Tutorial:
Probabilistic Model Checking

Christel Baier
Technische Universität Dresden

Probability elsewhere

Probability elsewhere

- randomized algorithms [Rabin 1960]
  symmetry breaking, fingerprint techniques,
  random choice of waiting times or IP addresses, ...
- stochastic control theory [Bellman 1957]
  operations research
- performance modeling [Markov, Erlang, Kolm., $\sim$ 1900]
- biological systems
- resilient systems

  ⋮

Probability elsewhere

- randomized algorithms [Rabin 1960]

  symmetry breaking, fingerprint techniques,
  random choice of waiting times or IP addresses, ...

- stochastic control theory [Bellman 1957]

  operations research

- performance modeling [Markov, Erlang, Kolm., $\sim$ 1900]

- biological systems

- resilient systems

discrete or continuous-time Markovian models

memoryless property: future system behavior depends only on the current state, but not on the past

Probabilistic models

Probabilistic models

	purely probabilistic	probabilistic and nondeterministic
discrete time		
continuous time		

Probabilistic models

	purely probabilistic	probabilistic and nondeterministic
discrete time	discrete-time Markov chain (DTMC)	Markov decision process (MDP)
continuous time		

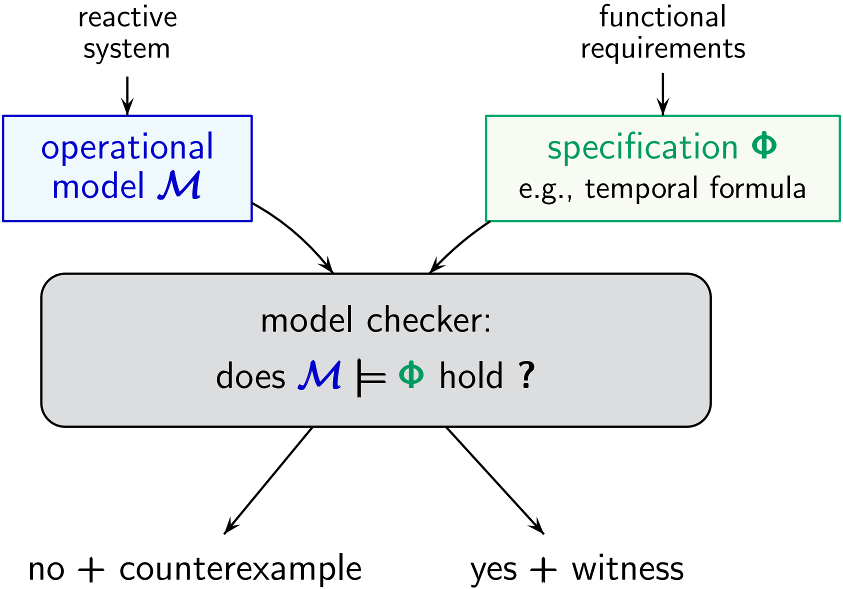
Probabilistic models

	purely probabilistic	probabilistic and nondeterministic
discrete time	discrete-time Markov chain (DTMC)	Markov decision process (MDP)
continuous time	continuous-time Markov chain (CTMC)	continuous-time MDP interactive Markov chains probabilistic timed automata stochastic automata ⋮

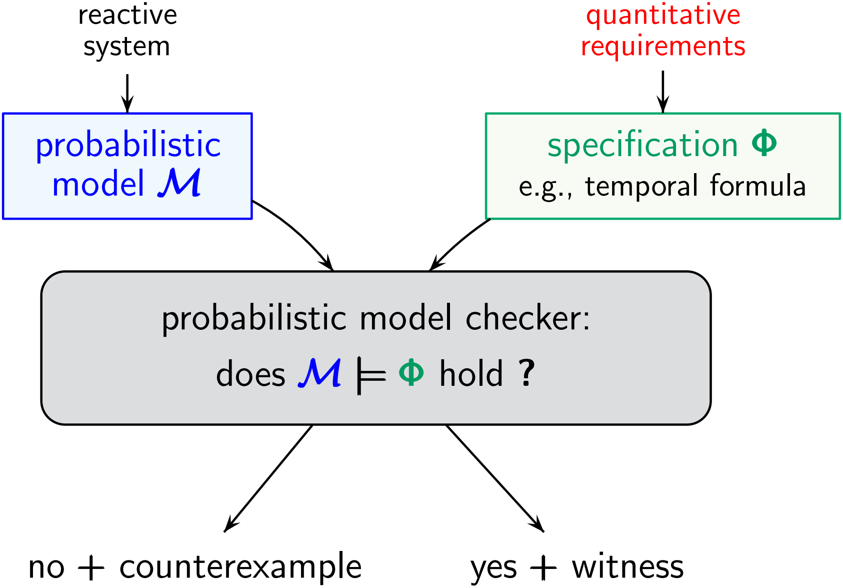
Probabilistic models

	purely probabilistic	probabilistic and nondeterministic
discrete time	discrete-time Markov chain (DTMC)	Markov decision process (MDP)
continuous time	continuous-time Markov chain (CTMC)	continuous-time MDP interactive Markov chains probabilistic timed automata stochastic automata ⋮

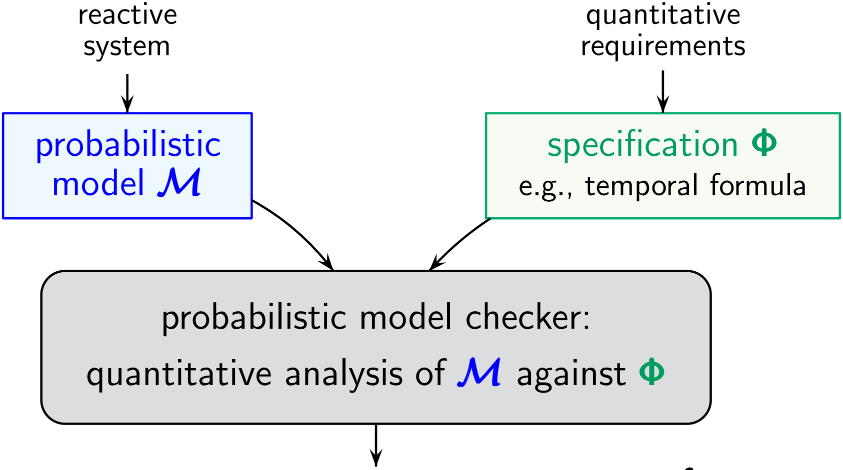
Model checking



Probabilistic model checking



Probabilistic model checking



probability for "bad behaviors" is $< \mathbf{10^{-6}}$
probability for "good behaviors" is $\mathbf{1}$
expected costs for ....

Probabilistic model checking

- termination of probabilistic programs [Hart/Sharir/Pnueli'83]
- qualitative linear time properties [Vardi/Wolper'86]
  for discrete-time Markov models [Courcoubetis/Yannak.'88]

Probabilistic model checking

- termination of probabilistic programs
  [Hart/Sharir/Pnueli'83]
- qualitative linear time properties [Vardi/Wolper'86]
  for discrete-time Markov models [Courcoubetis/Yannak.'88]
- probabilistic computation tree logic [Hansson/Jonsson'94]
  for discrete-time Markov models [Bianco/de Alfaro'95]
- continuous stochastic logic [Aziz et al'96]
  for continuous-time Markov chains [Baier et al'99]
- probabilistic timed automata [Jensen'96]
  [Kwiatkowska et al'00]

  ⋮

tools: PRISM, MRMC, STORM, IscasMC, PASS,
ProbDiVinE, MARCIE, YMER, ...

Discrete-time Markov chains (DTMC)

* basic definitions
* probabilistic computation tree logic PCTL/PCTL*
* rewards, cost-utility ratios, weights
* conditional probabilities

Markov decision processes (MDP)

* basic definitions
* PCTL/PCTL* model checking
* fairness
* conditional probabilities
* rewards, quantiles
* mean-payoff
* expected accumulated weights

Tutorial: Probabilistic Model Checking

Discrete-time Markov chains (DTMC)

* basic definitions
* probabilistic computation tree logic PCTL/PCTL*
* rewards, cost-utility ratios, weights
* conditional probabilities

Markov decision processes (MDP)

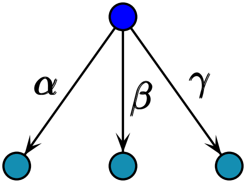
* basic definitions
* PCTL/PCTL* model checking
* fairness
* conditional probabilities
* rewards, quantiles
* mean-payoff
* expected accumulated weights

Markov chains

... transition systems with probabilistic distributions
for the successor states

Markov chains

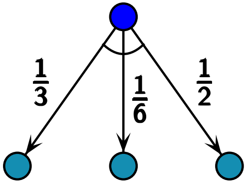
... transition systems with probabilistic distributions for the successor states



transition system

nondeterministic branching

choice between action-labeled transitions



Markov chain

probabilistic branching

discrete-time

Discrete-time Markov chain (DTMC)

$$\mathcal{M} = (S, P, \ldots)$$

Discrete-time Markov chain (DTMC)

$$\mathcal{M} = (S, P, \ldots)$$

- countable state space S

Discrete-time Markov chain (DTMC)

$$\mathcal{M} = (S, P, \ldots)$$

- countable state space S $\longleftarrow$ here: finite

Discrete-time Markov chain (DTMC)

$$\mathcal{M} = (S, P, \ldots)$$

- countable state space S $\longleftarrow$ here: finite
- transition probability function $P : S \times S \to [0, 1]$

$$\text{s.t.} \quad \sum_{s' \in S} P(s, s') = 1$$

Discrete-time Markov chain (DTMC)

$$\mathcal{M} = (S, P, \ldots)$$

- countable state space S $\longleftarrow$ here: finite
- transition probability function $P : S \times S \to [0, 1]$

$$\text{s.t.} \quad \sum_{s' \in S} P(s, s') = 1$$

$\uparrow$

discrete-time or time-abstract:

probability for the step $s \longrightarrow s'$

Discrete-time Markov chain (DTMC)

$\mathcal{M} = (S, P, AP, L, \ldots)$

- countable state space S $\longleftarrow$ here: finite
- transition probability function $P : S \times S \to [0, 1]$

  s.t. $\sum_{s' \in S} P(s, s') = 1$
- AP set of atomic propositions
- labeling function $L : S \to 2^{AP}$

Discrete-time Markov chain (DTMC)

$$\mathcal{M} = (S, P, AP, L, \ldots)$$

- countable state space S $\longleftarrow$ here: finite
- transition probability function $P : S \times S \to [0,1]$

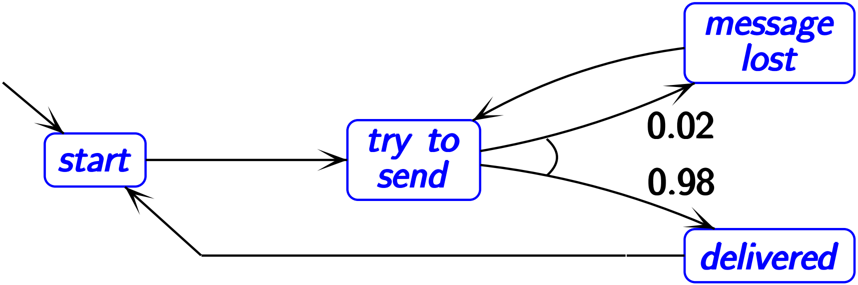
  s.t. $\sum_{s' \in S} P(s, s') = 1$
- AP set of atomic propositions
- labeling function $L : S \to 2^{AP}$
- $\mu : S \to [0,1]$ initial distribution
- $wgt : S \to \mathbb{Z}$ where $wgt(s)$ is the reward (or weight) earned per visit of state s

Example: DTMC for communication protocol

$$\mathcal{M} = (S, P, AP, L, \ldots)$$

- countable state space S $\longleftarrow$ here: finite
- transition probability function $P : S \times S \to [0, 1]$

  s.t. $\sum_{s' \in S} P(s, s') = 1$

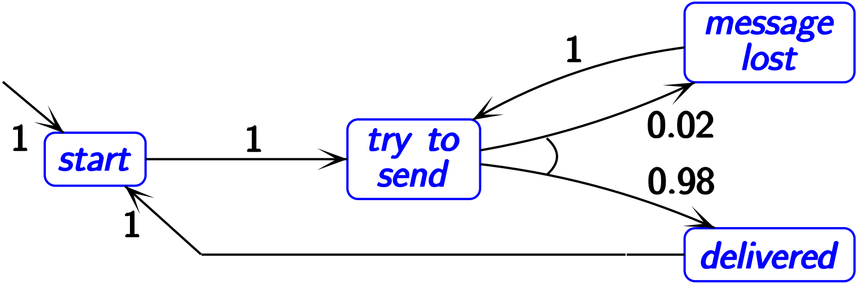


Example: DTMC for communication protocol

$$\mathcal{M} = (S, P, AP, L, \ldots)$$

- countable state space S $\longleftarrow$ here: finite
- transition probability function $P : S \times S \to [0,1]$

  s.t. $\sum_{s' \in S} P(s, s') = 1$



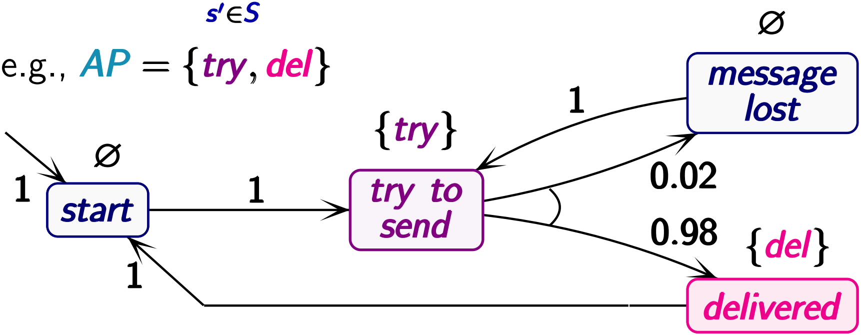
Example: DTMC for communication protocol

$$\mathcal{M} = (S, P, AP, L, \ldots)$$

- countable state space S $\longleftarrow$ here: finite
- transition probability function $P : S \times S \to [0, 1]$

  s.t. $\sum_{s' \in S} P(s, s') = 1$

e.g., $AP = \{try, del\}$



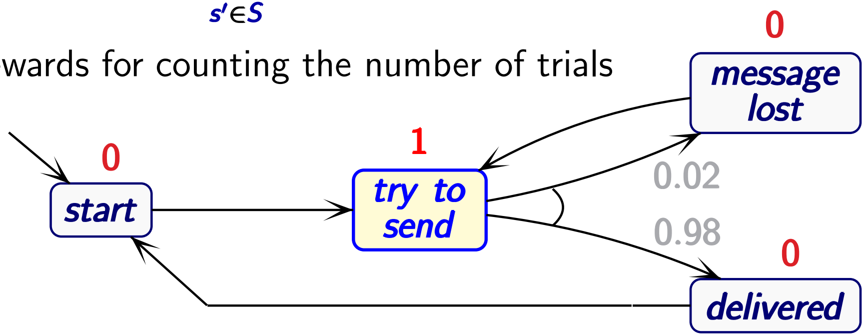
Example: DTMC for communication protocol

$$\mathcal{M} = (S, P, AP, L, \ldots)$$

- countable state space S
- transition probability function $P : S \times S \to [0,1]$

$$\text{s.t.} \quad \sum_{s' \in S} P(s, s') = 1$$

rewards for counting the number of trials



Probability measure of a Markov chain

Probability measure of a Markov chain

$$\mathcal{M} = (S, P, AP, L, \mu) \quad \text{where} \quad \mu : S \to [0,1]$$

↑ initial distribution

Probability measure of a Markov chain

$\mathcal{M} = (S, P, AP, L, \mu)$ where $\mu : S \to [0,1]$

↑ initial distribution

probability measure for measurable sets of paths:

Probability measure of a Markov chain

$\mathcal{M} = (S, P, AP, L, \mu)$ where $\mu : S \to [0, 1]$

↑ initial distribution

probability measure for measurable sets of paths:

consider the σ-algebra generated by cylinder sets

$\Delta(s_0\, s_1 \ldots s_n)$ = set of infinite paths $s_0\, s_1 \ldots s_n\, s_{n+1}\, s_{n+2}\, s_{n+3} \ldots$

↑ finite path

Probability measure of a Markov chain

$$\mathcal{M} = (S, P, AP, L, \mu) \quad \text{where} \quad \mu : S \to [0, 1]$$

↑ initial distribution

probability measure for measurable sets of paths:

consider the σ-algebra generated by cylinder sets

$$\Delta(s_0\, s_1 \ldots s_n) \;=\; \text{set of infinite paths} \ldots$$

σ-algebra on universe $\mathcal{U}$: set $\mathcal{V} \subseteq 2^{\mathcal{U}}$ s.t.

1. $\mathcal{U} \in \mathcal{V}$
2. if $T \in \mathcal{V}$ then $\mathcal{U} \setminus T \in \mathcal{V}$
3. if $T_i \in \mathcal{V}$ for $i \in \mathbb{N}$ then $\bigcup_{i \in \mathbb{N}} T_i \in \mathcal{V}$

Probability measure of a Markov chain

$\mathcal{M} = (S, P, AP, L, \mu)$ where $\mu : S \to [0, 1]$

↑ initial distribution

probability measure for measurable sets of paths:

consider the σ-algebra generated by cylinder sets

$$\Delta(s_0\, s_1 \ldots s_n) = \text{set of infinite paths} \ldots$$

here: $\mathcal{U}$ = set of infinite paths $\subseteq S^\omega$

$\mathcal{V}$ = smallest subset of $2^{\mathcal{U}}$ that contains all cylinder sets and is closed under complement and countable unions

Probability measure of a Markov chain

$$\mathcal{M} = (S, P, AP, L, \mu) \quad \text{where} \quad \mu : S \to [0,1]$$

initial distribution

probability measure for measurable sets of paths:

consider the σ-algebra generated by cylinder sets

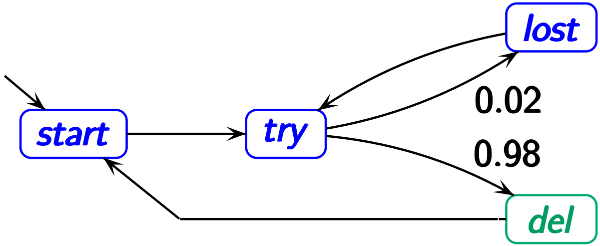
$$\Delta(s_0\, s_1 \ldots s_n) \;=\; \text{set of infinite paths}$$

$$s_0\, s_1 \ldots s_n\, s_{n+1}\, s_{n+2}\, s_{n+3} \ldots$$

probability measure is given by:

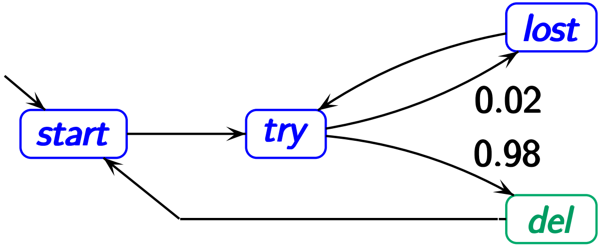
$$\Pr^{\mathcal{M}}\big(\, \Delta(s_0\, s_1 \ldots s_n)\,\big) \;=\; \mu(s_0) \cdot \prod_{1 \leqslant i \leqslant n} P(s_{i-1}, s_i)$$

Example: Markov chain



probability for delivering the message within **5** steps:

Example: Markov chain

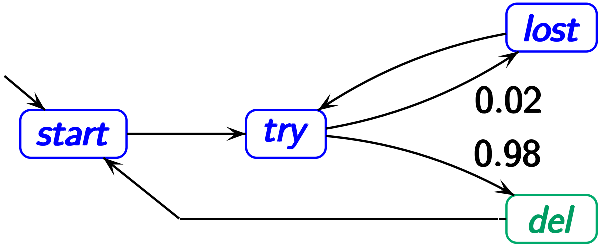


probability for delivering the message within **5** steps:

$$= \Pr^{\mathcal{M}}(\textit{start try del}) + \Pr^{\mathcal{M}}(\textit{start try lost try del})$$

notation: $\Pr^{\mathcal{M}}(s_0\, s_1 \ldots s_n) = \Pr^{\mathcal{M}}(\Delta(s_0\, s_1 \ldots s_n))$

Example: Markov chain



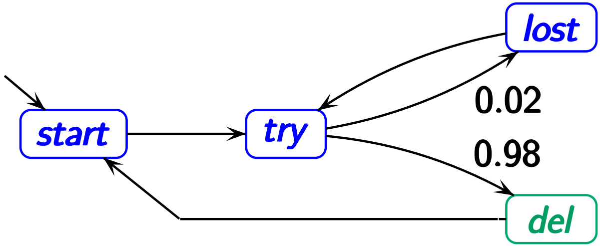
probability for delivering the message within **5** steps:

$$= \Pr^{\mathcal{M}}(\textit{start try del}) + \Pr^{\mathcal{M}}(\textit{start try lost try del})$$

$$= 0.98 + 0.02 \cdot 0.98 = 0.9996$$

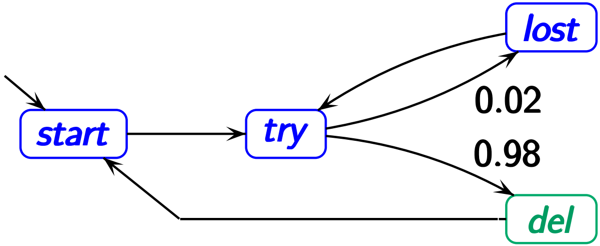
notation: $\Pr^{\mathcal{M}}(s_0\, s_1 \ldots s_n) = \Pr^{\mathcal{M}}(\Delta(s_0\, s_1 \ldots s_n))$

Example: Markov chain



probability for eventually delivering the message:

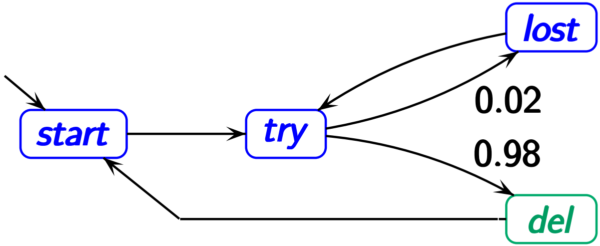
Example: Markov chain



probability for eventually delivering the message:

$$= \sum_{n=0}^{\infty} \Pr^{\mathcal{M}}(\textit{start}\ \textit{try}\ (\textit{lost}\ \textit{try})^n\ \textit{del})$$

Example: Markov chain



probability for eventually delivering the message:

$$= \sum_{n=0}^{\infty} \Pr^{\mathcal{M}}(\textit{start try } (\textit{lost try})^n \textit{ del})$$

$$= \sum_{n=0}^{\infty} 0.02^n \cdot 0.98 = 1$$

Measurability of classical properties

Measurability of classical properties

A σ-algebra is a pair $(\mathcal{U}, \mathcal{V})$ where $\mathcal{U}$ is a set and $\mathcal{V} \subseteq 2^{\mathcal{U}}$ such that:

1. $\mathcal{U} \in \mathcal{V}$
2. if $T \in \mathcal{V}$ then $\mathcal{U} \setminus T \in \mathcal{V}$
3. if $T_i \in \mathcal{V}$ for $i \in \mathbb{N}$ then $\bigcup_{i \in \mathbb{N}} T_i \in \mathcal{V}$

The elements of $\mathcal{V}$ are called events.

Measurability of classical properties

A σ-algebra is a pair $(\mathcal{U}, \mathcal{V})$ where $\mathcal{U}$ is a set and $\mathcal{V} \subseteq 2^{\mathcal{U}}$ such that:

1. $\mathcal{U} \in \mathcal{V}$
2. if $T \in \mathcal{V}$ then $\mathcal{U} \setminus T \in \mathcal{V}$
3. if $T_i \in \mathcal{V}$ for $i \in \mathbb{N}$ then $\bigcup_{i \in \mathbb{N}} T_i \in \mathcal{V}$

The elements of $\mathcal{V}$ are called events.

DTMCs: $\mathcal{U}$ = set of infinite paths

$$\mathcal{V} = \left\{ \begin{array}{l} \sigma\text{-algebra generated by the} \\ \text{cylinder sets} \end{array} \right.$$

$$\Delta(s_0\, s_1 \ldots s_n) = \left\{ \begin{array}{l} \text{set of infinite paths } \pi \text{ of the form} \\ s_0\, s_1 \ldots s_n\, s_{n+1}\, s_{n+2}\, s_{n+3} \ldots \end{array} \right.$$

Measurability of classical properties

A σ-algebra is a pair $(\mathcal{U}, \mathcal{V})$ where $\mathcal{U}$ is a set and $\mathcal{V} \subseteq 2^{\mathcal{U}}$ such that:

1. $\mathcal{U} \in \mathcal{V}$
2. if $T \in \mathcal{V}$ then $\mathcal{U} \setminus T \in \mathcal{V}$
3. if $T_i \in \mathcal{V}$ for $i \in \mathbb{N}$ then $\bigcup_{i \in \mathbb{N}} T_i \in \mathcal{V}$

The elements of $\mathcal{V}$ are called events.

step-bounded reachability: "visit G within n steps"

$$\Diamond^{\leqslant n} G \;=\; \bigcup_{0 \leqslant i \leqslant n} \; \bigcup_{s_0, \ldots, s_i} \Delta\big(s_0\, s_1 \ldots s_{i-1}\, s_i\big)$$

where $s_i \in G$

Measurability of classical properties

A σ-algebra is a pair $(\mathcal{U}, \mathcal{V})$ where $\mathcal{U}$ is a set and $\mathcal{V} \subseteq 2^{\mathcal{U}}$ such that:

1. $\mathcal{U} \in \mathcal{V}$
2. if $T \in \mathcal{V}$ then $\mathcal{U} \setminus T \in \mathcal{V}$
3. if $T_i \in \mathcal{V}$ for $i \in \mathbb{N}$ then $\bigcup_{i \in \mathbb{N}} T_i \in \mathcal{V}$

The elements of $\mathcal{V}$ are called events.

step-bounded reachability: "visit G within n steps"

$$\Diamond^{\leqslant n} G \;=\; \bigcup_{0 \leqslant i \leqslant n} \; \bigcup_{s_0, \ldots, s_i} \Delta\big(s_0\, s_1 \ldots s_{i-1}\, s_i\big)$$

where $s_i \in G$ and $s_0, \ldots, s_{i-1} \notin G$

Measurability of classical properties

A σ-algebra is a pair $(\mathcal{U}, \mathcal{V})$ where $\mathcal{U}$ is a set and $\mathcal{V} \subseteq 2^{\mathcal{U}}$ such that:

1. $\mathcal{U} \in \mathcal{V}$
2. if $T \in \mathcal{V}$ then $\mathcal{U} \setminus T \in \mathcal{V}$
3. if $T_i \in \mathcal{V}$ for $i \in \mathbb{N}$ then $\bigcup_{i \in \mathbb{N}} T_i \in \mathcal{V}$

The elements of $\mathcal{V}$ are called events.

step-bounded reachability: "visit G within n steps"

$$\Diamond^{\leqslant n} G = \bigcup_{0 \leqslant i \leqslant n} \ \bigcup_{s_0, \ldots, s_i} \Delta\big(s_0\, s_1 \ldots s_{i-1}\, s_i\big)$$

$$\Pr^{\mathcal{M}}(\Diamond^{\leqslant n} G) = \sum_{0 \leqslant i \leqslant n} \ \sum_{s_0, \ldots, s_i} \Pr^{\mathcal{M}}\big(s_0\, s_1 \ldots s_{i-1}\, s_i\big)$$

Measurability of classical properties

A σ-algebra is a pair $(\mathcal{U}, \mathcal{V})$ where $\mathcal{U}$ is a set and $\mathcal{V} \subseteq 2^{\mathcal{U}}$ such that:

1. $\mathcal{U} \in \mathcal{V}$
2. if $T \in \mathcal{V}$ then $\mathcal{U} \setminus T \in \mathcal{V}$
3. if $T_i \in \mathcal{V}$ for $i \in \mathbb{N}$ then $\bigcup_{i \in \mathbb{N}} T_i \in \mathcal{V}$

The elements of $\mathcal{V}$ are called events.

unbounded reachability: "visit G eventually"

$$\Diamond G = \bigcup_{i \in \mathbb{N}} \bigcup_{s_0, \ldots, s_i} \Delta\big(s_0\, s_1 \ldots s_{i-1}\, s_i\big)$$

where $s_i \in G$ and $s_0, \ldots, s_{i-1} \notin G$

Measurability of classical properties

A σ-algebra is a pair $(\mathcal{U}, \mathcal{V})$ where $\mathcal{U}$ is a set and $\mathcal{V} \subseteq 2^{\mathcal{U}}$ such that:

1. $\mathcal{U} \in \mathcal{V}$
2. if $T \in \mathcal{V}$ then $\mathcal{U} \setminus T \in \mathcal{V}$
3. if $T_i \in \mathcal{V}$ for $i \in \mathbb{N}$ then $\bigcup_{i \in \mathbb{N}} T_i \in \mathcal{V}$

The elements of $\mathcal{V}$ are called events.

unbounded reachability: "visit G eventually"

$$\Diamond G = \bigcup_{i \in \mathbb{N}} \bigcup_{s_0, \ldots, s_i} \Delta(s_0 s_1 \ldots s_{i-1} s_i)$$

$$\Pr^{\mathcal{M}}(\Diamond G) = \sum_{i \in \mathbb{N}} \sum_{s_0, \ldots, s_i} \Pr^{\mathcal{M}}(s_0 s_1 \ldots s_{i-1} s_i)$$

Measurability of classical properties

A σ-algebra is a pair $(\mathcal{U}, \mathcal{V})$ where $\mathcal{U}$ is a set and $\mathcal{V} \subseteq 2^{\mathcal{U}}$ such that:

1. $\mathcal{U} \in \mathcal{V}$
2. if $T \in \mathcal{V}$ then $\mathcal{U} \setminus T \in \mathcal{V}$
3. if $T_i \in \mathcal{V}$ for $i \in \mathbb{N}$ then $\bigcup\limits_{i \in \mathbb{N}} T_i \in \mathcal{V}$

The elements of $\mathcal{V}$ are called events.

repeated reachability: "visit G infinitely often"

$$\Box\Diamond G \;=\; \bigcap_{n \in \mathbb{N}} \; \bigcup_{i \geqslant n} \; \bigcup_{s_0,\ldots,s_i} \Delta\big(s_0\, s_1 \ldots s_{i-1}\, s_i\big)$$

where $s_i \in G$

Measurability of classical properties

A σ-algebra is a pair $(\mathcal{U}, \mathcal{V})$ where $\mathcal{U}$ is a set and $\mathcal{V} \subseteq 2^{\mathcal{U}}$ such that:

1. $\mathcal{U} \in \mathcal{V}$
2. if $T \in \mathcal{V}$ then $\mathcal{U} \setminus T \in \mathcal{V}$
3. if $T_i \in \mathcal{V}$ for $i \in \mathbb{N}$ then $\bigcup_{i \in \mathbb{N}} T_i \in \mathcal{V}$

The elements of $\mathcal{V}$ are called events.

repeated reachability: "visit G infinitely often"

$$\Box\Diamond G = \bigcap_{n \in \mathbb{N}} \bigcup_{i \geqslant n} \bigcup_{s_0,\ldots,s_i} \Delta(s_0\, s_1 \ldots s_{i-1}\, s_i)$$

where $s_i \in G$, but possibly $s_j \in G$ for some $j < i$

Measurability of classical properties

A σ-algebra is a pair $(\mathcal{U}, \mathcal{V})$ where $\mathcal{U}$ is a set and $\mathcal{V} \subseteq 2^{\mathcal{U}}$ such that:

1. $\mathcal{U} \in \mathcal{V}$
2. if $T \in \mathcal{V}$ then $\mathcal{U} \setminus T \in \mathcal{V}$
3. if $T_i \in \mathcal{V}$ for $i \in \mathbb{N}$ then $\bigcup_{i \in \mathbb{N}} T_i \in \mathcal{V}$

The elements of $\mathcal{V}$ are called events.

persistence: "from some moment on always G"

$$\Diamond\Box G \;=\; Paths^{\mathcal{M}} \setminus \Box\Diamond\neg G$$

Measurability of classical properties

A σ-algebra is a pair $(\mathcal{U}, \mathcal{V})$ where $\mathcal{U}$ is a set and $\mathcal{V} \subseteq 2^{\mathcal{U}}$ such that:

1. $\mathcal{U} \in \mathcal{V}$
2. if $T \in \mathcal{V}$ then $\mathcal{U} \setminus T \in \mathcal{V}$
3. if $T_i \in \mathcal{V}$ for $i \in \mathbb{N}$ then $\bigcup_{i \in \mathbb{N}} T_i \in \mathcal{V}$

The elements of $\mathcal{V}$ are called events.

persistence: "from some moment on always G"

$$\Diamond\Box G = \mathit{Paths}^{\mathcal{M}} \setminus \Box\Diamond\neg G$$

$$\Pr^{\mathcal{M}}(\Diamond\Box G) = 1 - \Pr^{\mathcal{M}}(\Box\Diamond\neg G)$$

Stochastic process of a Markov chain

Stochastic process of a Markov chain

general definition of a stochastic process:

family $\left(X_t\right)_{t \in Time}$ of random variables $X_t : \mathcal{U} \to S$

Stochastic process of a Markov chain

general definition of a stochastic process:

family $(X_t)_{t \in Time}$ of random variables $X_t : \mathcal{U} \to S$

- $Time$ is a time domain, e.g., $\mathbb{N}$ or $\mathbb{R}_{\geqslant 0}$
- S is a set with fixed σ-algebra
- $\mathcal{U}$ is a sample space with fixed σ-algebra

Stochastic process of a Markov chain

DTMC $\mathcal{M} = (S, P, \ldots)$

family $\left(X_t\right)_{t \in Time}$ of random variables $X_t : \mathcal{U} \to S$

- $Time$ is a time domain $\longleftarrow$ $Time = \mathbb{N}$
- S is a set $\longleftarrow$ state space
- $\mathcal{U}$ is a sample space $\longleftarrow$ set of infinite paths

Stochastic process of a Markov chain

DTMC $\mathcal{M} = (S, P, \ldots)$

family $(X_t)_{t \in Time}$ of random variables $X_t : \mathcal{U} \to S$

- $Time$ is a time domain $\longleftarrow$ $Time = \mathbb{N}$
- S is a set $\longleftarrow$ state space
- $\mathcal{U}$ is a sample space $\longleftarrow$ set of infinite paths

---

If $t \in \mathbb{N}$ and $\pi = s_0\, s_1\, s_2\, s_3\, \ldots\, s_t \ldots$ then $X_t(\pi) = s_t$.

Stochastic process of a Markov chain

DTMC $\mathcal{M} = (S, P, \ldots)$

family $(X_t)_{t \in Time}$ of random variables $X_t : \mathcal{U} \to S$

- $Time$ is a time domain $\longleftarrow$ $Time = \mathbb{N}$
- S is a set $\longleftarrow$ state space
- $\mathcal{U}$ is a sample space $\longleftarrow$ set of infinite paths

---

If $t \in \mathbb{N}$ and $\pi = s_0\, s_1 \ldots s_{t-2}\, u\, s_t \ldots$ then $X_t(\pi) = s_t$.

Markov property:

$$\Pr{}^{\mathcal{M}}(X_t = s \mid X_{t-1} = u) =$$
$$\Pr{}^{\mathcal{M}}(X_t = s \mid X_{t-1} = u,\, X_{t-2} = s_{t-2},\, \ldots,\, X_0 = s_0)$$

Stochastic process of a Markov chain

DTMC $\mathcal{M} = (S, P, \ldots)$

family $(X_t)_{t \in Time}$ of random variables $X_t : \mathcal{U} \to S$

- $Time$ is a time domain $\longleftarrow$ $Time = \mathbb{N}$
- S is a set $\longleftarrow$ state space
- $\mathcal{U}$ is a sample space $\longleftarrow$ set of infinite paths

---

If $t \in \mathbb{N}$ and $\pi = s_0\, s_1 \ldots s_{t-2}\, u\, s_t \ldots$ then $X_t(\pi) = s_t$.

Markov property:

$$\Pr^{\mathcal{M}}\left(X_t = s \,\middle|\, X_{t-1} = u \right) = P(u, s) =$$

$$\Pr^{\mathcal{M}}\left(X_t = s \,\middle|\, X_{t-1} = u,\, X_{t-2} = s_{t-2}, \ldots, X_0 = s_0 \right)$$

Stochastic process of a Markov chain

DTMC $\mathcal{M} = (S, P, \ldots)$

family $\left(X_t\right)_{t \in Time}$ of random variables $X_t : \mathcal{U} \to S$

- $Time$ is a time domain $\longleftarrow$ $Time = \mathbb{N}$
- S is a set $\longleftarrow$ state space
- $\mathcal{U}$ is a sample space $\longleftarrow$ set of infinite paths

---

If $t \in \mathbb{N}$ and $\pi = s_0\, s_1 \ldots s_{t-2}\, u\, s_t \ldots$ then $X_t(\pi) = s_t$.

Markov property:

$$\Pr^{\mathcal{M}}\left(X_t = s \,\middle|\, X_{t-1} = u \right) \;=\; P(u, s) \;=$$

$$\Pr^{\mathcal{M}}\left(X_1 = s \,\middle|\, X_0 = u \right) \qquad \text{time-homogeneous}$$

Transient and long-run distribution

Transient and long-run distribution

transient: … refers to a fixed time point t

long-run: … when time tends to infinity

Transient distribution

Let $\mathcal{M} = (S, P, \mu, \ldots)$ be a DTMC, $t \in \mathbb{N}$ and $s \in S$.

transient state probability:

$$
\begin{aligned}
\mu_t(s) &= \Pr^{\mathcal{M}}\{s_0\, s_1\, s_2 \ldots \in \mathit{Paths}^{\mathcal{M}} : s_t = s\} \\
&= \Pr^{\mathcal{M}}(\, X_t = s\,)
\end{aligned}
$$

Transient distribution

Let $\mathcal{M} = (S, P, \mu, \ldots)$ be a DTMC, $t \in \mathbb{N}$ and $s \in S$.

transient state probability:

$$\begin{aligned} \mu_t(s) &= \Pr^{\mathcal{M}}\{s_0\, s_1\, s_2 \ldots \in Paths^{\mathcal{M}} : s_t = s\} \\ &= \mu \cdot P^t \cdot id_s \end{aligned}$$

↑ initial distribution (row vector)

Transient distribution

Let $\mathcal{M} = (S, P, \mu, \ldots)$ be a DTMC, $t \in \mathbb{N}$ and $s \in S$.

transient state probability:

$$\begin{aligned} \mu_t(s) &= \Pr^{\mathcal{M}}\{s_0\, s_1\, s_2 \ldots \in Paths^{\mathcal{M}} : s_t = s\} \\ &= \mu \cdot P^t \cdot id_s \end{aligned}$$

↑ t-th power of transition probability matrix

$$P^t = P^{t-1} \cdot P$$

Transient distribution

Let $\mathcal{M} = (S, P, \mu, \ldots)$ be a DTMC, $t \in \mathbb{N}$ and $s \in S$.

transient state probability:

$$\begin{aligned} \mu_t(s) &= \Pr^{\mathcal{M}}\{s_0\, s_1\, s_2 \ldots \in \mathit{Paths}^{\mathcal{M}} : s_t = s\} \\ &= \mu \cdot P^t \cdot \mathit{id}_s \end{aligned}$$

column vector $(0 \ldots 0, 1, 0, \ldots 0)$
representing Dirac distribution
for state s

Transient distribution

Let $\mathcal{M} = (S, P, \mu, \ldots)$ be a DTMC, $t \in \mathbb{N}$ and $s \in S$.

transient state probability:

$$\begin{aligned} \mu_t(s) &= \Pr^{\mathcal{M}}\{s_0\, s_1\, s_2 \ldots \in Paths^{\mathcal{M}} : s_t = s\} \\ &= \mu \cdot P^t \cdot id_s \;=\; \mu_{t-1} \cdot P \cdot id_s \end{aligned}$$

↑ column vector $(0 \ldots 0, 1, 0, \ldots 0)$ representing Dirac distribution for state s

Transient distribution

Let $\mathcal{M} = (S, P, \mu, \ldots)$ be a DTMC, $t \in \mathbb{N}$ and $s \in S$.

transient state probability:

$$\begin{aligned} \mu_t(s) &= \Pr^{\mathcal{M}}\{s_0\, s_1\, s_2 \ldots \in Paths^{\mathcal{M}} : s_t = s\} \\ &= \mu \cdot P^t \cdot id_s \;=\; \mu_{t-1} \cdot P \cdot id_s \end{aligned}$$

↑ transient state distribution for time point $t{-}1$

Thus: $\mu_0 = \mu$ initial distribution

$\mu_t = \mu_{t-1} \cdot P$ for $t \geqslant 1$

Long-run distributions

Let $\mathcal{M} = (S, P, \mu, \ldots)$ be a DTMC.

steady-state probability: $\widetilde{\mu}(s) = \lim_{t \to \infty} \mu_t(s)$

$\mu_t(s)$ probability for being in state s after t steps

Long-run distributions

Let $\mathcal{M} = (S, P, \mu, \ldots)$ be a DTMC.

steady-state probability: $\widetilde{\mu}(s) = \lim\limits_{t \to \infty} \mu_t(s)$

- limit may not exist

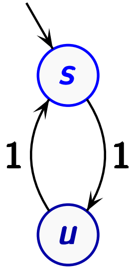
$\mu_t(s)$ probability for being in state s after t steps

Long-run distributions

Let $\mathcal{M} = (S, P, \mu, \ldots)$ be a DTMC.

steady-state probability: $$\widetilde{\mu}(s) = \lim_{t \to \infty} \mu_t(s)$$

- limit may not exist



$\mu_{2t}(s) = 1$ $\mu_{2t+1}(s) = 0$

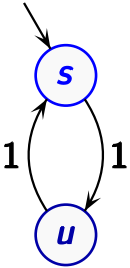
even time points   odd time points

Long-run distributions

Let $\mathcal{M} = (S, P, \mu, \ldots)$ be a DTMC.

steady-state probability: $\widetilde{\mu}(s) = \lim_{t \to \infty} \mu_t(s)$

- limit may not exist or depend on the initial distribution μ



$\mu_{2t}(s) = 1$ ↑ even time points

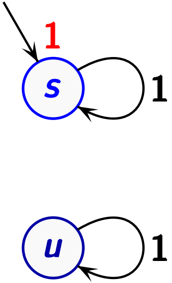
$\mu_{2t+1}(s) = 0$ ↑ odd time points

Long-run distributions

Let $\mathcal{M} = (S, P, \mu, \ldots)$ be a DTMC.

steady-state probability: $\widetilde{\mu}(s) = \lim_{t \to \infty} \mu_t(s)$

- limit may not exist or depend on the initial distribution μ



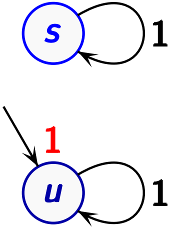
If $\mu(s) = 1$ then: $\widetilde{\mu}(s) = 1$

Long-run distributions

Let $\mathcal{M} = (S, P, \mu, \ldots)$ be a DTMC.

steady-state probability: $\widetilde{\mu}(s) = \lim_{t\to\infty} \mu_t(s)$

- limit may not exist or depend on the initial distribution μ



If $\mu(s) = 1$ then: $\widetilde{\mu}(s) = 1$

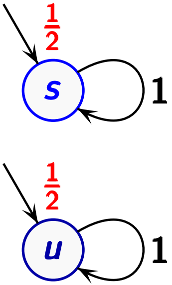
If $\mu(u) = 1$ then: $\widetilde{\mu}(s) = 0$

Long-run distributions

Let $\mathcal{M} = (S, P, \mu, \ldots)$ be a DTMC.

steady-state probability: $\quad \widetilde{\mu}(s) = \lim_{t\to\infty} \mu_t(s)$

- limit may not exist or depend on the initial distribution μ



If $\mu(s) = 1$ then: $\widetilde{\mu}(s) = 1$

If $\mu(u) = 1$ then: $\widetilde{\mu}(s) = 0$

If $\mu(s) = \mu(u) = \frac{1}{2}$ then:

$$\widetilde{\mu}(s) = \frac{1}{2}$$

Long-run distributions

Let $\mathcal{M} = (S, P, \mu, \ldots)$ be a DTMC.

steady-state probability: $\widetilde{\mu}(s) = \lim\limits_{t \to \infty} \mu_t(s)$

- limit may not exist or depend on the initial distribution μ
- if existing for all states s then $\widetilde{\mu} = \widetilde{\mu} \cdot P$

$\uparrow$ balance equation

Long-run distributions

Let $\mathcal{M} = (S, P, \mu, \ldots)$ be a DTMC.

steady-state probability: $\widetilde{\mu}(s) = \lim_{t \to \infty} \mu_t(s)$

long-run fraction of being in state s (Cesàro limit):

$\mu_t(s)$ probability for being in state s after t steps

Long-run distributions

Let $\mathcal{M} = (S, P, \mu, \ldots)$ be a DTMC.

steady-state probability: $\widetilde{\mu}(s) = \lim\limits_{t \to \infty} \mu_t(s)$

long-run fraction of being in state s (Cesàro limit):

$$\theta(s) \;=\; \lim_{T \to \infty} \frac{1}{T+1} \cdot \sum_{t=0}^{T} \mu_t(s)$$

$\mu_t(s)$ probability for being in state s after t steps

Long-run distributions

Let $\mathcal{M} = (S, P, \mu, \ldots)$ be a DTMC.

steady-state probability: $\widetilde{\mu}(s) = \lim\limits_{t\to\infty} \mu_t(s)$

long-run fraction of being in state s (Cesàro limit):

$$\theta(s) \;=\; \lim_{T\to\infty} \frac{1}{T+1} \cdot \sum_{t=0}^{T} \mu_t(s)$$

- Cesàro limit always exists

$\mu_t(s)$ probability for being in state s after t steps

Long-run distributions

Let $\mathcal{M} = (S, P, \mu, \ldots)$ be a DTMC.

steady-state probability: $\widetilde{\mu}(s) = \lim\limits_{t \to \infty} \mu_t(s)$

long-run fraction of being in state s (Cesàro limit):

$$\theta(s) \ = \ \lim_{T \to \infty} \frac{1}{T+1} \cdot \sum_{t=0}^{T} \mu_t(s)$$

- Cesàro limit always exists
- if the steady-state probabilities exists: $\widetilde{\mu}(s) = \theta(s)$

$\mu_t(s)$ probability for being in state s after t steps

Long-run distributions

Let $\mathcal{M} = (S, P, \mu, \ldots)$ be a DTMC.

steady-state probability: $\widetilde{\mu}(s) = \lim_{t\to\infty} \mu_t(s)$

long-run fraction of being in state s (Cesàro limit):

$$\theta(s) \ = \ \lim_{T\to\infty} \frac{1}{T+1} \cdot \sum_{t=0}^{T} \mu_t(s)$$

- Cesàro limit always exists
- if the steady-state probabilities exists: $\widetilde{\mu}(s) = \theta(s)$
- if $\mathcal{M}$ is strongly connected: θ is computable via the balance equation $\theta = \theta \cdot P$ where $\sum_{s\in S} \theta(s) = 1$

Fundamental property of finite Markov chains

Fundamental property of finite Markov chains

Almost surely, i.e., with probability **1**:

A bottom strongly connected component will be reached and all its states visited infinitely often.

Fundamental property of finite Markov chains

Almost surely, i.e., with probability **1**:

A bottom strongly connected component will be reached and all its states visited infinitely often.

$$\Pr^{\mathcal{M}}\Big\{ s_0\, s_1\, s_2 \ldots \in \mathit{Paths}^{\mathcal{M}} :$$

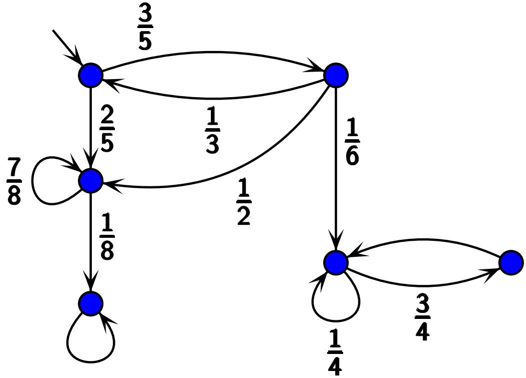
there exists $i \geqslant 0$ and a BSCC C s.t.

$$\underbrace{\forall j \geqslant i.\ s_j \in C}_{\text{eventually forever } C} \ \wedge\ \underbrace{\forall s \in C\ \overset{\infty}{\exists} j.\ s_j = s}_{\text{visit each state in } C \text{ infinitely often}} \Big\} = 1$$

Fundamental property of finite Markov chains

Almost surely, i.e., with probability **1**:

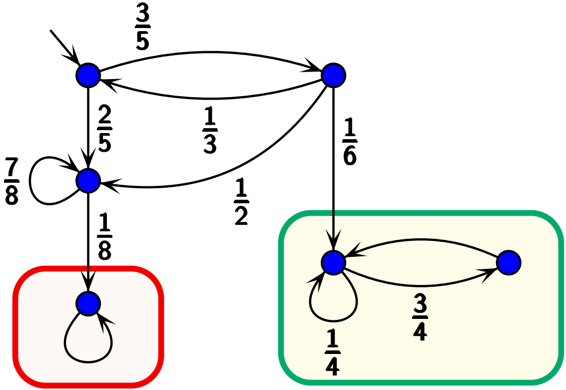
A bottom strongly connected component will be reached and all its states visited infinitely often.



Fundamental property of finite Markov chains

Almost surely, i.e., with probability **1**:

A bottom strongly connected component will be reached and all its states visited infinitely often.



2 BSCCs

Fundamental property of finite Markov chains

> Almost surely, i.e., with probability **1**:
>
> A bottom strongly connected component will be reached and all its states visited infinitely often.

long-run distribution:

$$\theta(s) > 0 \quad \text{iff} \quad s \text{ belongs to some BSCC}$$

Fundamental property of finite Markov chains

Almost surely, i.e., with probability **1**:

A bottom strongly connected component will be reached and all its states visited infinitely often.

long-run distribution:

- $\theta(s) > 0$ iff s belongs to some BSCC
- if s is a state of BSCC B then:

$$\theta(s) = \underbrace{\Pr^{\mathcal{M}}(\Diamond B)}_{\text{probability for reaching B}} \cdot \theta^{B}(s)$$

$\theta^{B}(s)$: long-run probability for state s inside B

Tutorial: Probabilistic Model Checking

Discrete-time Markov chains (DTMC)

* basic definitions
* probabilistic computation tree logic PCTL/PCTL*
* rewards, cost-utility ratios, weights
* conditional probabilities

Markov decision processes (MDP)

* basic definitions
* PCTL/PCTL* model checking
* fairness
* conditional probabilities
* rewards, quantiles
* mean-payoff
* expected accumulated weights

Probabilistic computation tree logic

Probabilistic computation tree logic

PCTL/PCTL* [Hansson/Jonsson 1994]

- probabilistic variants of CTL/CTL*
- contains a probabilistic operator $\mathbb{P}$
  to specify lower/upper probability bounds

Probabilistic computation tree logic

PCTL/PCTL* [Hansson/Jonsson 1994]

- probabilistic variants of CTL/CTL*
- contains a probabilistic operator $\mathbb{P}$
  to specify lower/upper probability bounds
- operators for expected costs, long-run averages, ...
  will be considered later

Syntax of PCTL*

state formulas:

$$\Phi \ ::= \ \mathit{true} \mid a \mid \Phi_1 \wedge \Phi_2 \mid \neg\Phi \mid \quad \ldots$$

path formulas:

$$\varphi \ ::= \ \ldots$$

Syntax of PCTL*

state formulas:

$$\Phi \ ::= \ \mathit{true} \mid a \mid \Phi_1 \wedge \Phi_2 \mid \neg\Phi \mid \mathbb{P}_{\mathrm{I}}(\varphi)$$

path formulas:

$$\varphi \ ::= \ \ldots$$

where $a \in AP$ is an atomic proposition

$\mathrm{I} \subseteq [0,1]$ is a probability interval

Syntax of PCTL*

state formulas:

$$\Phi \ ::= \ true \mid a \mid \Phi_1 \wedge \Phi_2 \mid \neg\Phi \mid \mathbb{P}_{\mathrm{I}}(\varphi)$$

path formulas:

$$\varphi \ ::= \ \ldots$$

where $a \in AP$ is an atomic proposition

$\mathrm{I} \subseteq [0,1]$ is a probability interval

qualitative properties: $\mathbb{P}_{>0}(\varphi)$ or $\mathbb{P}_{=1}(\varphi)$

quantitative properties: e.g., $\mathbb{P}_{>0.5}(\varphi)$ or $\mathbb{P}_{\leqslant 0.01}(\varphi)$

Syntax of PCTL* path formulas

state formulas:

$$\Phi ::= \mathit{true} \mid a \mid \Phi_1 \wedge \Phi_2 \mid \neg\Phi \mid \mathbb{P}_I(\varphi)$$

path formulas:

$$\varphi ::= \Phi \mid \varphi_1 \wedge \varphi_2 \mid \neg\varphi \mid \ldots$$

↑ state formula

Syntax of PCTL* path formulas

state formulas:

$$\Phi \;::=\; \mathit{true} \;\big|\; a \;\big|\; \Phi_1 \wedge \Phi_2 \;\big|\; \neg\Phi \;\big|\; \mathbb{P}_{\mathrm{I}}(\varphi)$$

path formulas:

$$\varphi \;::=\; \Phi \;\big|\; \varphi_1 \wedge \varphi_2 \;\big|\; \neg\varphi \;\big|\; \bigcirc\varphi \;\big|\; \ldots$$

$\bigcirc \mathrel{\widehat{=}}$ next

Syntax of PCTL* path formulas

state formulas:

$$\Phi \; ::= \; \mathit{true} \;\big|\; a \;\big|\; \Phi_1 \wedge \Phi_2 \;\big|\; \neg\Phi \;\big|\; \mathbb{P}_I(\varphi)$$

path formulas:

$$\varphi \; ::= \; \Phi \;\big|\; \varphi_1 \wedge \varphi_2 \;\big|\; \neg\varphi \;\big|\; \bigcirc\varphi \;\big|\; \varphi_1 \, \mathbf{U} \, \varphi_2$$

$\bigcirc \mathrel{\widehat{=}}$ next $\qquad$ $\mathbf{U} \mathrel{\widehat{=}}$ until

Syntax of PCTL* path formulas

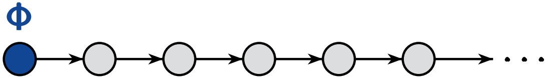
state formulas:

$$\Phi \ ::= \ true \mid a \mid \Phi_1 \wedge \Phi_2 \mid \neg\Phi \mid \mathbb{P}_I(\varphi)$$

path formulas:

$$\varphi \ ::= \ \Phi \mid \varphi_1 \wedge \varphi_2 \mid \neg\varphi \mid \bigcirc\varphi \mid \varphi_1 \, \mathsf{U} \, \varphi_2$$

state formula



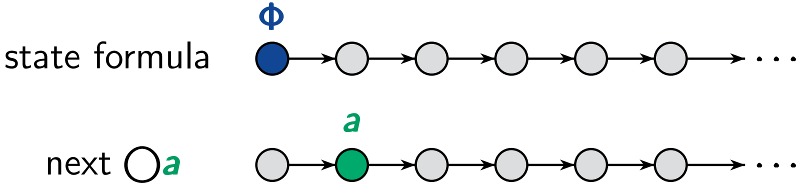
Syntax of PCTL* path formulas

state formulas:

$$\Phi \ ::= \ \mathit{true} \ \big| \ a \ \big| \ \Phi_1 \wedge \Phi_2 \ \big| \ \neg\Phi \ \big| \ \mathbb{P}_I(\varphi)$$

path formulas:

$$\varphi \ ::= \ \Phi \ \big| \ \varphi_1 \wedge \varphi_2 \ \big| \ \neg\varphi \ \big| \ \bigcirc\varphi \ \big| \ \varphi_1 \, \mathbf{U} \, \varphi_2$$



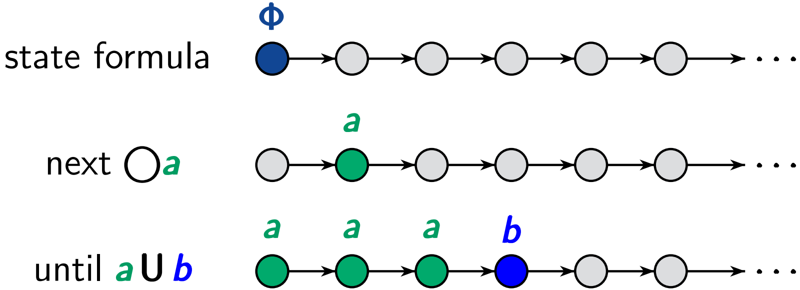
Syntax of PCTL* path formulas

state formulas:

$$\Phi \ ::= \ \mathit{true} \ \big| \ a \ \big| \ \Phi_1 \wedge \Phi_2 \ \big| \ \neg\Phi \ \big| \ \mathbb{P}_I(\varphi)$$

path formulas:

$$\varphi \ ::= \ \Phi \ \big| \ \varphi_1 \wedge \varphi_2 \ \big| \ \neg\varphi \ \big| \ \bigcirc\varphi \ \big| \ \varphi_1 \, \mathsf{U} \, \varphi_2$$



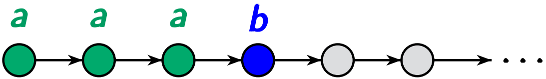
Derived path operators: eventually, always

Derived path operators: eventually, always

syntax of path formulas:

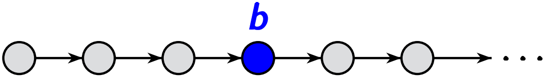
$$\varphi \ ::= \ \Phi \ \big| \ \varphi_1 \wedge \varphi_2 \ \big| \ \neg\varphi \ \big| \ \bigcirc\varphi \ \big| \ \varphi_1 \, \mathsf{U} \, \varphi_2$$

until $a \, \mathsf{U} \, b$



eventually

$\Diamond b \stackrel{\text{def}}{=} \mathit{true} \, \mathsf{U} \, b$

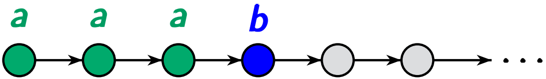


Derived path operators: eventually, always

syntax of path formulas:

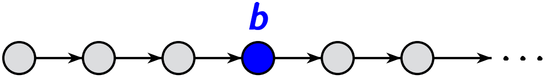
$$\varphi \ ::= \ \Phi \ \big| \ \varphi_1 \wedge \varphi_2 \ \big| \ \neg\varphi \ \big| \ \bigcirc\varphi \ \big| \ \varphi_1 \,\mathsf{U}\, \varphi_2$$

until $a \,\mathsf{U}\, b$



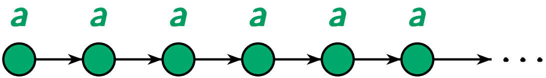
eventually

$\Diamond b \overset{\text{def}}{=} \mathit{true} \,\mathsf{U}\, b$



always

$\Box a \overset{\text{def}}{=} \neg\Diamond\neg a$



Semantics of PCTL*

Semantics of PCTL*

Let $\mathcal{M} = (S, P, AP, L)$ be a Markov chain.

