d_2} \exp\Big(- \Big(\sum_{e' \in \mathbf{E}(s)} \lambda_{e'} \Big) \Big) \mathrm{d}\tau$$

Generalized semi-Markov Processes (GSMP)



Distributions on activated events:

- Bounded-support distrib. for e_1
- Exponential distrib. for e_2

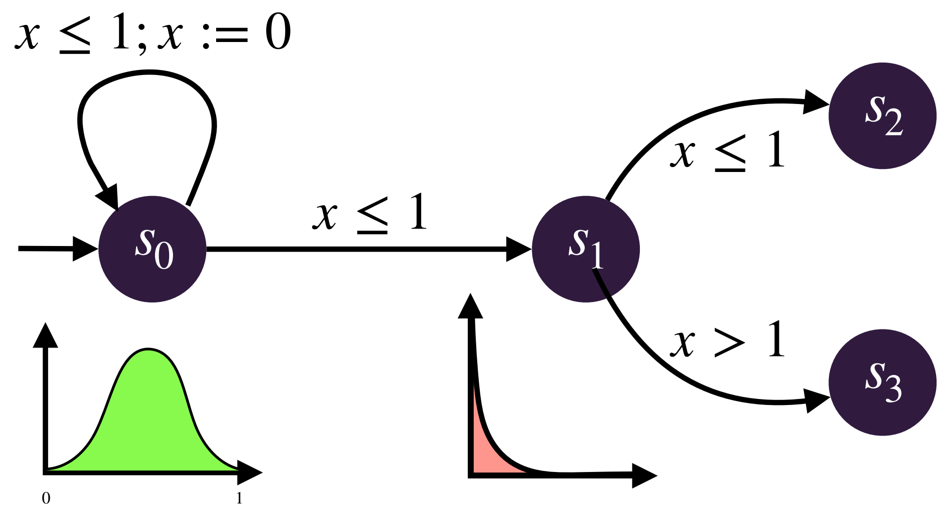
- Semantics from state $\gamma = (s, \nu)$ with $\nu(e) = \perp$ if $e \notin \mathbf{E}(s)$ and $\nu(e) \in \mathbb{R}_+$ otherwise (the remaining time before expiring):
  - Pick the event e_0 with the shortest expiring delay
  - Go to state s' s.t. $s \xrightarrow{e_0} s'$ and set $\gamma' = (s', \nu')$
  - Shift all remaining delays: $\nu'(e) = \nu(e) - \nu(e_0)$ if $e \in \big(\mathbf{E}(s') \cap \mathbf{E}(s)\big) \setminus \{e_0\}$ and sample newly activated events using their nominal distributions

- Kernel at $\gamma = (s, \nu)$ for $B = \{s'\} \times B'$:

$$\kappa(\gamma, B) = \delta(s, e_0)(s') \int_{(t_1, \ldots, t_p) \in B'} \left(\prod_{e \in \mathbf{E}(s')} g_e(t_e) \right) \, \mathrm{d}t_{e_1} \cdots \mathrm{d}t_{e_p}$$

Stochastic timed automata (STA)

- Stochastic timed automata = timed automata with random delays



Distributions on possible delays:

- Bounded-support distrib. in s_0
- Exponential distrib. in s_1

- Semantics from state $\gamma = (s, v)$:
  - Pick a delay d according to distribution μ in s at v
  - Choose at random an available edge

- Kernel at $\gamma = (s, v)$:

$$\kappa(\gamma, B) = \sum_{e=(s,g,Y,s')} \int_\tau \mathbb{1}_B((s', [Y](v+\tau))) \cdot p_{\gamma+\tau}(e) \; \mathrm{d}\mu(\tau)$$

[BBB+14] N. Bertrand, P. Bouyer, Th. Brihaye, Q. Menet, C. Baier, M. Größer, M. Jurdzinski. Stochastic Timed Automata (LMCS, 2014)

Decisiveness of STSs

- New ☹ needs to be defined
- ☹ $= \{s \in S \mid \mathbb{P}_s(\mathbf{F}\,\text{☺}) = 0\}$

Decisiveness of STSs

An STS $\mathscr{T}$ is decisive w.r.t. ☺ if for all distribution μ, $\mathbb{P}_\mu(\mathbf{F}\,\text{☺} \vee \mathbf{F}\,\text{☹}) = 1$

- How to perform approximate quantitative analysis of decisive STSs?

