Faster Statistical Model Checking for Unbounded Temporal Properties

Przemysław Daca Thomas A. Henzinger Jan Křetínský Tatjana Petrov

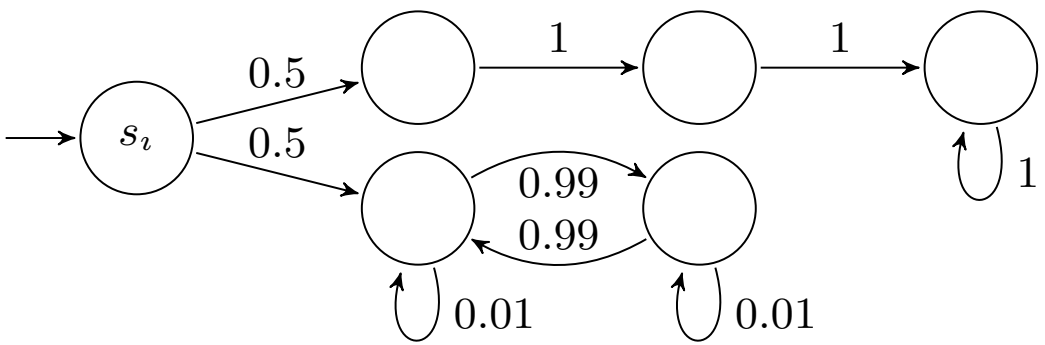
April 4, 2016

TACAS'16

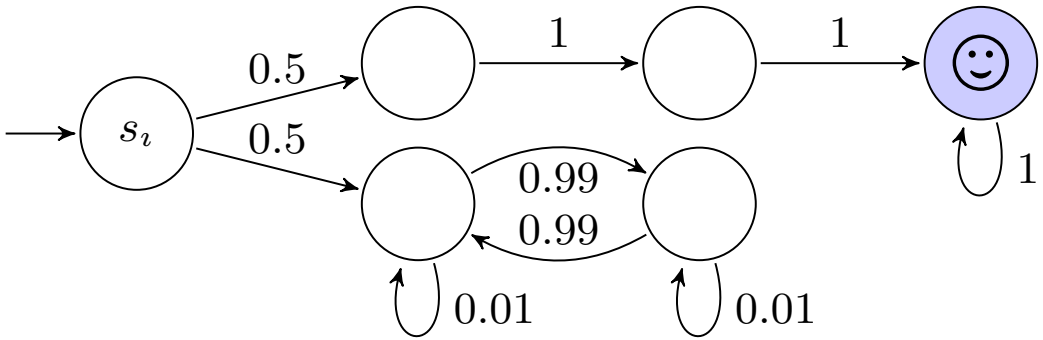
Markov chains

A *Markov chain (MC)* is a tuple $\mathcal{M} = (S, \mathbf{P}, s_i)$, where

- S is a finite set of states,
- $\mathbf{P} : S \times S \to [0, 1]$ is the transition probability matrix, s.t. $\forall s \in S : \sum_{s' \in S} \mathbf{P}(s, s') = 1$,
- $s_i \in S$ is the initial state.



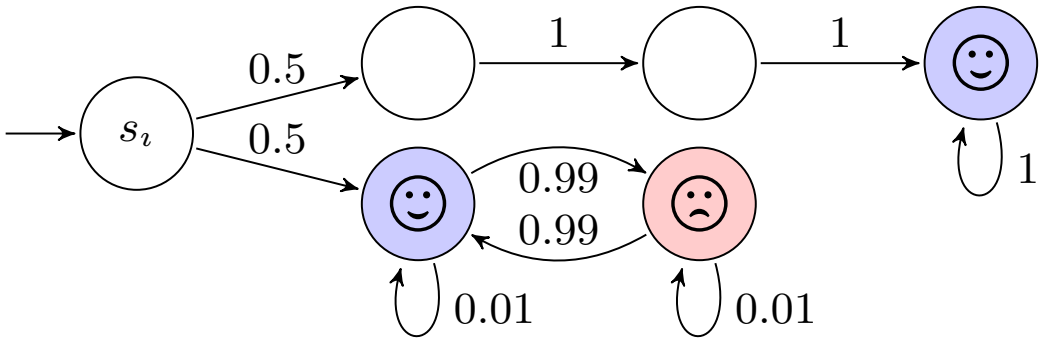
Unbounded properties



1. **Reachability**, e.g.:

$$\mathbb{P}(\Diamond ☺) = ?$$

Unbounded properties



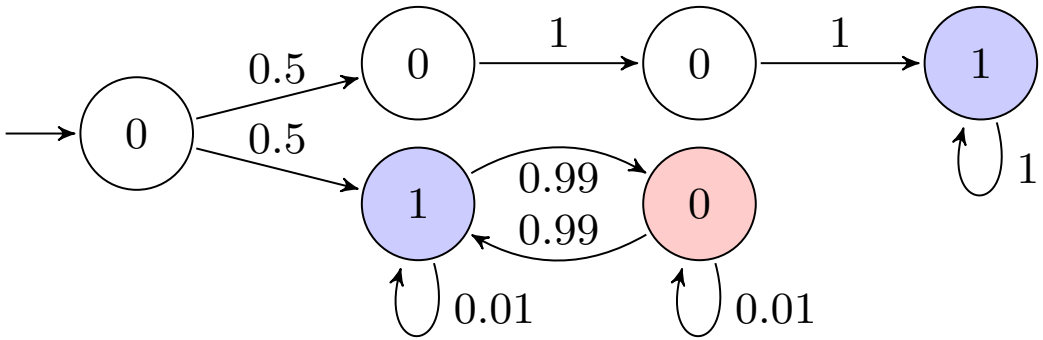
1. **Reachability**, e.g.:

$$\mathbb{P}(\Diamond \text{☺}) = ?$$

2. **Linear temporal logic**, e.g.:

$$\mathbb{P}(\Diamond \Box \text{☹}) = ?$$

Unbounded properties



1. **Reachability**, e.g.:

$$\mathbb{P}(\Diamond \text{☺}) =?$$

2. **Linear temporal logic**, e.g.:

$$\mathbb{P}(\Diamond \Box \text{☹}) =?$$

3. **Mean payoff**. State rewards $r : S \to [0, 1]$.

$$\text{MP} := \lim_{n \to \infty} \mathbb{E}\left[\frac{1}{n} \sum_{i=1}^{n} r(S_i)\right] =?$$

Model checking vs. statistical model checking

Two approaches to checking properties of Markov chains.

Numerical (model checking)	**Statistical model checking (SMC)**
by solving equations	by sampling random paths
precise answer	approximate answer
complexity grows with state space	complexity depends on the precision of answer
requires full knowledge of the system	requires none/partial information about the system

Statistical Model Checking

Input

- a Markov chain $\mathcal{M}$, property φ
- threshold p, indifference region $\epsilon > 0$,
- type I, II errors $\alpha, \beta > 0$.

Output

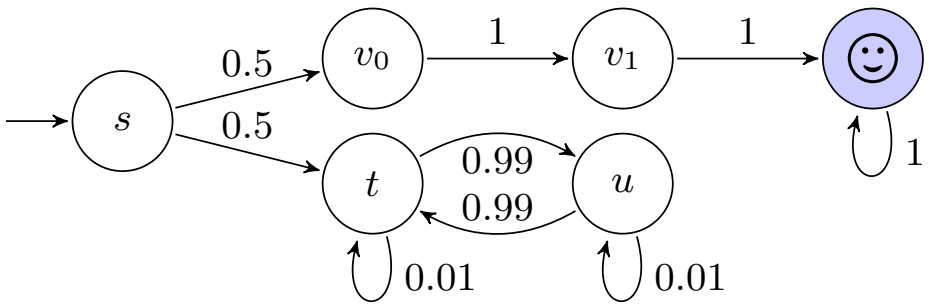
- if $\mathbb{P}(\varphi) \geq p + \epsilon$, return YES with probability at least $1 - \alpha$.
- if $\mathbb{P}(\varphi) \leq p - \epsilon$, return NO with probability at least $1 - \beta$.

