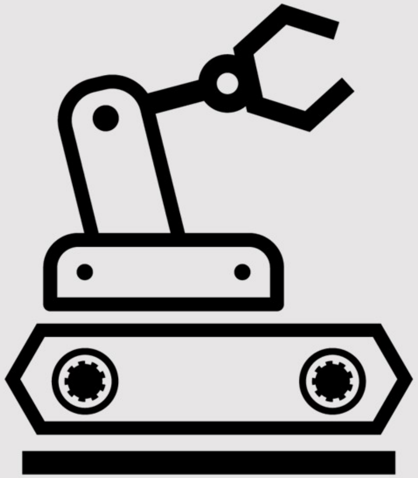
Ranking and Repulsing Supermartingales for Reachability in Probabilistic Programs

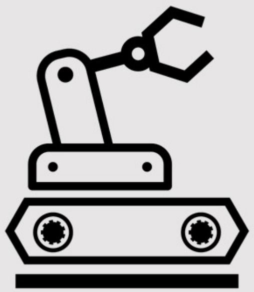
Toru Takisaka, Yuichiro Oyabu, Natsuki Urabe, Ichiro Hasuo

ERATO 蓮尾メタ数理システムデザインプロジェクト
ERATO Metamathematics for Systems Design Project
国立情報学研究所 & 科学技術振興機構 National Institute of Informatics & Japan Science and Technology Agency