Analysis of decisive STSs

Approximation scheme for reach.

Given $\varepsilon > 0$:

$$\begin{cases} p_n^{\text{yes}} & = & \mathbb{P}(\mathbf{F}_{\leq n} \text{🙂}) \\ p_n^{\text{no}} & = & \mathbb{P}(\neg \text{🙂} \, \mathbf{U}_{\leq n} \text{☹}) \end{cases}$$

until $p_n^{\text{yes}} + p_n^{\text{no}} \geq 1 - \varepsilon$

If $\mathcal{T}$ is decisive w.r.t. 🙂 then the approximation scheme converges and is correct:

$(p_n^{\text{yes}})_n$ and $(1 - p_n^{\text{no}})_n$ both converge to $\mathbb{P}(\mathbf{F}\text{🙂})$

- Applicability: the approximation scheme is effective when
  - ☹ can be computed
  - One can evaluate the values p_n^{yes} and p_n^{no}, i.e. one can compute (or approximate) probabilities of cylinders of the form $\mathsf{Cyl}(SS\ldots S\text{🙂})$ and $\mathsf{Cyl}(\neg\text{🙂}\ldots\neg\text{🙂}\text{☹})$

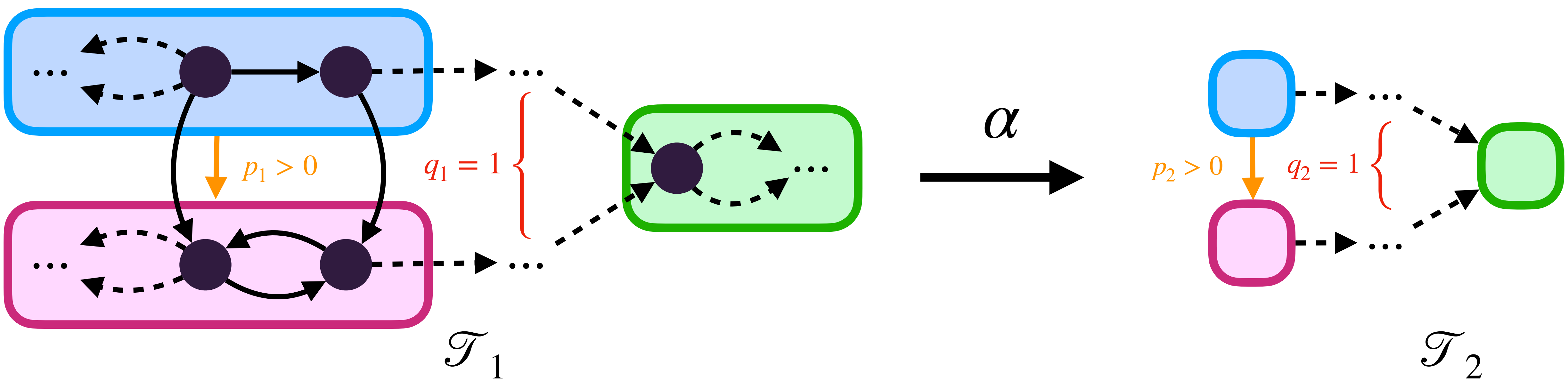
Other approximation schemes also apply

Is that all?

- Decisiveness is hard to check in general
- One needs:
  - To design methods to avoid proving directly decisiveness
  - And/or to identify subclasses of systems which are decisive
- Standard approach for real-time systems:
  - Use of abstractions?