Generic algorithm

repeat
sample finite path π_i from $\mathcal{M}$,
$x_i = \begin{cases} 1 & \pi_i \text{ satisfies } \varphi \\ 0 & \pi_i \text{ does not satisfies } \varphi \end{cases}$
until $x_0, \cdots, x_n$ are enough to conclude YES or NO.

Time-bounded property

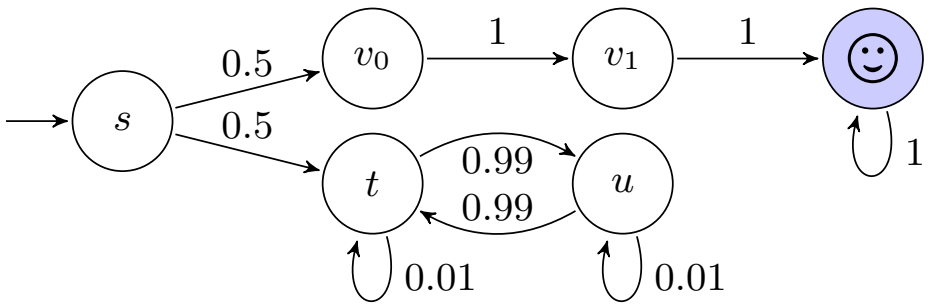
$$\mathbb{P}(\Diamond^{\leq 5} ☺) \leq 0.24 \quad \text{vs.} \quad \mathbb{P}(\Diamond^{\leq 5} ☺) \geq 0.26$$



random path	result
$s, v_0, v_1, ☺$	$x_0 = 1$
	...

Time-bounded property

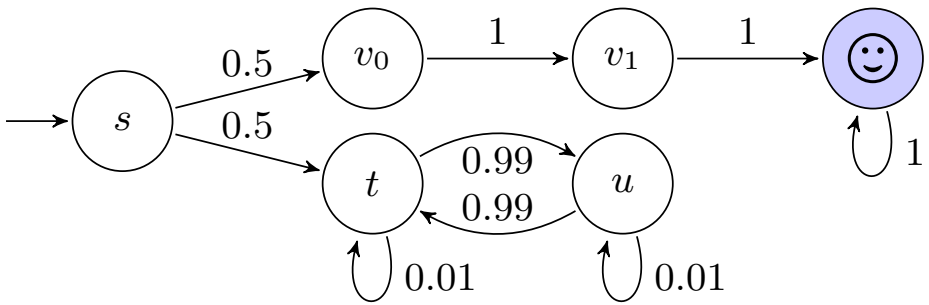
$$\mathbb{P}(\Diamond^{\leq 5} ☺) \leq 0.24 \quad \text{vs.} \quad \mathbb{P}(\Diamond^{\leq 5} ☺) \geq 0.26$$



random path	result
$s, v_0, v_1, ☺$	$x_0 = 1$
s, t, u, t, u	$x_1 = 0$
	...

Time-bounded property

$$\mathbb{P}(\Diamond^{\leq 5} ☺) \leq 0.24 \quad \text{vs.} \quad \mathbb{P}(\Diamond^{\leq 5} ☺) \geq 0.26$$