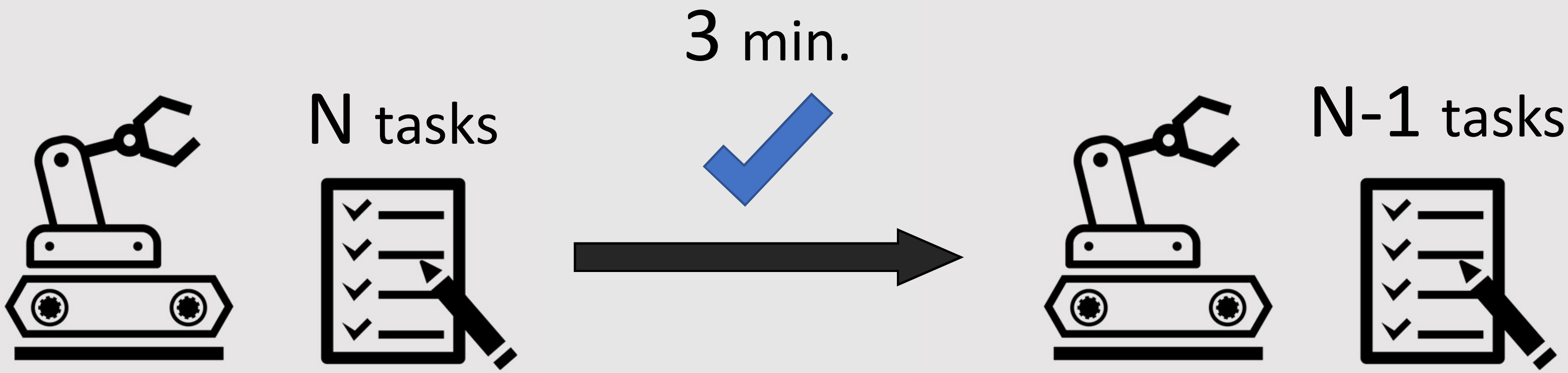
A robot resolves a set of tasks

Mode 1: safe mode



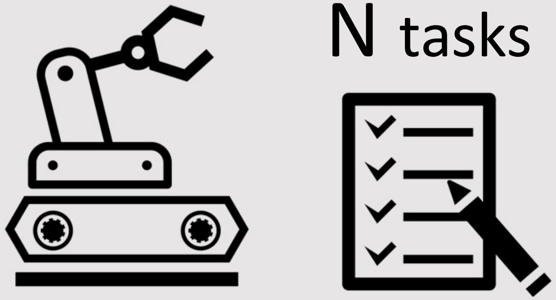
N tasks

Mode 1: safe mode

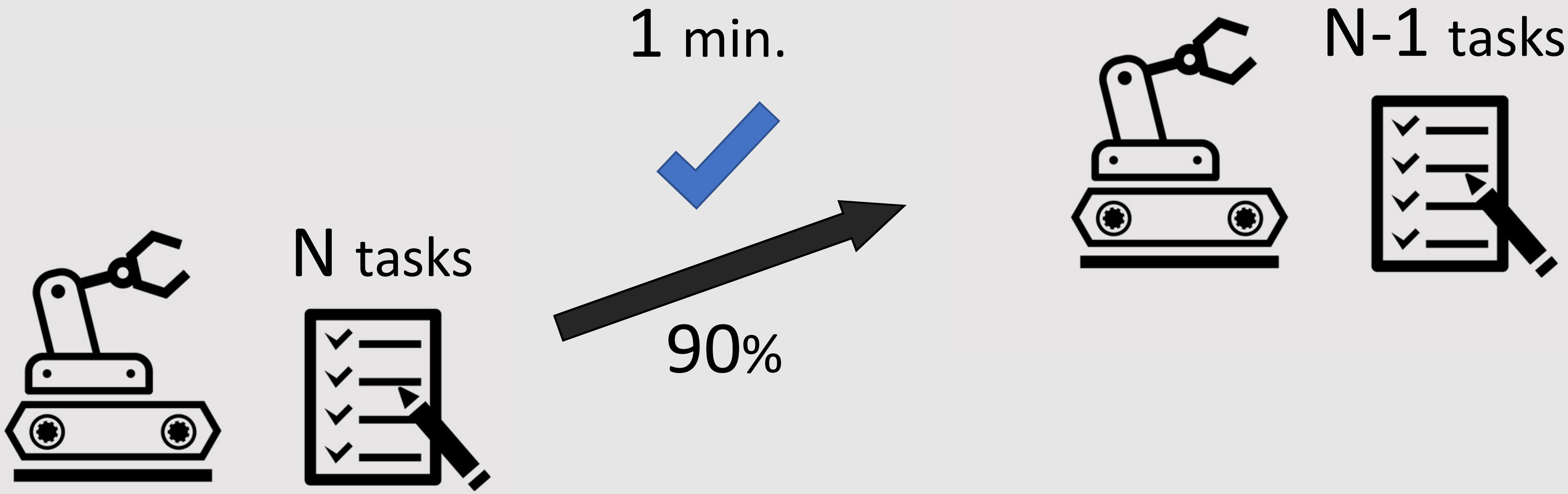


Mode 2: urgent mode

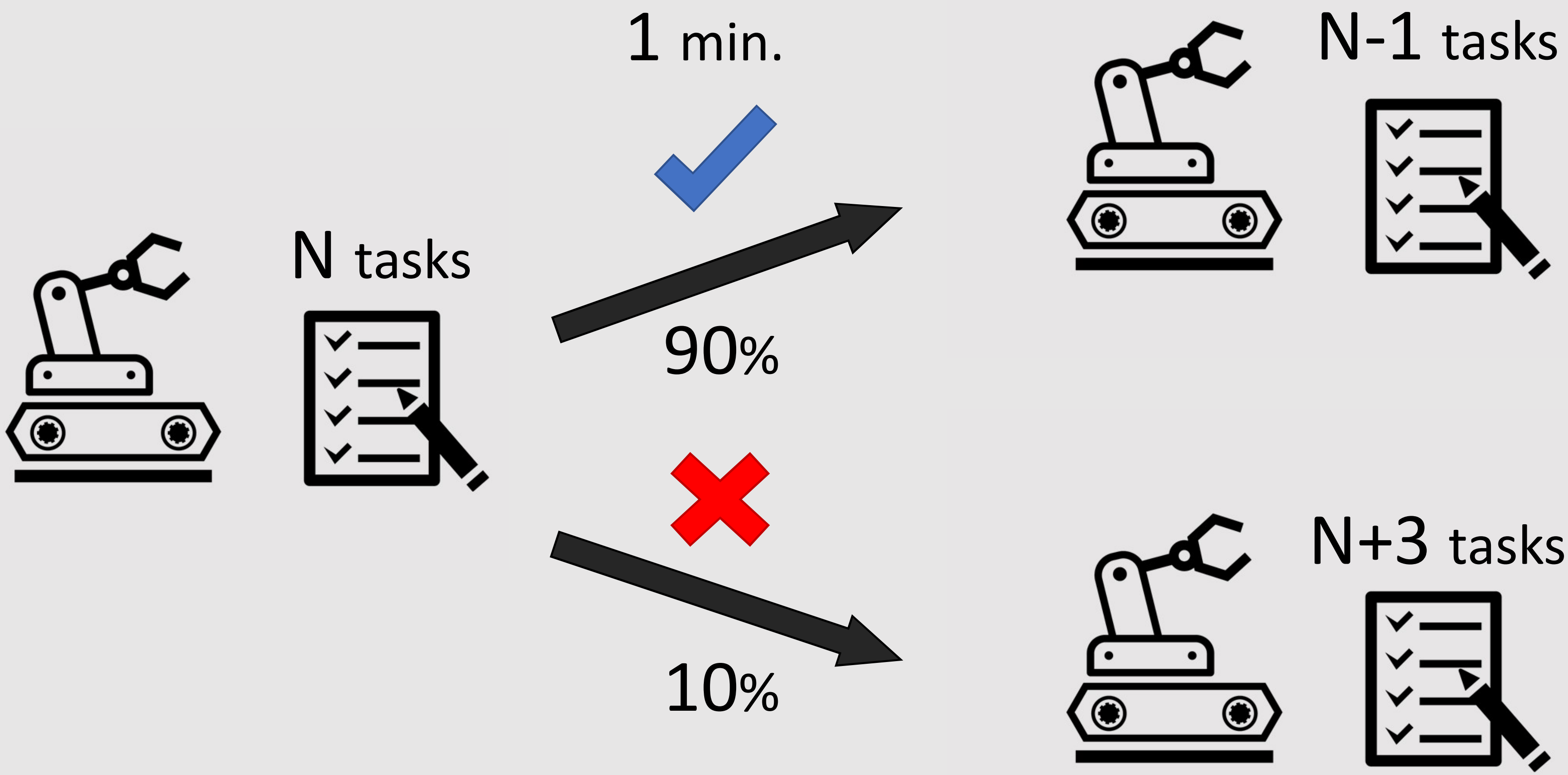
N tasks

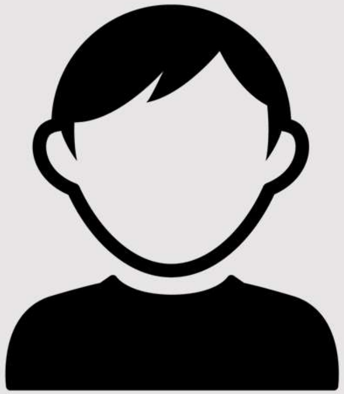


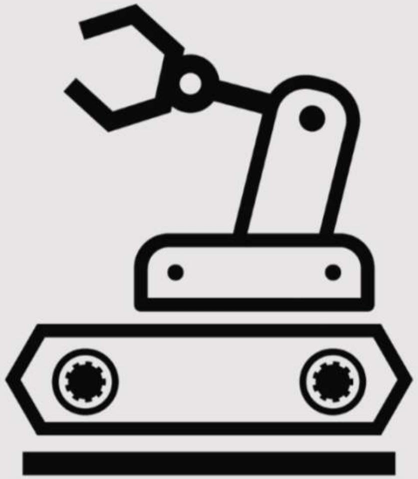
Mode 2: urgent mode

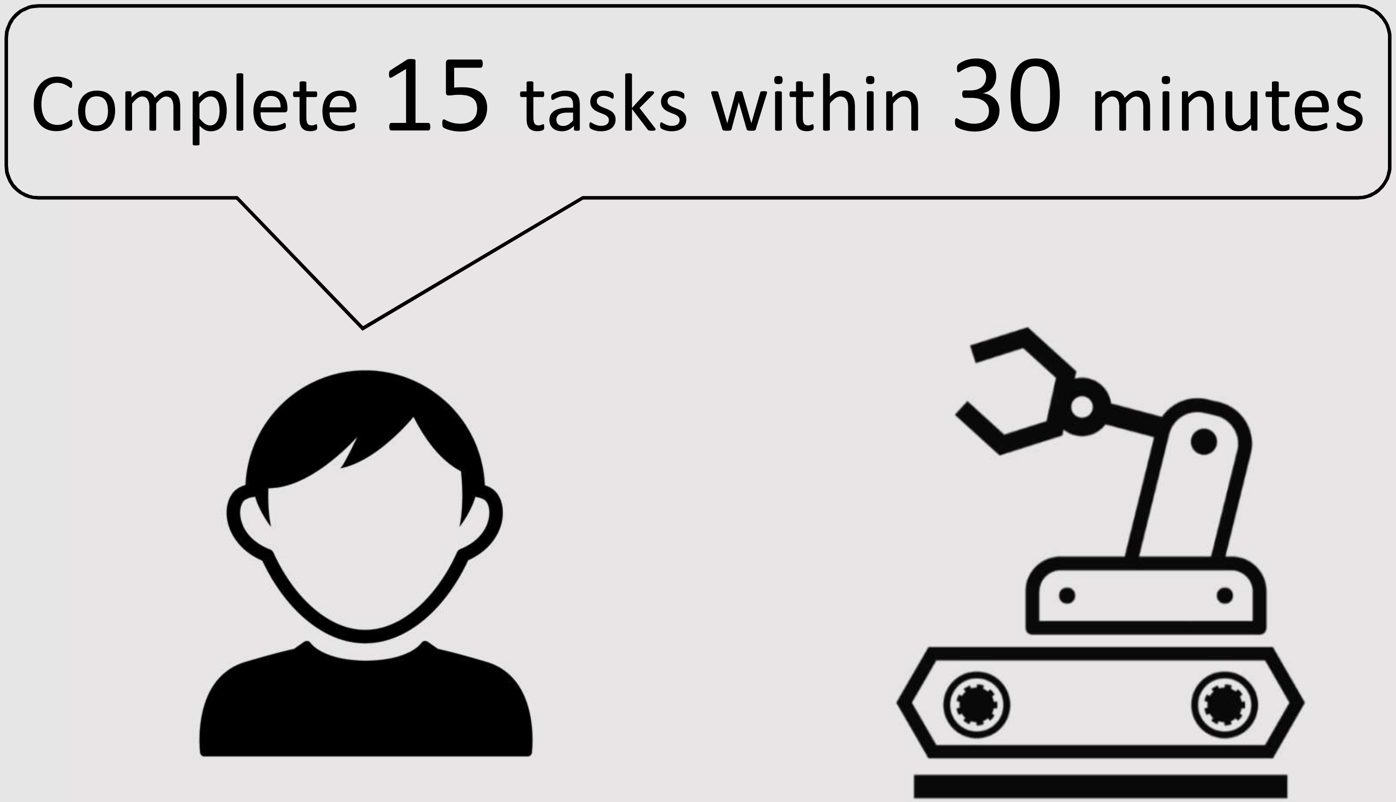


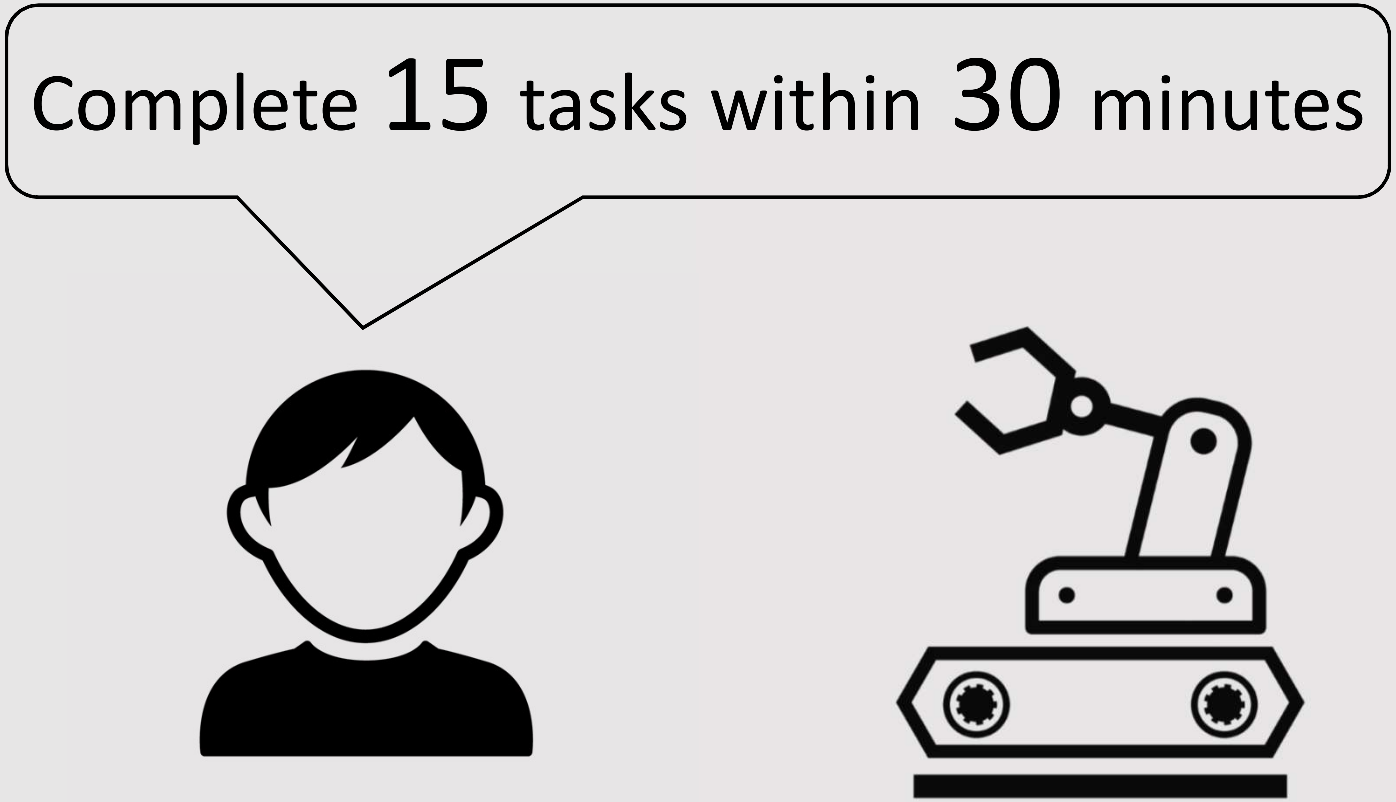
Mode 2: urgent mode








Complete 15 tasks within 30 minutes



What is the probability that
the robot completes the tasks?

Problem formulation

Input: probabilistic program

```
x := 15; t := 0;
p := {0.9:1, 0.1:-3};
while x > 0 do
  if * then
    t := t + 3;
    x := x - 1
  else
    t := t + 1;
    x := x - p
  fi
refute (t > 30)
```

Problem formulation

Input: probabilistic program

```
x := 15; t := 0;
p := {0.9:1 , 0.1:-3};
while x > 0 do
  if * then
    t := t + 3;
    x := x - 1
  else
    t := t + 1;
    x := x - p
  fi
refute (t > 30)
```

Nondet. / Prob. branching

Problem formulation

Input: probabilistic program

```
x := 15; t := 0;
p := {0.9:1 , 0.1:−3};
while x > 0 do
  if * then
    t := t + 3;
    x := x − 1