Abstractions

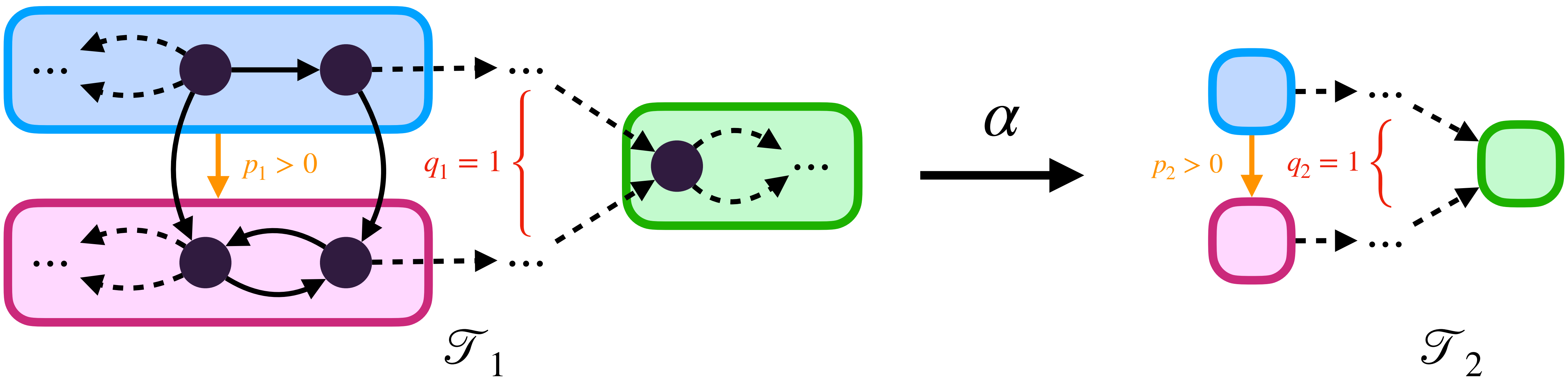
For two STSs $\mathscr{T}_1 = (S_1, \Sigma_1, \kappa_1)$ and $\mathscr{T}_2 = (S_2, \Sigma_2, \kappa_2)$, and $\alpha : (S_1, \Sigma_1) \to (S_2, \Sigma_2)$ a measurable function:



- $\mathscr{T}_2$ is an α-abstraction of $\mathscr{T}_1$ whenever $p_1 > 0$ is equivalent to $p_2 > 0$
- $\mathscr{T}_2$ is a sound α-abstraction of $\mathscr{T}_1$ whenever for each $B \in \Sigma_2$:

$$q_2 = \mathbb{P}^{\mathscr{T}_2}(\mathbf{F}B) = 1 \text{ implies } q_1 = \mathbb{P}^{\mathscr{T}_1}(\mathbf{F}\alpha^{-1}(B)) = 1$$

Abstractions, decisiveness and attractors



If $\mathcal{T}_2$ is a sound α-abstraction of $\mathcal{T}_1$, then:

- $\mathcal{T}_2$ decisive w.r.t. 😊 implies $\mathcal{T}_1$ decisive w.r.t. α^{-1}(😊)
- Attr attractor for $\mathcal{T}_2$ implies $\alpha^{-1}(\text{Attr})$ attractor for $\mathcal{T}_1$

Example of application of the approach

- Setting:
  - $\mathcal{T}_1$ general STS
  - $\mathcal{T}_2$ countable Markov chain with a finite attractor
  - $\mathcal{T}_2$ sound α-abstraction of $\mathcal{T}_1$