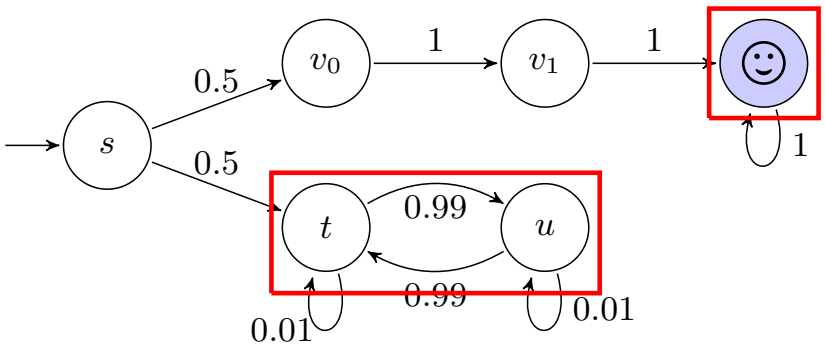


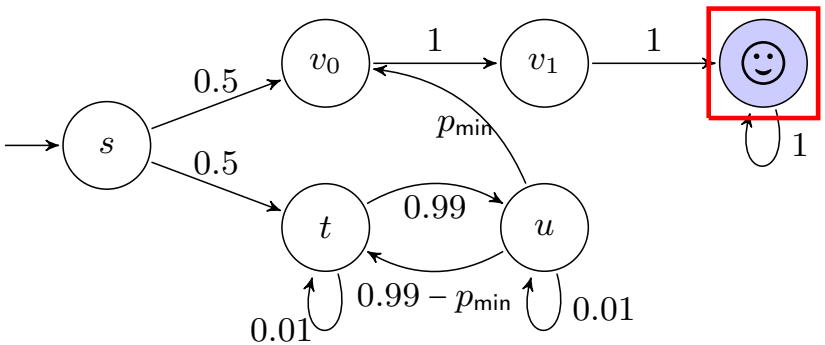
random path	result
$s, v_0, v_1, ☺$	$x_0 = 1$
s, t, u, t, u	$x_1 = 0$
s, t, u, t, u	$x_2 = 0$
$s, v_0, v_1, ☺$	$x_3 = 1$
	$\ldots$

SMC of unbouded properties

Unbounded reachability:

$$\mathbb{P}(\Diamond ☺) \leq 0.24 \quad \text{vs.} \quad \mathbb{P}(\Diamond ☺) \geq 0.26$$





random path	result
$s, t, u, t, u, t, u, t, u, \cdots$	can we still reach ☺?

Problem: what is the stopping criterion for paths?

We detect bottom strongly connected components (BSCC).

Contribution

- Monitor paths on-the-fly to detect BSCCs,
- intuition: if a path is stuck in set C of states, then with high confidence S is a BSCC,
- requires the minimum transition probability $p_{\min}$.

Applications

SMC algorithms for:

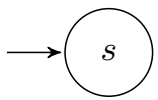
- reachability, e.g. $\mathbb{P}(\Diamond ☺)$,
- linear temporal logic, e.g. $\mathbb{P}(\Diamond \Box ☹)$
- mean-payoff objectives MP $:= \lim_{n\to\infty} \mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n} r(S_i)\right]$

Candidate

Set C of state is a k-candidate if:

1) C is the BSCC of the path
2) every state $s \in C$ is visited $\geq k$ times on the path.

The graph of $\pi = s$ is:

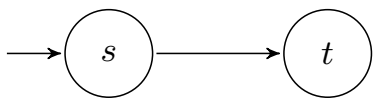


Candidate

Set C of state is a k-candidate if:

1) C is the BSCC of the path
2) every state $s \in C$ is visited $\geq k$ times on the path.

The graph of $\pi = s, t$ is:

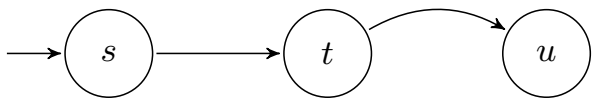


Candidate

Set C of state is a k-candidate if:

1) C is the BSCC of the path
2) every state $s \in C$ is visited $\geq k$ times on the path.

The graph of $\pi = s, t, u$ is:

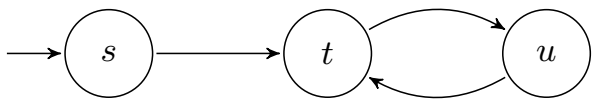


Candidate

Set C of state is a k-candidate if:

1) C is the BSCC of the path
2) every state $s \in C$ is visited $\geq k$ times on the path.

The graph of $\pi = s, t, u, t$ is:

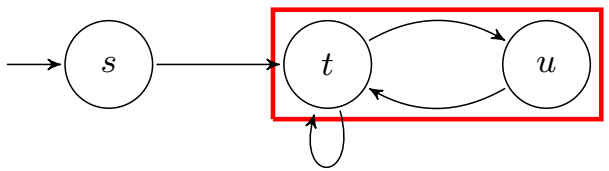


Candidate

Set C of state is a k-candidate if:

1) C is the BSCC of the path
2) every state $s \in C$ is visited $\geq k$ times on the path.