Define by structural induction:

- a satisfaction relation $\models$ for states $s \in S$ and **PCTL*** state formulas
- a satisfaction relation $\models$ for infinite paths π in $\mathcal{M}$ and **PCTL*** path formulas

Semantics of PCTL*

$$
\begin{array}{lll}
s \models true & & \\
s \models a & \text{iff} & a \in L(s) \\
s \models \neg\Phi & \text{iff} & s \not\models \Phi \\
s \models \Phi_1 \wedge \Phi_2 & \text{iff} & s \models \Phi_1 \text{ and } s \models \Phi_2 \\
s \models \mathbb{P}_{\mathrm{I}}(\varphi) & \text{iff} & \Pr_s^{\mathcal{M}}(\varphi) \in \mathrm{I}
\end{array}
$$

Semantics of PCTL*

$s \models true$

$s \models a$ iff $a \in L(s)$

$s \models \neg\Phi$ iff $s \not\models \Phi$

$s \models \Phi_1 \wedge \Phi_2$ iff $s \models \Phi_1$ and $s \models \Phi_2$

$s \models \mathbb{P}_I(\varphi)$ iff $\Pr_s^{\mathcal{M}}(\varphi) \in I$

probability measure of the set of paths π with $\pi \models \varphi$

when s is viewed as the unique starting state

Semantics of PCTL* path formulas

let $\pi = s_0\, s_1\, s_2\, s_3 \dots$ be an infinite path in $\mathcal{M}$

Semantics of PCTL* path formulas

let $\pi = s_0\, s_1\, s_2\, s_3 \ldots$ be an infinite path in $\mathcal{M}$

$\pi \models \Phi$	iff	$s_0 \models \Phi$
$\pi \models \neg\varphi$	iff	$\pi \not\models \varphi$
$\pi \models \varphi_1 \wedge \varphi_2$	iff	$\pi \models \varphi_1$ and $\pi \models \varphi_2$
$\pi \models \bigcirc\varphi$	iff	$s_1\, s_2\, s_3 \ldots \models \varphi$
$\pi \models \varphi_1 \,\mathbf{U}\, \varphi_2$	iff	there exists $\ell \geqslant 0$ such that
		$s_\ell\, s_{\ell+1}\, s_{\ell+2} \ldots \models \varphi_2$
		$s_i\, s_{i+1}\, s_{i+2} \ldots \models \varphi_1$ for $0 \leqslant i < \ell$

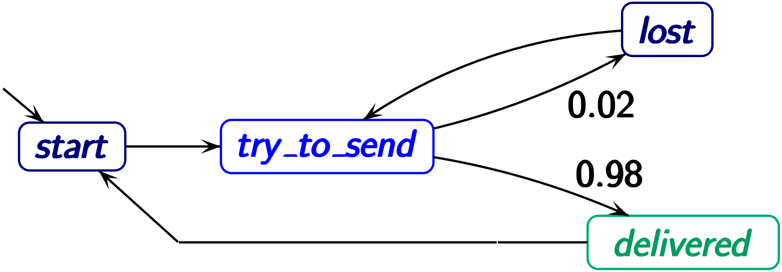
Examples for PCTL*-specifications

Examples for PCTL*-specifications

communication protocol:

$$\mathbb{P}_{=1}\big(\,\Box(\,\textit{try\_to\_send}\;\longrightarrow\;\mathbb{P}_{\geqslant 0.9}(\bigcirc \textit{delivered})\,)\,\big)$$

$$\mathbb{P}_{=1}\big(\,\Box(\,\textit{try\_to\_send}\;\longrightarrow\;\neg \textit{start}\;\mathsf{U}\;\textit{delivered}\,)\,\big)$$



Examples for PCTL*-specifications

communication protocol:

$$\mathbb{P}_{=1}\big(\ \Box(\ \mathit{try\_to\_send}\ \longrightarrow\ \mathbb{P}_{\geqslant 0.9}(\bigcirc \mathit{delivered})\ )\ \big)$$

$$\mathbb{P}_{=1}\big(\ \Box(\ \mathit{try\_to\_send}\ \longrightarrow\ \neg \mathit{start}\ \mathsf{U}\ \mathit{delivered}\ )\ \big)$$

leader election protocol for n processes:

$$\mathbb{P}_{=1}\big(\ \Diamond\ \mathit{leader\_elected}\ \big)$$

$$\mathbb{P}_{\geqslant 0.9}\big(\ \bigvee_{i \leqslant n} \bigcirc^{i}\ \mathit{leader\_elected}\ \big)$$

PCTL* model checking for DTMC

PCTL* model checking for DTMC

given: Markov chain $\mathcal{M} = (S, P, AP, L, s_0)$
PCTL* state formula Φ

task: check whether $s_0 \models \Phi$

PCTL* model checking for DTMC

given: Markov chain $\mathcal{M} = (S, P, AP, L, s_0)$
PCTL* state formula Φ

task: check whether $s_0 \models \Phi$

main procedure as for CTL*:

recursively compute the satisfaction sets

$$Sat(\Psi) \;=\; \{\, s \in S : s \models \Psi \,\}$$

for all state subformulas Ψ of Φ

Recursive computation of the satisfaction sets

Recursive computation of the satisfaction sets

$$\begin{aligned}
Sat(true) &= S \quad \text{state space of } \mathcal{M} \\
Sat(a) &= \{\, s \in S : a \in L(s) \,\} \\
Sat(\Phi_1 \wedge \Phi_2) &= Sat(\Phi_1) \cap Sat(\Phi_2) \\
Sat(\neg \Phi) &= S \setminus Sat(\Phi)
\end{aligned}$$

Recursive computation of the satisfaction sets

$$
\begin{aligned}
Sat(true) &= S \quad \text{state space of } \mathcal{M} \\
Sat(a) &= \{\, s \in S : a \in L(s) \,\} \\
Sat(\Phi_1 \wedge \Phi_2) &= Sat(\Phi_1) \cap Sat(\Phi_2) \\
Sat(\neg \Phi) &= S \setminus Sat(\Phi) \\
Sat(\,\mathbb{P}_{\mathrm{I}}(\varphi)\,) &= \{\, s \in S : \Pr_s^{\mathcal{M}}(\varphi) \in \mathrm{I} \,\}
\end{aligned}
$$

Recursive computation of the satisfaction sets

$$Sat(true) = S \quad \text{state space of } \mathcal{M}$$

$$Sat(a) = \{\, s \in S : a \in L(s) \,\}$$

$$Sat(\Phi_1 \wedge \Phi_2) = Sat(\Phi_1) \cap Sat(\Phi_2)$$

$$Sat(\neg \Phi) = S \setminus Sat(\Phi)$$

$$Sat(\, \mathbb{P}_{\mathrm{I}}(\varphi)\,) = \{\, s \in S : \mathrm{Pr}_s^{\mathcal{M}}(\varphi) \in \mathrm{I} \,\}$$

special case: $\varphi = \Diamond \Phi$

Recursive computation of the satisfaction sets

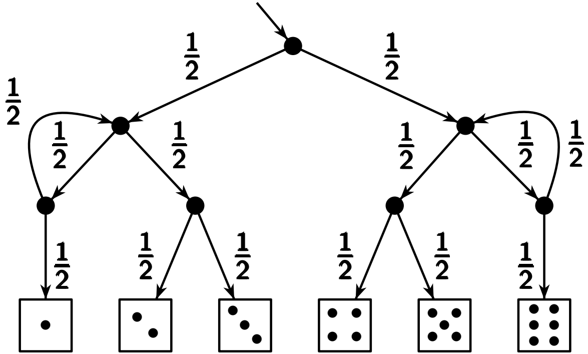
$$\begin{aligned}
Sat(true) &= S \quad \text{state space of } \mathcal{M} \\
Sat(a) &= \{\, s \in S : a \in L(s) \,\} \\
Sat(\Phi_1 \wedge \Phi_2) &= Sat(\Phi_1) \cap Sat(\Phi_2) \\
Sat(\neg \Phi) &= S \setminus Sat(\Phi) \\
Sat(\,\mathbb{P}_{\mathrm{I}}(\varphi)\,) &= \{\, s \in S : \Pr_s^{\mathcal{M}}(\varphi) \in \mathrm{I} \,\}
\end{aligned}$$

special case: $\varphi = \Diamond \Phi$

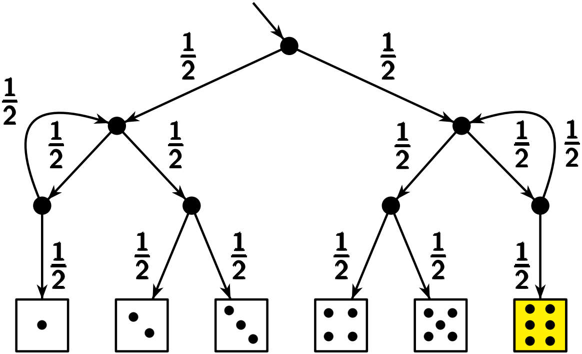
1. compute recursively $Sat(\Phi)$
2. compute $x_s = \Pr_s^{\mathcal{M}}(\Diamond \Phi)$ by solving a linear equation system

Simulating a dice by a coin

[Knuth]



Simulating a dice by a coin [KNUTH]

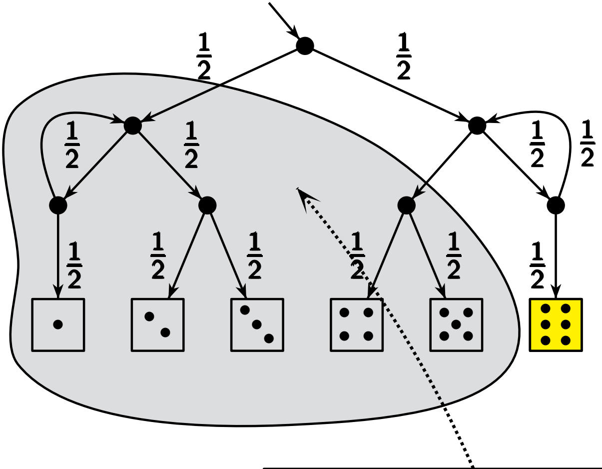


probability for the outcome **six**

$$Pr^{\mathcal{M}}(\Diamond\ \textsf{six}\ ) = \textbf{?}$$

Simulating a dice by a coin

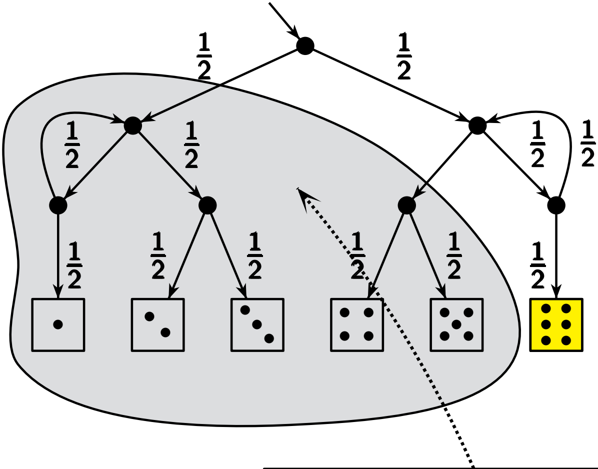
[KNUTH]



outcome **six** unreachable

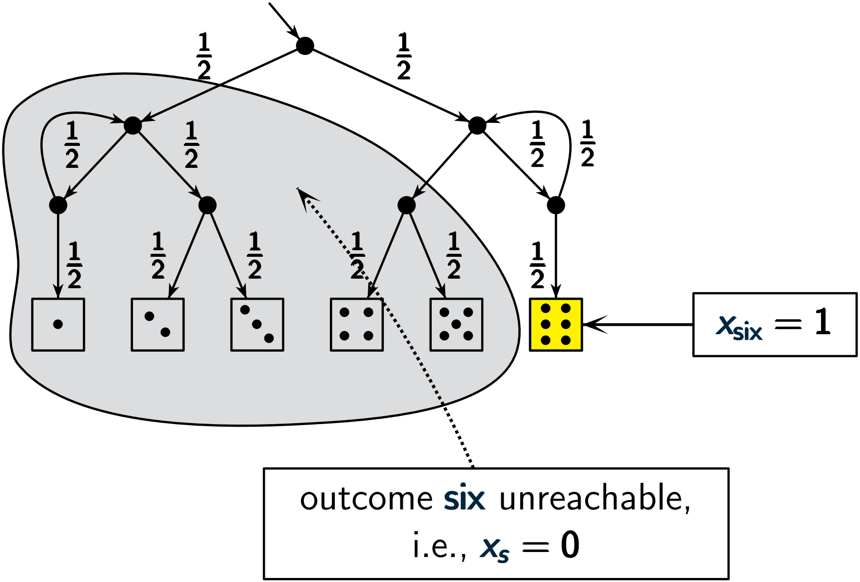
Simulating a dice by a coin

[KNUTH]



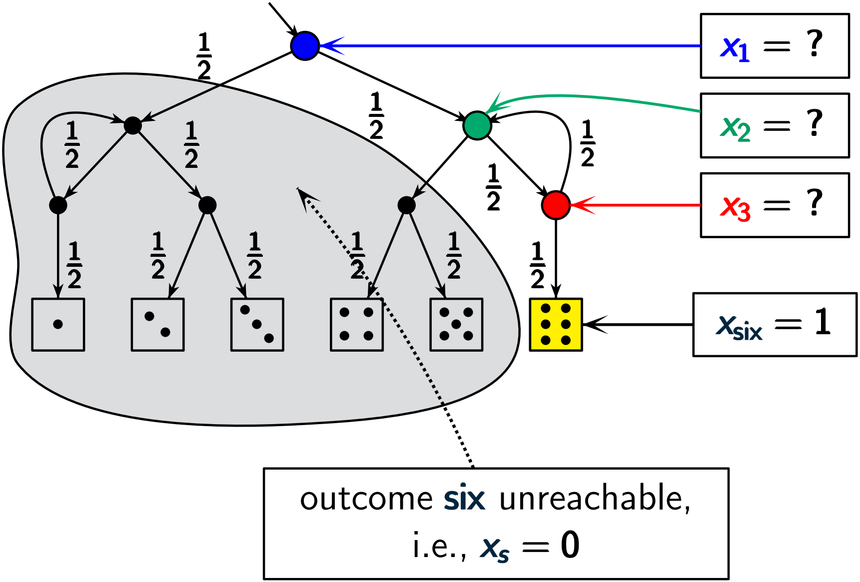
outcome **six** unreachable,
i.e., $x_s = 0$

Simulating a dice by a coin [KNUTH]



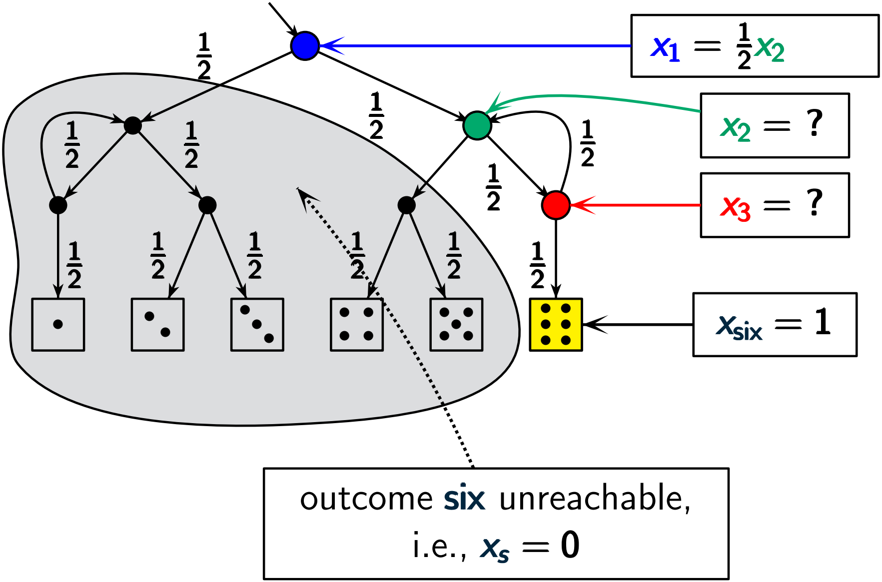
Simulating a dice by a coin

[KNUTH]



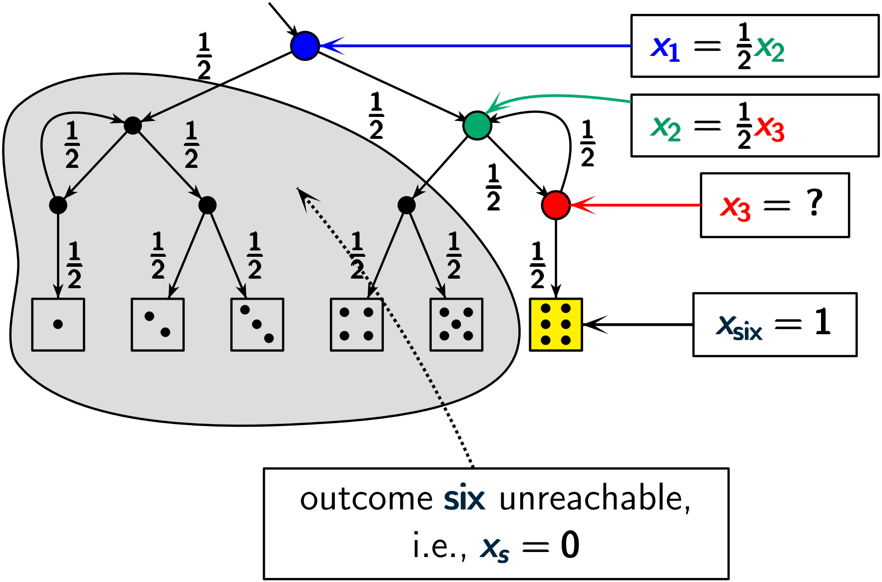
Simulating a dice by a coin

[Knuth]



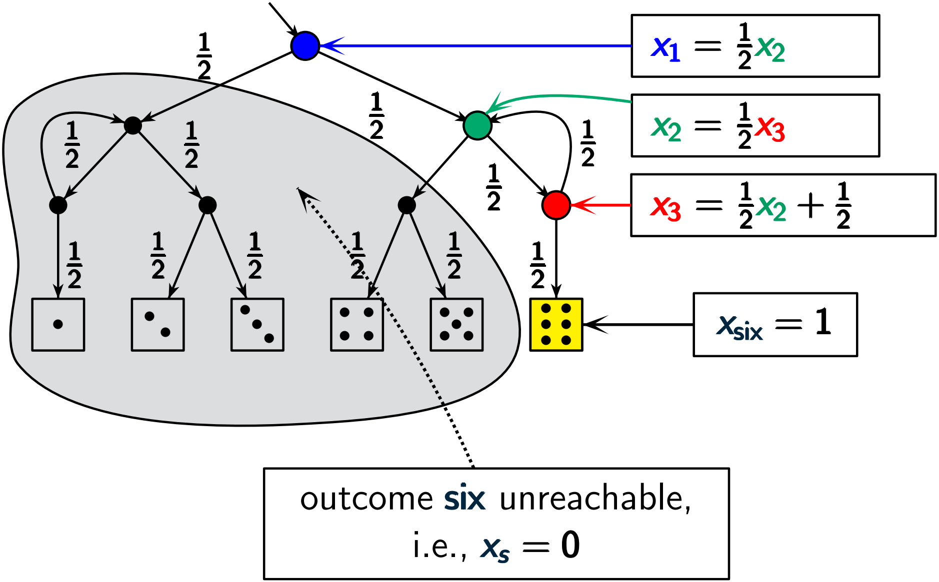
Simulating a dice by a coin

[Knuth]



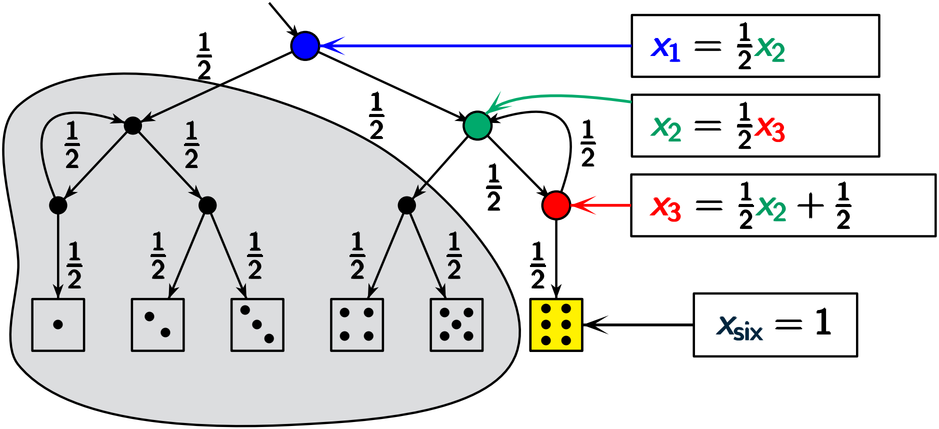
Simulating a dice by a coin

[Knuth]



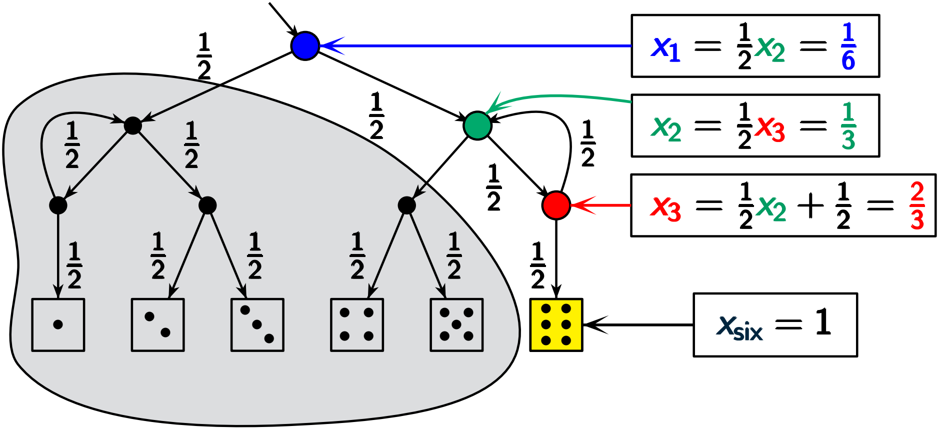
Simulating a dice by a coin

[Knuth]



$$\begin{pmatrix} 1 & -\frac{1}{2} & 0 \\ 0 & 1 & -\frac{1}{2} \\ 0 & -\frac{1}{2} & 1 \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \frac{1}{2} \end{pmatrix}$$

Simulating a dice by a coin [KNUTH]



$$\Pr^{\mathcal{M}}(\Diamond \text{ six }) = x_1 = \frac{1}{6}$$

$$\begin{pmatrix} 1 & -\frac{1}{2} & 0 \\ 0 & 1 & -\frac{1}{2} \\ 0 & -\frac{1}{2} & 1 \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \frac{1}{2} \end{pmatrix}$$

Computing reachability probabilities

Computing reachability probabilities

given: DTMC $\mathcal{M} = (S, P, \ldots)$ and $T \subseteq S$

task: compute $x_s = \Pr_s^{\mathcal{M}}(\Diamond T)$ for all $s \in S$

$\Diamond T$ "eventually reaching T"

Computing reachability probabilities

given: DTMC $\mathcal{M} = (S, P, \ldots)$ and $T \subseteq S$

task: compute $x_s = \Pr^{\mathcal{M}}_s(\Diamond T)$ for all $s \in S$

1. compute S^0 and S^1

$$S^0 = \{s \in S : x_s = 0\}$$
$$S^1 = \{s \in S : x_s = 1\}$$

2. ...

$\Diamond T$ "eventually reaching T"

Computing reachability probabilities

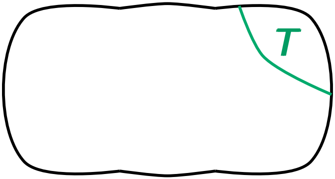
given: DTMC $\mathcal{M} = (S, P, \ldots)$ and $T \subseteq S$

task: compute $x_s = \Pr_s^{\mathcal{M}}(\Diamond T)$ for all $s \in S$

1. compute S^0 and S^1

$$S^0 = \{s \in S : x_s = 0\}$$
$$S^1 = \{s \in S : x_s = 1\}$$

2. ...



state space S

Computing reachability probabilities

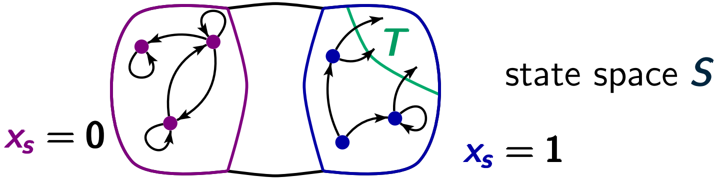
given: DTMC $\mathcal{M} = (S, P, \ldots)$ and $T \subseteq S$

task: compute $x_s = \Pr_s^{\mathcal{M}}(\Diamond T)$ for all $s \in S$

1. compute S^0 and S^1

$$S^0 = \{s \in S : x_s = 0\}$$
$$S^1 = \{s \in S : x_s = 1\}$$

2. ...



Computing reachability probabilities

given: DTMC $\mathcal{M} = (S, P, \ldots)$ and $T \subseteq S$

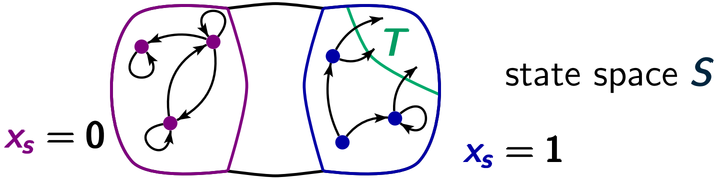
task: compute $x_s = \Pr_s^{\mathcal{M}}(\Diamond T)$ for all $s \in S$

1. compute S^0 and S^1

$$S^0 = \{s \in S : x_s = 0\} = \{s : s \not\models \exists \Diamond T\}$$

$$S^1 = \{s \in S : x_s = 1\}$$

2. …



Computing reachability probabilities

given: DTMC $\mathcal{M} = (S, P, \ldots)$ and $T \subseteq S$

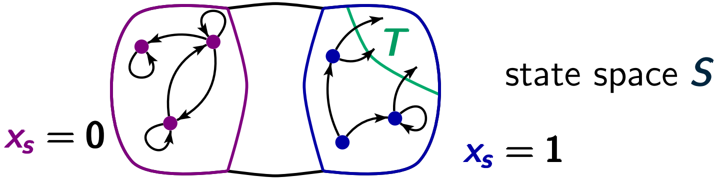
task: compute $x_s = \Pr_s^{\mathcal{M}}(\Diamond T)$ for all $s \in S$

1. compute S^0 and S^1

$$S^0 = \{s \in S : x_s = 0\} = \{s : s \not\models \exists \Diamond T\}$$

$$S^1 = \{s \in S : x_s = 1\} = \{s : s \not\models \exists (\neg T) \, \mathsf{U} \, S^0\}$$

2. …



Computing reachability probabilities

given: DTMC $\mathcal{M} = (S, P, \ldots)$ and $T \subseteq S$

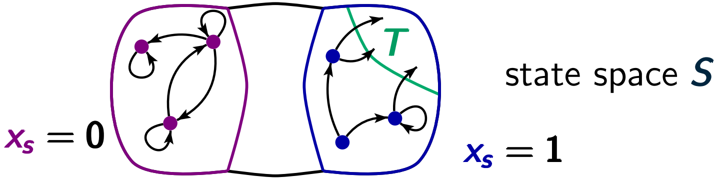
task: compute $x_s = \Pr_s^{\mathcal{M}}(\Diamond T)$ for all $s \in S$

1. compute S^0 and S^1 ⟵ graph algorithms

$$S^0 = \{s \in S : x_s = 0\} = \{s : s \not\models \exists \Diamond T\}$$

$$S^1 = \{s \in S : x_s = 1\} = \{s : s \not\models \exists (\neg T) \,\mathsf{U}\, S^0\}$$

2. ...