- How to model-check Muller properties?
  - Almost-sure model checking of a Muller property in $\mathcal{T}_1$ reduces to almost-sure model checking of a reachability property in $\mathcal{T}_2$
  - Computation of the probability of Muller properties in $\mathcal{T}_1$ reduces to computation of a reachability probability in $\mathcal{T}_1$

$$\mathbb{P}_{\mathcal{T}_1}(\mathsf{Inf} \in \mathcal{F}) = \sum_{C \text{ good BSCC in } \mathcal{T}_2} \mathbb{P}_{\mathcal{T}_1}(\mathbf{F}\alpha^{-1}(C))$$

Specific results for real-time stochastic systems

- The state-space includes a time component: $\widehat{S} = S \times \mathbb{R}_+$
- Time elapses almost-surely: $\kappa((s,t), \{(s',t') \in \widehat{S} \mid t' > t\}) = 1$

- If $\mathcal{T}$ is almost-surely non-Zeno, then $A_\Delta = \{(s,t) \in \widehat{S} \mid t > \Delta\}$ is an attractor.
- $\mathcal{T}$ is decisive w.r.t. time-bounded sets.

- One gets immediately approximation schemes for time-bounded properties like $B_1 \ \mathbf{U}_I \ B_2$ where I is a bounded interval.

Laboratoire Méthodes Formelles

université PARIS-SACLAY

cnrs

école normale supérieure paris—saclay

Applications

—

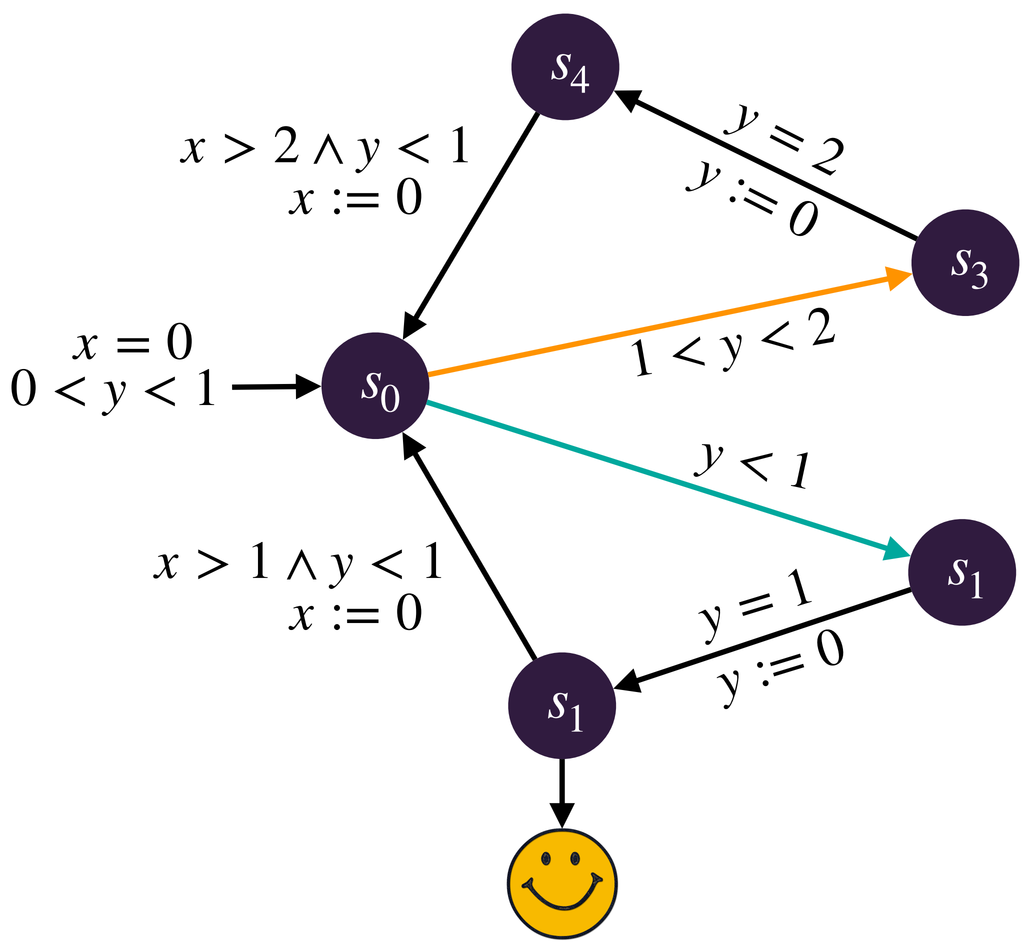
Application to Stochastic Timed Automata