  else
    t := t + 1;
    x := x − p
  fi
refute (t > 30)
```

Nondet. / Prob. branching

Nondet. / Prob. assignment

Problem formulation

Input: probabilistic program

```
x := 15; t := 0;
p := {0.9:1 , 0.1:−3};
while x > 0 do
  if * then
    t := t + 3;
    x := x − 1
  else
    t := t + 1;
    x := x − p
  fi
refute (t > 30)
```

Nondet. / Prob. branching

Nondet. / Prob. assignment

Problem

What is the probability that the program terminates?

(under angelic/demonic scheduler)

We admit continuous variable

⇒Generally one can't compute this value efficiently

Problem formulation

Input: probabilistic program

```
x := 15; t := 0;
p := {0.9:1 , 0.1:-3};
while x > 0 do
  if * then
    t := t + 3;
    x := x - 1
  else
    t := t + 1;
    x := x - p
  fi
refute (t > 30)
```

Nondet. / Prob. branching

Nondet. / Prob. assignment

Problem

What is the probability that the program terminates?

(under angelic/demonic scheduler)

We admit continuous variable

⇒Generally one can't compute this value efficiently

⇒ Certification by supermartingale

Certification by supermartingale

Probabilistic modification of real-world benchmarks

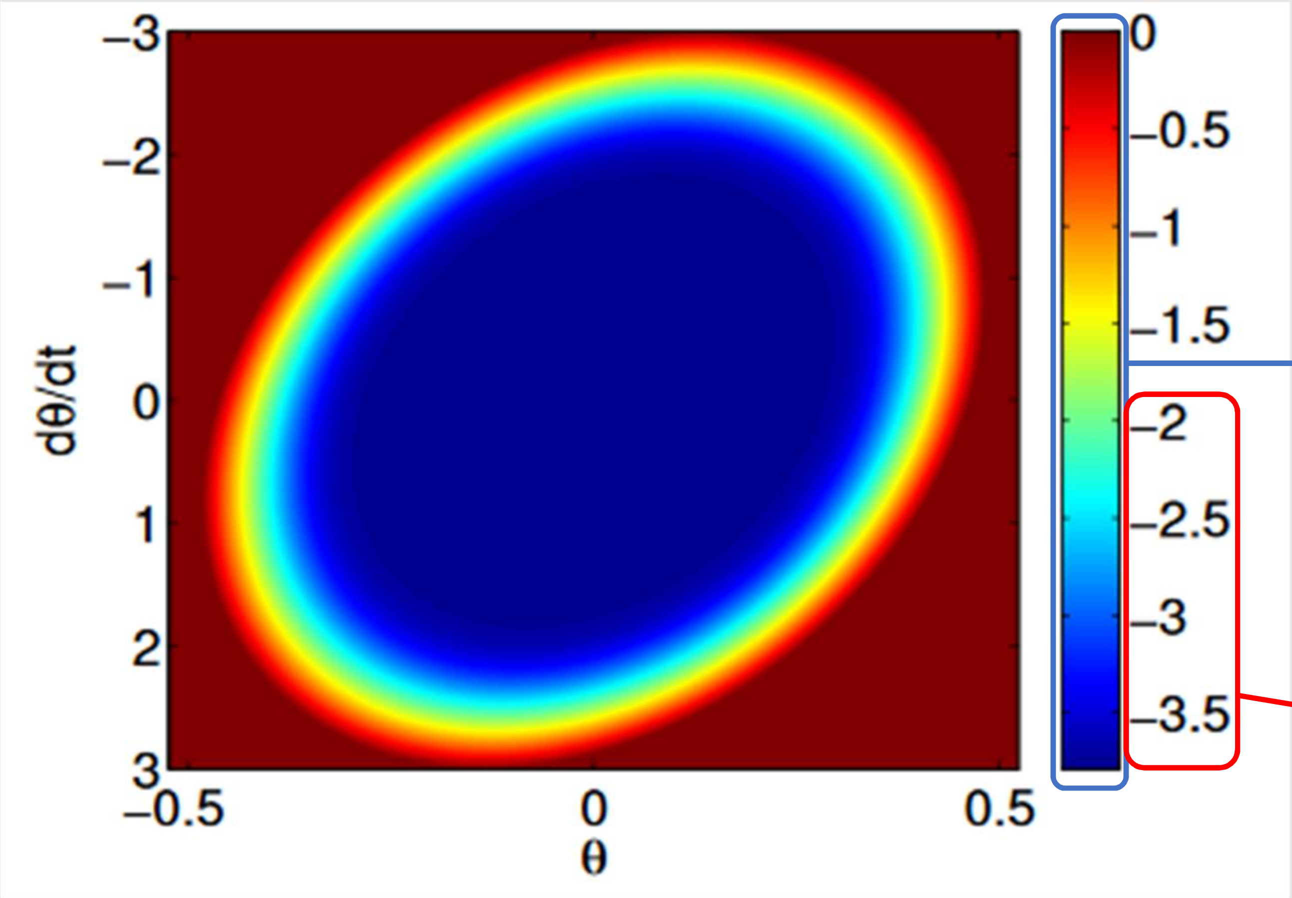
(in Alias+, SAS'10)