The graph of $\pi = s, t, u, t, t, u, t, u$ is:



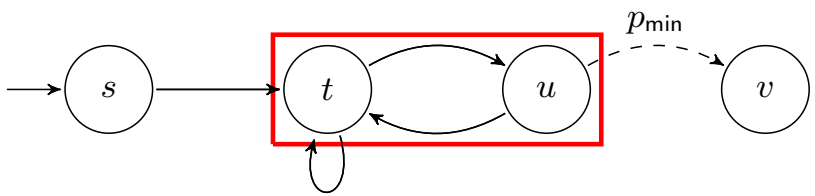
$\{t, u\}$ is the 2-candidate of the path π.

Candidate

Set C of state is a k-candidate if:

1) C is the BSCC of the path
2) every state $s \in C$ is visited $\geq k$ times on the path.

The graph of $\pi = s, t, u, t, t, u, t, u$ is:



$\{t, u\}$ is the 2-candidate of the path π.

$\{t, u\}$ **is not a BSCC** with probability $\leq (1 - p_{\min})^2$

Candidate error

Lemma

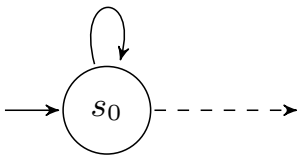
For every set C of states that is **not a BSCC**, $k \in \mathbb{N}$:

$$\mathbb{P}(C \text{ becomes a } k\text{-candidate}) \leq (1 - p_{\min})^k$$

BSCC detection

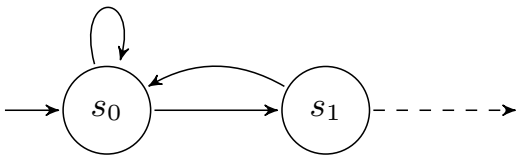
Declare k-candidate a BSCC if k is large enough.

Evolution of candidates



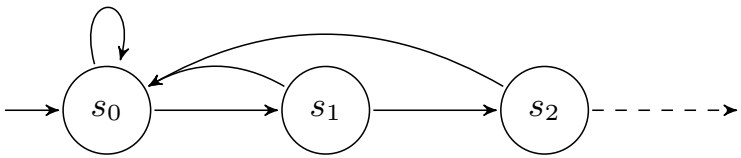
path	probability of false candidate
s_0^+	$(1 - p_{\mathsf{min}})^k$

Evolution of candidates



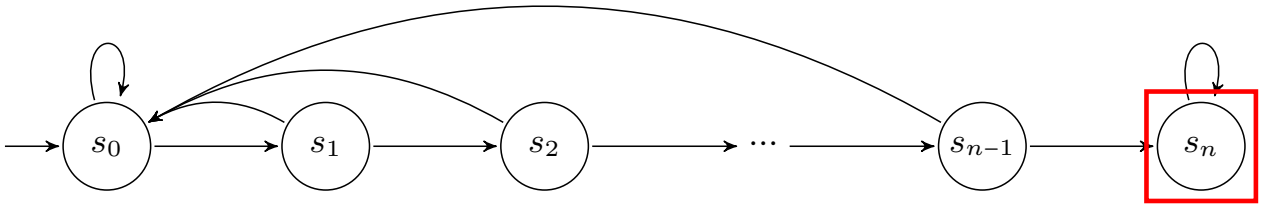
path	probability of false candidate
s_0^+	$(1-p_{\mathsf{min}})^k$
$\{s_0, s_1\}^+$	$(1-p_{\mathsf{min}})^{k'}$

Evolution of candidates



path	probability of false candidate
s_0^+	$(1-p_{\mathsf{min}})^k$
$\{s_0, s_1\}^+$	$(1-p_{\mathsf{min}})^{k'}$
$\{s_0, s_1, s_2\}^+$	$(1-p_{\mathsf{min}})^{k''}$

Evolution of candidates



path	probability of false candidate
s_0^+	$(1-p_{\min})^k$
$\{s_0, s_1\}^+$	$(1-p_{\min})^{k'}$
$\{s_0, s_1, s_2\}^+$	$(1-p_{\min})^{k''}$
$\cdots$	$\cdots$
$\{s_0, \cdots, s_n\}^+$	detect BSCC $\{s_n\}$

Choose k's s.t. total probability of declaring false candidate is $\leq \delta$.