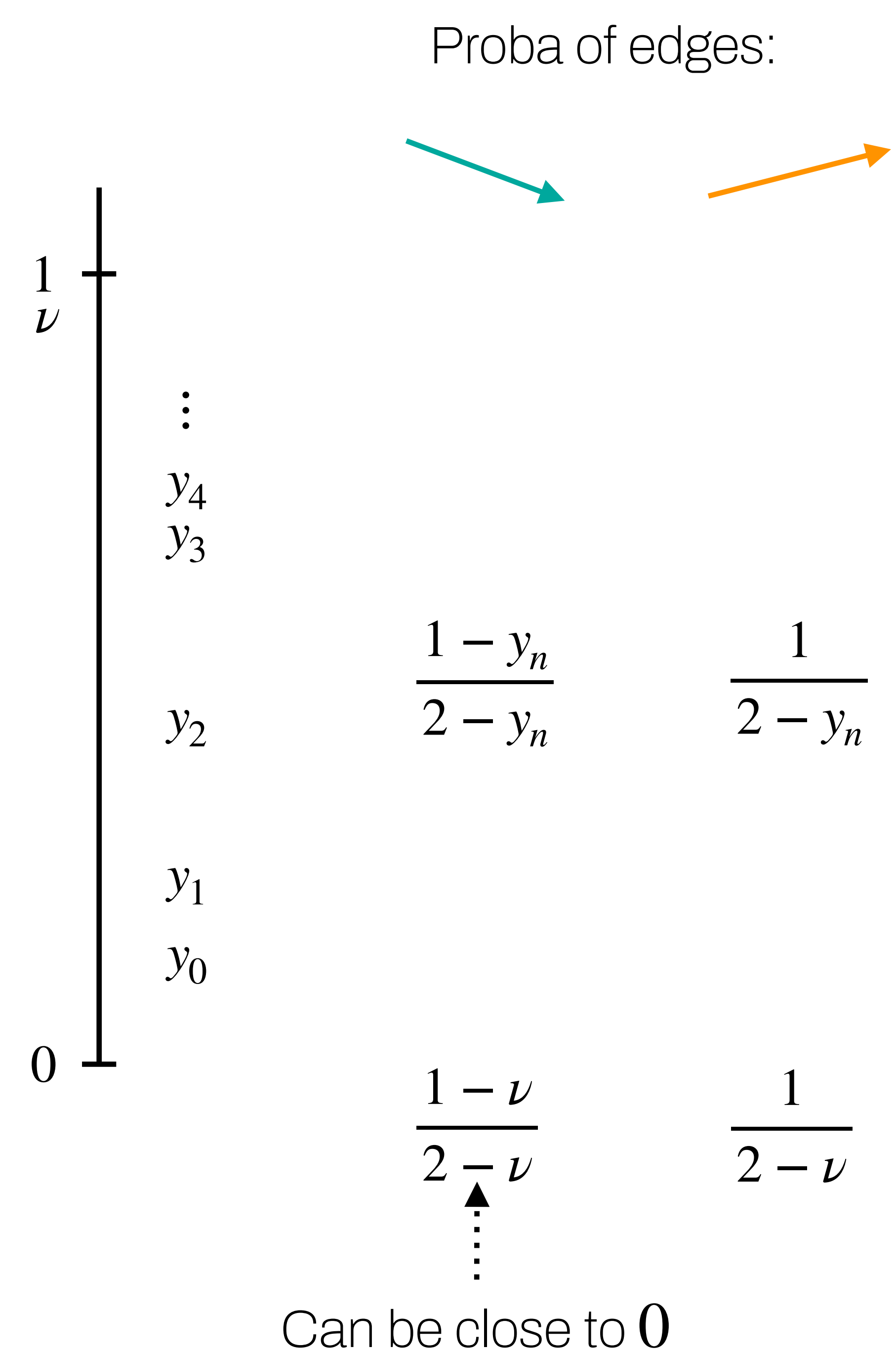
- Natural abstraction:
  - Markov chain built on region automaton