Computing reachability probabilities

given: DTMC $\mathcal{M} = (S, P, \ldots)$ and $T \subseteq S$

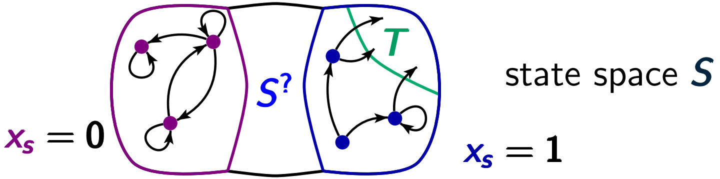
task: compute $x_s = \Pr_s^{\mathcal{M}}(\Diamond T)$ for all $s \in S$

1. compute S^0 and S^1 $\longleftarrow$ graph algorithms

$$S^0 = \{s \in S : x_s = 0\} = \{s : s \not\models \exists \Diamond T\}$$

$$S^1 = \{s \in S : x_s = 1\} = \{s : s \not\models \exists(\neg T) \, \mathsf{U} \, S^0\}$$

2. compute x_s for $s \in S^? = S \setminus (S^0 \cup S^1)$



Computing reachability probabilities

given: DTMC $\mathcal{M} = (S, P, \ldots)$ and $T \subseteq S$

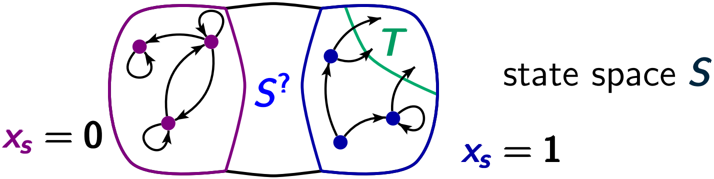
task: compute $x_s = \Pr_s^{\mathcal{M}}(\Diamond T)$ for all $s \in S$

1. compute S^0 and S^1 $\longleftarrow$ graph algorithms

$$S^0 = \{ s \in S : x_s = 0 \} = \{ s : s \not\models \exists \Diamond T \}$$

$$S^1 = \{ s \in S : x_s = 1 \} = \{ s : s \not\models \exists (\neg T) \, \mathsf{U} \, S^0 \}$$

2. compute x_s for $s \in S^? = \{ s : 0 < x_s < 1 \}$



Computing reachability probabilities

given: DTMC $\mathcal{M} = (S, P, \ldots)$ and $T \subseteq S$

task: compute $x_s = \Pr_s^{\mathcal{M}}(\Diamond T)$ for all $s \in S$

1. compute S^0 and S^1

$$S^0 = \{s \in S : x_s = 0\} = \{s : s \not\models \exists \Diamond T\}$$

$$S^1 = \{s \in S : x_s = 1\} = \{s : s \not\models \exists (\neg T) \, \mathrm{U} \, S^0\}$$

2. compute x_s for $s \in S^? = \{s : 0 < x_s < 1\}$

↑

by solving a linear equation system

Computing reachability probabilities

task: compute $x_s = \Pr_s^{\mathcal{M}}(\Diamond T)$ for all $s \in S^?$

by solving the equation system:

$$x_s = \sum_{s' \in S^?} P(s, s') \cdot x_{s'} + P(s, S^1)$$

$$P(s, S^1) = \sum_{u \in S^1} P(s, u)$$

Computing reachability probabilities

task: compute $x_s = \Pr^{\mathcal{M}}_s(\Diamond T)$ for all $s \in S^?$

by solving the equation system:

$$x_s = \sum_{s' \in S^?} P(s,s') \cdot x_{s'} + \underbrace{P(s, S^1)}$$

↑

probability for paths of the form

$s\ \underbrace{u_1\, u_2 \ldots u_k}_{u_j \in S^1}\ t$ with $t \in T$

Computing reachability probabilities

task: compute $x_s = \Pr^{\mathcal{M}}_s(\Diamond T)$ for all $s \in S^?$

by solving the equation system:

$$x_s = \underbrace{\sum_{s' \in S^?} P(s,s') \cdot x_{s'}}_{} + P(s, S^1)$$

↑

probability for paths of the form

$$\underbrace{s\ s_1\ s_2 \ldots s_m}_{s_i \in S^?}\ \underbrace{u_1\ u_2 \ldots u_k}_{u_j \in S^1}\ t \quad \text{with } t \in T \qquad m \geqslant 1$$

Computing reachability probabilities

task: compute $x_s = \Pr_s^{\mathcal{M}}(\Diamond T)$ for all $s \in S^?$

by solving the equation system:

$$x_s = \sum_{s' \in S^?} P(s,s') \cdot x_{s'} + P(s, S^1)$$

$$x = A \cdot x + b$$

Computing reachability probabilities

task: compute $x_s = \Pr^{\mathcal{M}}_s(\Diamond T)$ for all $s \in S^?$

by solving the equation system:

$$x_s \;=\; \sum_{s' \in S^?} P(s,s') \cdot x_{s'} \;+\; P(s, S^1)$$

$$x = A \cdot x + b$$

matrix $A = \big(P(s,s')\big)_{s,s' \in S^?}$

vectors $x = \big(x_s\big)_{s \in S^?}$

$b = \big(P(s,S^1)\big)_{s \in S^?}$

Computing reachability probabilities

task: compute $x_s = \Pr^{\mathcal{M}}_s(\Diamond T)$ for all $s \in S^?$

by solving the equation system:

$$x_s \;=\; \sum_{s' \in S^?} P(s,s') \cdot x_{s'} \;+\; P(s,S^1)$$

$$x = A \cdot x + b$$

iff

$$(\mathrm{I} - A) \cdot x = b$$

matrix $A = \big(P(s,s')\big)_{s,s' \in S^?}$

vectors $x = \big(x_s\big)_{s \in S^?}$

$b = \big(P(s,S^1)\big)_{s \in S^?}$

identity matrix I

Computing reachability probabilities

task: compute $x_s = \Pr^{\mathcal{M}}_s(\Diamond T)$ for all $s \in S^?$

by solving the equation system:

$$x_s = \sum_{s' \in S^?} P(s,s') \cdot x_{s'} + P(s, S^1)$$

$$x = A \cdot x + b$$
iff
$$(\mathrm{I} - A) \cdot x = b$$

linear equation system with non-singular matrix $\mathrm{I} - A$

Computing reachability probabilities

task: compute $x_s = \Pr_s^{\mathcal{M}}(\Diamond T)$ for all $s \in S^?$

by solving the equation system:

$$x_s = \sum_{s' \in S^?} P(s, s') \cdot x_{s'} + P(s, S^1)$$

$$x = A \cdot x + b$$

iff

$$(\mathrm{I} - A) \cdot x = b$$

linear equation system with non-singular matrix $\mathrm{I} - A$

$\Downarrow$

unique solution

PCTL

sublogic of PCTL* where only path formulas of the form $\bigcirc \Phi$ and $\Phi_1 \, \mathsf{U} \, \Phi_2$ are allowed

PCTL

sublogic of PCTL* where only path formulas of the form $\bigcirc \Phi$ and $\Phi_1 \, \mathsf{U} \, \Phi_2$ are allowed

state formulas:

$$\Phi \;::=\; \mathit{true} \;\big|\; a \;\big|\; \Phi_1 \wedge \Phi_2 \;\big|\; \neg\Phi \;\big|\; \mathbb{P}_{\mathrm{I}}(\varphi)$$

path formulas:

$$\varphi \;::=\; \bigcirc \Phi \;\big|\; \Phi_1 \, \mathsf{U} \, \Phi_2$$

PCTL

sublogic of PCTL* where only path formulas of the form $\bigcirc \Phi$ and $\Phi_1 \, \mathsf{U} \, \Phi_2$ are allowed

state formulas:

$$\Phi \;::=\; \mathit{true} \mid a \mid \Phi_1 \wedge \Phi_2 \mid \neg\Phi \mid \mathbb{P}_I(\varphi)$$

path formulas:

$$\varphi \;::=\; \bigcirc\Phi \mid \Phi_1 \, \mathsf{U} \, \Phi_2 \mid \Diamond\Phi \mid \Box\Phi$$

$$\mathbb{P}_I(\Diamond\Phi) \stackrel{\text{def}}{=} \mathbb{P}_I(\mathit{true} \, \mathsf{U} \, \Phi)$$

PCTL

sublogic of PCTL* where only path formulas of the form $\bigcirc\Phi$ and $\Phi_1 \,\mathsf{U}\, \Phi_2$ are allowed

state formulas:

$$\Phi \quad ::= \quad \mathit{true} \;\big|\; a \;\big|\; \Phi_1 \wedge \Phi_2 \;\big|\; \neg\Phi \;\big|\; \mathbb{P}_I(\varphi)$$

path formulas:

$$\varphi \quad ::= \quad \bigcirc\Phi \;\big|\; \Phi_1 \,\mathsf{U}\, \Phi_2 \;\big|\; \Diamond\Phi \;\big|\; \Box\Phi$$

$$\mathbb{P}_I(\Diamond\Phi) \;\stackrel{\text{def}}{=}\; \mathbb{P}_I(\mathit{true}\,\mathsf{U}\,\Phi)$$

e.g., $\mathbb{P}_{<0.4}(\Box\Phi) \;\stackrel{\text{def}}{=}\; \mathbb{P}_{>0.6}(\Diamond\neg\Phi)$

note: $\Pr^{\mathcal{M}}(s, \Box\Phi) \;=\; 1 - \Pr^{\mathcal{M}}(s, \Diamond\neg\Phi)$

PCTL model checking

given: Markov chain $\mathcal{M} = (S, P, AP, L, s_0)$
PCTL state formula Φ

task: check whether $s_0 \models \Phi$

PCTL model checking

given: Markov chain $\mathcal{M} = (S, P, AP, L, s_0)$
PCTL state formula Φ

task: check whether $s_0 \models \Phi$

recursive computation of $Sat(\Psi) = \{ s \in S : s \models \Psi \}$
for all state subformulas Ψ of Φ

in bottom-up manner, i.e.,
inner subformulas first

PCTL model checking

given: Markov chain $\mathcal{M} = (S, P, AP, L, s_0)$
PCTL state formula Φ

task: check whether $s_0 \models \Phi$

recursive computation of $Sat(\Psi) = \{ s \in S : s \models \Psi \}$
for all state subformulas Ψ of Φ

- treatment of propositional logic fragment:

PCTL model checking

given: Markov chain $\mathcal{M} = (S, P, AP, L, s_0)$
PCTL state formula Φ

task: check whether $s_0 \models \Phi$

recursive computation of $Sat(\Psi) = \{s \in S : s \models \Psi\}$
for all state subformulas Ψ of Φ

- treatment of propositional logic fragment: $\surd$

$$
\begin{aligned}
Sat(\mathit{true}) &= S \\
Sat(a) &= \{s \in S : a \in L(s)\} \\
Sat(\neg\Psi) &= S \setminus Sat(\Psi) \\
Sat(\Psi_1 \wedge \Psi_2) &= Sat(\Psi_1) \cap Sat(\Psi_2)
\end{aligned}
$$

PCTL model checking

given: Markov chain $\mathcal{M} = (S, P, AP, L, s_0)$
PCTL state formula Φ

task: check whether $s_0 \models \Phi$

recursive computation of $Sat(\Psi) = \{s \in S : s \models \Psi\}$
for all state subformulas Ψ of Φ

- treatment of propositional logic fragment: $\surd$
- treatment of the probability operator $\mathbb{P}_{\mathrm{I}}(\varphi)$

PCTL model checking

given: Markov chain $\mathcal{M} = (S, P, AP, L, s_0)$
PCTL state formula Φ

task: check whether $s_0 \models \Phi$

recursive computation of $Sat(\Psi) = \{ s \in S : s \models \Psi \}$
for all state subformulas Ψ of Φ

- treatment of propositional logic fragment: $\surd$
- treatment of the probability operator $\mathbb{P}_I(\varphi)$

compute $\Pr_s^{\mathcal{M}}(\varphi)$ for all states s and return

$$Sat(\mathbb{P}_I(\varphi)) = \{ s \in S : \Pr_s^{\mathcal{M}}(\varphi) \in I \}$$

PCTL model checking

given: Markov chain $\mathcal{M} = (S, P, AP, L, s_0)$
PCTL state formula Φ

task: check whether $s_0 \models \Phi$

recursive computation of $Sat(\Psi) = \{s \in S : s \models \Psi\}$
for all state subformulas Ψ of Φ

- treatment of propositional logic fragment: $\surd$
- treatment of the probability operator $\mathbb{P}_I(\varphi)$

graph algorithms + matrix/vector operations

↑

next:	matrix/vector multiplication
until:	linear equation system

PCTL model checking

given: Markov chain $\mathcal{M} = (S, P, AP, L, s_0)$
PCTL state formula Φ

task: check whether $s_0 \models \Phi$

recursive computation of $Sat(\Psi) = \{s \in S : s \models \Psi\}$
for all state subformulas Ψ of Φ

- treatment of propositional logic fragment: $\surd$
- treatment of the probability operator $\mathbb{P}_{\mathrm{I}}(\varphi)$

  graph algorithms + matrix/vector operations

time complexity: $\mathcal{O}\big(\,\mathbf{poly}(\mathcal{M}) \cdot |\Phi|\,\big)$

PCTL* model checking

given: Markov chain $\mathcal{M} = (S, P, AP, L, s_0)$
PCTL* state formula Φ

task: check whether $s_0 \models \Phi$

recursive computation of $Sat(\Psi) = \{ s \in S : s \models \Psi \}$
for all state subformulas Ψ of Φ

- treatment of propositional logic fragment: $\surd$
- treatment of the probability operator $\mathbb{P}_{\mathrm{I}}(\varphi)$

PCTL* path formula φ $\rightsquigarrow$ LTL formula φ'
$\uparrow$
path formula without probability operator

PCTL* model checking

given: Markov chain $\mathcal{M} = (S, P, AP, L, s_0)$
PCTL* state formula Φ

task: check whether $s_0 \models \Phi$

recursive computation of $Sat(\Psi) = \{s \in S : s \models \Psi\}$
for all state subformulas Ψ of Φ

- treatment of propositional logic fragment: $\surd$
- treatment of the probability operator $\mathbb{P}_I(\varphi)$

PCTL* path formula $\varphi \rightsquigarrow$ LTL formula φ'

... automata-based approach for φ' ...

PCTL* model checking

given: Markov chain $\mathcal{M} = (S, P, AP, L, s_0)$
PCTL* state formula Φ

task: check whether $s_0 \models \Phi$

treatment of the probability operator $\mathbb{P}_I(\varphi)$

PCTL* path formula $\varphi \rightsquigarrow$ LTL formula φ'

by replacing each maximal state subformula
with a fresh atomic proposition

PCTL* model checking

given: Markov chain $\mathcal{M} = (S, P, AP, L, s_0)$
PCTL* state formula Φ

task: check whether $s_0 \models \Phi$

treatment of the probability operator $\mathbb{P}_I(\varphi)$

PCTL* path formula $\varphi \rightsquigarrow$ LTL formula φ'

by replacing each maximal state subformula
with a fresh atomic proposition

$$\Diamond\big(a \, \mathsf{U} \, \mathbb{P}_{\geqslant 0.7}(\Box\Diamond b) \;\wedge \Box\; \mathbb{P}_{<0.3}(\bigcirc\Box c)\;\big)$$

PCTL* model checking

given: Markov chain $\mathcal{M} = (S, P, AP, L, s_0)$
PCTL* state formula Φ

task: check whether $s_0 \models \Phi$

treatment of the probability operator $\mathbb{P}_I(\varphi)$

PCTL* path formula $\varphi \rightsquigarrow$ LTL formula φ'

by replacing each maximal state subformula with a fresh atomic proposition

$$\Diamond\big(a \,\mathsf{U}\, \boxed{\mathbb{P}_{\geqslant 0.7}(\Box\Diamond b)} \wedge \Box\, \boxed{\mathbb{P}_{<0.3}(\bigcirc\Box c)}\,\big)$$

PCTL* model checking

given: Markov chain $\mathcal{M} = (S, P, AP, L, s_0)$
PCTL* state formula Φ

task: check whether $s_0 \models \Phi$

treatment of the probability operator $\mathbb{P}_{\mathrm{I}}(\varphi)$

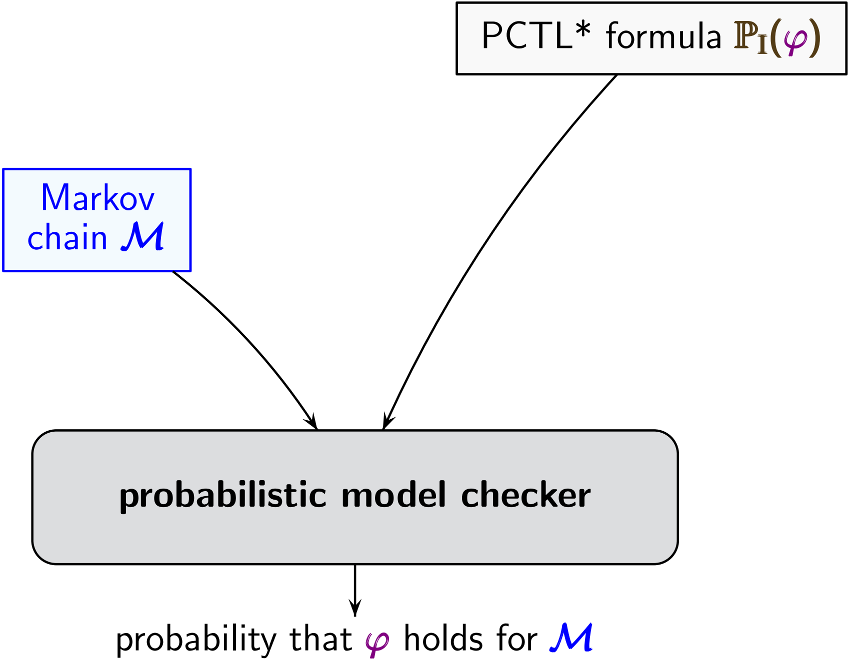
PCTL* path formula $\varphi \rightsquigarrow$ LTL formula φ'

by replacing each maximal state subformula
with a fresh atomic proposition

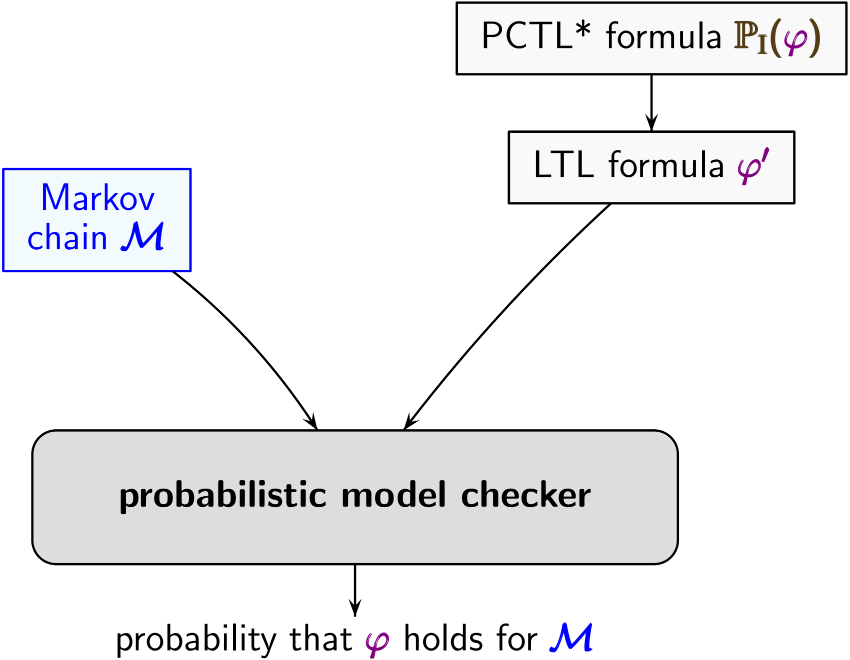
$$\Diamond\big(a \,\mathsf{U}\, \mathbb{P}_{\geqslant 0.7}(\Box\Diamond b) \wedge \Box\, \mathbb{P}_{<0.3}(\bigcirc\Box c)\,\big)$$

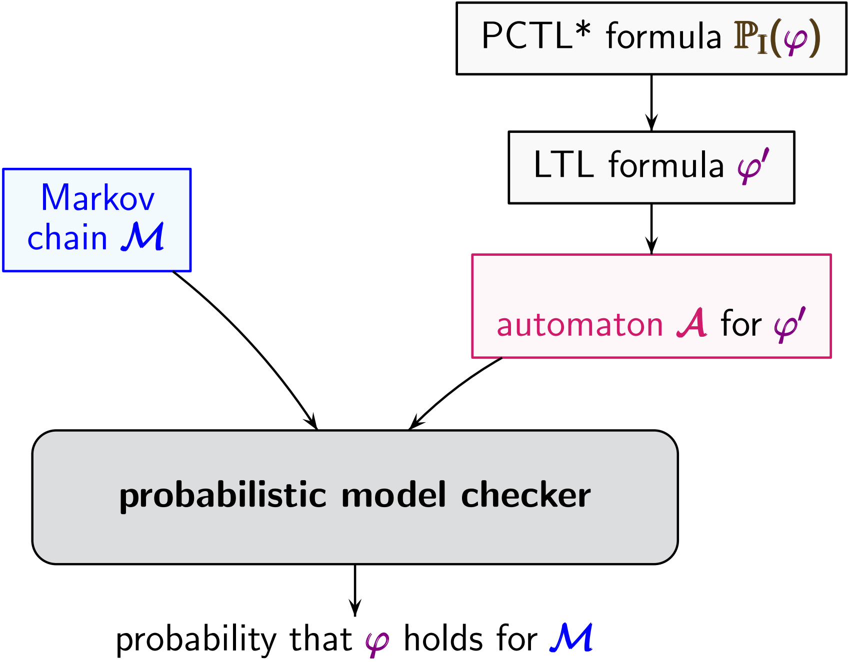
$$\Downarrow$$

$$\Diamond\big(\, a \,\mathsf{U}\, d \;\wedge\; \Box\, e\,\big)$$

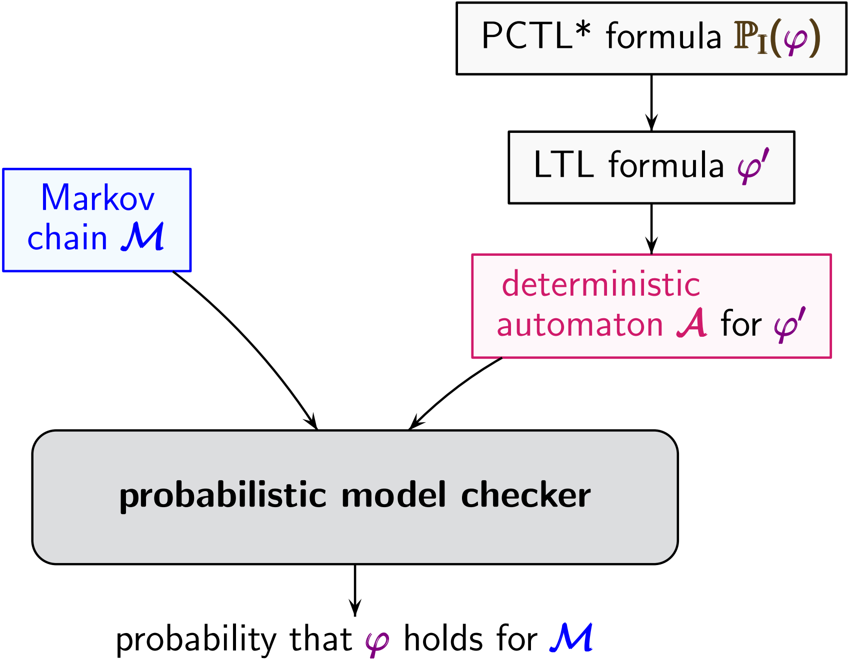


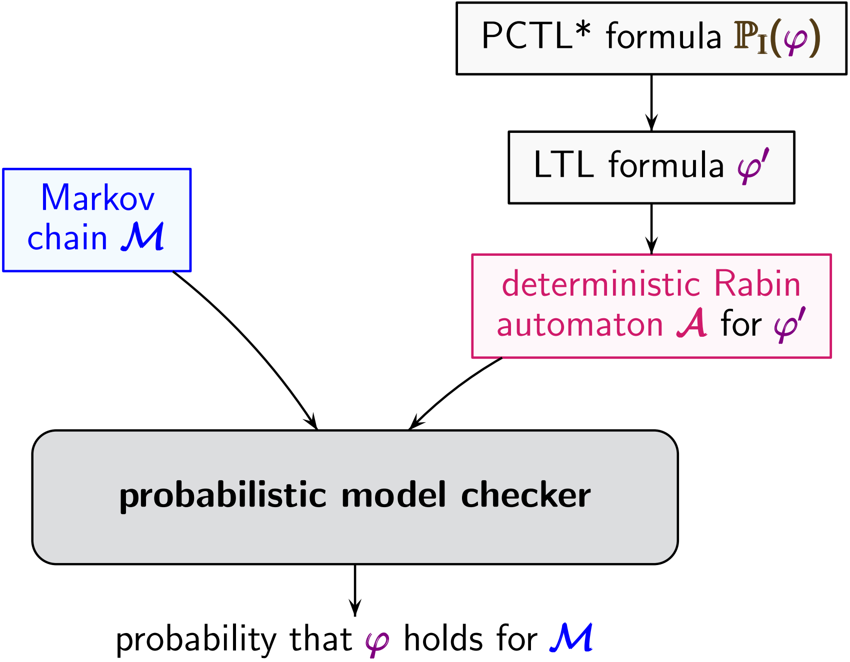
PCTL* formula $\mathbb{P}_{\mathbf{I}}(\varphi)$
Markov chain $\mathcal{M}$
probabilistic model checker
probability that φ holds for $\mathcal{M}$


PCTL* formula $\mathbb{P}_I(\varphi)$
LTL formula φ'
Markov chain $\mathcal{M}$
probabilistic model checker
probability that φ holds for $\mathcal{M}$



PCTL* formula $\mathbb{P}_I(\varphi)$
LTL formula φ'
Markov chain $\mathcal{M}$
automaton $\mathcal{A}$ for φ'
probabilistic model checker
probability that φ holds for $\mathcal{M}$


PCTL* formula $\mathbb{P}_I(\varphi)$
LTL formula φ'
Markov chain $\mathcal{M}$
deterministic automaton $\mathcal{A}$ for φ'
probabilistic model checker
probability that φ holds for $\mathcal{M}$


PCTL* formula $\mathbb{P}_I(\varphi)$
LTL formula φ'
Markov chain $\mathcal{M}$
deterministic Rabin automaton $\mathcal{A}$ for φ'
probabilistic model checker
probability that φ holds for $\mathcal{M}$

Deterministic Rabin automata (DRA)

Deterministic Rabin automata (DRA)

A DRA is a tuple $\mathcal{A} = (Q, \Sigma, \delta, q_0, Acc)$ where

- Q finite state space
- $q_0 \in Q$ initial state
- Σ alphabet
- $\delta : Q \times \Sigma \longrightarrow Q$ deterministic transition function

Deterministic Rabin automata (DRA)

A DRA is a tuple $\mathcal{A} = (Q, \Sigma, \delta, q_0, Acc)$ where

- Q finite state space
- $q_0 \in Q$ initial state
- Σ alphabet
- $\delta : Q \times \Sigma \longrightarrow Q$ deterministic transition function
- acceptance condition Acc is a set of pairs (L, U) with $L, U \subseteq Q$

Deterministic Rabin automata (DRA)

A DRA is a tuple $\mathcal{A} = (Q, \Sigma, \delta, q_0, Acc)$ where

- Q finite state space
- $q_0 \in Q$ initial state
- Σ alphabet
- $\delta : Q \times \Sigma \longrightarrow Q$ deterministic transition function
- acceptance condition Acc is a set of pairs (L, U) with $L, U \subseteq Q$, say $Acc = \{(L_1, U_1), ..., (L_k, U_k)\}$

semantics of the acceptance condition:

$$\bigvee_{1 \leqslant i \leqslant k} (\Diamond\Box\neg L_i \ \wedge \ \Box\Diamond U_i)$$

Accepted language of a DRA

A DRA is a tuple $\mathcal{A} = (Q, \Sigma, \delta, q_0, Acc)$ where

$$Acc = \{(L_1, U_1), \ldots, (L_k, U_k)\} \qquad L_i, U_i \subseteq Q$$

accepted language:

$$\mathcal{L}_\omega(\mathcal{A}) = \{\ \sigma \in \Sigma^\omega : \text{ the run for } \sigma \text{ in } \mathcal{A} \text{ fulfills } Acc\ \}$$

Accepted language of a DRA

A DRA is a tuple $\mathcal{A} = (Q, \Sigma, \delta, q_0, Acc)$ where

$$Acc \;=\; \{(L_1, U_1), \ldots, (L_k, U_k)\} \qquad L_i, U_i \subseteq Q$$

accepted language:

$$\mathcal{L}_\omega(\mathcal{A}) = \{\, \sigma \in \Sigma^\omega : \text{ the run for } \sigma \text{ in } \mathcal{A} \text{ fulfills } Acc \,\}$$

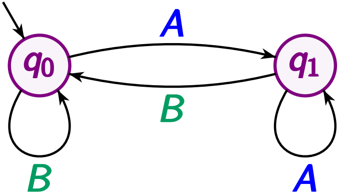
Let $\rho = q_0\, q_1\, q_2 \ldots$ be the run for some infinite word σ.