SMC for unbounded reachability

Algorithm

repeat

  repeat sample finite path π_i from $\mathcal{M}$,

  until π_i reaches ☺ or reaches a BSCC without ☺

  $$x_i = \begin{cases} 1 & \pi_i \text{ reaches } ☺ \\ 0 & \pi_i \text{ doesn't reach } ☺ \end{cases}$$

until $x_0, \cdots, x_n$ are enough to conclude YES or NO.

Extension to Linear Temporal Logic

Given an LTL formula φ, we are interested in $\mathbb{P}(\varphi)$.

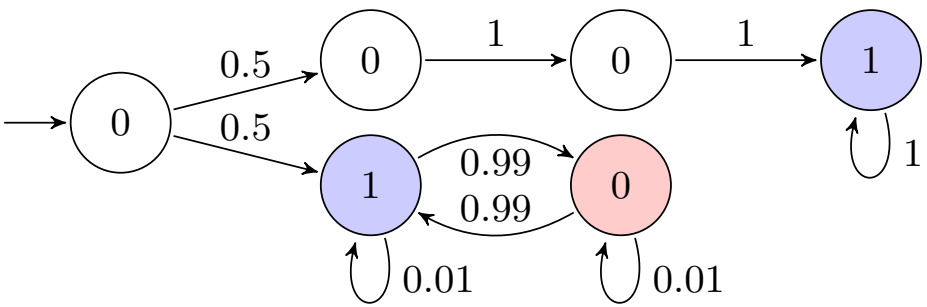
Solution (sketch)

- translate φ to a deterministic Rabin automata $\mathcal{A}_\varphi$,
- use $\mathcal{A}_\varphi$ to find accepting BSCCs in $\mathcal{M} \otimes \mathcal{A}_\varphi$,
- $\mathbb{P}(\varphi)$ = probability of reaching an accepting BSCCs.

Mean payoff

MC with state rewards $r : S \to [0, 1]$.

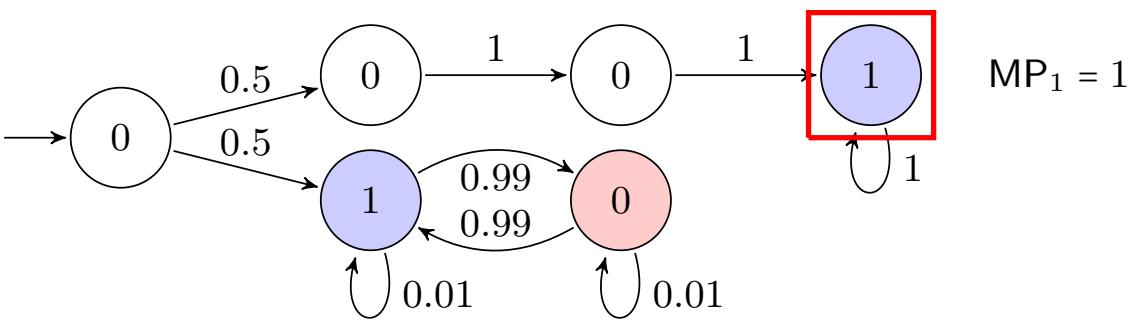
$$\text{MP} := \lim_{n \to \infty} \mathbb{E}\left[\frac{1}{n} \sum_{i=1}^{n} r(S_i)\right]$$



Mean payoff

MC with state rewards $r : S \to [0, 1]$.