- STA with an attractor, hence decisive
  - Single-clock STA:
    $\mathsf{Attr} = \{(\ell,0)\} \cup \{(\ell,r) \mid r = (M,+\infty)\}$
  - Reactive STA, i.e. complete w.r.t. delays
    $\mathsf{Attr} = \{(\ell,r) \mid \forall x,\ x = 0 \text{ or } x > M \text{ in } r\}$

- Model-checking STA
  - We recover all known decidability/approximability results...
  - ... and extend them, e.g. for Muller properties

STA — A counterexample





One can show that this STA is not decisive, and standard region automaton does not correctly evaluate the probability of reaching 🙂

Application to Generalized Semi-Markov Processes

- We consider GSMP **with no** fixed-delay events
- Natural abstraction:
  - Markov chain built on a refined region abstraction
- An attractor based on these refined regions exist
  - The abstraction is sound!
  - Hence GSMP with no fixed-delay events are decisive!
- Model-checking GSMP:
  - Decidability of qualitative analysis for rich properties
  - Approximate analysis for rich properties as well
- Warning: with fixed-delay events, this is no more the case!
  This was pinpointed in [BKKR11]

[BKKR11] T. Brázdil, J. Krcál, J. Kretínský, V. Rehák. Fixed-Delay Events in Generalized Semi-Markov Processes Revisited (CONCUR'11)