ρ fulfills Acc iff

$$\exists i \in \{1, \ldots, k\}.\ \inf(\rho) \cap L_i = \varnothing \wedge \inf(\rho) \cap U_i \neq \varnothing$$

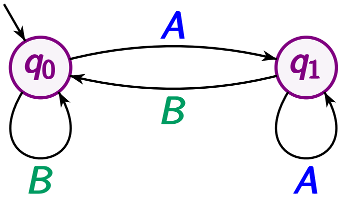
where $\inf(\rho) \;=\; \{q \in Q : \overset{\infty}{\exists}\, \ell \in \mathbb{N}.\ q = q_\ell\}$

Example: DRA



$$Acc = \{(\{q_0\}, \{q_1\})\}$$

Example: DRA



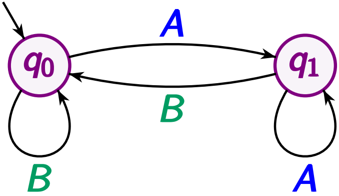
$$Acc = \{(\{q_0\}, \{q_1\})\}$$

$$\hat{=} \Diamond\Box\neg q_0 \ \wedge\ \Box\Diamond q_1$$

$\Diamond\Box$ "eventually forever"

$\Box\Diamond$ "infinitely often"

Example: DRA



$Acc = \{(\{q_0\}, \{q_1\})\}$

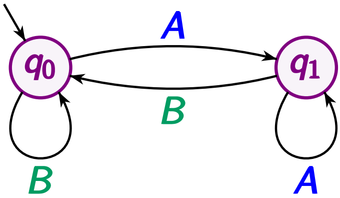
$\mathrel{\widehat{=}} \Diamond\Box\neg q_0 \;\wedge\; \Box\Diamond q_1$

accepted language: $(A + B)^* A^\omega$

$\Diamond\Box$ "eventually forever"

$\Box\Diamond$ "infinitely often"

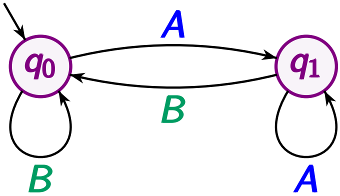
Example: DRA



$Acc = \{(\{q_0\}, \{q_1\})\}$

$\widehat{=}\ \Diamond\Box\neg q_0 \ \wedge\ \Box\Diamond q_1$

accepted language: $(A + B)^* A^\omega$

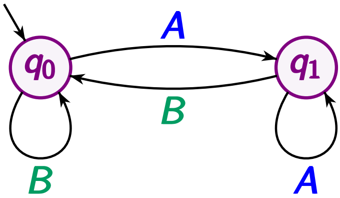


$Acc = \{(\varnothing, \{q_1\})\}$

$\widehat{=}\ \Box\Diamond q_1$

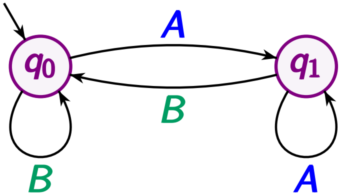
$\Box\Diamond$ "infinitely often"

Example: DRA



$$Acc = \{(\{q_0\}, \{q_1\})\}$$
$$\widehat{=}\ \Diamond\Box\neg q_0 \ \wedge\ \Box\Diamond q_1$$

accepted language: $(A + B)^* A^\omega$



$$Acc = \{(\varnothing, \{q_1\})\}$$
$$\widehat{=}\ \Box\Diamond q_1$$

accepted language: $(B^* A)^\omega$

Fundamental result: LTL-2-DRA

Fundamental result: LTL-2-DRA

For each LTL formula φ over AP there exists a DRA $\mathcal{A}$ with the alphabet $\Sigma = 2^{AP}$ s.t.

$$\mathcal{L}_\omega(\mathcal{A}) \;=\; \big\{\sigma \in \Sigma^\omega : \sigma \models \varphi\big\}$$

Fundamental result: LTL-2-DRA

For each LTL formula φ over AP there exists a DRA $\mathcal{A}$ with the alphabet $\Sigma = 2^{AP}$ s.t.

$$\mathcal{L}_\omega(\mathcal{A}) \;=\; \{\sigma \in \Sigma^\omega : \sigma \models \varphi\}$$

LTL formula

↓

NBA

↓ determinization [SAFRA'88]

DRA

Fundamental result: LTL-2-DRA

For each LTL formula φ over AP there exists a DRA $\mathcal{A}$ with the alphabet $\Sigma = 2^{AP}$ s.t.

$$\mathcal{L}_\omega(\mathcal{A}) \;=\; \{\sigma \in \Sigma^\omega : \sigma \models \varphi\}$$

LTL formula

↓

NBA

↓ determinization [SAFRA'88]

DRA

LTL formula

↓ compositional [ESPARZA/KRETINSKY'14]

DRA

Fundamental result: LTL-2-DRA

For each LTL formula φ over AP there exists a DRA $\mathcal{A}$ with the alphabet $\Sigma = 2^{AP}$ s.t.

$$\mathcal{L}_\omega(\mathcal{A}) \;=\; \{\sigma \in \Sigma^\omega : \sigma \models \varphi\}$$

Example: $AP = \{a, b\}$

Fundamental result: LTL-2-DRA

For each LTL formula φ over AP there exists a DRA $\mathcal{A}$ with the alphabet $\Sigma = 2^{AP}$ s.t.

$$\mathcal{L}_\omega(\mathcal{A}) \;=\; \{\sigma \in \Sigma^\omega : \sigma \models \varphi\}$$

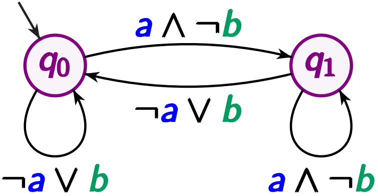
Example: $AP = \{a, b\} \;\rightsquigarrow\; \Sigma = \Big\{\varnothing, \{a\}, \{b\}, \{a, b\}\Big\}$

Fundamental result: LTL-2-DRA

For each LTL formula φ over AP there exists a DRA $\mathcal{A}$ with the alphabet $\Sigma = 2^{AP}$ s.t.

$$\mathcal{L}_\omega(\mathcal{A}) \;=\; \big\{\sigma \in \Sigma^\omega : \sigma \models \varphi\big\}$$

Example: $AP = \{a, b\} \;\rightsquigarrow\; \Sigma = \Big\{\varnothing, \{a\}, \{b\}, \{a, b\}\Big\}$



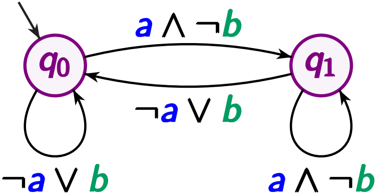
acceptance condition:
$\Diamond\Box\neg q_0 \;\wedge\; \Box\Diamond q_1$

Fundamental result: LTL-2-DRA

For each LTL formula φ over AP there exists a DRA $\mathcal{A}$ with the alphabet $\Sigma = 2^{AP}$ s.t.

$$\mathcal{L}_\omega(\mathcal{A}) \;=\; \{\sigma \in \Sigma^\omega : \sigma \models \varphi\}$$

Example: $AP = \{a, b\} \;\rightsquigarrow\; \Sigma = \Big\{\varnothing, \{a\}, \{b\}, \{a, b\}\Big\}$



acceptance condition:
$\Diamond\Box\neg q_0 \;\wedge\; \Box\Diamond q_1$

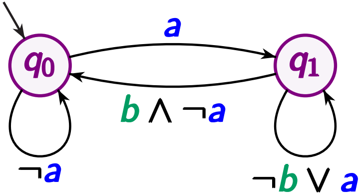
LTL formula $\Diamond\Box(a \wedge \neg b)$

Fundamental result: LTL-2-DRA

For each LTL formula φ over AP there exists a DRA $\mathcal{A}$ with the alphabet $\Sigma = 2^{AP}$ s.t.

$$\mathcal{L}_\omega(\mathcal{A}) \;=\; \{\sigma \in \Sigma^\omega : \sigma \models \varphi\}$$

Example: $AP = \{a, b\} \;\rightsquigarrow\; \Sigma = \big\{\varnothing, \{a\}, \{b\}, \{a, b\}\big\}$



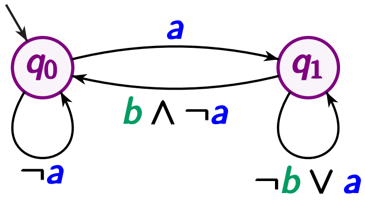
acceptance condition:
$\Diamond\Box\neg q_1 \;\wedge\; \Box\Diamond q_0$

Fundamental result: LTL-2-DRA

For each LTL formula φ over AP there exists a DRA $\mathcal{A}$ with the alphabet $\Sigma = 2^{AP}$ s.t.

$$\mathcal{L}_\omega(\mathcal{A}) \;=\; \{\sigma \in \Sigma^\omega : \sigma \models \varphi\}$$

Example: $AP = \{a, b\} \;\rightsquigarrow\; \Sigma = \Big\{\varnothing, \{a\}, \{b\}, \{a, b\}\Big\}$

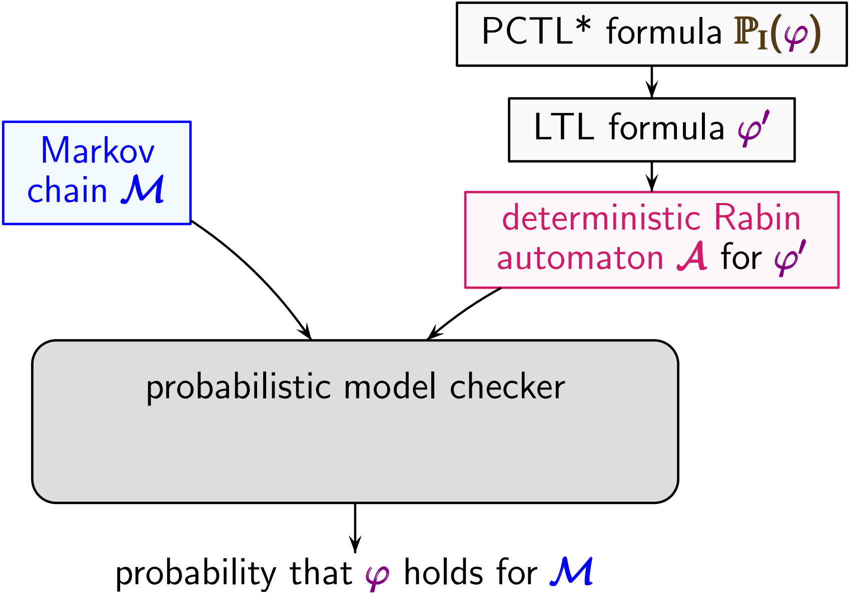


acceptance condition:
$\Diamond\Box\neg q_1 \;\wedge\; \Box\Diamond q_0$

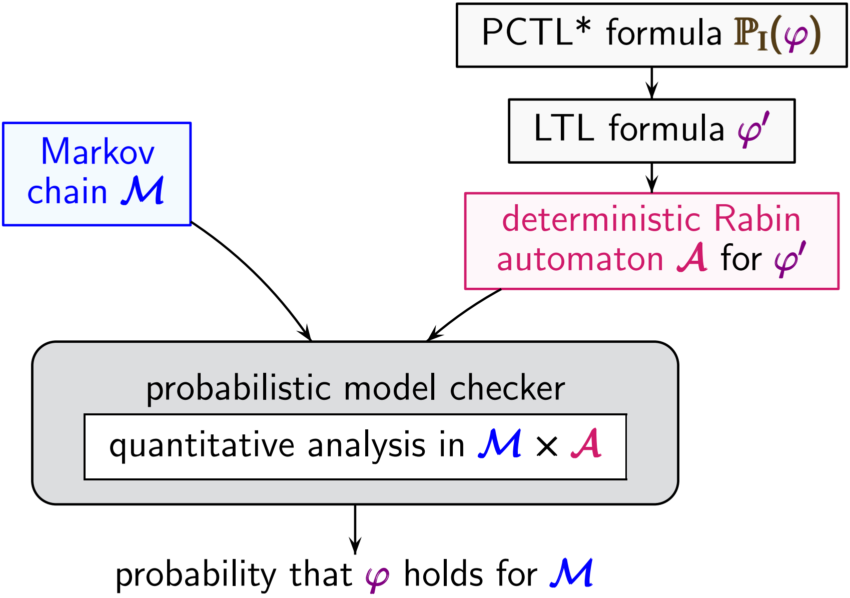
LTL formula
$\Box(a \rightarrow \Diamond(b \wedge \neg a)) \;\wedge\; \Diamond\Box\neg a$

PCTL* model checking

PCTL* model checking



PCTL* model checking



Product of a Markov chain and a DRA

Product of a Markov chain and a DRA

given: Markov chain $\mathcal{M} = (S, P, AP, L)$

DRA $\mathcal{A} = (Q, 2^{AP}, \delta, q_0, Acc)$

goal: define a Markov chain $\mathcal{M} \times \mathcal{A}$

Product of a Markov chain and a DRA

given: Markov chain $\mathcal{M} = (S, P, AP, L)$

DRA $\mathcal{A} = (Q, 2^{AP}, \delta, q_0, Acc)$

goal: define a Markov chain $\mathcal{M} \times \mathcal{A}$ s.t.

$$\Pr_s^{\mathcal{M}}(\mathcal{A}) \;=\; \Pr^{\mathcal{M}}\{\pi \in Paths(s) : trace(\pi) \in \mathcal{L}_\omega(\mathcal{A})\}$$

can be derived by a probabilistic reachability analysis in the product-chain $\mathcal{M} \times \mathcal{A}$

$$trace(s_0\, s_1\, s_2 \ldots) \;=\; L(s_0)\; L(s_1)\; L(s_2) \ldots \in \left(2^{AP}\right)^{\omega}$$

Product of a Markov chain and a DRA

given: Markov chain $\mathcal{M} = (S, P, AP, L)$

DRA $\mathcal{A} = (Q, 2^{AP}, \delta, q_0, Acc)$

idea: define a Markov chain $\mathcal{M} \times \mathcal{A}$ s.t. ...

path π in $\mathcal{M}$

$$s_0 \downarrow s_1 \downarrow s_2 \downarrow \vdots$$

Product of a Markov chain and a DRA

given: Markov chain $\mathcal{M} = (S, P, AP, L)$

DRA $\mathcal{A} = (Q, 2^{AP}, \delta, q_0, Acc)$

idea: define a Markov chain $\mathcal{M} \times \mathcal{A}$ s.t. ...

path π in $\mathcal{M}$

$$s_0 \downarrow s_1 \downarrow s_2 \downarrow \vdots$$

run for $trace(\pi)$ in $\mathcal{A}$

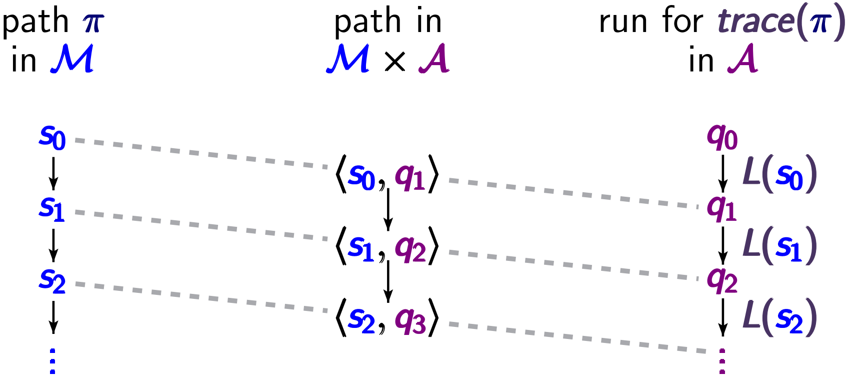
$$q_0 \xrightarrow{L(s_0)} q_1 \xrightarrow{L(s_1)} q_2 \xrightarrow{L(s_2)} \vdots$$

Product of a Markov chain and a DRA

given: Markov chain $\mathcal{M} = (S, P, AP, L)$

DRA $\mathcal{A} = (Q, 2^{AP}, \delta, q_0, Acc)$

idea: define a Markov chain $\mathcal{M} \times \mathcal{A}$ s.t. ...



Fundamental property of the product

Fundamental property of the product

given: Markov chain $\mathcal{M}$ and DRA $\mathcal{A}$ where

$$Acc = \{ (L_1, U_1), (L_2, U_2), \ldots, (L_k, U_k) \}$$

Fundamental property of the product

given: Markov chain $\mathcal{M}$ and DRA $\mathcal{A}$ where

$$Acc = \{ (L_1, U_1), (L_2, U_2), \ldots, (L_k, U_k) \}$$

For each state s in $\mathcal{M}$, let $q_s = \delta\big(q_0, L(s)\big)$.

↑

successor state in $\mathcal{A}$ of the initial DRA-state q_0 for the input symbol $L(s) \in 2^{AP}$

Fundamental property of the product

given: Markov chain $\mathcal{M}$ and DRA $\mathcal{A}$ where

$$Acc = \{ (L_1, U_1), (L_2, U_2), \ldots, (L_k, U_k) \}$$

For each state s in $\mathcal{M}$, let $q_s = \delta\big(q_0, L(s)\big)$.

$\Pr_s^{\mathcal{M}}(\mathcal{A})$

probability measure of all paths $\pi \in Paths^{\mathcal{M}}(s)$ such that $trace(\pi) \in \mathcal{L}_\omega(\mathcal{A})$

Fundamental property of the product

given: Markov chain $\mathcal{M}$ and DRA $\mathcal{A}$ where

$$Acc = \{ (L_1, U_1), (L_2, U_2), \ldots, (L_k, U_k) \}$$

For each state s in $\mathcal{M}$, let $q_s = \delta\big(q_0, L(s)\big)$.

$$\Pr_s^{\mathcal{M}}(\mathcal{A})$$

$$= \Pr_{\langle s, q_s \rangle}^{\mathcal{M} \times \mathcal{A}}\Big(\bigvee_{1 \leqslant i \leqslant k} (\Diamond\Box\neg L_i \wedge \Box\Diamond U_i) \Big)$$

probability measure of all paths π in the product s.t. $\pi\big|_{\mathcal{A}}$ satisfies the acceptance condition of $\mathcal{A}$

Fundamental property of the product

given: Markov chain $\mathcal{M}$ and DRA $\mathcal{A}$ where

$$Acc = \{ (L_1, U_1), (L_2, U_2), \ldots, (L_k, U_k) \}$$

For each state s in $\mathcal{M}$, let $q_s = \delta(q_0, L(s))$.

$$\begin{aligned} & \Pr_s^{\mathcal{M}}(\mathcal{A}) \\ = \; & \Pr_{\langle s, q_s \rangle}^{\mathcal{M} \times \mathcal{A}}\Big(\bigvee_{1 \leqslant i \leqslant k} (\Diamond\Box\neg L_i \wedge \Box\Diamond U_i) \Big) \\ = \; & \Pr_{\langle s, q_s \rangle}^{\mathcal{M} \times \mathcal{A}}(\Diamond \, accBSCC) \end{aligned}$$

Fundamental property of the product

given: Markov chain $\mathcal{M}$ and DRA $\mathcal{A}$ where

$$Acc = \{ (L_1, U_1), (L_2, U_2), \ldots, (L_k, U_k) \}$$

For each state s in $\mathcal{M}$, let $q_s = \delta\big(q_0, L(s)\big)$.

$$\Pr_s^{\mathcal{M}}(\mathcal{A})$$

$$= \Pr_{\langle s, q_s \rangle}^{\mathcal{M} \times \mathcal{A}}(\lozenge\, accBSCC)$$

↑

union of accepting BSCCs in $\mathcal{M} \times \mathcal{A}$ i.e., BSCC C s.t.

$$\exists i \in \{1, \ldots, k\}. \;\; C \cap L_i = \varnothing \;\; \wedge \;\; C \cap U_i \neq \varnothing$$

Summary: PCTL* model checking

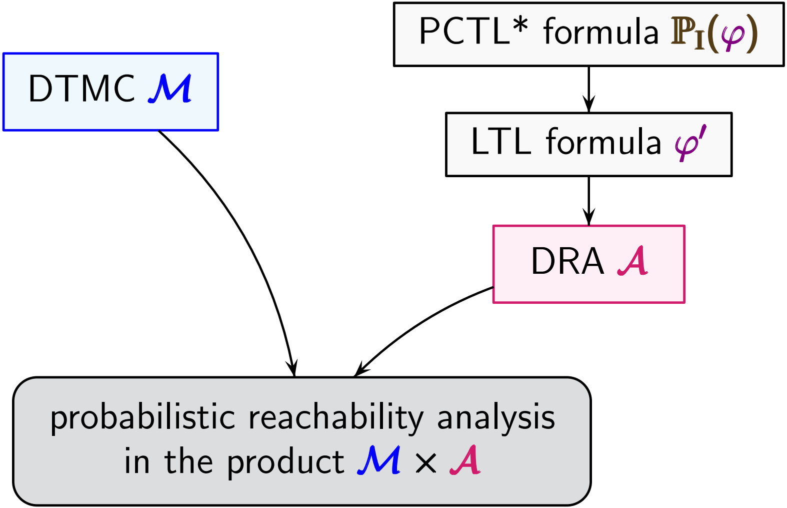
given: Markov chain $\mathcal{M} = (S, P, AP, L, s_0)$
PCTL* state formula Φ

task: check whether $\mathcal{M} \models \Phi$

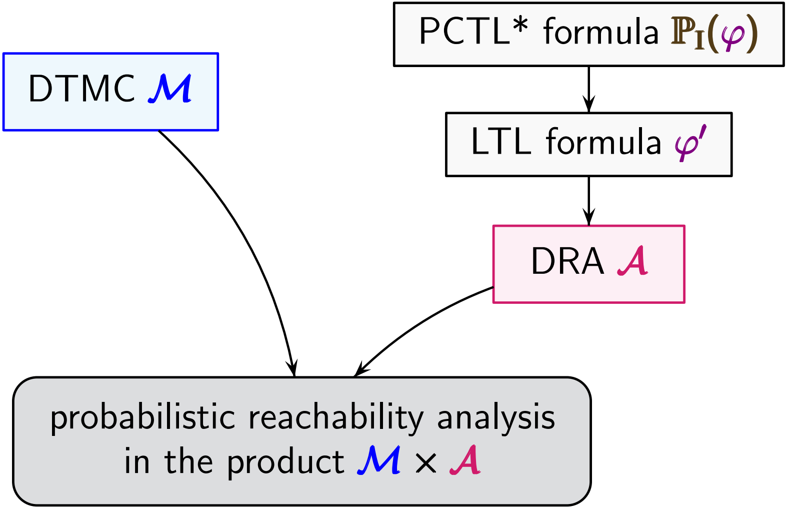
method: bottom-up treatment of state subformulas Ψ to compute

$$Sat(\Psi) = \{ s \in S : s \models \Psi \}$$

- propositional logic fragment: obvious
- probability operator $\mathbb{P}_{\mathrm{I}}(\varphi)$ via
  - construction of a DRA $\mathcal{A}$ for φ
  - probabilistic reachability analysis in $\mathcal{M} \times \mathcal{A}$



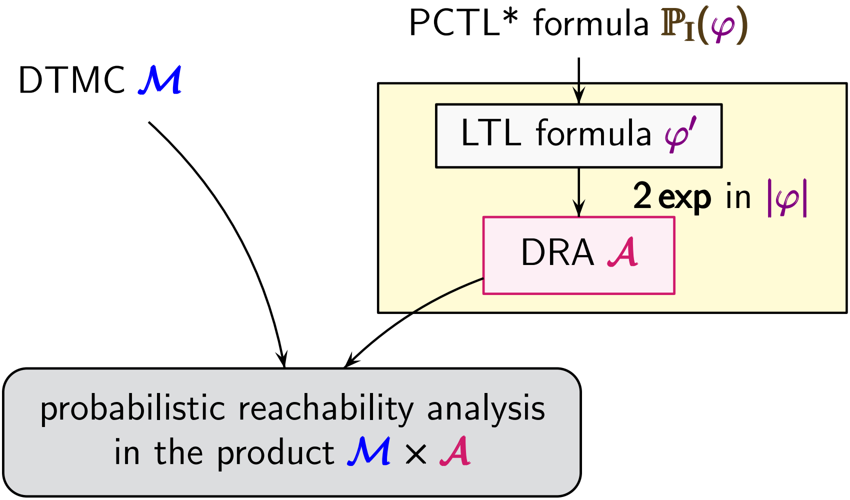
1. graph analysis to compute the accepting BSCCs of the product
2. linear equation system for the probabilities to reach an accepting BSCC



1. graph analysis to compute the accepting BSCCs of the product
2. linear equation system

time complexity:

polynomial in the sizes of $\mathcal{M}$ and $\mathcal{A}$



1. graph analysis to compute the accepting BSCCs of the product
2. linear equation system

time complexity:

polynomial in the sizes of $\mathcal{M}$ and $\mathcal{A}$

Exponential-time algorithms for DTMC and LTL

Exponential-time algorithms for DTMC and LTL

given: Markov chain $\mathcal{M}$, LTL formula φ

task: compute $\Pr^{\mathcal{M}}(\varphi)$

single exponential-time algorithms:

- iterative, automata-less approach

  [Courcoubetis/Yannakakis'88]

- using weak alternating automata

  [Bustan/Rubin/Vardi'04]

- using separated Büchi automata

  [Couvreur/Saheb/Sutre'03]

- using unambiguous Büchi automata

  [Baier/Kiefer/Klein/Klüppelholz/Müller/Worrell'16]

Tutorial: Probabilistic Model Checking

Discrete-time Markov chains (DTMC)

- basic definitions
- probabilistic computation tree logic PCTL/PCTL*
- rewards, cost-utility ratios, weights
- conditional probabilities

Markov decision processes (MDP)

- basic definitions
- PCTL/PCTL* model checking
- fairness
- conditional probabilities
- rewards, quantiles
- mean-payoff
- expected accumulated weights

Markov reward model (MRM)

Markov reward model (MRM)

Markov chain $\mathcal{M} = (S, P, AP, L, rew)$ with a reward function for the states:

$$rew : S \to \mathbb{N}$$

idea: reward $rew(s)$ will be earned when leaving s

analogously: rewards for edges $rew : S \times S \to \mathbb{N}$

Markov reward model (MRM)

Markov chain $\mathcal{M} = (S, P, AP, L, rew)$ with a reward function for the states:

$$rew : S \to \mathbb{N}$$

idea: reward $rew(s)$ will be earned when leaving s

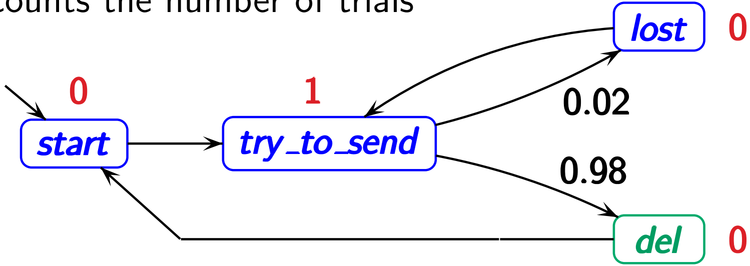
formalization by accumulated rewards of finite paths

$$rew(s_0\, s_1 \ldots s_n) \quad = \quad \sum_{0 \leqslant i < n} rew(s_i)$$

analogously: rewards for edges $rew : S \times S \to \mathbb{N}$

Example: Markov reward model

communication protocol with reward function that counts the number of trials

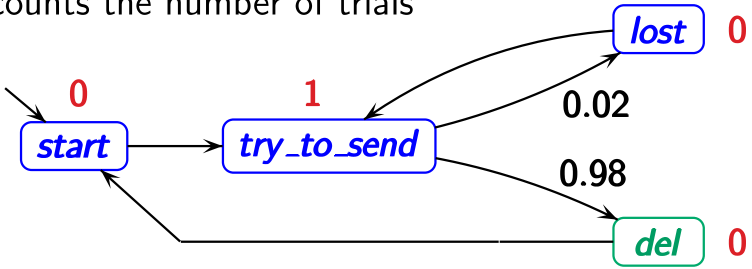


accumulated reward of finite paths, e.g.,

$$rew(start\ try\ lost\ try\ del) = 2$$

Example: Markov reward model

communication protocol with reward function that counts the number of trials



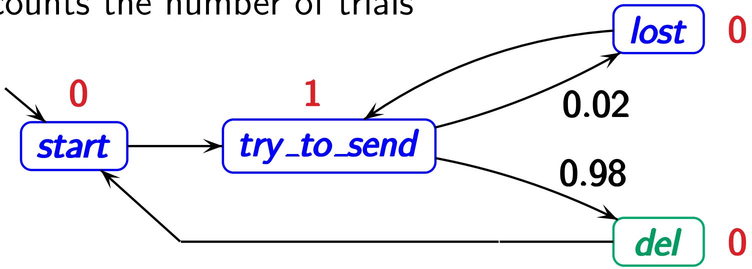
measures of interest, e.g.,

$\Pr^{\mathcal{M}}(\lozenge^{\leqslant 3} \mathit{del})$ probability to deliver a message within at most three trials

reachability with reward bound $\leqslant 3$

Example: Markov reward model

communication protocol with reward function that counts the number of trials



measures of interest, e.g.,

$\Pr^{\mathcal{M}}(\Diamond^{\leqslant 3} del)$ probability to deliver a message within at most three trials

$\mathbb{E}(\oplus\, del)$ expected number of trials until delivered

Reward-based extension of PCTL

Reward-based extension of PCTL

probability operator for reward-bounded path formulas:

$\mathbb{P}_I(\Phi_1 \mathsf{U}^{\leqslant r} \Phi_2)$ until with upper reward bound

Reward-based extension of PCTL

probability operator for reward-bounded path formulas:

$\mathbb{P}_{\mathrm{I}}(\,\Phi_1\, \mathsf{U}^{\leqslant r}\, \Phi_2\,)$ until with upper reward bound

expected accumulated reward operator: $\mathbb{E}_{\leqslant r}(\lozenge \Phi)$

$s \models \mathbb{E}_{\leqslant r}(\lozenge \Phi)$ iff $\begin{cases} \text{expected accumulated reward on} \\ \text{paths from } s \text{ to a } \Phi\text{-state is} \leqslant r \end{cases}$

Reward-based extension of PCTL

probability operator for reward-bounded path formulas:

$\mathbb{P}_I(\Phi_1 \mathsf{U}^{\leqslant r} \Phi_2)$ until with upper reward bound

expected accumulated reward operator: $\mathbb{E}_{\leqslant r}(\Diamond\!\!\!\!+ \Phi)$

$s \models \mathbb{E}_{\leqslant r}(\Diamond\!\!\!\!+ \Phi)$ iff $\left\{ \begin{array}{l} \text{expected accumulated reward on} \\ \text{paths from } s \text{ to a } \Phi\text{-state is } \leqslant r \end{array} \right.$

example: communication protocol

$\mathbb{P}_{\geqslant 0.9}(\Diamond^{\leqslant 3} del)$ probability for delivering the message within at most three trials is at least 0.9

$\mathbb{E}_{\leqslant 5}(\Diamond\!\!\!\!+ del)$ average number of trials is less or equal 5

Model checking reward-based properties

Model checking reward-based properties

treatment of $\mathbb{P}_I(\Phi_1 \, \mathsf{U}^{\leqslant r} \, \Phi_2)$ where $r \in \mathbb{N}$

compute $\Pr_s^{\mathcal{M}}(\Phi_1 \, \mathsf{U}^{\leqslant i} \, \Phi_2)$ iteratively
for increasing reward bound $i = 0, 1, 2, \ldots, r$

Model checking reward-based properties

treatment of $\mathbb{P}_{\mathrm{I}}(\Phi_1 \, \mathsf{U}^{\leqslant r} \, \Phi_2)$ where $r \in \mathbb{N}$

compute $\mathrm{Pr}_s^{\mathcal{M}}(\Phi_1 \, \mathsf{U}^{\leqslant i} \, \Phi_2)$ iteratively
for increasing reward bound $i = 0, 1, 2, \ldots, r$

Let $x_{s,i} = \mathrm{Pr}_s^{\mathcal{M}}(\Phi_1 \, \mathsf{U}^{\leqslant i} \, \Phi_2)$. Then:

Model checking reward-based properties

treatment of $\mathbb{P}_I(\Phi_1 \, \mathsf{U}^{\leqslant r} \, \Phi_2)$ where $r \in \mathbb{N}$

compute $\Pr_s^{\mathcal{M}}(\Phi_1 \, \mathsf{U}^{\leqslant i} \, \Phi_2)$ iteratively
for increasing reward bound $i = 0, 1, 2, \ldots, r$

Let $x_{s,i} = \Pr_s^{\mathcal{M}}(\Phi_1 \, \mathsf{U}^{\leqslant i} \, \Phi_2)$. Then:

if $s \models \exists(\Phi_1 \, \mathsf{U} \, \Phi_2) \wedge \neg\Phi_2$ and $i \geqslant \mathit{rew}(s)$ then

$$x_{s,i} \;=\; \sum_{s' \in S} P(s, s') \cdot x_{s', i - \mathit{rew}(s)}$$

Model checking reward-based properties

treatment of $\mathbb{P}_I(\Phi_1 \, \mathsf{U}^{\leqslant r} \, \Phi_2)$ where $r \in \mathbb{N}$

compute $\Pr^{\mathcal{M}}_s(\Phi_1 \, \mathsf{U}^{\leqslant i} \, \Phi_2)$ iteratively
for increasing reward bound $i = 0, 1, 2, \ldots, r$

Let $x_{s,i} = \Pr^{\mathcal{M}}_s(\Phi_1 \, \mathsf{U}^{\leqslant i} \, \Phi_2)$. Then:

if $s \models \exists(\Phi_1 \, \mathsf{U} \, \Phi_2) \wedge \neg\Phi_2$ and $i \geqslant rew(s)$ then

$$x_{s,i} = \sum_{s' \in S} P(s, s') \cdot x_{s', i - rew(s)}$$

if $s \models \Phi_2$ then: $x_{s,i} = 1$

in all other cases: $x_{s,i} = 0$

Model checking reward-based properties

treatment of $\mathbb{P}_{I}(\Phi_1 \, \mathsf{U}^{\leqslant r} \, \Phi_2)$ where $r \in \mathbb{N}$

compute $\Pr_s^{\mathcal{M}}(\Phi_1 \, \mathsf{U}^{\leqslant i} \, \Phi_2)$ iteratively
for increasing reward bound $i = 0, 1, 2, \ldots, r$

treatment of the $\mathbb{E}_{\leqslant r}(\diamondplus \Phi)$

compute the expected accumulated rewards
by solving the linear equation system

Model checking reward-based properties

treatment of $\mathbb{P}_I(\Phi_1 \, \mathsf{U}^{\leqslant r} \, \Phi_2)$ where $r \in \mathbb{N}$

compute $\Pr_s^{\mathcal{M}}(\Phi_1 \, \mathsf{U}^{\leqslant i} \, \Phi_2)$ iteratively
for increasing reward bound $i = 0, 1, 2, \ldots, r$

treatment of the $\mathbb{E}_{\leqslant r}(\diamondplus \Phi)$, assuming $\Pr^{\mathcal{M}}(\Diamond \Phi) = 1$

compute the expected accumulated rewards
by solving the linear equation system

$$
\begin{aligned}
x_s &= rew(s) + \sum_{s' \in S} P(s, s') \cdot x_{s'} && \text{if } s \not\models \Phi \\
x_s &= 0 && \text{if } s \models \Phi
\end{aligned}
$$

Model checking reward-based properties

treatment of $\mathbb{P}_{I}(\Phi_1 \, \mathsf{U}^{\leqslant r} \, \Phi_2)$ where $r \in \mathbb{N}$

compute $\Pr_s^{\mathcal{M}}(\Phi_1 \, \mathsf{U}^{\leqslant i} \, \Phi_2)$ iteratively
for increasing reward bound $i = 0, 1, 2, \ldots, r$

treatment of the $\mathbb{E}_{\leqslant r}(\diamondplus \Phi)$, assuming $\Pr^{\mathcal{M}}(\Diamond \Phi) = 1$

compute the expected accumulated rewards
by solving the linear equation system

$$x_s = rew(s) + \sum_{s' \in S} P(s, s') \cdot x_{s'} \quad \text{if } s \not\models \Phi$$

also applicable for rational-valued weight fct.