Almost-sure termination is certified in 20/28 examples

(Agrawal+, POPL'18)

Benchmark	Time (s)	Solution	Dimension	Prob. loops	Prob. Assignments
alain	0.11	yes	2	yes	yes
catmouse	0.08	yes	2	yes	yes
counterex1a	0.1	no		no	no
counterex1c	0.11	yes	3	yes	yes
easy1	0.09	yes	1	yes	yes
exmini	0.09	yes	2	yes	yes
insertsort	0.1	yes	3	yes	yes
ndecr	0.09	yes	2	yes	yes
perfect	0.11	yes	3	yes	yes
perfect2	0.1	yes	3	yes	no
	0.11	no		yes	yes
real2	0.09	no		no	no
realbubble	0.22	yes	3	yes	yes
realselect	0.11	yes	3	yes	yes
realshellsort	0.09	no		yes	no
serpent	0.1	yes	1	yes	yes
sipmabubble	0.1	yes	3	yes	yes
speedDis2	0.09	no		no	no
speedNestedMultiple	0.1	yes	3	yes	yes
speedpldi2	0.09	yes	2	yes	yes
speedpldi4	0.09	yes	3	yes	yes
speedSimpleMultipleDep	0.09	no		no	no
speedSingleSingle2	0.12	yes	2	yes	no
	0.1	no		yes	yes
unperfect	0.1	yes	2	yes	no
	0.16	no		yes	yes
wcet1	0.11	yes	2	yes	yes
while2	0.1	yes	3	yes	yes

Certification by supermartingale



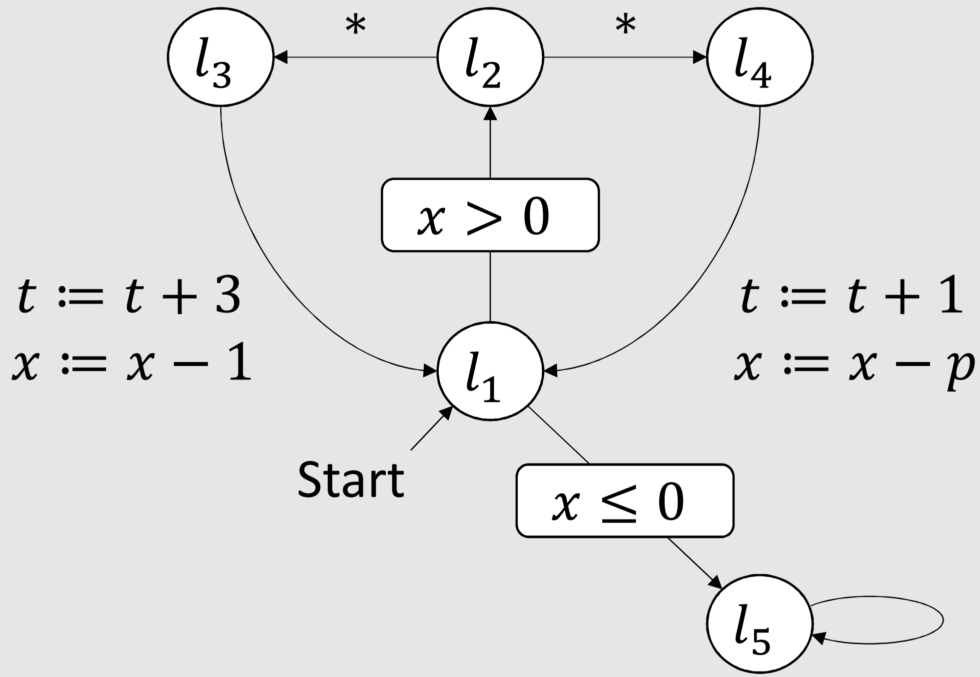
System: a pendulum under Gaussian noise

The log-base-10 of the failure probability
(failure = $|\theta| > \pi/6$ within 1h)

>99% safety is guaranteed
(Pr(enter a bad state) <1%)

(Steinhardt-Tedrake, IJRR'12)

Control flow graph



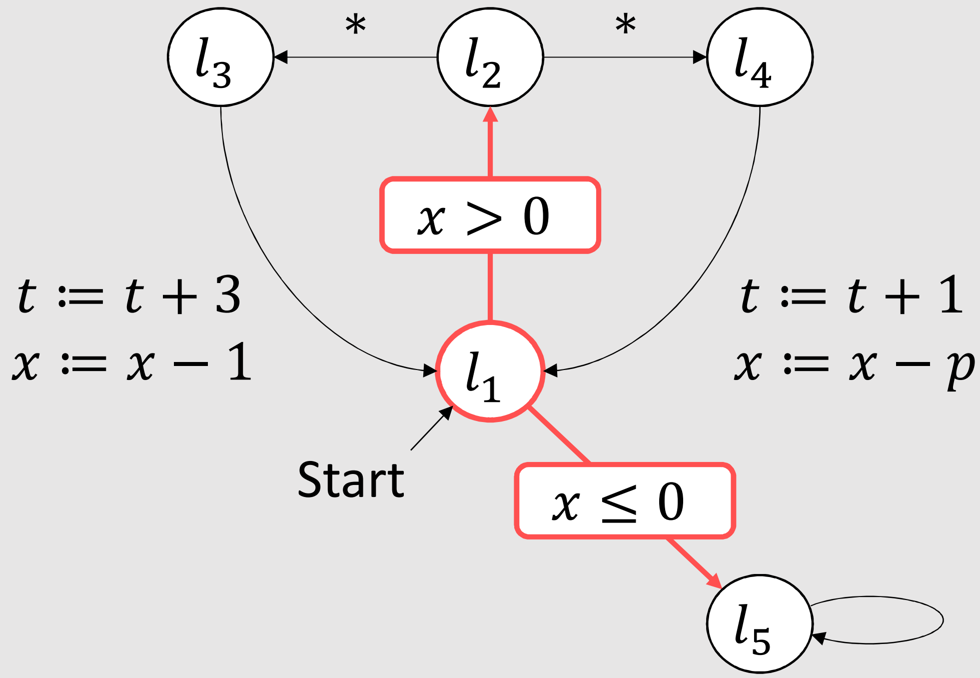

```
x := 15; t := 0;
p := {0.9:1 , 0.1:−3};
while x > 0 do
  if ∗ then
    t := t + 3;
    x := x − 1
  else
    t := t + 1;
    x := x − p
  fi
refute (t > 30)
```

- A state is a pair (program location, memory state)
- As powerful as MDP

finite

$\mathbb{R}^V$

Control flow graph



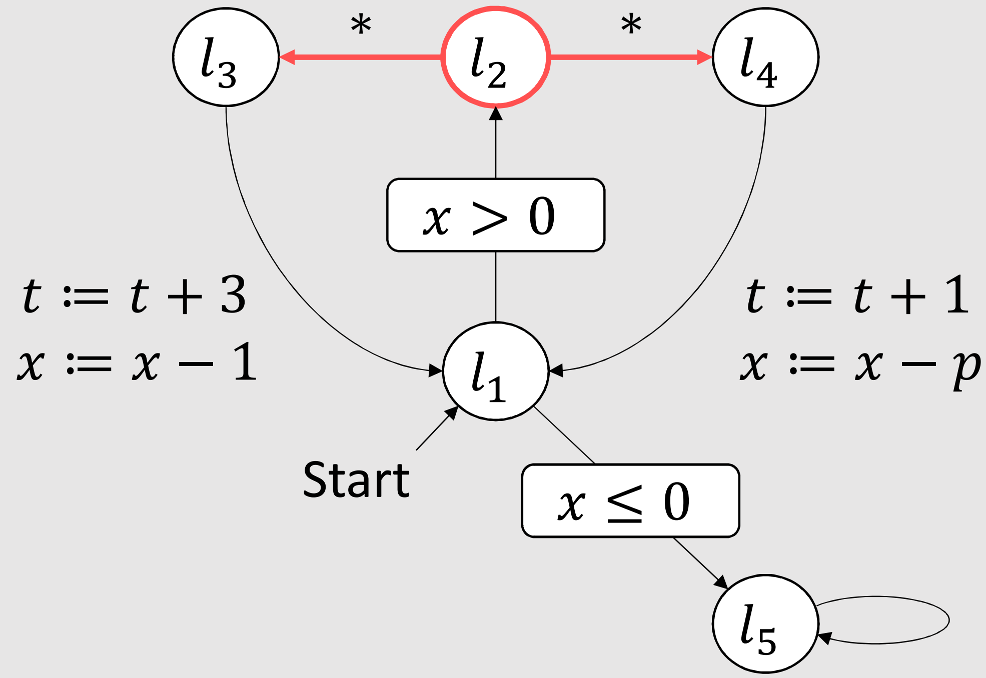

```
x := 15; t := 0;
p := {0.9:1, 0.1:−3};
while x > 0 do
  if * then
    t := t + 3;
    x := x − 1
  else
    t := t + 1;
    x := x − p
  fi
refute (t > 30)
```