GSMP — Counterexample

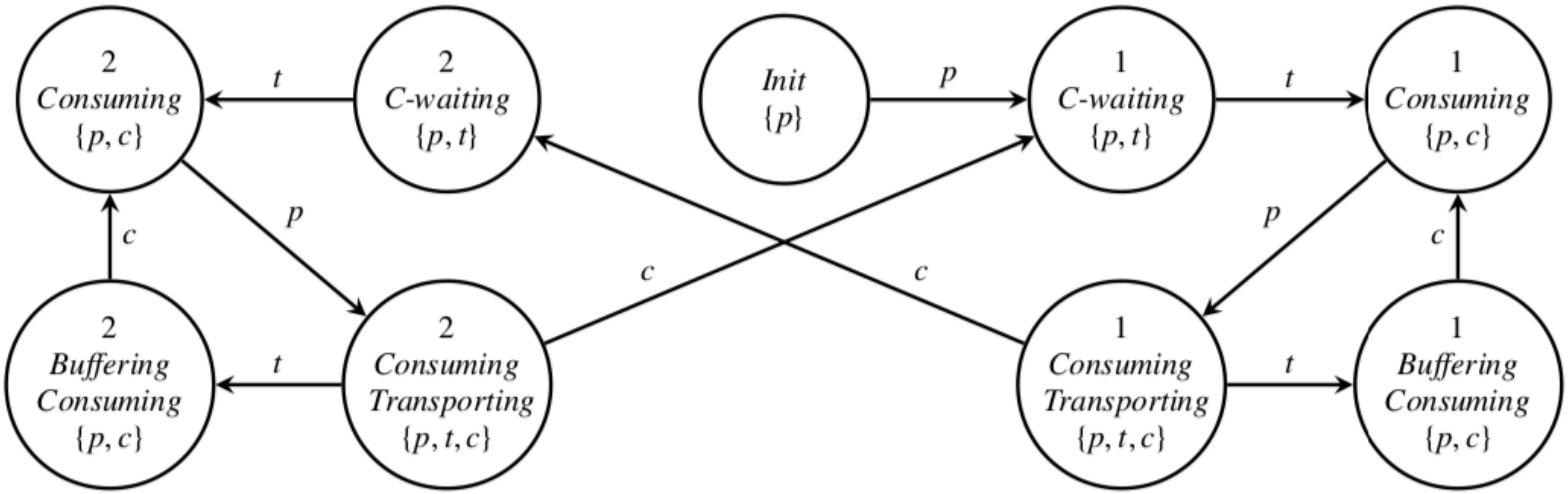


Fig. 2. A GSMP of a producer-consumer system. The events p, t, and c model that a packet production, transport, and consumption is finished, respectively. Below each state label, there is the set of scheduled events. The fixed-delay events p and c have $l_p = u_p = l_c = u_c = 1$ and the uniformly distributed variable-delay event t has $l_t = 0$ and $u_t = 1$.

[BKKR11] T. Brázdil, J. Krcál, J. Kretínský, V. Rehák. Fixed-Delay Events in Generalized Semi-Markov Processes Revisited (CONCUR'11)

Stochastic Petri nets

- Petri nets in which stochastic delays are attached to transitions [ACB84]

- Restricted setting to fit our framework:
  - Bounded Petri net
  - Markov regenerative: regeneration points are encountered infinitely often almost-surely [HPRV12,PHV16]

The remaining durations of enabled transitions are all memoryless

- Regeneration points form a finite attractor
- Abstraction: standard state-class graph
- Regenerative Petri nets are decisive!

- Approximate analysis can be done, provided numerical computations are amenable
- We recover the classes that were analyzed (though the authors had a focus on efficient computations)

[ACB84] M. Ajmone Marsan, G. Conte, G. Balbo, A class of generalized stochastic Petri nets for the performance evaluation of multiprocessor systems (ACM Trans. Comput. Syst. 1984)
[HPRV12] A. Horváth, M. Paolieri, L. Ridi, E. Vicario, Transient analysis of non-Markovian models using stochastic state classes (Perform. Eval. 2012)
[PHV16] M. Paolieri, A. Horváth, E. Vicario, Probabilistic model checking of regenerative concurrent systems (IEEE Trans. Softw. 2016)

Conclusion

—

Thoughts on SMC

What we did

- A generic approach to approximate analysis of stochastic processes with possibly continuous state-space, based on finite-horizon computations
  - With hypotheses (existence of an attractor, decisiveness, ...) and guarantees!
- It requires numerical computability properties to be effective (that we did not consider here)
- It applies to many classes of real-time stochastic systems
  - Classes of STA
  - Classes of GSMPs
  - Regenerative Petri nets
  - ...
- The decisiveness property is in the core of the approach
  - Tools like attractors and abstractions are very helpful to ensure decisiveness

[BBBC18] N. Bertrand, P. Bouyer, Th. Brihaye, P. Carlier. When are stochastic transition systems tameable? (J. Log. Algebraic Methods Program, 2018)

Going further: statistical model-checking

- Monte-Carlo simulation:
  - Sample a large number of realizations of a random variable X, and compute the mean
  - This is an estimator of $\mathbb{E}(X)$, with guarantees given as confidence intervals
- In our case:
  - A realization = an (infinite) execution
  - X evaluates a property ϕ over executions

- Everything works fine with time-bounded properties [YS06]
  - Finite executions are sufficient
- Time-unbounded properties require some attention [YCZ11]
  - Compute prior to simulations
  - The executions will almost-surely be finite (and end in or in)
  - This is applicable to finite Markov chains only

The only
required assumption
is a decisiveness
property!

[YS06] H.L.S. Younes and R.G. Simmons, Statistical probabilistic model checking with a focus on time-bound… 2006)
[YCZ11] H.L.S. Younes, E.M. Clarke and P. Zuliani, Statistical Verification of Probabilistic Properties with Unbounded … 11)