Model checking reward-based properties

treatment of $\mathbb{P}_I(\Phi_1 \, \mathsf{U}^{\leqslant r} \, \Phi_2)$ where $r \in \mathbb{N}$

compute $\Pr_s^{\mathcal{M}}(\Phi_1 \, \mathsf{U}^{\leqslant i} \, \Phi_2)$ iteratively
for increasing reward bound $i = 0, 1, 2, \ldots, r$

treatment of the $\mathbb{E}_{\leqslant r}(\Diamond\!\!\!\!+ \Phi)$, assuming $\Pr^{\mathcal{M}}(\Diamond \Phi) = 1$

compute the expected accumulated rewards
by solving the linear equation system

---

time complexity:

expected rewards: polynomial in $\mathit{size}(\mathcal{M})$

reward-bounded until: polynomial in $\mathit{size}(\mathcal{M})$ and r

Model checking reward-based properties

treatment of $\mathbb{P}_I(\Phi_1 \,\mathsf{U}^{\leqslant r}\, \Phi_2)$ where $r \in \mathbb{N}$

compute $\Pr_s^{\mathcal{M}}(\Phi_1 \,\mathsf{U}^{\leqslant i}\, \Phi_2)$ iteratively
for increasing reward bound $i = 0, 1, 2, \ldots, r$

treatment of the $\mathbb{E}_{\leqslant r}(\Diamond\!\!\!\!+ \Phi)$, assuming $\Pr^{\mathcal{M}}(\Diamond \Phi) = 1$

compute the expected accumulated rewards
by solving the linear equation system

---

time complexity:

pseudo-polynomial

reward-bounded until: polynomial in $size(\mathcal{M})$ and r

Complexity: reward-bounded until

treatment of $\mathbb{P}_I(\Phi_1 \, \mathsf{U}^{\leqslant r} \, \Phi_2)$ where $r \in \mathbb{N}$

compute $\Pr_s^{\mathcal{M}}(\Phi_1 \, \mathsf{U}^{\leqslant i} \, \Phi_2)$ iteratively
for increasing reward bound $i = 0, 1, 2, \ldots, r$

---

unit rewards: polynomial in $size(\mathcal{M})$ and $\log r$
repeated squaring

general case: polynomial in $size(\mathcal{M})$ and r
"pseudo-polynomial"

Complexity: reward-bounded until

treatment of $\mathbb{P}_{\mathrm{I}}(\Phi_1 \,\mathsf{U}^{\leqslant r}\, \Phi_2)$ where $r \in \mathbb{N}$

compute $\Pr_s^{\mathcal{M}}(\Phi_1 \,\mathsf{U}^{\leqslant i}\, \Phi_2)$ iteratively
for increasing reward bound $i = 0, 1, 2, \ldots, r$

---

unit rewards: polynomial in $size(\mathcal{M})$ and $\log r$
repeated squaring

general case: polynomial in $size(\mathcal{M})$ and r

decision problem "does $\Pr_s^{\mathcal{M}}(\Phi_1 \,\mathsf{U}^{\leqslant r}\, \Phi_2) > q$ hold ?"

NP-hard [Laroussinie/Sproston'05]

PosSLP-hard, in PSPACE [Haase/Kiefer'15]

Complexity: reward-bounded until

treatment of $\mathbb{P}_{\mathrm{I}}(\Phi_1 \, \mathsf{U}^{\leqslant r} \, \Phi_2)$ where $r \in \mathbb{N}$

compute $\mathrm{Pr}_s^{\mathcal{M}}(\Phi_1 \, \mathsf{U}^{\leqslant i} \, \Phi_2)$ iteratively
for increasing reward bound $i = 0, 1, 2, \ldots, r$

---

unit rewards: polynomial in $size(\mathcal{M})$ and $\log r$
repeated squaring

general case: polynomial in $size(\mathcal{M})$ and r

decision problem "does $\mathrm{Pr}_s^{\mathcal{M}}(\Phi_1 \, \mathsf{U}^{\leqslant r} \, \Phi_2) > q$ hold ?"

NP-hard [Laroussinie/Sproston'05]

PosSLP-hard, in PSPACE [Haase/Kiefer'15]

NP-hardness [Laroussinie/Sproston'05]

NP-hardness

[Laroussinie/Sproston'05]

The threshold problem for Markov chains is NP-hard:

given: Markov chain $\mathcal{M} = (S, P, s_{init}, rew)$,
$G \subseteq S$, $r \in \mathbb{N}$ and $q \in \,]0,1[\cap \mathbb{Q}$

task: check whether $\Pr_{s_{init}}(\Diamond^{\leqslant r} G) \geqslant q$

NP-hardness

[Laroussinie/Sproston'05]

The threshold problem for Markov chains is NP-hard:

given: Markov chain $\mathcal{M} = (S, P, s_{init}, rew)$,
$G \subseteq S$, $r \in \mathbb{N}$ and $q \in]0,1[\cap \mathbb{Q}$

task: check whether $\Pr_{s_{init}}(\Diamond^{\leqslant r} G) \geqslant q$

Polynomial reduction from counting variant of SUBSUM:

given: $x_1, \ldots, x_n, y, k \in \mathbb{N}$

task: check whether there are at least k subsets N
of $\{1, \ldots, n\}$ s.t. $\sum_{i \in N} x_i \leqslant y$

Polynomial reduction [Laroussinie/Sproston'05]

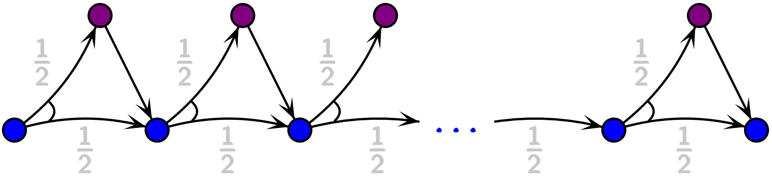
Polynomial reduction [Laroussinie/Sproston'05]

counting variant of SUBSUM:

given: $x_1, \ldots, x_n, y, k \in \mathbb{N}$

task: check whether there are at least k subsets N of $\{1, \ldots, n\}$ s.t. $\sum_{i \in N} x_i \leqslant y$

Markov chain: $2n+1$ states



Polynomial reduction

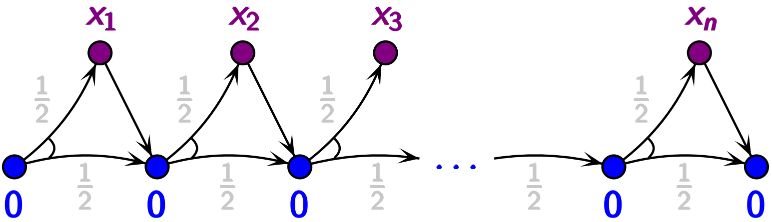
[Laroussinie/Sproston'05]

counting variant of SUBSUM:

given: $x_1, \ldots, x_n, y, k \in \mathbb{N}$

task: check whether there are at least k subsets N of $\{1, \ldots, n\}$ s.t. $\sum_{i \in N} x_i \leqslant y$

Markov chain: $2n+1$ states and rewards for the states



Polynomial reduction

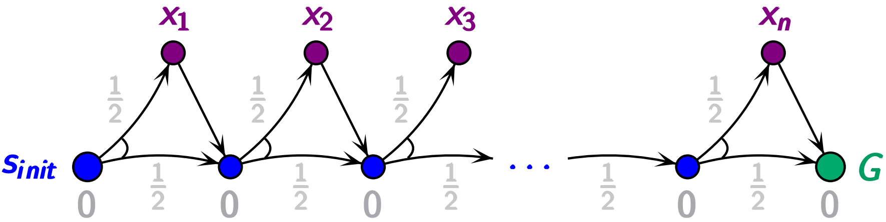
[Laroussinie/Sproston'05]

counting variant of SUBSUM:

given: $x_1, \ldots, x_n, y, k \in \mathbb{N}$

task: check whether there are at least k subsets N of $\{1, \ldots, n\}$ s.t. $\sum_{i \in N} x_i \leqslant y$

Markov chain: $2n+1$ states and rewards for the states

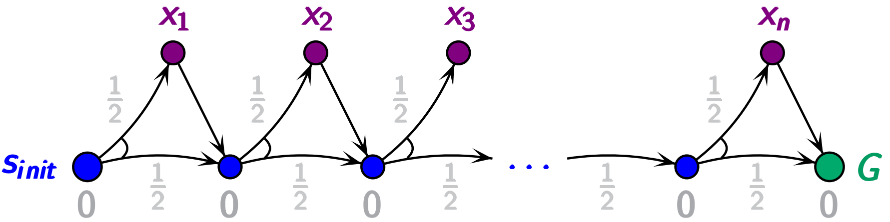


counting variant of SUBSUM:

given: $x_1, \ldots, x_n, y, k \in \mathbb{N}$

task: check whether there are at least k subsets N of $\{1, \ldots, n\}$ s.t. $\sum_{i \in N} x_i \leqslant y$

$\mathrm{Pr}_{s_{init}}(\Diamond^{\leqslant y} G) \geqslant \frac{k}{2^n}$ iff there are at least k subsets ....



Mean-payoff (a.k.a. long-rung average)

Mean-payoff (a.k.a. long-rung average)

given: a weighted graph without trap states

mean-payoff functions $\overline{\mathrm{MP}}, \underline{\mathrm{MP}} : \mathit{InfPaths} \to \mathbb{R}$:

$$\overline{\mathrm{MP}}(s_0\, s_1\, s_2 \ldots) = \limsup_{n\to\infty} \frac{1}{n+1} \cdot \sum_{i=0}^{n} wgt(s_i)$$

$$\underline{\mathrm{MP}}(s_0\, s_1\, s_2 \ldots) = \liminf_{n\to\infty} \frac{1}{n+1} \cdot \sum_{i=0}^{n} wgt(s_i)$$

Mean-payoff (a.k.a. long-rung average)

given: a weighted graph without trap states

mean-payoff functions $\overline{\mathrm{MP}}, \underline{\mathrm{MP}} : \mathit{InfPaths} \to \mathbb{R}$:

$$\overline{\mathrm{MP}}(s_0\, s_1\, s_2 \ldots) \;=\; \limsup_{n\to\infty} \frac{1}{n+1} \cdot \sum_{i=0}^{n} \mathit{wgt}(s_i)$$

$$\underline{\mathrm{MP}}(s_0\, s_1\, s_2 \ldots) \;=\; \liminf_{n\to\infty} \frac{1}{n+1} \cdot \sum_{i=0}^{n} \mathit{wgt}(s_i)$$

if $\mathit{wgt}(s) = +1$, $\mathit{wgt}(t) = -1$ then there exists $n_1, n_2, \ldots$ and $k_1, k_2, \ldots \in \mathbb{N}$ s.t. for $\pi = s^{n_1}\, t^{k_1}\, s^{n_2}\, t^{k_2} \ldots$:

$$\underline{\mathrm{MP}}(\pi) \;<\; 0 \;<\; \overline{\mathrm{MP}}(\pi)$$

Expected mean-payoff in finite MC

fundamental results:

in finite MC: $\mathbb{E}_s(\underline{\mathrm{MP}}) = \mathbb{E}_s(\overline{\mathrm{MP}})$

notation: $\mathbb{E}_s(\mathrm{MP})$ rather than $\mathbb{E}_s(\underline{\mathrm{MP}})$ resp. $\mathbb{E}_s(\overline{\mathrm{MP}})$

Expected mean-payoff in finite MC

fundamental results:

in finite MC: $\mathbb{E}_s(\underline{\mathrm{MP}}) = \mathbb{E}_s(\overline{\mathrm{MP}})$

notation: $\mathbb{E}_s(\mathrm{MP})$ rather than $\mathbb{E}_s(\underline{\mathrm{MP}})$ resp. $\mathbb{E}_s(\overline{\mathrm{MP}})$

Almost all paths eventually enter a BSCC and visit all its states infinitely often.

BSCC: bottom strongly connected component

Expected mean-payoff in finite MC

fundamental results:

in finite MC: $\mathbb{E}_s(\underline{\mathrm{MP}}) = \mathbb{E}_s(\overline{\mathrm{MP}})$

notation: $\mathbb{E}_s(\mathrm{MP})$ rather than $\mathbb{E}_s(\underline{\mathrm{MP}})$ resp. $\mathbb{E}_s(\overline{\mathrm{MP}})$

Almost all paths eventually enter a BSCC and visit all its states infinitely often ...

... with the same long-run frequencies ...

BSCC: bottom strongly connected component

Long-run frequencies in finite MC

steady-state probabilities in BSCC B of a finite MC:

$$\theta^B(s) \;=\; \lim_{n\to\infty} \frac{1}{n} \cdot \sum_{i=1}^{n} \Pr_t(\bigcirc^i s) \quad \text{for each } t \in B$$

$\bigcirc^i s \;\triangleq\;$ "after i steps in state s"

Long-run frequencies in finite MC

steady-state probabilities in BSCC B of a finite MC:

$$\theta^B(s) \;=\; \lim_{n\to\infty} \frac{1}{n} \cdot \sum_{i=1}^{n} \Pr_t(\bigcirc^i s) \quad \text{for each } t \in B$$

computable by a linear equation system:

$$\theta^B(s) \;=\; \sum_{t\in B} \theta^B(t) \cdot P(t,s)$$

"balance equations"

$\bigcirc^i s \;\hat{=}\;$ "after i steps in state s"

Long-run frequencies in finite MC

steady-state probabilities in BSCC B of a finite MC:

$$\theta^B(s) \;=\; \lim_{n\to\infty} \frac{1}{n} \cdot \sum_{i=1}^{n} \Pr_t(\bigcirc^i s) \quad \text{for each } t \in B$$

computable by a linear equation system:

$$\theta^B(s) \;=\; \sum_{t\in B} \theta^B(t) \cdot P(t,s)$$

$$\sum_{s\in B} \theta^B(s) \;=\; 1$$

$\bigcirc^i s \;\hat{=}\;$ "after i steps in state s"

Long-run frequencies in finite MC

steady-state probabilities in BSCC B of a finite MC:

$$\theta^B(s) \;=\; \lim_{n\to\infty} \frac{1}{n} \cdot \sum_{i=1}^{n} \Pr_t(\bigcirc^i s) \quad \text{for each } t \in B$$

computable by a linear equation system:

$$\theta^B(s) \;=\; \sum_{t\in B} \theta^B(t) \cdot P(t,s)$$

$$\sum_{s\in B} \theta^B(s) \;=\; 1$$

unique solution of the linear equation system

$$x = x \cdot P|_B$$

$$\sum_{s\in B} x_s = 1$$

$\bigcirc^i s \;\widehat{=}\;$ "after i steps in state s"

Long-run frequencies in finite MC

steady-state probabilities in BSCC B of a finite MC:

$$\theta^B(s) \;=\; \lim_{n\to\infty} \frac{1}{n} \cdot \sum_{i=1}^{n} \Pr_t\big(\bigcirc^i s\big) \quad \text{for each } t \in B$$

for almost all paths $\pi = s_0\, s_1\, s_2 \ldots$ with $\pi \models \lozenge B$:

$$\theta^B(s) \;=\; \underbrace{\lim_{n\to\infty} \frac{1}{n+1} \cdot \mathit{freq}(s, s_0\, s_1 \ldots s_n)}_{\text{long-run frequency of state } s \text{ in path } \pi}$$

... limit exists for almost all paths ...

Long-run frequencies in finite MC

steady-state probabilities in BSCC B of a finite MC:

$$\theta^B(s) \;=\; \lim_{n\to\infty} \frac{1}{n} \cdot \sum_{i=1}^{n} \Pr_t(\bigcirc^i s) \quad \text{for each } t \in B$$

for almost all paths $\pi = s_0\, s_1\, s_2 \ldots$ with $\pi \models \Diamond B$:

$$\theta^B(s) \;=\; \underbrace{\lim_{n\to\infty} \frac{1}{n+1} \cdot \mathit{freq}(s, s_0\, s_1 \ldots s_n)}_{\text{long-run frequency of state } s \text{ in path } \pi}$$

$$\mathit{freq}(s, s_0\, s_1 \ldots s_n) \;=\; \left\{ \begin{array}{l} \text{number of occurrences of } s \\ \text{in the sequence } s_0\, s_1 \ldots s_n \end{array} \right.$$

Mean-payoff in finite weighted MC

steady-state probabilities in BSCC B of a finite MC:

$$\theta^B(s) \;=\; \lim_{n\to\infty} \frac{1}{n} \cdot \sum_{i=1}^{n} \Pr_t\big(\bigcirc^i s\big) \quad \text{for each } t \in B$$

for almost all paths $\pi = s_0\, s_1\, s_2 \ldots$ with $\pi \models \Diamond B$:

$$\theta^B(s) \;=\; \lim_{n\to\infty} \frac{1}{n+1} \cdot \mathit{freq}(s, s_0\, s_1 \ldots s_n)$$

if $\pi \models \Diamond B$ where B is a BSCC then almost surely

$$\mathrm{MP}(\pi) \;=\; \sum_{s\in B} \theta^B(s) \cdot \mathit{wgt}(s)$$

Mean-payoff in finite weighted MC

steady-state probabilities in BSCC B of a finite MC:

$$\theta^B(s) \;=\; \lim_{n\to\infty} \frac{1}{n} \cdot \sum_{i=1}^{n} \Pr_t\big(\bigcirc^i s\big) \quad \text{for each } t \in B$$

for almost all paths $\pi = s_0\, s_1\, s_2 \ldots$ with $\pi \models \Diamond B$:

$$\theta^B(s) \;=\; \lim_{n\to\infty} \frac{1}{n+1} \cdot \mathit{freq}(s, s_0\, s_1 \ldots s_n)$$

if $\pi \models \Diamond B$ where B is a BSCC then almost surely

$$\mathrm{MP}(\pi) \;=\; \underbrace{\sum_{s\in B} \theta^B(s) \cdot \mathit{wgt}(s)}_{\text{only depends on } B}$$

Mean-payoff in finite weighted MC

steady-state probabilities in BSCC B of a finite MC:

$$\theta^B(s) \;=\; \lim_{n\to\infty} \frac{1}{n} \cdot \sum_{i=1}^{n} \Pr_t\big(\bigcirc^i s\big) \quad \text{for each } t \in B$$

for almost all paths $\pi = s_0\, s_1\, s_2 \ldots$ with $\pi \models \Diamond B$:

$$\theta^B(s) \;=\; \lim_{n\to\infty} \frac{1}{n+1} \cdot \mathit{freq}(s, s_0\, s_1 \ldots s_n)$$

if $\pi \models \Diamond B$ where B is a BSCC then almost surely

$$\mathrm{MP}(\pi) \;=\; \underbrace{\sum_{s\in B} \theta^B(s) \cdot \mathit{wgt}(s)}_{\text{only depends on } B} \;\stackrel{\text{def}}{=}\; \mathrm{MP}(B)$$

Mean-payoff in finite weighted MC

steady-state probabilities in BSCC B of a finite MC:

$$\theta^B(s) \;=\; \lim_{n\to\infty} \frac{1}{n} \cdot \sum_{i=1}^{n} \Pr_t\big(\bigcirc^i s\big) \quad \text{for each } t \in B$$

for almost all paths $\pi = s_0\, s_1\, s_2 \ldots$ with $\pi \models \Diamond B$:

$$\theta^B(s) \;=\; \lim_{n\to\infty} \frac{1}{n+1} \cdot \mathit{freq}(s, s_0\, s_1 \ldots s_n)$$

if $\pi \models \Diamond B$ where B is a BSCC then almost surely

$$\mathrm{MP}(\pi) \;=\; \sum_{s\in B} \theta^B(s) \cdot \mathit{wgt}(s) \;\stackrel{\text{def}}{=}\; \mathrm{MP}(B)$$

expected mean-payoff: $\displaystyle\sum_{B} \Pr_{s_0}(\Diamond B) \cdot \mathrm{MP}(B)$

Long-run ratios in finite MC

MC with two reward functions $cost, util : S \to \mathbb{N}$

Examples:

- energy-utility ratio
- number of SLA violations per day
- recovery time per failure

Long-run ratios in finite MC

MC with two reward functions $cost, util : S \to \mathbb{N}$

long-run cost-utility ratio $lrrat : InfPaths \to \mathbb{R}$

$$lrrat(s_0\, s_1\, s_2 \ldots) \;=\; \lim_{n\to\infty} \frac{cost(s_0\, s_1 \ldots s_n)}{util(s_0\, s_1 \ldots s_n)}$$

Examples:

- energy-utility ratio
- number of SLA violations per day
- recovery time per failure

Long-run ratios in finite MC

MC with two reward functions $\mathit{cost}, \mathit{util} : S \to \mathbb{N}$

long-run cost-utility ratio $\mathit{lrrat} : \mathit{InfPaths} \to \mathbb{R}$

$$\mathit{lrrat}(s_0\, s_1\, s_2 \ldots) \;=\; \lim_{n\to\infty} \frac{\mathit{cost}(s_0\, s_1 \ldots s_n)}{\mathit{util}(s_0\, s_1 \ldots s_n)}$$

does the limit exist for almost all paths ?

- energy-utility ratio
- number of SLA violations per day
- recovery time per failure

Long-run ratios in finite MC

MC with two reward functions $\mathit{cost}, \mathit{util} : S \to \mathbb{N}$

long-run cost-utility ratio $\mathit{lrrat} : \mathit{InfPaths} \to \mathbb{R}$

$$\mathit{lrrat}(s_0\, s_1\, s_2 \ldots) = \lim_{n\to\infty} \frac{\mathit{cost}(s_0\, s_1 \ldots s_n)}{\mathit{util}(s_0\, s_1 \ldots s_n)}$$

$$= \lim_{n\to\infty} \frac{\frac{1}{n+1} \cdot \sum_{i=0}^{n} \mathit{cost}(s_i)}{\frac{1}{n+1} \cdot \sum_{i=0}^{n} \mathit{util}(s_i)}$$

Long-run ratios in finite MC

MC with two reward functions $cost, util : S \to \mathbb{N}$

long-run cost-utility ratio $lrrat : InfPaths \to \mathbb{R}$

$$lrrat(s_0\, s_1\, s_2 \ldots) = \lim_{n\to\infty} \frac{cost(s_0\, s_1 \ldots s_n)}{util(s_0\, s_1 \ldots s_n)}$$

$$= \lim_{n\to\infty} \frac{\frac{1}{n+1} \cdot \sum_{i=0}^{n} cost(s_i)}{\frac{1}{n+1} \cdot \sum_{i=0}^{n} util(s_i)}$$

$$= \frac{\mathrm{MP}[cost](s_0\, s_1\, s_2 \ldots)}{\mathrm{MP}[util](s_0\, s_1\, s_2 \ldots)}$$

Long-run ratios in finite MC

MC with two reward functions $cost, util : S \to \mathbb{N}$

long-run cost-utility ratio $lrrat : InfPaths \to \mathbb{R}$

$$lrrat(s_0\, s_1\, s_2 \ldots) = \lim_{n\to\infty} \frac{cost(s_0\, s_1 \ldots s_n)}{util(s_0\, s_1 \ldots s_n)}$$

$$= \lim_{n\to\infty} \frac{\frac{1}{n+1} \cdot \sum_{i=0}^{n} cost(s_i)}{\frac{1}{n+1} \cdot \sum_{i=0}^{n} util(s_i)}$$

$$= \frac{\mathrm{MP}[cost](s_0\, s_1\, s_2 \ldots)}{\mathrm{MP}[util](s_0\, s_1\, s_2 \ldots)}$$

in particular:
limit exists for
almost all paths

Long-run ratios in finite MC

MC with two reward functions $cost, util : S \to \mathbb{N}$

long-run cost-utility ratio $lrrat : InfPaths \to \mathbb{R}$

$$lrrat(s_0\, s_1\, s_2 \ldots) \; = \; \lim_{n \to \infty} \frac{cost(s_0\, s_1 \ldots s_n)}{util(s_0\, s_1 \ldots s_n)}$$

if $\pi \models \Diamond B$ where B is a BSCC then almost surely

$$lrrat(\pi) \; = \; \frac{\mathrm{MP}[cost](B)}{\mathrm{MP}[util](B)}$$

$$MP[wgt](B) = \sum_{s \in B} \theta^B(s) \cdot wgt(s)$$ mean-payoff for weight function

Long-run ratios in finite MC

MC with two reward functions $\mathit{cost}, \mathit{util} : S \to \mathbb{N}$

long-run cost-utility ratio $\mathit{lrrat} : \mathit{InfPaths} \to \mathbb{R}$

$$\mathit{lrrat}(s_0\, s_1\, s_2 \ldots) \;=\; \lim_{n\to\infty} \frac{\mathit{cost}(s_0\, s_1 \ldots s_n)}{\mathit{util}(s_0\, s_1 \ldots s_n)}$$

if $\pi \models \Diamond B$ where B is a BSCC then almost surely

$$\mathit{lrrat}(\pi) \;=\; \underbrace{\frac{\mathrm{MP}[\mathit{cost}](B)}{\mathrm{MP}[\mathit{util}](B)}}_{\text{only depends on } B} \;\stackrel{\text{def}}{=}\; \mathit{lrrat}(B)$$

Long-run ratios in finite MC

MC with two reward functions $cost, util : S \to \mathbb{N}$

long-run cost-utility ratio $lrrat : InfPaths \to \mathbb{R}$

$$lrrat(s_0\, s_1\, s_2 \ldots) \;=\; \lim_{n\to\infty} \frac{cost(s_0\, s_1 \ldots s_n)}{util(s_0\, s_1 \ldots s_n)}$$

if $\pi \models \Diamond B$ where B is a BSCC then almost surely

$$lrrat(\pi) \;=\; \frac{\mathrm{MP}[cost](B)}{\mathrm{MP}[util](B)} \;\stackrel{\text{def}}{=}\; lrrat(B)$$

expected long-run ratio: $\displaystyle\sum_{B} \Pr^{\mathcal{M}}(\Diamond B) \cdot lrrat(B)$

Best threshold for long-run ratios

given: MC with reward functions $\mathit{cost}, \mathit{util} : S \to \mathbb{N}$
rational probability bound p

compute $r_{opt} \;=\; \inf\left\{\, r \in \mathbb{R} : \Pr^{\mathcal{M}}(\, \mathit{lrrat} \leqslant r \,) > p \,\right\}$

↑

random variable for the
long-run cost-utility ratio
(as before)