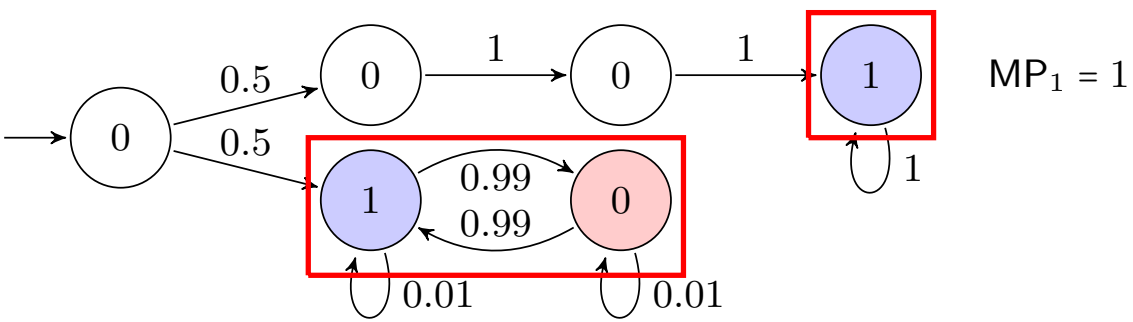
$$\mathsf{MP} := \lim_{n \to \infty} \mathbb{E}\left[\frac{1}{n} \sum_{i=1}^{n} r(S_i)\right]$$



Mean payoff

MC with state rewards $r : S \to [0, 1]$.

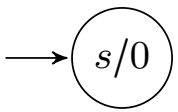
$$\mathsf{MP} := \lim_{n \to \infty} \mathbb{E}\left[\frac{1}{n} \sum_{i=1}^{n} r(S_i)\right]$$



$$\mathsf{MP} = 0.5 \cdot \mathsf{MP}_1 + 0.5 \cdot \mathsf{MP}_2$$

Mean payoff

The graph of $\pi = s$, is:

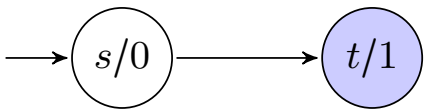


MP for a path

- Estimate transtition probabilities
- Compute empirical mean payoff for the BSCC.
- Use robustness theorem to bound error (Chaterjee'12).

Mean payoff

The graph of $\pi = s, t$ is:

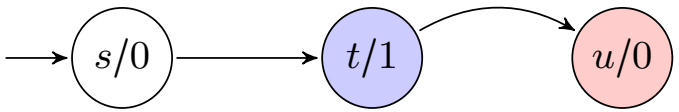


MP for a path

- Estimate transtition probabilities
- Compute empirical mean payoff for the BSCC.
- Use robustness theorem to bound error (Chaterjee'12).

Mean payoff

The graph of $\pi = s, t, u$ is:

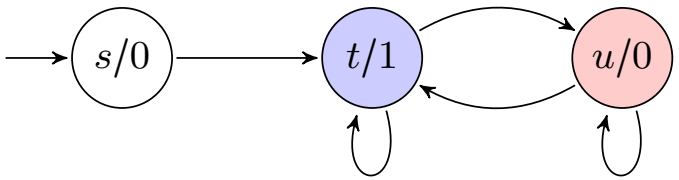


MP for a path

- Estimate transtition probabilities
- Compute empirical mean payoff for the BSCC.
- Use robustness theorem to bound error (Chaterjee'12).

Mean payoff

The graph of $\pi = s, t, u, t, t, u, u$ is:

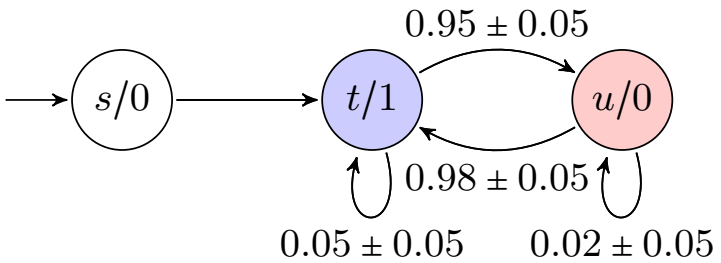


MP for a path

- Estimate transtition probabilities
- Compute empirical mean payoff for the BSCC.
- Use robustness theorem to bound error (Chaterjee'12).