- A state is a pair (program location, memory state)
- As powerful as MDP

finite

$\mathbb{R}^V$

Control flow graph



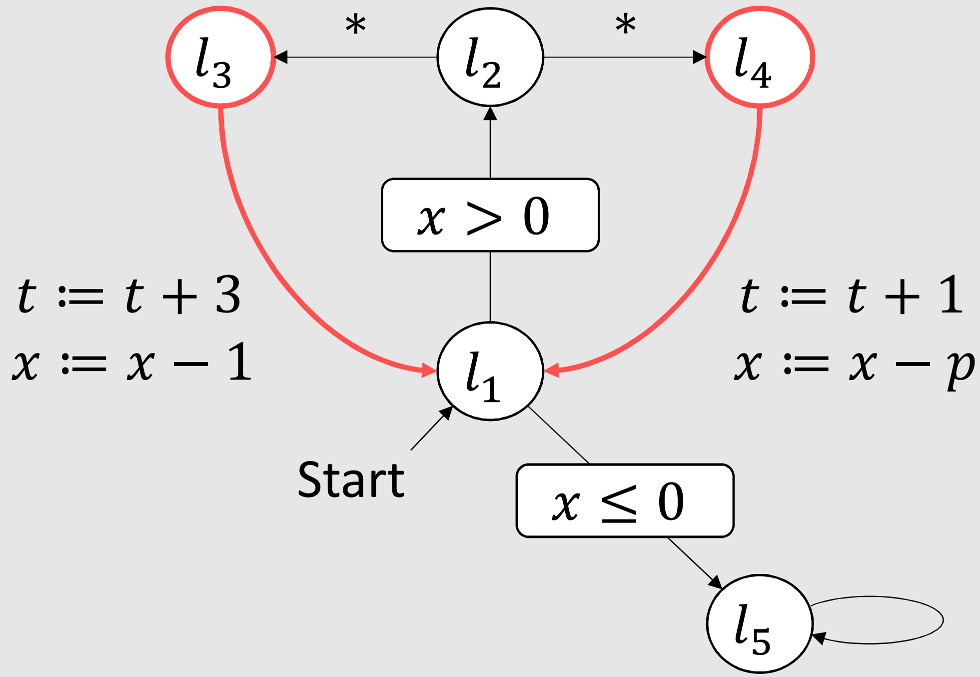

```
x := 15; t := 0;
p := {0.9:1, 0.1:-3};
while x > 0 do
  if * then
    t := t + 3;
    x := x - 1
  else
    t := t + 1;
    x := x - p
  fi
refute (t > 30)
```

- A state is a pair (program location, memory state)
- As powerful as MDP

finite

$\mathbb{R}^V$

Control flow graph



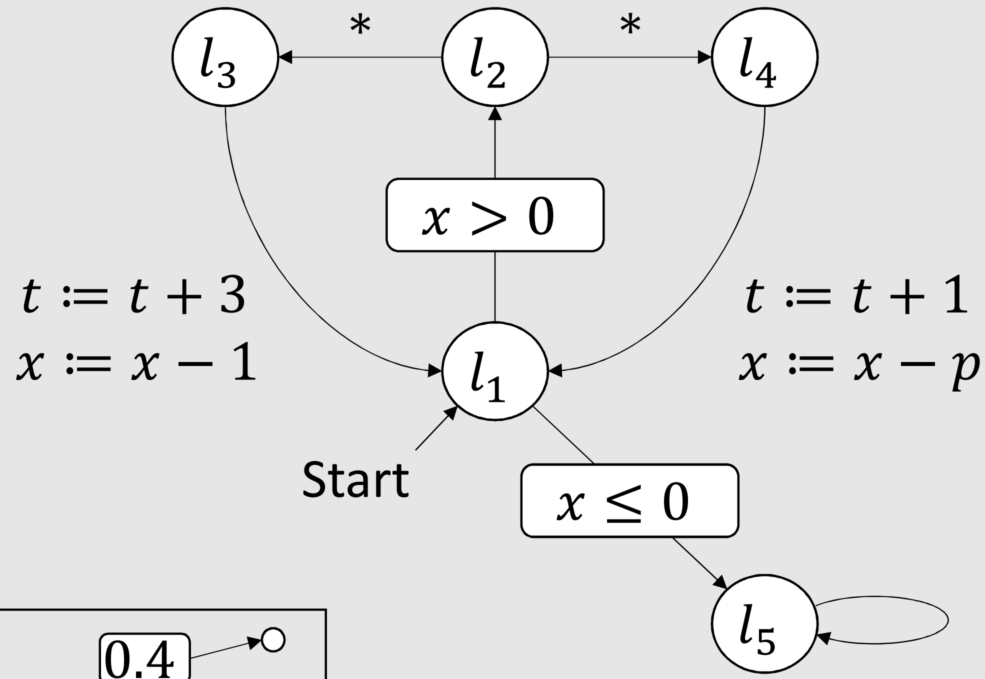

```
x := 15; t := 0;
p := {0.9:1, 0.1:−3};
while x > 0 do
   if * then
      t := t + 3;
      x := x − 1
   else
      t := t + 1;
      x := x − p
   fi
refute (t > 30)
```

- A state is a pair (program location, memory state)
- As powerful as MDP

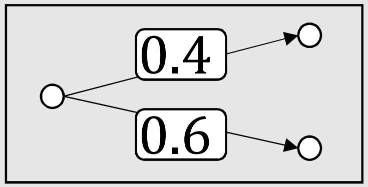
finite

$\mathbb{R}^V$

Control flow graph




```
x := 15; t := 0;
p := {0.9:1, 0.1:−3};
while x > 0 do
  if * then
    t := t + 3;
    x := x − 1
  else
    t := t + 1;
    x := x − p
  fi
refute (t > 30)
```

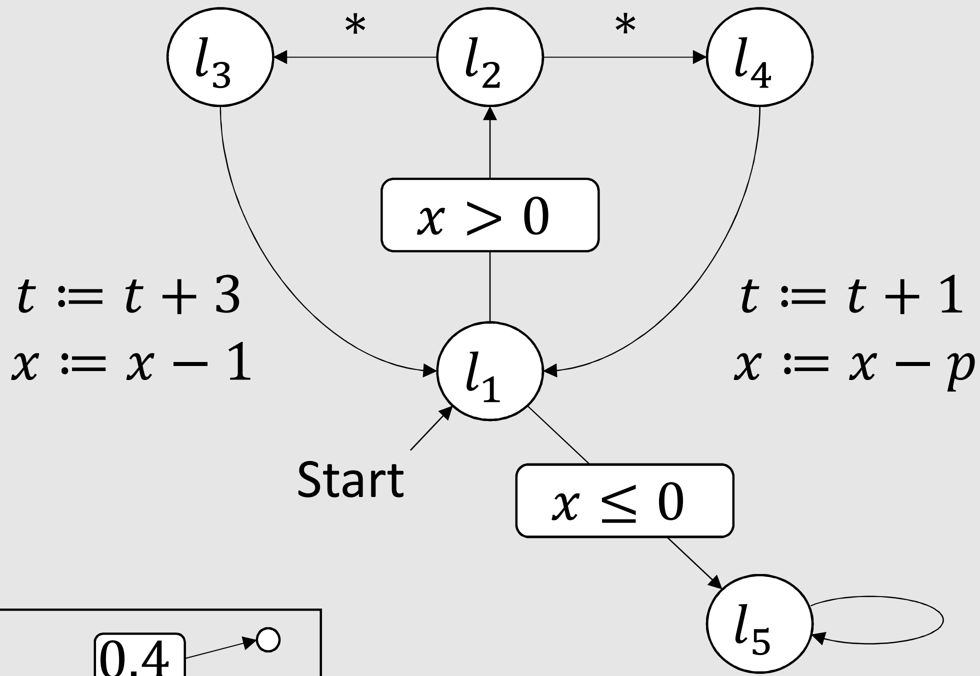


- A state is a pair (program location, memory state)
- As powerful as MDP

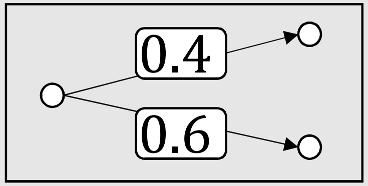
finite

$\mathbb{R}^V$

Control flow graph




```
x := 15; t := 0;
p := {0.9:1, 0.1:-3};
while x > 0 do
  if * then
    t := t + 3;
    x := x - 1
  else
    t := t + 1;
    x := x - p
  fi
refute (t > 30)
```



- A state is a pair (program location, memory state)
- As powerful as MDP

finite

$\mathbb{R}^V$

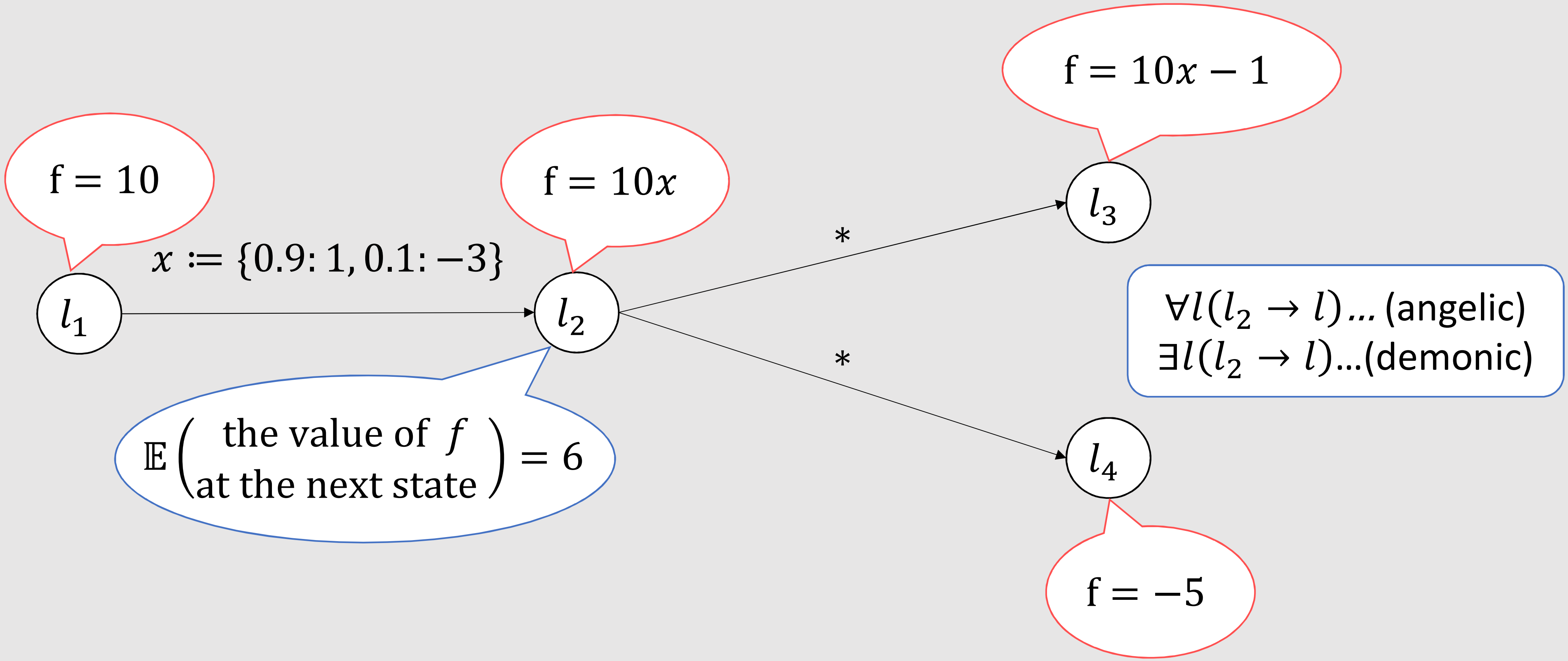
Problem

$$C \coloneqq \{l_5\} \times (\mathbb{R} \times (30, \infty))$$

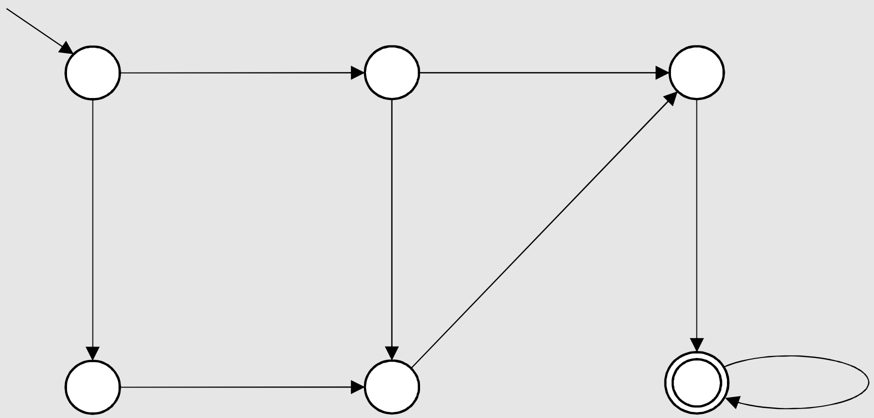
$$\subseteq (\text{Locations}) \times (\text{Variables})$$

⇒Pr(the system eventually visits the region C)?