Best threshold for long-run ratios

given: MC with reward functions $cost, util : S \to \mathbb{N}$
rational probability bound p

compute $r_{opt} = \inf \{ r \in \mathbb{R} : \Pr^{\mathcal{M}}(lrrat \leqslant r) > p \}$

$$r_{opt} = \inf \{ r \in \mathbb{R} : \Pr^{\mathcal{M}}(\Box\Diamond(\frac{cost}{util} \leqslant r)) > p \}$$

if $\pi = s_0\, s_1\, s_2 \ldots$ is an infinite path then

$$\pi \models \Box\Diamond(\frac{cost}{util} \leqslant r) \quad \text{iff} \quad \overset{\infty}{\exists}\, n \text{ s.t. } \frac{cost(s_0\, s_1 \ldots s_n)}{util(s_0\, s_1 \ldots s_n)} \leqslant r$$

Best threshold for long-run ratios

given: MC with reward functions $cost, util : S \to \mathbb{N}$
rational probability bound p

compute $r_{opt} = \inf \{ r \in \mathbb{R} : \Pr^{\mathcal{M}}(\mathit{lrrat} \leqslant r) > p \}$

$$r_{opt} = \inf \{ r \in \mathbb{R} : \Pr^{\mathcal{M}}(\Box\Diamond(\tfrac{cost}{util} \leqslant r)) > p \}$$

$$= \inf \{ r \in \mathbb{R} : \Pr^{\mathcal{M}}(\Diamond\Box(\tfrac{cost}{util} \leqslant r)) > p \}$$

$\pi \models \Box\Diamond(\frac{cost}{util} \leqslant r)$ iff $\overset{\infty}{\exists}\, n$ s.t. $\frac{cost(s_0\, s_1 \ldots s_n)}{util(s_0\, s_1 \ldots s_n)} \leqslant r$

Best threshold for long-run ratios

given: MC with reward functions $\mathit{cost}, \mathit{util} : S \to \mathbb{N}$
rational probability bound p

compute $\quad r_{opt} \quad = \quad \inf \{ r \in \mathbb{R} : \Pr^{\mathcal{M}}(\mathit{lrrat} \leqslant r) > p \}$

$$\begin{aligned} r_{opt} &= \inf \{ r \in \mathbb{R} : \Pr^{\mathcal{M}}(\Box\Diamond(\tfrac{\mathit{cost}}{\mathit{util}} \leqslant r)) > p \} \\ &= \inf \{ r \in \mathbb{R} : \Pr^{\mathcal{M}}(\Diamond\Box(\tfrac{\mathit{cost}}{\mathit{util}} \leqslant r)) > p \} \\ &= \min \{ r \in \mathbb{Q} : \Pr^{\mathcal{M}}(\Diamond C_r) > p \} \end{aligned}$$

where C_r = union of all BSCCs B with $\mathit{lrrat}(B) \leqslant r$

Best threshold for long-run ratios

given: MC with reward functions $cost, util : S \to \mathbb{N}$
rational probability bound p

compute $$r_{opt} = \inf\{\, r \in \mathbb{R} : \Pr^{\mathcal{M}}(\, lrrat \leqslant r\,) > p \,\}$$
$$= \min\{\, r \in \mathbb{Q} : \Pr^{\mathcal{M}}(\Diamond C_r) > p \,\}$$

where C_r = union of all BSCCs B with $lrrat(B) \leqslant r$

↑ expected long-run ratio of B

Best threshold for long-run ratios

given: MC with reward functions $cost, util : S \to \mathbb{N}$
rational probability bound p

compute $$r_{opt} = \inf \{ r \in \mathbb{R} : \Pr^{\mathcal{M}}(lrrat \leqslant r) > p \}$$
$$= \min \{ r \in \mathbb{Q} : \Pr^{\mathcal{M}}(\Diamond C_r) > p \}$$

where C_r = union of all BSCCs B with $lrrat(B) \leqslant r$

1. compute the BSCCs $B_1, \ldots, B_k$ and $r_i = lrrat(B_i)$

Best threshold for long-run ratios

given: MC with reward functions $cost, util : S \to \mathbb{N}$
rational probability bound p

compute $r_{opt} = \inf \{ r \in \mathbb{R} : \Pr^{\mathcal{M}}(lrrat \leqslant r) > p \}$
$= \min \{ r \in \mathbb{Q} : \Pr^{\mathcal{M}}(\Diamond C_r) > p \}$

where C_r = union of all BSCCs B with $lrrat(B) \leqslant r$

1. compute the BSCCs $B_1, \ldots, B_k$ and $r_i = lrrat(B_i)$

   w.l.o.g. $r_1 < r_2 < \ldots < r_k$

Best threshold for long-run ratios

given: MC with reward functions $cost, util : S \to \mathbb{N}$
rational probability bound p

compute $r_{opt} = \inf \{ r \in \mathbb{R} : \Pr^{\mathcal{M}}(lrrat \leqslant r) > p \}$
$= \min \{ r \in \mathbb{Q} : \Pr^{\mathcal{M}}(\Diamond C_r) > p \}$

where C_r = union of all BSCCs B with $lrrat(B) \leqslant r$

1. compute the BSCCs $B_1, \ldots, B_k$ and $r_i = lrrat(B_i)$
   w.l.o.g. $r_1 < r_2 < \ldots < r_k$
2. determine the minimal $i \in \{1, \ldots, k\}$ such that
   $\Pr^{\mathcal{M}}(\Diamond B_1) + \ldots + \Pr^{\mathcal{M}}(\Diamond B_i) > p$ and return r_i

Cost-utility ratios: invariances

Given an MC with two positive reward functions $\mathit{cost}, \mathit{util} : S \to \mathbb{N}$, consider their ratio:

$$\mathit{ratio} \;=\; \frac{\mathit{cost}}{\mathit{util}} : \mathit{FinPaths} \to \mathbb{Q}$$

$$\mathit{ratio}(\pi) \;=\; \frac{\mathit{cost}(\pi)}{\mathit{util}(\pi)} \quad \text{for all finite paths } \pi$$

decision problems: given an ω-regular property φ and probability bound $q \in [0,1[$, ratio threshold $r \in \mathbb{Q}$:

- does $\Pr^{\mathcal{M}}(\Box(\mathit{ratio} \leqslant r) \wedge \varphi) > q$ hold ?
- does $\Pr^{\mathcal{M}}(\Box(\mathit{ratio} \leqslant r) \wedge \varphi) = 1$ hold ?

Cost-utility ratio via weight functions

Given an MC with two positive reward functions $cost, util : S \to \mathbb{N}$, consider their ratio:

$$ratio = \frac{cost}{util} : FinPaths \to \mathbb{Q}$$

$$ratio(\pi) = \frac{cost(\pi)}{util(\pi)} \quad \text{for all finite paths } \pi$$

replace ratio by weight constraints:

$$\Box(\, ratio \leqslant r \,) \quad \equiv \quad \Box(\, wgt \leqslant 0 \,)$$

Cost-utility ratio via weight functions

Given an MC with two positive reward functions $cost, util : S \to \mathbb{N}$, consider their ratio:

$$ratio = \frac{cost}{util} : FinPaths \to \mathbb{Q}$$

$$ratio \leqslant r \quad \text{iff} \quad wgt \leqslant 0$$

$$\text{where} \quad wgt = cost - r \cdot util$$

$$\Box(ratio \leqslant r) \equiv \Box(wgt \leqslant 0)$$

Cost-utility ratio via weight functions

Given an MC with two positive reward functions $cost, util : S \to \mathbb{N}$, consider their ratio:

$$ratio = \frac{cost}{util} : FinPaths \to \mathbb{Q}$$

$$ratio \leqslant r \quad \text{iff} \quad wgt \leqslant 0$$

$$\text{where} \quad wgt = cost - r \cdot util \in \mathbb{Q}$$

$$\Box(ratio \leqslant r) \equiv \Box(wgt \leqslant 0)$$

Cost-utility ratio via weight functions

Given an MC with two positive reward functions $cost, util : S \to \mathbb{N}$, consider their ratio:

$$ratio = \frac{cost}{util} : FinPaths \to \mathbb{Q}$$

$$ratio \leqslant r \quad \text{iff} \quad wgt > 0$$

where $wgt = (cost - r \cdot util) \cdot const \in \mathbb{Z}$

↑ integer-valued weight function

$$\Box(\, ratio \leqslant r \,) \quad \equiv \quad \Box(\, wgt > 0 \,)$$

Weight invariances for MC

Given an MC with a weight function $\boldsymbol{wgt} : \boldsymbol{S} \to \mathbb{Z}$.

Weight invariances for MC

Given an MC with a weight function $wgt : S \to \mathbb{Z}$.

almost-sure problem:

does $\Pr^{\mathcal{M}}_{s_0}\big(\Box(wgt > 0) \wedge \varphi\big) = 1$ hold ?

positive problem:

does $\Pr^{\mathcal{M}}_{s_0}\big(\Box(wgt > 0) \wedge \varphi\big) > 0$ hold ?

quantitative problems, e.g.:

does $\Pr^{\mathcal{M}}_{s_0}\big(\Box(wgt > 0) \wedge \varphi\big) > \frac{1}{2}$ hold ?

Weight invariances for MC

Given an MC with a weight function $wgt : S \to \mathbb{Z}$.

almost-sure problem:

does $\Pr^{\mathcal{M}}_{s_0}\big(\Box(wgt > 0) \wedge \varphi\big) = 1$ hold ?

simple

positive problem:

does $\Pr^{\mathcal{M}}_{s_0}\big(\Box(wgt > 0) \wedge \varphi\big) > 0$ hold ?

quantitative problems, e.g.:

does $\Pr^{\mathcal{M}}_{s_0}\big(\Box(wgt > 0) \wedge \varphi\big) > \frac{1}{2}$ hold ?

difficult

Almost-sure weight invariances

$$\Pr_{s}^{\mathcal{M}}\big(\Box(wgt > 0) \wedge \varphi\big) = 1$$

Almost-sure weight invariances

$$\Pr_s^{\mathcal{M}}\big(\Box(wgt > 0) \wedge \varphi\big) = 1$$

iff $\Pr_s^{\mathcal{M}}\big(\Box(wgt > 0)\big) = 1$ and $\Pr_s^{\mathcal{M}}\big(\varphi\big) = 1$

Almost-sure weight invariances

$$\Pr_s^{\mathcal{M}}\big(\Box(wgt > 0) \wedge \varphi\big) = 1$$

iff $\Pr_s^{\mathcal{M}}\big(\Box(wgt > 0)\big) = 1$ and $\Pr_s^{\mathcal{M}}\big(\varphi\big) = 1$

iff $s \not\models \exists\Diamond(wgt \leqslant 0)$ and $\Pr_s^{\mathcal{M}}\big(\varphi\big) = 1$

Almost-sure weight invariances

$$\Pr_s^{\mathcal{M}}\big(\Box(wgt > 0) \wedge \varphi\big) = 1$$

iff $\Pr_s^{\mathcal{M}}\big(\Box(wgt > 0)\big) = 1$ and $\Pr_s^{\mathcal{M}}(\varphi) = 1$

iff $s \not\models \exists\Diamond(wgt \leqslant 0)$ and $\Pr_s^{\mathcal{M}}(\varphi) = 1$

↑

solvable by standard
shortest-path algorithms

Almost-sure weight invariances

$$\Pr_s^{\mathcal{M}}\big(\Box(wgt > 0) \wedge \varphi\big) = 1$$

iff $\Pr_s^{\mathcal{M}}\big(\Box(wgt > 0)\big) = 1$ and $\Pr_s^{\mathcal{M}}(\varphi) = 1$

iff $s \not\models \exists\Diamond(wgt \leqslant 0)$ and $\Pr_s^{\mathcal{M}}(\varphi) = 1$

solvable by standard shortest-path algorithms

standard methods for ω-regular path properties

polynomially time-bounded for reachability or Büchi properties

Almost-sure weight invariances

$$\Pr^{\mathcal{M}}_{s}\big(\Box(wgt > 0) \wedge \varphi\big) = 1$$

iff $\Pr^{\mathcal{M}}_{s}\big(\Box(wgt > 0)\big) = 1$ and $\Pr^{\mathcal{M}}_{s}(\varphi) = 1$

iff $s \not\models \exists\Diamond(wgt \leqslant 0)$ and $\Pr^{\mathcal{M}}_{s}(\varphi) = 1$

Best threshold computable by shortest-path algorithms:

$$\textbf{sup}\ \{\ r \in \mathbb{Z}\ :\ \Pr^{\mathcal{M}}_{s}\big(\Box(wgt > r) \wedge \varphi\big) = 1\ \}$$

$1 +$ length of a shortest path starting in state s,
provided that φ holds almost surely and there are no negative cycles

Weight invariances for MC

Given an MC with a weight function $wgt : S \to \mathbb{Z}$.

almost-sure problem:

does $\Pr^{\mathcal{M}}_{s_0}\big(\Box(wgt > 0) \wedge \varphi\big) = 1$ hold ?

simple

positive problem:

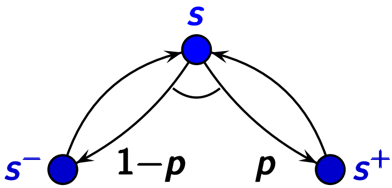
does $\Pr^{\mathcal{M}}_{s_0}\big(\Box(wgt > 0) \wedge \varphi\big) > 0$ hold ?

quantitative problems, e.g.:

does $\Pr^{\mathcal{M}}_{s_0}\big(\Box(wgt > 0) \wedge \varphi\big) > \frac{1}{2}$ hold ?

difficult

Markov chain with weight function



$wgt(s) = +1$

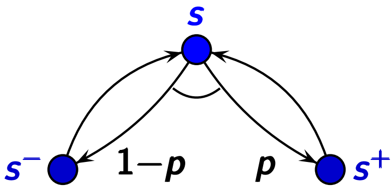
$wgt(s^-) = -2$

$wgt(s^+) = 0$

probability parameter

$0 < p < 1$

Markov chain with weight function

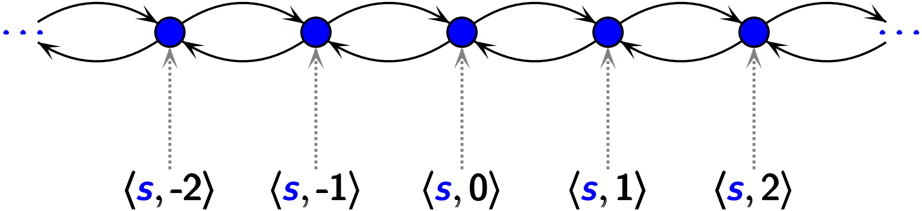


$wgt(s) = +1$

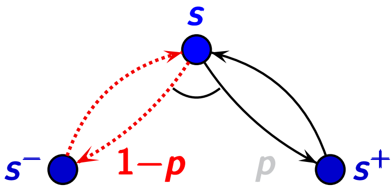
$wgt(s^-) = -2$

$wgt(s^+) = 0$

random walk:



Markov chain with weight function



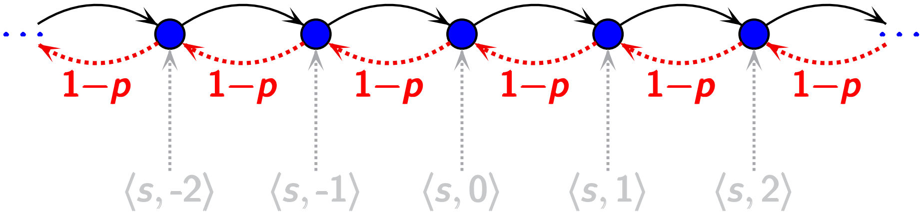
$wgt(s) = +1$

$wgt(s^-) = -2$

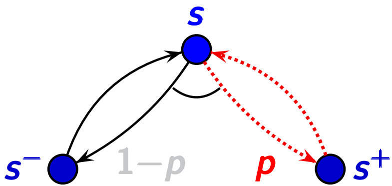
$wgt(s^+) = 0$

weight $-\mathbf{1}$ for the cycle $s\ s^-\ s$

random walk:



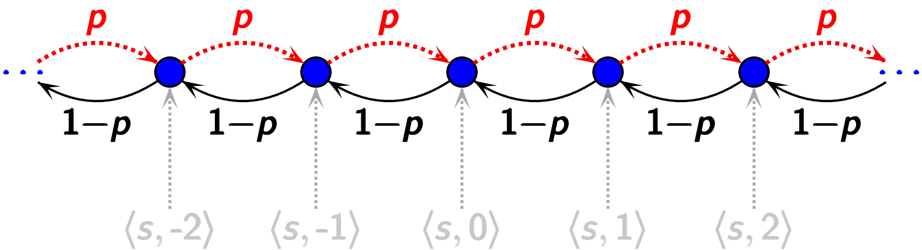
Markov chain with weight function



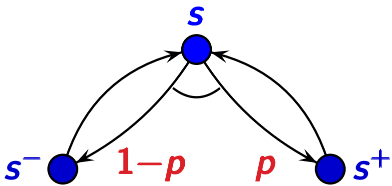
$$wgt(s) = +1$$
$$wgt(s^-) = -2$$
$$wgt(s^+) = 0$$

weight **+1** for the cycle $s\ s^+\ s$

random walk:



Markov chain with weight function



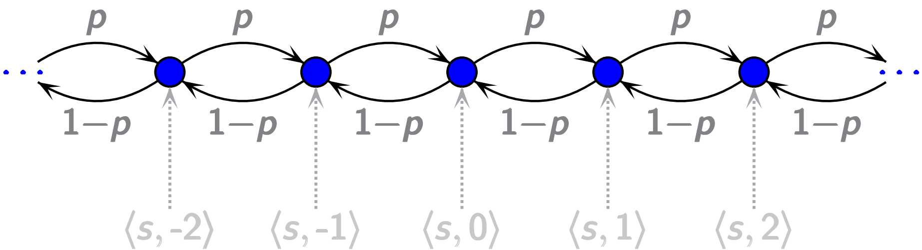
$wgt(s) = +1$

$wgt(s^-) = -2$

$wgt(s^+) = 0$

$$\Pr_s(\Box(wgt > 0)) > 0 \quad \text{iff} \quad p > \frac{1}{2}$$

random walk:



Weight invariance problem: positive case

The problem "does $\Pr_s(\Box(wgt > r) \wedge \varphi) > 0$ hold ?"

- depends on the concrete transition probabilities

where φ is a ω-regular property and $0 \leqslant q < 1$

Weight invariance problem: positive case

The problem "does $\mathrm{Pr}_s(\Box(wgt > r) \wedge \varphi) > 0$ hold ?"

- depends on the concrete transition probabilities
- is solvable in polynomial time

  BSCC-analysis and variants of shortest-paths algorithms, assuming φ is a Rabin or Streett or reachability condition

  [Brázdil/Kiefer/Kučera/Novotný/Katoen'14]
  [Krähmann/Schubert/Baier/Dubslaff'15]

where φ is a ω-regular property and $0 \leqslant q < 1$

Weight invariance problem: positive case

The problem "does $\mathbf{Pr}_s(\Box(wgt > r) \wedge \varphi) > 0$ hold ?"

- depends on the concrete transition probabilities
- is solvable in polynomial time

  BSCC-analysis and variants of shortest-paths algorithms, assuming φ is a Rabin or Streett condition

check whether there exists a good BSCC B s.t.

1. $\mathrm{MP}(B) > 0$ or $\mathrm{MP}(B) = 0$ & no negative cycle in B
2. there is a path π from s to B s.t. π and its prefixes have sufficiently high weight

Weight invariance problem: quantitative case

The problem "does $\Pr_s(\Box(wgt > r) \land \varphi) > 0$ hold ?"

- depends on the concrete transition probabilities
- is solvable in polynomial time

  BSCC-analysis and variants of shortest-paths algorithms, assuming φ is a Rabin or Streett condition

The problem "does $\Pr_s(\Box(wgt > 0) \land \varphi) > q$ hold ?"

- is reducible to the threshold problem for probabilistic pushdown automata (exponential blowup)

Weight invariance problem: quantitative case

The problem "does $\mathrm{Pr}_s(\Box(wgt > r) \wedge \varphi) > 0$ hold ?"

- depends on the concrete transition probabilities
- is solvable in polynomial time

  BSCC-analysis and variants of shortest-paths algorithms, assuming φ is a Rabin or Streett condition

The problem "does $\mathrm{Pr}_s(\Box(wgt > 0) \wedge \varphi) > q$ hold ?"

- is reducible to the threshold problem for probabilistic pushdown automata (exponential blowup)
- is PosSLP-hard, even for unit weights and $\varphi = true$

  [Etessami/Yannak.'09], [Brázdil/Brozek/Etes./Kučera/Wojt.'10]

Weight invariance problem: almost-sure case

The problem "does $\mathrm{Pr}_s(\Box(wgt > r) \wedge \varphi) = 1$ hold ?"

- independent from the concrete transition probabilities
- is solvable in polynomial time

Weight invariance problem: almost-sure case

The problem "does $\Pr_s(\Box(wgt > r) \wedge \varphi) = 1$ hold ?"

- independent from the concrete transition probabilities
- is solvable in polynomial time

$$\Pr_s(\Box(wgt > r) \wedge \varphi) = 1$$

iff $\Pr_s(\Box(wgt > r)) = 1$ and $\Pr_s(\varphi) = 1$

Weight invariance problem: almost-sure case

The problem "does $\Pr_s(\Box(wgt > r) \wedge \varphi) = 1$ hold ?"

- independent from the concrete transition probabilities
- is solvable in polynomial time

$$\Pr_s(\Box(wgt > r) \wedge \varphi) = 1$$

iff $\Pr_s(\Box(wgt > r)) = 1$ and $\boxed{\Pr_s(\varphi) = 1}$

↑

standard algorithm

polynomial-time for

reachability, Rabin or Streett

Weight invariance problem: almost-sure case

The problem "does $\mathbf{Pr}_s(\Box(wgt > r) \wedge \varphi) = 1$ hold ?"

- independent from the concrete transition probabilities
- is solvable in polynomial time

$$\mathbf{Pr}_s(\Box(wgt > r) \wedge \varphi) = 1$$

iff $\boxed{\mathbf{Pr}_s(\Box(wgt > r)) = 1}$ and $\mathbf{Pr}_s(\varphi) = 1$

↑

shortest-path algorithm

check chether the weight of a shortest path from s is at least r+1

Weight-bounded reachability in MC

Weight-bounded reachability in MC

given: weighted MC $\mathcal{M}$, weight bound $r \in \mathbb{Z}$
and a distinguished states s, $goal$

decision problems:

positive prob: does $\Pr_s(\Diamond^{\leqslant r} goal) > 0$ hold ?

almost-sure: does $\Pr_s(\Diamond^{\leqslant r} goal) = 1$ hold ?

quantitative: does $\Pr_s(\Diamond^{\leqslant r} goal) > \frac{1}{2}$ hold ?

Weight-bounded reachability in MC

given: weighted MC $\mathcal{M}$, weight bound $r \in \mathbb{Z}$
and a distinguished states s, *goal*

decision problems:

positive prob: does $\Pr_s(\Diamond^{\leqslant r} goal) > 0$ hold ?

solvable in poly-time using shortest-path algorithms

almost-sure: does $\Pr_s(\Diamond^{\leqslant r} goal) = 1$ hold ?

solvable in poly-time using shortest-path algorithms;
a bit tricky if goal is not a trap

quantitative: does $\Pr_s(\Diamond^{\leqslant r} goal) > \frac{1}{2}$ hold ?

solvable in poly-space using algorithms for prob PDA

Weight-bounded reachability in MC

given: weighted MC $\mathcal{M}$, weight bound $r \in \mathbb{Z}$
and a distinguished states s, *goal*

decision problems:

positive prob: does $\Pr_s(\Diamond^{\leqslant r} goal) > 0$ hold ?

solvable in poly-time using shortest-path algorithms

almost-sure: does $\Pr_s(\Diamond^{\leqslant r} goal) = 1$ hold ?

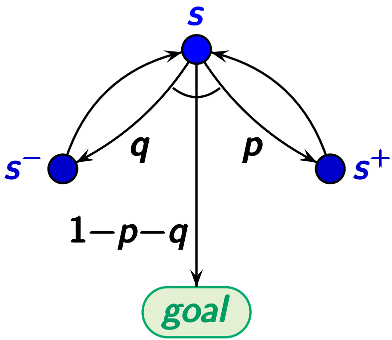
solvable in poly-time using shortest-path algorithms;
a bit tricky if goal is not a trap

quantitative: does $\Pr_s(\Diamond^{\leqslant r} goal) > \frac{1}{2}$ hold ?

solvable in poly-space using algorithms for prob PDA

Is there an algorithm to compute $\Pr_s(\Diamond^{\leqslant r} goal)$?

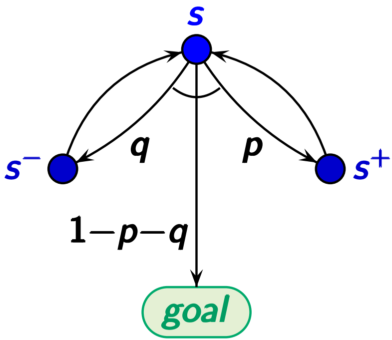
Weight-bounded reachability in MC [Kiefer'17]



$wgt(s) = 0$
$wgt(s^-) = -1$
$wgt(s^+) = +1$

probability parameters
p and q with $0 < p, q < 1$
and $p + q < 1$

Weight-bounded reachability in MC [Kiefer'17]



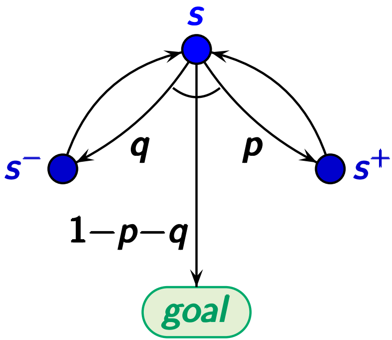
$wgt(s) = 0$

$wgt(s^-) = -1$

$wgt(s^+) = +1$

$$\Pr_s(\Diamond^{=0} goal) = (1-p-q) \cdot \sum_{n=0}^{\infty} \binom{2n}{n} \cdot p^n \cdot q^n$$

Weight-bounded reachability in MC [KIEFER'17]



$$wgt(s) = 0$$
$$wgt(s^-) = -1$$
$$wgt(s^+) = +1$$

$$\Pr_s(\Diamond^{=0} goal) = (1-p-q) \cdot \sum_{n=0}^{\infty} \binom{2n}{n} \cdot p^n \cdot q^n$$

$$= \frac{1-p-q}{\sqrt{1-4 \cdot p \cdot q}}$$ ... irrational

Best threshold for ratio invariances

Given a Markov chain $\mathcal{M}$ with two reward functions $rew_1, rew_2 : S \to \mathbb{N}$ with $rew_2 > \mathbf{0}$, consider their ratio

$$ratio : FinPaths \to \mathbb{Q}, \quad ratio(\pi) = \frac{rew_1(\pi)}{rew_2(\pi)}$$

examples:

- energy-utility ratio
- cost of repair mechanisms per failure
- SLA violations per day

Best threshold for ratio invariances

Given a Markov chain $\mathcal{M}$ with two reward functions $rew_1, rew_2 : S \to \mathbb{N}$ with $rew_2 > 0$, consider their ratio:

$$ratio : FinPaths \to \mathbb{Q}, \quad ratio(\pi) \;=\; \frac{rew_1(\pi)}{rew_2(\pi)}$$

$$\Pr_s(\,\Box(ratio > r)\,)$$

examples:

- energy-utility ratio
- cost of repair mechanisms per failure
- SLA violations per day

Best threshold for ratio invariances

Given a Markov chain $\mathcal{M}$ with two reward functions $rew_1, rew_2 : S \to \mathbb{N}$ with $rew_2 > \mathbf{0}$, consider their ratio:

$$ratio : FinPaths \to \mathbb{Q}, \quad ratio(\pi) = \frac{rew_1(\pi)}{rew_2(\pi)}$$

best threshold for qualitative ratio invariances:

$$\sup \{ \ r \in \mathbb{Q} : \ \Pr_s(\Box(ratio > r)) > 0 \ \}$$

$$\sup \{ \ r \in \mathbb{Q} : \ \Pr_s(\Box(ratio > r)) = 1 \ \}$$

examples:

- energy-utility ratio
- cost of repair mechanisms per failure
- SLA violations per day

Best threshold for ratio invariances

Given a Markov chain $\mathcal{M}$ with two reward functions $rew_1, rew_2 : S \to \mathbb{N}$ with $rew_2 > \mathbf{0}$, consider their ratio:

$$ratio : FinPaths \to \mathbb{Q}, \quad ratio(\pi) = \frac{rew_1(\pi)}{rew_2(\pi)}$$

best threshold for qualitative ratio invariances:

$$\textbf{sup} \left\{ r \in \mathbb{Q} : \Pr_s(\Box(ratio > r)) > 0 \right\}$$

$$\textbf{sup} \left\{ r \in \mathbb{Q} : \Pr_s(\Box(ratio > r)) = 1 \right\}$$

... are computable in polynomial time ...

[Krähmann/Schubert/Baier/Dubslaff'15]

Best threshold for ratio invariances

Given a Markov chain $\mathcal{M}$ with two reward functions $rew_1, rew_2 : S \to \mathbb{N}$ with $rew_2 > \mathbf{0}$, consider their ratio:

$$ratio : FinPaths \to \mathbb{Q}, \quad ratio(\pi) \;=\; \frac{rew_1(\pi)}{rew_2(\pi)}$$

best threshold for qualitative ratio invariances:

$$\textbf{sup} \left\{ \, r \in \mathbb{Q} : \; \Pr_s(\Box(ratio > r)) > 0 \, \right\}$$

$$\textbf{sup} \left\{ \, r \in \mathbb{Q} : \; \Pr_s(\Box(ratio > r)) = 1 \, \right\}$$

... are computable in polynomial time ...