Mean payoff

The graph of $\pi = s, t, u, t, t, u, u, t, u, t, u, t, \cdots$ is:

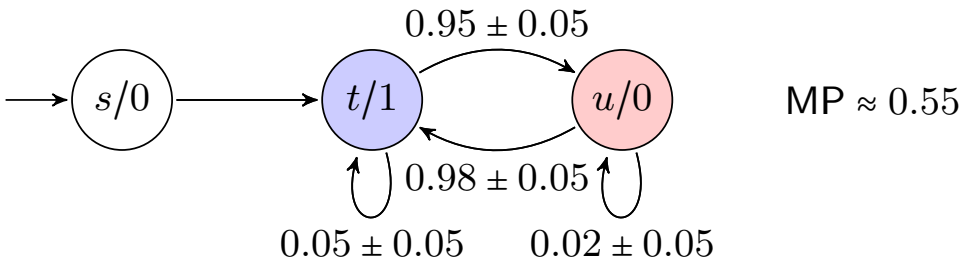


MP for a path

- Estimate transtition probabilities
- Compute empirical mean payoff for the BSCC.
- Use robustness theorem to bound error (Chaterjee'12).

Mean payoff

The graph of $\pi = s, t, u, t, t, u, u, t, u, t, u, t, \cdots$ is:



MP ≈ 0.55

MP for a path

- Estimate transtition probabilities
- Compute empirical mean payoff for the BSCC.
- Use robustness theorem to bound error (Chaterjee'12).

Experiments for reachability

- evaluated on PRISM benchmarks
- MC sizes: 10^6–10^{15} transitions,

Benchmark	max BSCC	Our method	SimTermination[YCZ10]	SimAnalysis[YCZ10]	PRISM numeric
bluetooth	1	5.0s	109.2s	TO	TO
brp	1	89.2s	15.8s	TO	TO
crowds	1	5.6s	340.0s	4.0s	TO
eql	1	31.4s	TO	TO	TO
herman	42	570.0s	TO	505.2s	1.4s
leader	1	153.0s	174.8s	TO	TO
nand	1	6.8s	370.2s	148.2s	TO
tandem	>501K	3.4s	72.4s	62.4s	59.4s
gridworld	250K	5.8s	109.4s	TO	TO

$\alpha = \beta = \epsilon = 0.01, \delta = 0.001, \mathsf{TO} = 900s$

Comparison with

- PRISM - numberic model checking.
- SimTermination - path randomly terminated; requires 2nd eigenvalue.
- SimAnalysis - prune states that cannot statisfy the property; requires information on topology.

Experiments for LTL and mean payoff

Benchmark	max BSCC	LTL		Mean payoff	
		Our method	PRISM numeric	Our method	PRISM numeric
bluetooth	1	8.0s	TO	3.0s	TO
brp	1	90.0s	TO	6.6s	TO
crowds	1	9.0s	TO	2.0s	TO
eql	1	7.0s	TO	2.6s	TO
herman	42	TO	2.0s	TO	3.0s
leader	1	277.0s	117.0s	48.5	576.0
nand	1	4.0s	TO	2.0s	294.0s
tandem	>501K	TO	221.0s	TO	191.0s
gridworld	250K	TO	110.4s	TO	58.1s

LTL: $\alpha = \beta = \epsilon = 0.01, \delta = 0.001$, TO = $900s$, Mean payoff: 95% – confidence interval of size 0.22

Performance of our SMC algorithm for LTL and MP strongly depends on topology.

Conclusions

- On-the-fly BSCC detection is a simple, but powerful method.
- First SMC algorithm for reachability with access only to $p_{\min}$.
- First SMC algorithm for full LTL and mean payoff.
- No silver bullet – performance strongly depends on the topology.

Conclusions

- On-the-fly BSCC detection is a simple, but powerful method.
- First SMC algorithm for reachability with access only to $p_{\min}$.
- First SMC algorithm for full LTL and mean payoff.
- No silver bullet – performance strongly depends on the topology.

Thank you!