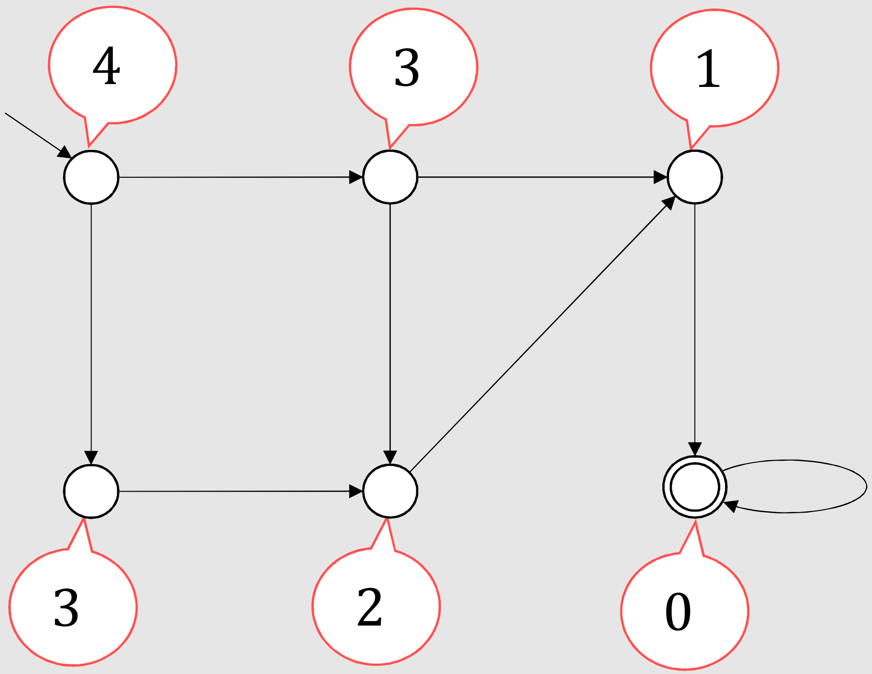
Supermartingale = a function over states that is "non-increasing" through transitions



Ranking function

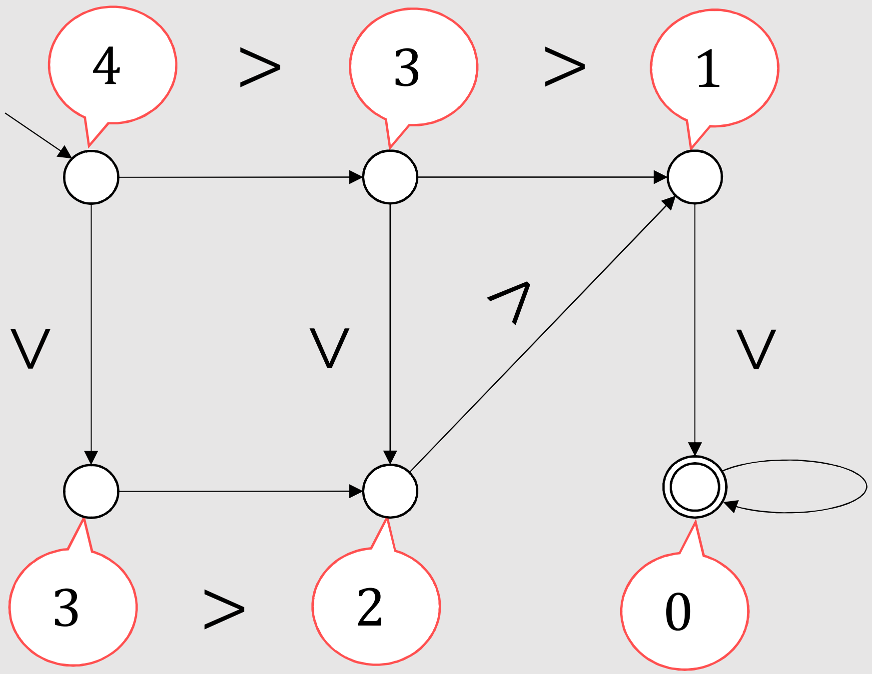


Ranking function



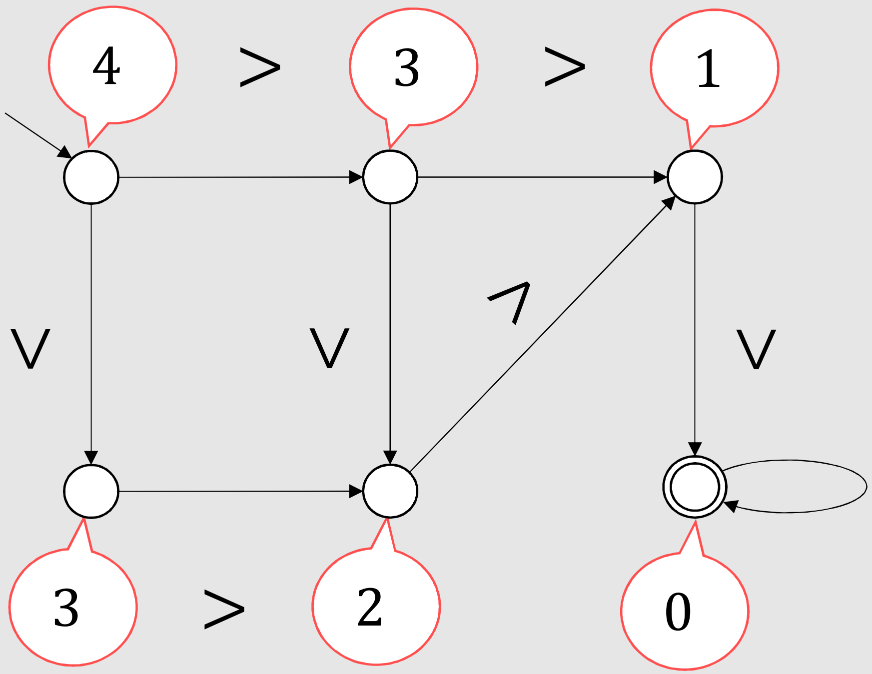
Ranking function

Int-valued



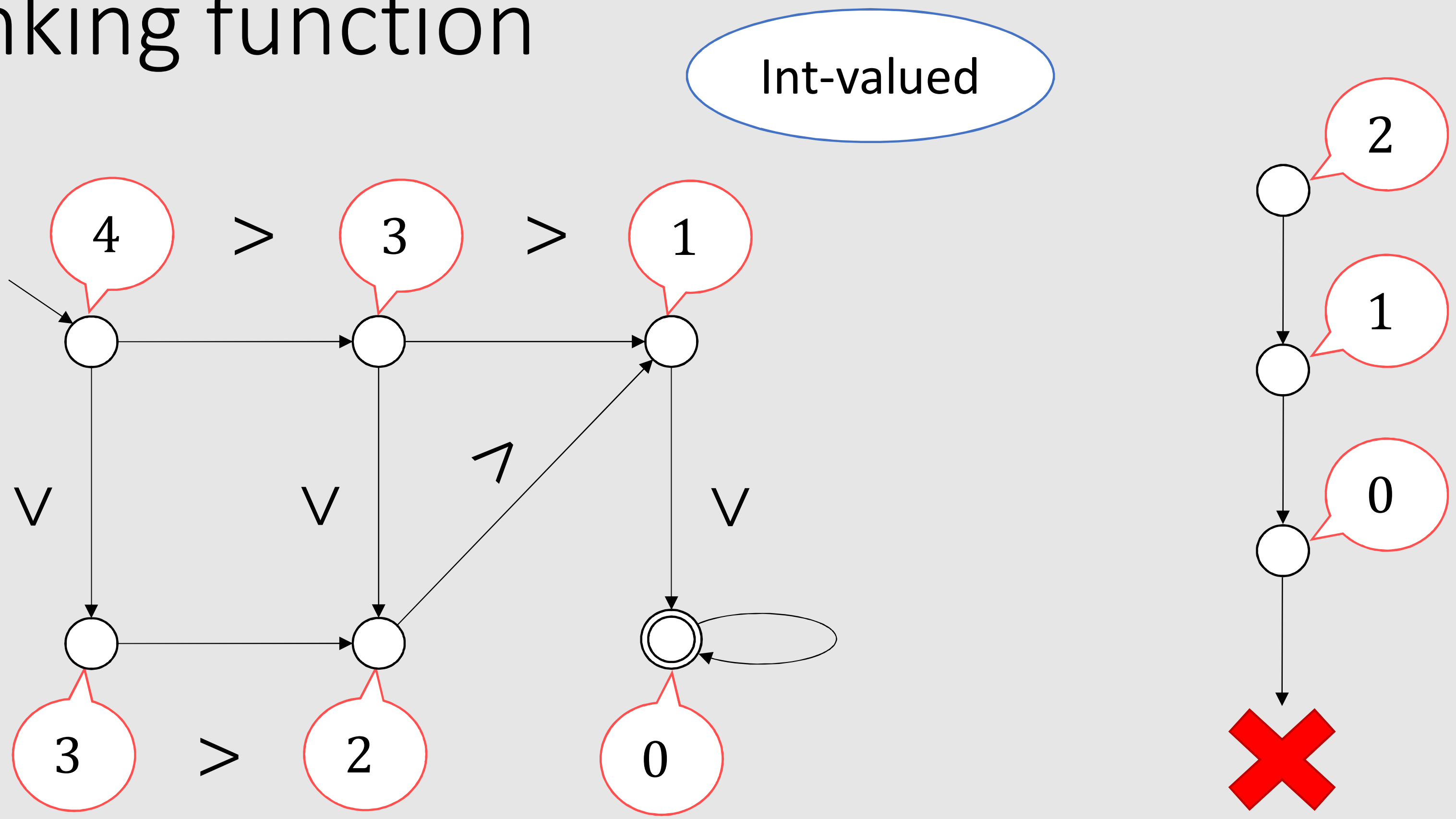
Ranking function

Int-valued



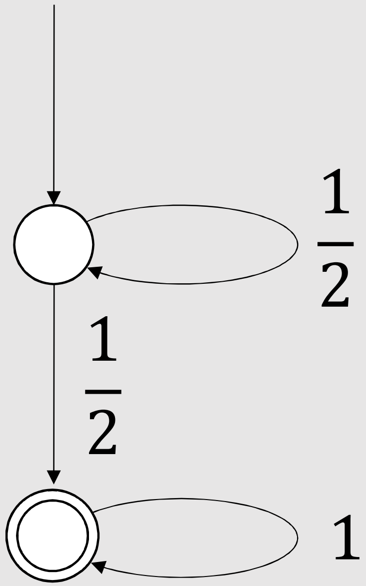
The system eventually visits ◎ (under any nondeterministic choice)