[Krähmann/Schubert/Baier/Dubslaff'15]

Positive ratio quantiles

$$\sup \left\{ r \in \mathbb{Q} : \ \Pr_s(\Box(ratio > r)) > 0 \ \right\}$$

$$ratio = \frac{rew_1}{rew_2} : FinPaths \to \mathbb{Q} \quad \text{where } rew_2 > 0$$

Positive ratio quantiles

$$\sup \left\{ r \in \mathbb{Q} : \boxed{\Pr_s(\Box(ratio > r)) > 0} \right\}$$

- inner decision problem for fixed r is solvable in polynomial time

$$ratio = \frac{rew_1}{rew_2} : FinPaths \to \mathbb{Q} \quad \text{where } rew_2 > 0$$

Positive ratio quantiles

$$\sup \left\{ r \in \mathbb{Q} : \boxed{\Pr_s(\Box(ratio > r)) > 0} \right\}$$

- inner decision problem for fixed r is solvable in polynomial time

reduction to positive weight invariances:

$$ratio > r \quad \text{iff} \quad rew_1 - r \cdot rew_2 > 0$$

$$ratio = \frac{rew_1}{rew_2} : FinPaths \to \mathbb{Q} \quad \text{where } rew_2 > 0$$

Positive ratio quantiles

$$\sup \left\{ r \in \mathbb{Q} : \ \Pr_s(\Box(ratio > r)) > 0 \ \right\}$$

- inner decision problem for fixed r is solvable in polynomial time

reduction to positive weight invariances:

$$ratio > r \quad \text{iff} \quad \underbrace{rew_1 - r \cdot rew_2}_{\text{weight function}} > 0$$

$$ratio = \frac{rew_1}{rew_2} : FinPaths \to \mathbb{Q} \quad \text{where } rew_2 > 0$$

Positive ratio quantiles

$$\sup \left\{ r \in \mathbb{Q} : \ \Pr_s(\Box(ratio > r)) > 0 \ \right\}$$

- inner decision problem for fixed r is solvable in polynomial time

reduction to positive weight invariances:

$$ratio > r \quad \text{iff} \quad \underbrace{rew_1 - r \cdot rew_2}_{\text{weight function}} > 0$$

If $r \in \mathbb{Q}$ then pick some $c \in \mathbb{N}$ such that $(rew_1 - r \cdot rew_2) \cdot c$ is an integer weight function.

Positive ratio quantiles

$$\mathbf{sup} \left\{ r \in \mathbb{Q} : \Pr_s(\Box(ratio > r)) > 0 \right\}$$

- inner decision problem for fixed r is solvable in polynomial time
- quantile can be approximated using a binary search

$$ratio = \frac{rew_1}{rew_2} : FinPaths \to \mathbb{Q} \quad \text{where } rew_2 > 0$$

Positive ratio quantiles

$$\sup \left\{ r \in \mathbb{Q} : \Pr_s(\Box(ratio > r)) > 0 \right\}$$

- inner decision problem for fixed r is solvable in polynomial time
- quantile can be approximated using a binary search

for all finite paths π:

$$0 \leqslant ratio(\pi) \leqslant \frac{\max rew_1}{\min rew_2}$$

$ratio = \frac{rew_1}{rew_2} : FinPaths \to \mathbb{Q}$ where $rew_2 > 0$

Positive ratio quantiles

$$\sup \{ \ r \in \mathbb{Q} : \ \Pr_s(\Box(ratio > r)) > 0 \ \}$$

- inner decision problem for fixed r is solvable in polynomial time
- quantile can be approximated using a binary search and is one of the values
  - expected long-run ratio of a BSCC

If B is a BSCC then the expected long-run ratio is:

$$\frac{\mathrm{MP}_B[rew_1]}{\mathrm{MP}_B[rew_2]} \quad \text{where} \quad \mathrm{MP}_B[rew] = \left\{ \begin{array}{l} \text{mean-payoff} \\ \text{of } rew \text{ in } B \end{array} \right.$$

Positive ratio quantiles

$$\textbf{sup} \left\{ \, r \in \mathbb{Q} : \; \Pr_s(\Box(\mathit{ratio} > r)) > 0 \, \right\}$$

- inner decision problem for fixed r is solvable in polynomial time
- quantile can be approximated using a binary search and is one of the values
  - expected long-run ratio of a BSCC
  - $\mathit{ratio}(\pi)$ for a simple path π from s
  - $\mathit{ratio}(\pi)$ for a simple cycle π reachable from s

Positive ratio quantiles

$$\sup \{ \ r \in \mathbb{Q} : \ \Pr_s(\Box(ratio > r)) > 0 \ \}$$

- inner decision problem for fixed r is solvable in polynomial time

- quantile can be approximated using a binary search and is one of the values

  - expected long-run ratio of a BSCC
  - $ratio(\pi)$ for a simple path π from s
  - $ratio(\pi)$ for a simple cycle π reachable from s

finitely many values

Positive ratio quantiles

$$\textbf{sup}\ \{\ r \in \mathbb{Q} :\ \Pr_s(\Box(\mathit{ratio} > r)) > 0\ \}$$

- inner decision problem for fixed r is solvable in polynomial time
- quantile can be approximated using a binary search and is one of the values ... and therefore rational
  - expected long-run ratio of a BSCC
  - $\mathit{ratio}(\pi)$ for a simple path π from s
  - $\mathit{ratio}(\pi)$ for a simple cycle π reachable from s

finitely many values

Positive ratio quantiles

$$\sup \{ \ r \in \mathbb{Q} : \ \Pr_s(\Box(ratio > r)) > 0 \ \}$$

- inner decision problem for fixed r is solvable in polynomial time
- quantile can be approximated using a binary search and is one of the values ... and therefore rational
  - expected long-run ratio of a BSCC
  - $ratio(\pi)$ for a simple path π from s
  - $ratio(\pi)$ for a simple cycle π reachable from s

finitely many values

- computation using the continued-fraction method

Positive ratio quantiles

$$\sup \left\{ \, r \in \mathbb{Q} : \; \Pr_s(\Box(ratio > r)) > 0 \, \right\} \; = \; \frac{c}{d}$$

where $c, d \in \mathbb{N}$ with $d > 0$

- quantile can be approximated using a binary search and is one of the values ... and therefore rational
  - expected long-run ratio of a BSCC
  - $ratio(\pi)$ for a simple path π from s
  - $ratio(\pi)$ for a simple cycle π reachable from s

  finitely many values
- computation using the continued-fraction method

Positive ratio quantiles

$$\mathbf{sup}\ \{\ r \in \mathbb{Q} :\ \Pr_s(\Box(\mathit{ratio} > r)) > 0\ \}\ =\ \frac{c}{d}$$

where $d \leqslant D = \max\{ \max_B d_B,\ |S| \cdot \max \mathit{rew}_2 \}$

- quantile can be approximated using a binary search and is one of the values ... and therefore rational
  - expected long-run ratio c_B/d_B of BSCC B
  - $\mathit{ratio}(\pi)$ for a simple path π from s
  - $\mathit{ratio}(\pi)$ for a simple cycle π reachable from s
- computation using the continued-fraction method

Positive ratio quantiles

$$\sup \left\{ \, r \in \mathbb{Q} : \; \Pr_s(\,\Box(ratio > r)\,) > 0 \, \right\} \; = \; \frac{c}{d}$$

where $d \leqslant D = \max\left\{ \max_B d_B, \; |S| \cdot \max rew_2 \right\}$

1. compute an approximation p of the quantile up to precision $\varepsilon = 1/2D^2$

$$\left| \frac{c}{d} - p \right| < \varepsilon$$

Positive ratio quantiles

$$\sup \left\{ r \in \mathbb{Q} : \Pr_s(\Box(ratio > r)) > 0 \right\} = \frac{c}{d}$$

where $d \leqslant D = \max\left\{ \max_B d_B, \ |S| \cdot \max rew_2 \right\}$

1. compute an approximation p of the quantile up to precision $\varepsilon = 1/2D^2$

The quantile is the best rational approximation of p with denominator at most D

$$\left|\frac{c}{d} - p\right| < \varepsilon$$

Positive ratio quantiles

$$\sup \{ r \in \mathbb{Q} : \Pr_s(\Box(ratio > r)) > 0 \} = \frac{c}{d}$$

where $d \leqslant D = \max\{ \max_B d_B, \ |S| \cdot \max rew_2 \}$

1. compute an approximation p of the quantile up to precision $\varepsilon = 1/2D^2$

The quantile is the best rational approximation of p with denominator at most D, i.e., if $a, b \in \mathbb{N}$ with $0 < b \leqslant D$ then:

$$\left| \frac{a}{b} - p \right| < \varepsilon \quad \text{iff} \quad \frac{a}{b} = \frac{c}{d}$$

Positive ratio quantiles

$$\sup \{ r \in \mathbb{Q} : \Pr_s(\Box(ratio > r)) > 0 \} = \frac{c}{d}$$

where $d \leqslant D = \max\{ \max_B d_B, \ |S| \cdot \max rew_2 \}$

1. compute an approximation p of the quantile up to precision $\varepsilon = 1/2D^2$
2. apply the continued-fraction method to p

The quantile is the best rational approximation of p with denominator at most D, i.e., if $a, b \in \mathbb{N}$ with $0 < b \leqslant D$ then:

$$\left|\frac{a}{b} - p\right| < \varepsilon \quad \text{iff} \quad \frac{a}{b} = \frac{c}{d}$$

Positive ratio quantiles

$$\sup \left\{ r \in \mathbb{Q} : \Pr_s(\Box(ratio > r)) > 0 \right\} = \frac{c}{d}$$

where $d \leqslant D = \max\left\{ \max_B d_B, \ |S| \cdot \max rew_2 \right\}$

1. compute an approximation p of the quantile up to precision $\varepsilon = 1/2D^2$

2. apply the continued-fraction method to p

$$p = p_1 + \cfrac{1}{p_2 + \cfrac{1}{p_3 + \cfrac{1}{p_4 + \cfrac{1}{\cdots}}}}$$

Positive ratio quantiles

$$\textbf{sup} \left\{ \, r \in \mathbb{Q} : \; \Pr_s(\Box(\mathit{ratio} > r)) > 0 \, \right\} \;=\; \frac{c}{d}$$

where $d \leqslant D = \max\left\{ \max_B d_B, \; |S| \cdot \max \mathit{rew}_2 \right\}$

1. compute an approximation p of the quantile up to precision $\varepsilon = 1/2D^2$

2. apply the continued-fraction method to p

   [Grötschel/Lovász/Schrijver'87]

Positive ratio quantiles

$$\sup \left\{ r \in \mathbb{Q} : \Pr_s(\Box(ratio > r)) > 0 \right\} = \frac{c}{d}$$

where $d \leqslant D = \max\left\{ \max_B d_B, \ |S| \cdot \max rew_2 \right\}$

1. compute an approximation p of the quantile up to precision $\varepsilon = 1/2D^2$

2. apply the continued-fraction method to p

   [Grötschel/Lovász/Schrijver'87]

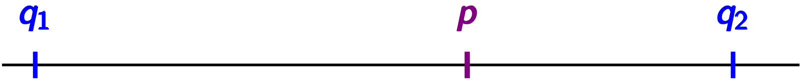
Positive ratio quantiles

$$\mathbf{sup}\ \{\ r \in \mathbb{Q} :\ \Pr_s(\Box(\mathit{ratio} > r)) > 0\ \}\ =\ \frac{c}{d}$$

where $d \leqslant D = \max\left\{ \max_B d_B,\ |S| \cdot \max \mathit{rew}_2 \right\}$

1. compute an approximation p of the quantile up to precision $\varepsilon = 1/2D^2$

2. apply the continued-fraction method to p
   [Grötschel/Lovász/Schrijver'87]



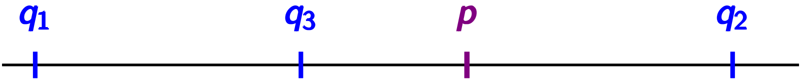
Positive ratio quantiles

$$\sup \left\{ r \in \mathbb{Q} : \Pr_s(\Box(ratio > r)) > 0 \right\} = \frac{c}{d}$$

where $d \leqslant D = \max\left\{ \max_B d_B, \; |S| \cdot \max rew_2 \right\}$

1. compute an approximation p of the quantile up to precision $\varepsilon = 1/2D^2$

2. apply the continued-fraction method to p
   [Grötschel/Lovász/Schrijver'87]



Positive ratio quantiles

$$\sup \left\{ r \in \mathbb{Q} : \Pr_s(\Box(ratio > r)) > 0 \right\} = \frac{c}{d}$$

where $d \leqslant D = \max\left\{ \max_B d_B, \ |S| \cdot \max rew_2 \right\}$

1. compute an approximation p of the quantile up to precision $\varepsilon = 1/2D^2$

2. apply the continued-fraction method to p

   [Grötschel/Lovász/Schrijver'87]

Positive ratio quantiles

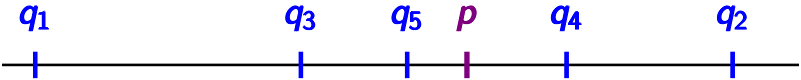
$$\sup \left\{ \, r \in \mathbb{Q} : \, \Pr_s(\Box(\mathit{ratio} > r)) > 0 \, \right\} \; = \; \frac{c}{d}$$

where $d \leqslant D = \max\left\{ \max_B d_B, \; |S| \cdot \max \mathit{rew}_2 \right\}$

1. compute an approximation p of the quantile up to precision $\varepsilon = 1/2D^2$

2. apply the continued-fraction method to p

   [Grötschel/Lovász/Schrijver'87]



Positive ratio quantiles

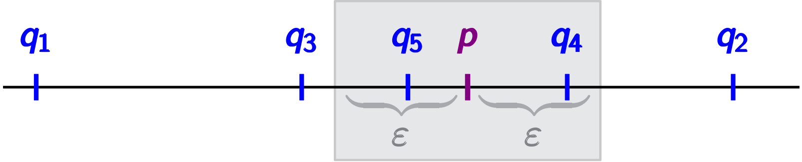
$$\sup \{ r \in \mathbb{Q} : \Pr_s(\Box(ratio > r)) > 0 \} = \frac{c}{d}$$

where $d \leqslant D = \max\{ \max_B d_B, \ |S| \cdot \max rew_2 \}$

1. compute an approximation p of the quantile up to precision $\varepsilon = 1/2D^2$

2. apply the continued-fraction method to p

[Grötschel/Lovász/Schrijver'87]



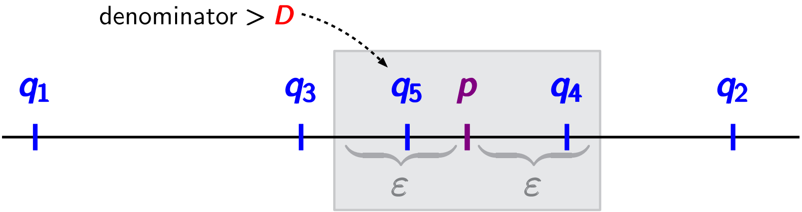
Positive ratio quantiles

$$\sup \{ \ r \in \mathbb{Q} : \ \Pr_s(\Box(ratio > r)) > 0 \ \} \ = \ \frac{c}{d}$$

where $d \leqslant D = \max\{ \max_B d_B, \ |S| \cdot \max rew_2 \}$

1. compute an approximation p of the quantile up to precision $\varepsilon = 1/2D^2$

2. apply the continued-fraction method to p



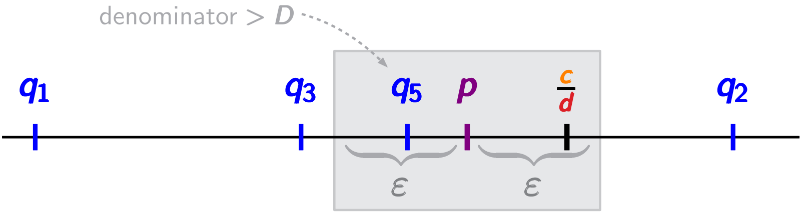
Positive ratio quantiles

$$\sup \left\{ r \in \mathbb{Q} : \Pr_s(\Box(ratio > r)) > 0 \right\} = \frac{c}{d}$$

where $d \leqslant D = \max\left\{ \max_B d_B,\ |S| \cdot \max rew_2 \right\}$

1. compute an approximation p of the quantile up to precision $\varepsilon = 1/2D^2$

2. apply the continued-fraction method to p



Polynomially computable ratio quantiles in MC

qualitative quantiles for ratio invariances:

$$\textbf{sup}\,\{\, r \in \mathbb{Q} \,:\, \Pr_s\big(\Box(\textit{ratio} > r) \wedge \varphi\,\big) \,>\, \mathbf{0}\,\}$$

$$\textbf{sup}\,\{\, r \in \mathbb{Q} \,:\, \Pr_s\big(\Box(\textit{ratio} > r) \wedge \varphi\,\big) \,=\, \mathbf{1}\,\}$$

where φ is a reachability, Rabin or Streett condition

Polynomially computable ratio quantiles in MC

qualitative quantiles for ratio invariances:

$$\textsf{sup}\left\{ r \in \mathbb{Q} \,:\, \Pr_s\big(\Box(\mathit{ratio} > r) \wedge \varphi\big) \,>\, 0 \right\}$$

$$\textsf{sup}\left\{ r \in \mathbb{Q} \,:\, \underbrace{\Pr_s\big(\Box(\mathit{ratio} > r) \wedge \varphi\big) \,=\, 1}_{\Pr_s(\varphi) = 1 \text{ and } s \not\models \exists\Diamond(\mathit{wgt}_r \leqslant 0)} \right\}$$

where $\mathit{wgt}_r = \mathit{cost} - r \cdot \mathit{util}$

... binary search for maximal r and shortest-path algorithms ...

where φ is a reachability, Rabin or Streett condition

Polynomially computable ratio quantiles in MC

qualitative quantiles for ratio invariances:

$$\sup \{ r \in \mathbb{Q} : \Pr_s(\Box(ratio > r) \wedge \varphi) > 0 \}$$

$$\sup \{ r \in \mathbb{Q} : \Pr_s(\Box(ratio > r) \wedge \varphi) = 1 \}$$

qualitative and quantitative quantiles for long-run ratios:

$$\sup \{ r \in \mathbb{Q} : \Pr_s(\Box\Diamond(ratio > r) \wedge \varphi) = 1 \}$$

$$\sup \{ r \in \mathbb{Q} : \Pr_s(\Box\Diamond(ratio > r) \wedge \varphi) > q \}$$

where φ is a reachability, Rabin or Streett condition

Polynomially computable ratio quantiles in MC

qualitative quantiles for ratio invariances:

$$\sup \{ r \in \mathbb{Q} : \Pr_s(\Box(ratio > r) \wedge \varphi) > 0 \}$$

$$\sup \{ r \in \mathbb{Q} : \Pr_s(\Box(ratio > r) \wedge \varphi) = 1 \}$$

qualitative and quantitative quantiles for long-run ratios:

$$\sup \{ r \in \mathbb{Q} : \Pr_s(\Box\Diamond(ratio > r) \wedge \varphi) = 1 \}$$

$$\sup \{ r \in \mathbb{Q} : \Pr_s(\Box\Diamond(ratio > r) \wedge \varphi) > q \}$$

$$= \min \{ r \in \mathbb{Q} : \Pr_s(\Diamond C_r) > q \}$$

where C_r = union of "good" BSCCs B with $lrrat(B) \geqslant r$

Tutorial: Probabilistic Model Checking

Discrete-time Markov chains (DTMC)

* basic definitions
* probabilistic computation tree logic PCTL/PCTL*
* rewards, cost-utility ratios, weights
* conditional probabilities

Markov decision processes (MDP)

* basic definitions
* PCTL/PCTL* model checking
* fairness
* conditional probabilities
* rewards, quantiles
* mean-payoff
* expected accumulated weights

Conditional probabilities

Conditional probabilities

- useful for various multi-objective properties

  e.g. analyze the gained utility for a given energy budget

$$\Pr_s\big(\Diamond_{\geqslant u}\, goal \mid \Diamond^{\leqslant e}\, goal\big) \quad \text{or}$$

$$\mathrm{ExpUtil}_s\big(\oplus\, goal \mid \Diamond^{\leqslant e}\, goal\big)$$

$\Diamond_{\geqslant u}\, goal$ "gained utility for reaching the goal is at least u"

$\Diamond^{\leqslant e}\, goal$ "consumed energy until reaching the goal is at most e"

Conditional probabilities

- useful for various multi-objective properties

  e.g. analyze the gained utility for a given energy budget

$$\Pr_s\big(\Diamond_{\geqslant u}\, goal \mid \Diamond^{\leqslant e}\, goal\big) \quad \text{or}$$

$$\mathrm{ExpUtil}_s\big(\oplus\, goal \mid \Diamond^{\leqslant e}\, goal\big)$$

- useful for failure diagnosis

  e.g. study the impact of failures and cost of repair mechanisms in resilient systems

$$\Pr_s\big(\Diamond goal \mid \Diamond failure\big) \quad \text{or}$$

$$\mathrm{ExpCost}_s\big(\oplus\, goal \mid \Diamond failure\big)$$

Conditional probabilities

for Markov chains:

$$\Pr_s^{\mathcal{M}}(\varphi \mid \psi) \;=\; \frac{\Pr_s^{\mathcal{M}}(\varphi \wedge \psi)}{\Pr_s^{\mathcal{M}}(\psi)}$$

provided $\Pr_s^{\mathcal{M}}(\psi) > 0$

Conditional probabilities

for Markov chains:

$$\mathrm{Pr}^{\mathcal{M}}_{s}(\varphi \mid \psi) = \frac{\mathrm{Pr}^{\mathcal{M}}_{s}(\varphi \wedge \psi)}{\mathrm{Pr}^{\mathcal{M}}_{s}(\psi)}$$

- discrete MCs and PCTL [Andrés/Rossum'08] [Ji/Wu/Chen'13]
- continuous-time MCs and CSL [Gao/Xu/Zhan/Zhang'13]

PCTL: probabilistic computation tree logic

CSL: continuous stochastic logic

Conditional probabilities

for Markov chains:

$$\Pr_s^{\mathcal{M}}(\varphi \mid \psi) \;=\; \frac{\Pr_s^{\mathcal{M}}(\varphi \wedge \psi)}{\Pr_s^{\mathcal{M}}(\psi)}$$

- discrete MCs and PCTL [Andrés/Rossum'08] [Ji/Wu/Chen'13]
- continuous-time MCs and CSL [Gao/Xu/Zhan/Zhang'13]

---

transformation-based approach for LTL conditions

MC $\mathcal{M} \rightsquigarrow$ MC $\mathcal{M}_\psi$: [Baier/Klein/Klüppelholz/Märcker'14]

$$\Pr_s^{\mathcal{M}}(\varphi \mid \psi) \;=\; \Pr_s^{\mathcal{M}_\psi}(\varphi)$$

Transformation-based approach for MC

given: Markov chain $\mathcal{M} = (S, P)$ and $\psi = \Diamond G$

define Markov chain $\mathcal{M}_\psi$ s.t. for all LTL formulas φ

$$\Pr_s^{\mathcal{M}}(\varphi \mid \Diamond G) \;=\; \Pr_s^{\mathcal{M}_\psi}(\varphi)$$

LTL: linear temporal logic

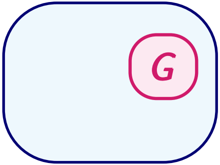
Transformation-based approach for MC

given: Markov chain $\mathcal{M} = (S, P)$ and $\psi = \Diamond G$

define Markov chain $\mathcal{M}_\psi$ s.t. for all LTL formulas φ

$$\Pr_s^{\mathcal{M}}(\varphi \mid \Diamond G) \;=\; \Pr_s^{\mathcal{M}_\psi}(\varphi)$$

MC $\mathcal{M}$



LTL: linear temporal logic

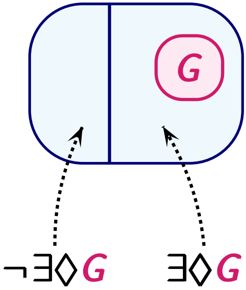
Transformation-based approach for MC

given: Markov chain $\mathcal{M} = (S, P)$ and $\psi = \Diamond G$

define Markov chain $\mathcal{M}_\psi$ s.t. for all LTL formulas φ

$$\Pr_s^{\mathcal{M}}(\varphi \mid \Diamond G) = \Pr_s^{\mathcal{M}_\psi}(\varphi)$$

MC $\mathcal{M}$



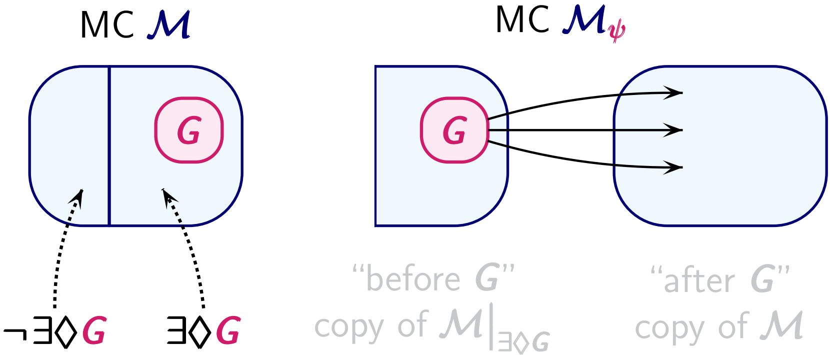
LTL: linear temporal logic

Transformation-based approach for MC

given: Markov chain $\mathcal{M} = (S, P)$ and $\psi = \Diamond G$

define Markov chain $\mathcal{M}_\psi$ s.t. for all LTL formulas φ

$$\Pr_s^{\mathcal{M}}(\varphi \mid \Diamond G) = \Pr_s^{\mathcal{M}_\psi}(\varphi)$$

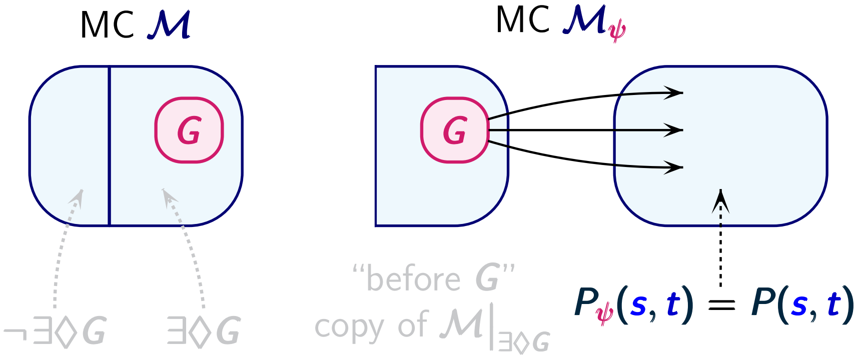


Transformation-based approach for MC

given: Markov chain $\mathcal{M} = (S, P)$ and $\psi = \Diamond G$

define Markov chain $\mathcal{M}_\psi$ s.t. for all LTL formulas φ

$$\Pr_s^{\mathcal{M}}(\varphi \mid \Diamond G) = \Pr_s^{\mathcal{M}_\psi}(\varphi)$$

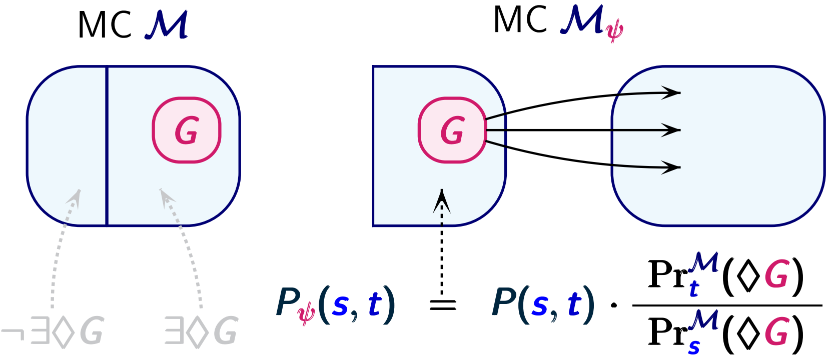


Transformation-based approach for MC

given: Markov chain $\mathcal{M} = (S, P)$ and $\psi = \Diamond G$

define Markov chain $\mathcal{M}_\psi$ s.t. for all LTL formulas φ

$$\Pr_s^{\mathcal{M}}(\varphi \mid \Diamond G) = \Pr_s^{\mathcal{M}_\psi}(\varphi)$$



$$P_\psi(s,t) = P(s,t) \cdot \frac{\Pr_t^{\mathcal{M}}(\Diamond G)}{\Pr_s^{\mathcal{M}}(\Diamond G)}$$

given: Markov chain $\mathcal{M} = (S, P)$ and $\psi = \Diamond G$

define Markov chain $\mathcal{M}_\psi$ s.t. for all LTL formulas φ

$$\Pr_s^{\mathcal{M}}(\varphi \mid \Diamond G) \;=\; \Pr_s^{\mathcal{M}_\psi}(\varphi)$$

... can be generalized for other temporal conditions ψ

either by adapting the definition of $\mathcal{M}_\psi$ or
by using an ω-automaton for LTL conditions

Transformation-based approach for MC

given: Markov chain $\mathcal{M} = (S, P)$ and $\psi = \Diamond G$

define Markov chain $\mathcal{M}_\psi$ s.t. for all LTL formulas φ

$$\Pr_s^{\mathcal{M}}(\varphi \mid \Diamond G) = \Pr_s^{\mathcal{M}_\psi}(\varphi)$$

... can be generalized for other temporal conditions ψ

same method applicable for conditional expectations

$$\mathbb{E}_s^{\mathcal{M}}(f \mid \psi) = \mathbb{E}_s^{\mathcal{M}_\psi}(f')$$

e.g.: $\mathbb{E}_s^{\mathcal{M}}($ "energy until reaching the goal" $\mid \Diamond goal$ $)$

Tutorial: Probabilistic Model Checking

Discrete-time Markov chains (DTMC)

* basic definitions
* probabilistic computation tree logic PCTL/PCTL*
* rewards, cost-utility ratios, weights
* conditional probabilities

Markov decision processes (MDP)

* basic definitions
* PCTL/PCTL* model checking
* fairness
* conditional probabilities
* rewards, quantiles
* mean-payoff
* expected accumulated weights