Ranking function

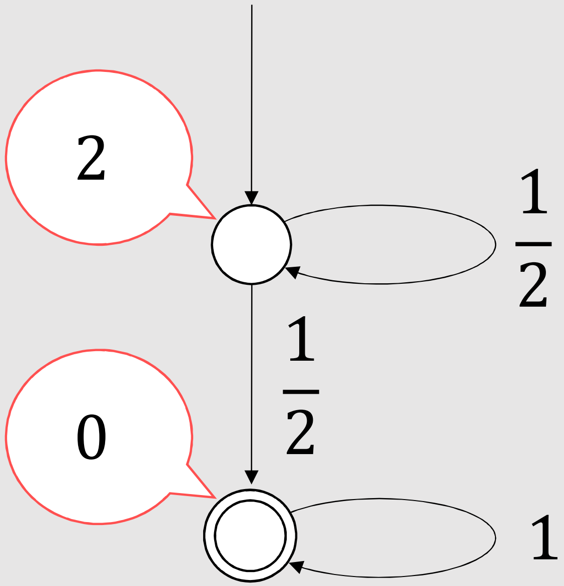


The system eventually visits ◎ (under any nondeterministic choice)

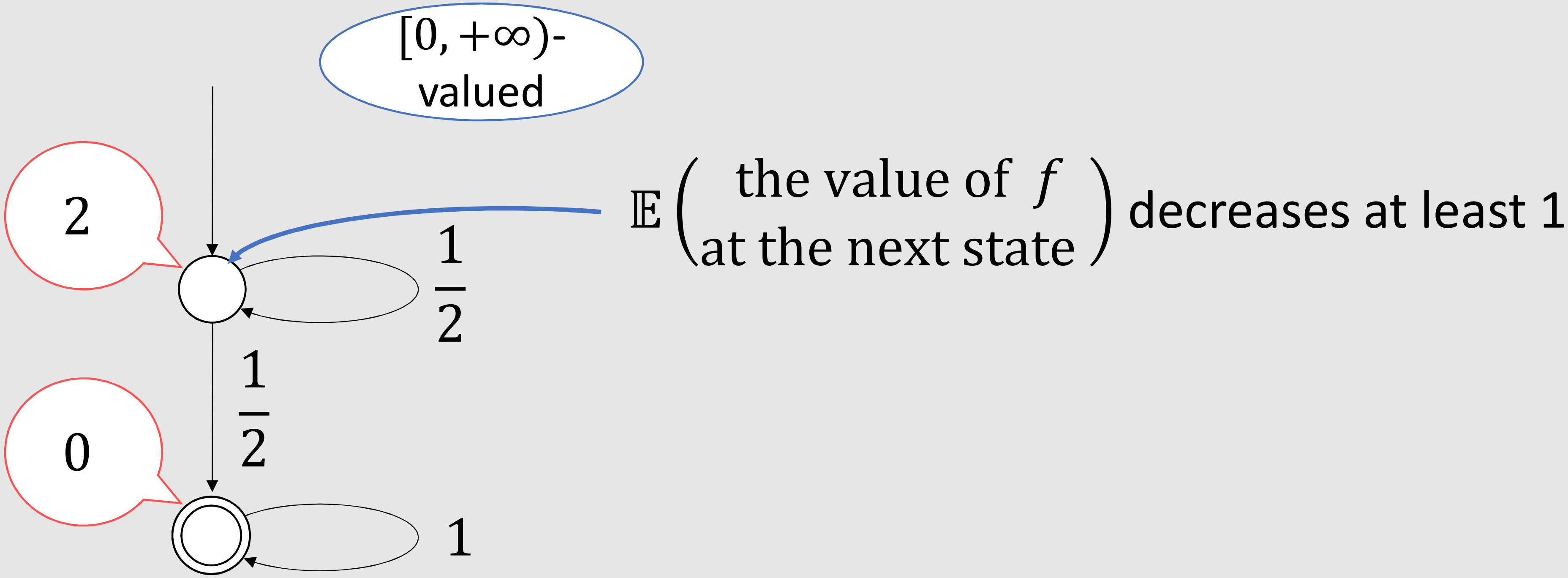
Ranking supermartingale



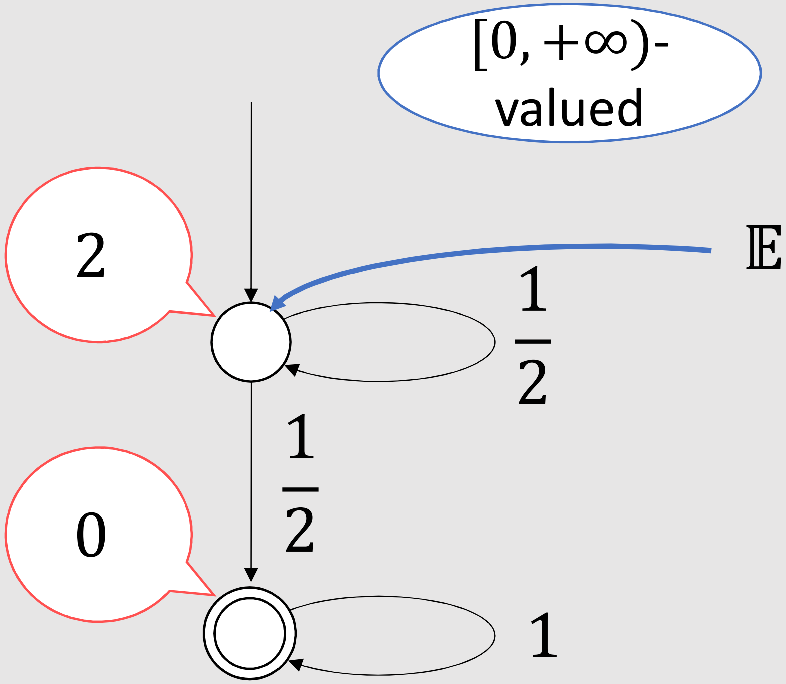
Ranking supermartingale



Ranking supermartingale



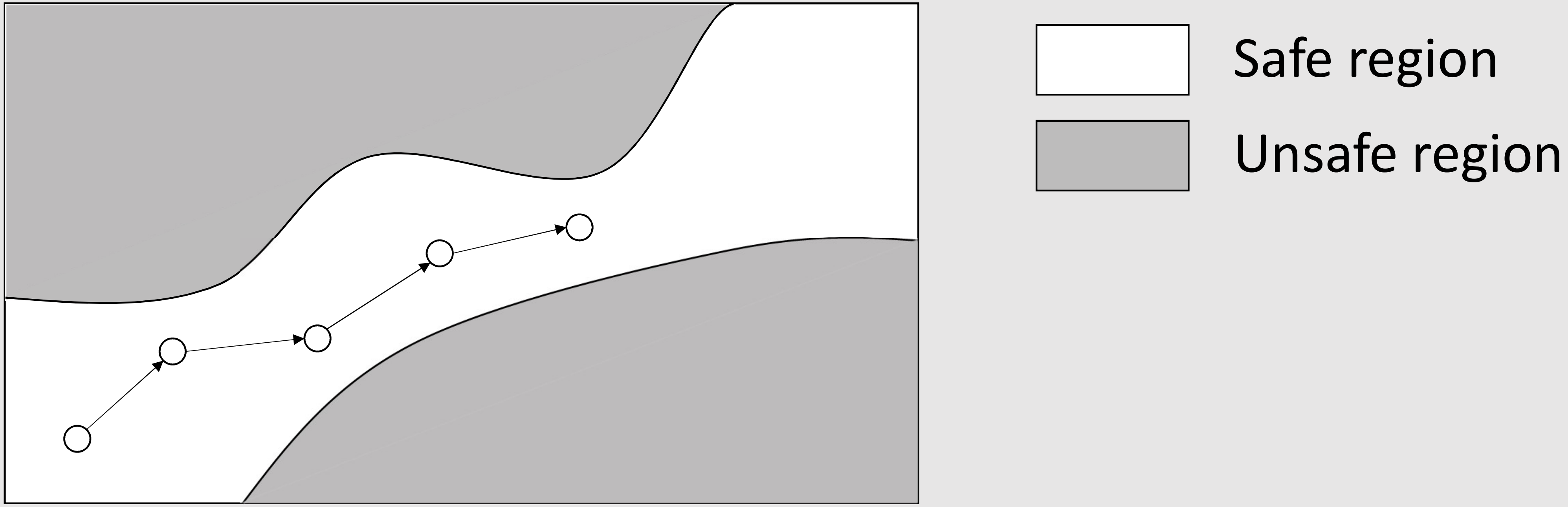
Ranking supermartingale



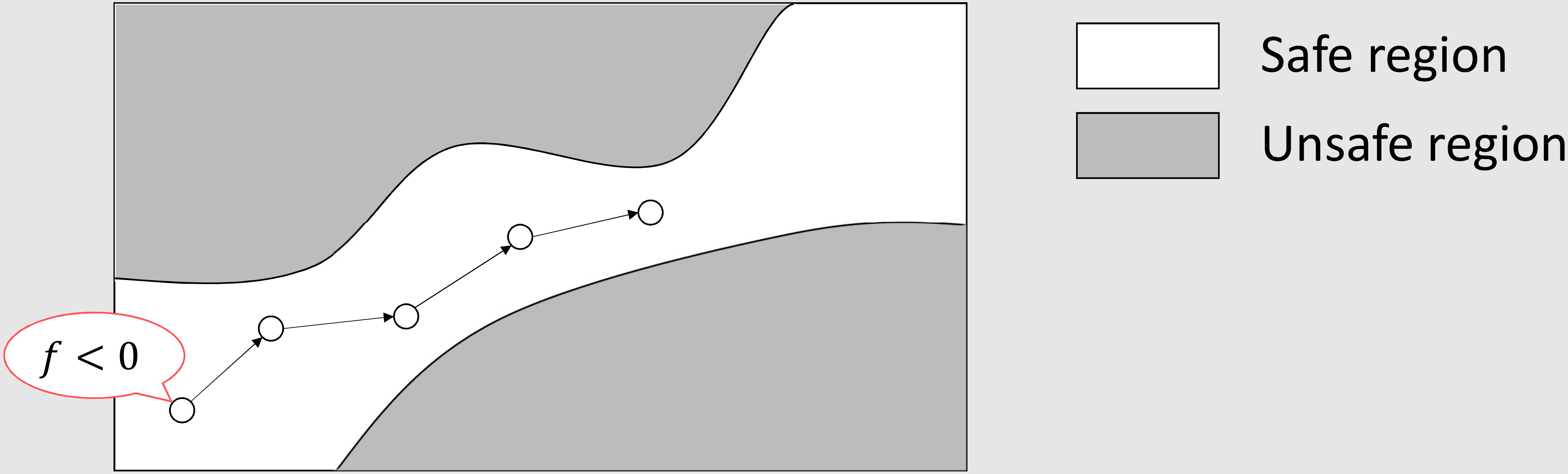
$\mathbb{E}\left(\begin{array}{c}\text{the value of } f\\ \text{at the next state}\end{array}\right)$ decreases at least 1

The system eventually visits ◎ almost surely

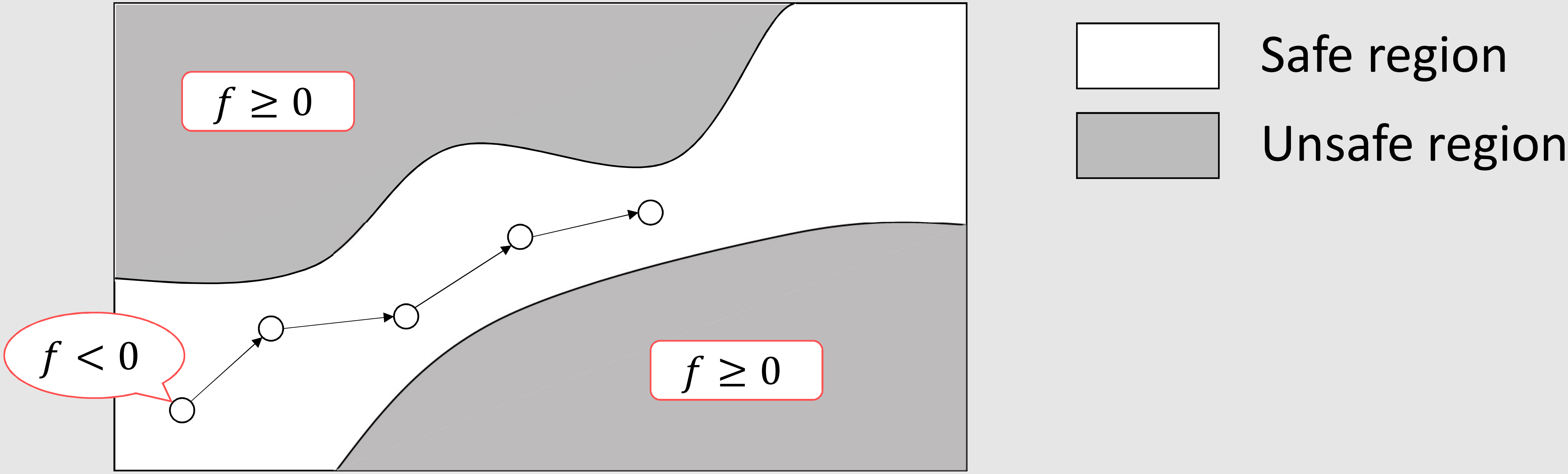
Barrier certificate



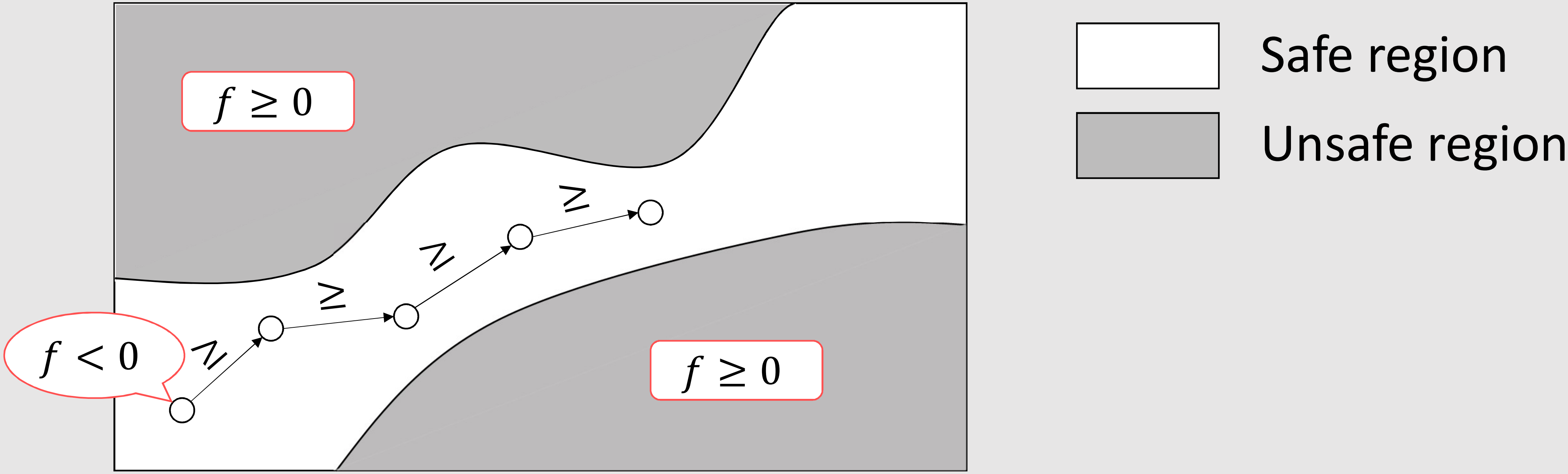
Barrier certificate



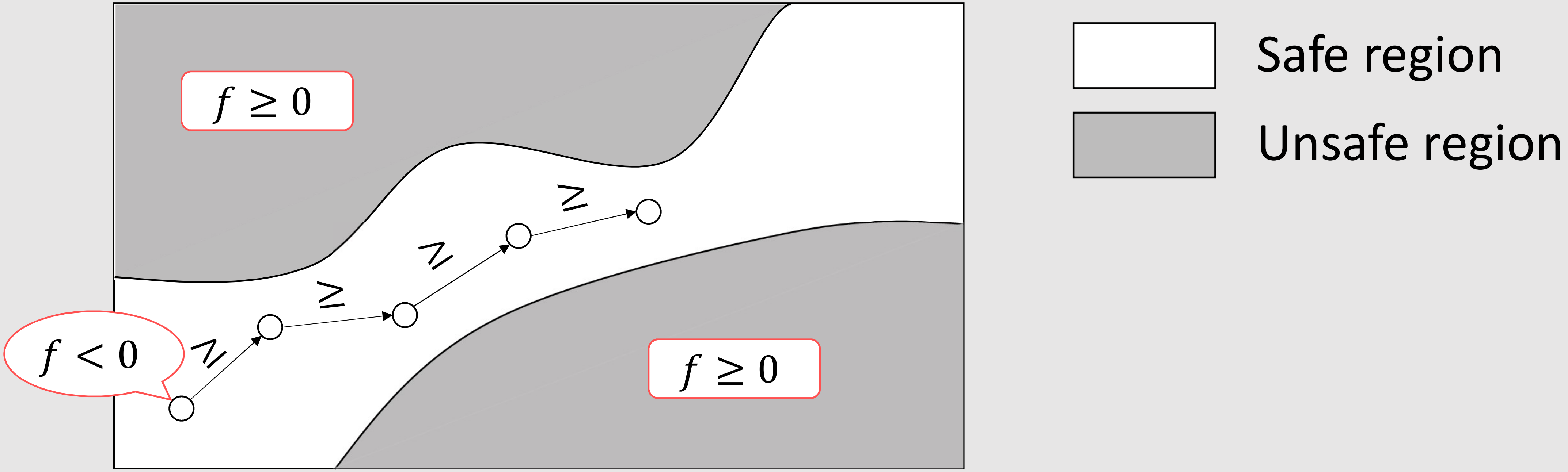
Barrier certificate



Barrier certificate

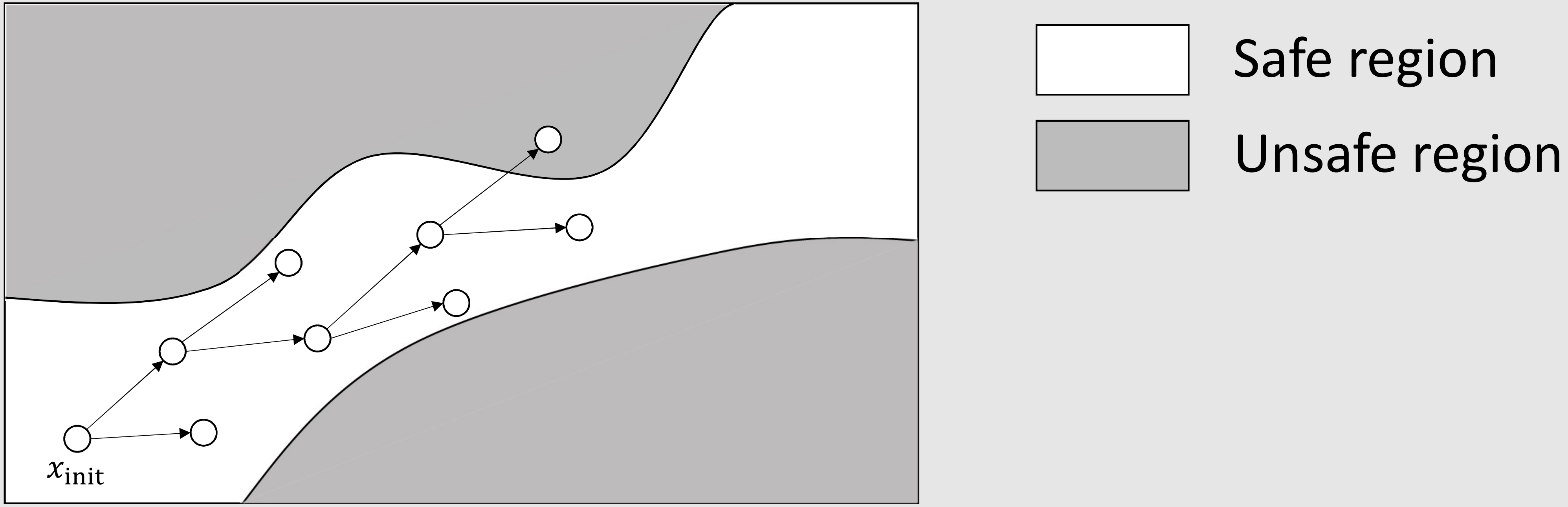


Barrier certificate

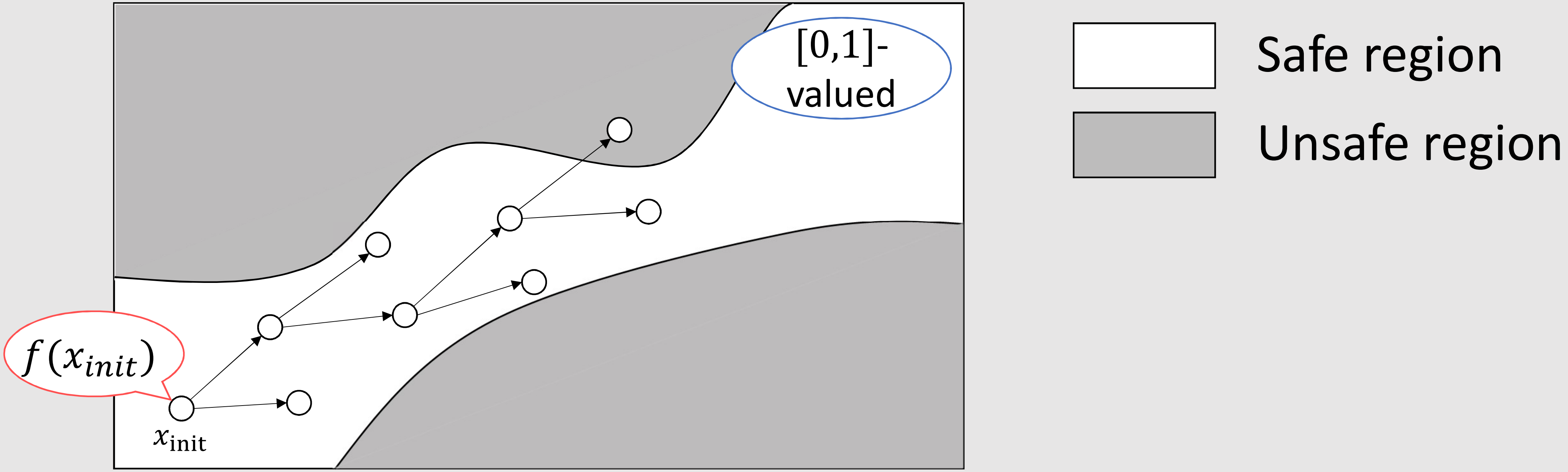


The system does not enter the unsafe region

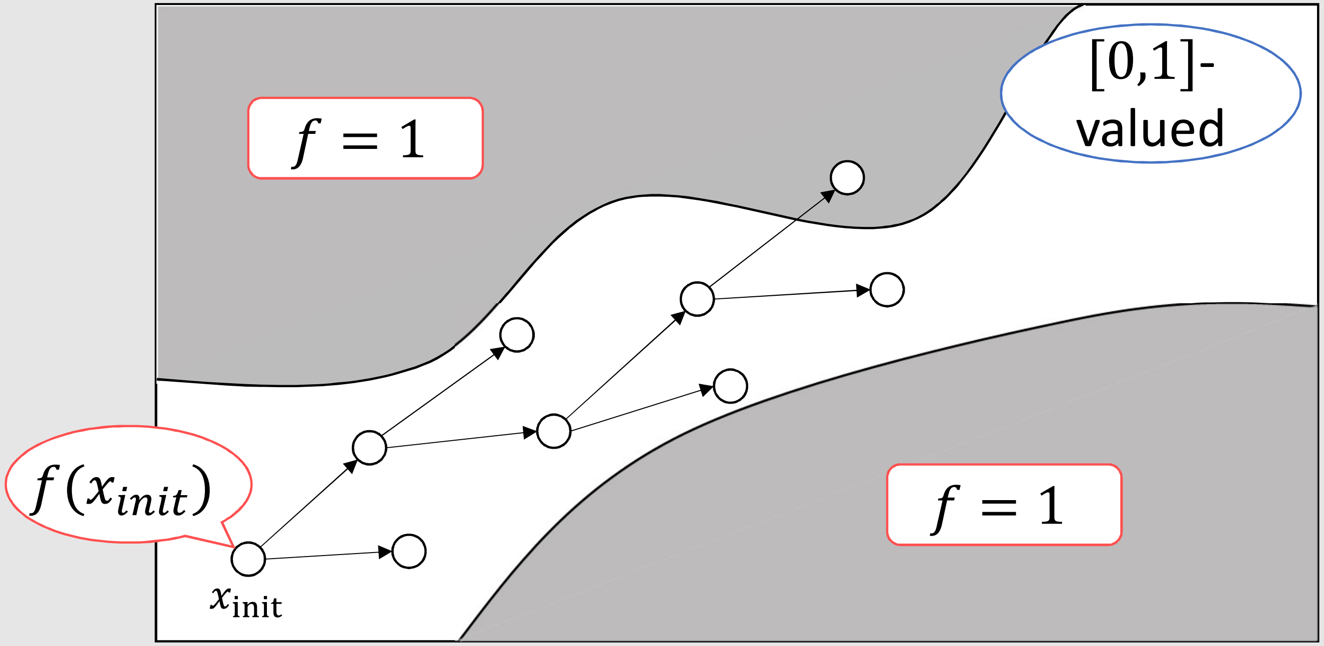
Probabilistic barrier certificate
(a.k.a. nonnegative repulsing supermartingale)

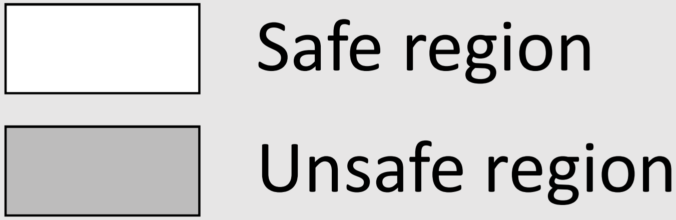


Probabilistic barrier certificate
(a.k.a. nonnegative repulsing supermartingale)

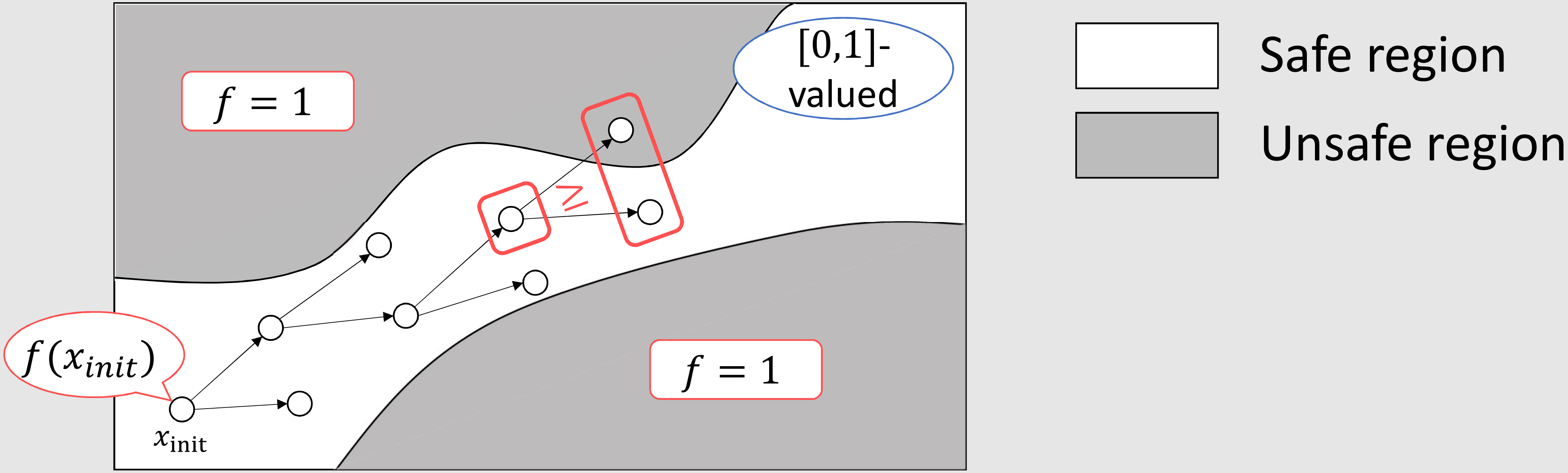


Probabilistic barrier certificate
(a.k.a. nonnegative repulsing supermartingale)

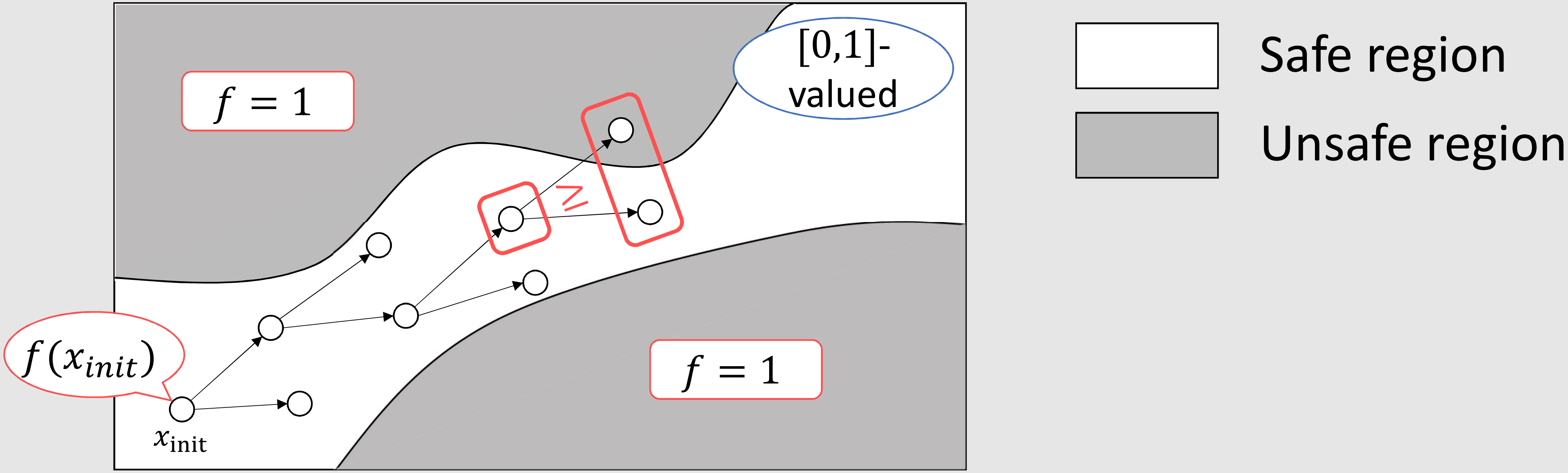




Probabilistic barrier certificate
(a.k.a. nonnegative repulsing supermartingale)



Probabilistic barrier certificate
(a.k.a. nonnegative repulsing supermartingale)



Pr(the system enters the unsafe region) $\leq f(x_{\mathrm{init}})$

Our contributions

Comprehensive account of martingale-based approximation methods via **fixed point argument**

Soundness/completeness for **uncountable-states MDPs**, under angelic/demonic nondeterminism

Implementation and **experiments**

Our contributions

Comprehensive account of martingale-based approximation methods via **fixed point argument**

Soundness/completeness for uncountable-states MDPs, under angelic/demonic nondeterminism

Implementation and experiments

Two objective functions

- Given: a control flow graph, and a subset C of its states
- $\mathbb{E}^{\text{steps}}: L \times \mathbb{R}^V \to [0, \infty]$ and $\mathbb{P}^{\text{reach}}: L \times \mathbb{R}^V \to [0,1]$ are

$$\mathbb{E}^{\text{steps}}: c \mapsto \mathbb{E}\left(\begin{matrix}\text{the number of steps from } c\\ \text{to the region } C\end{matrix}\right)$$

$$\mathbb{P}^{\text{reach}}: c \mapsto \mathbb{P}\left(\begin{matrix}\text{the system eventually visits}\\ \text{the region } C \text{ from } c\end{matrix}\right)$$

Two objective functions

- Given: a control flow graph, and a subset C of its states
- $\mathbb{E}^{\text{steps}}: L \times \mathbb{R}^V \to [0, \infty]$ and $\mathbb{P}^{\text{reach}}: L \times \mathbb{R}^V \to [0,1]$ are

$$\mathbb{E}^{\text{steps}}: c \mapsto \mathbb{E}\left(\begin{array}{c}\text{the number of steps from } c\\ \text{to the region } C\end{array}\right)$$

$$\mathbb{P}^{\text{reach}}: c \mapsto \mathbb{P}\left(\begin{array}{c}\text{the system eventually visits}\\ \text{the region } C \text{ from } c\end{array}\right)$$

…under angelic/demonic scheduler

Soundness/completeness

Ranking supermartingale

Soundness: $\exists(\mathrm{RankSM}) \quad \Rightarrow \quad \mathbb{E}^{\mathrm{steps}}(c_{init}) < \infty$

$(\Rightarrow \mathbb{P}^{\mathrm{reach}}(c_{init}) = 1)$

Completeness: $\mathbb{E}^{\mathrm{steps}}(c_{init}) < \infty \quad \Rightarrow \quad \exists(\mathrm{RankSM})$

Nonnegative repulsing supermartingale

Soundness: $\exists(\mathrm{RepSM}) \quad \Rightarrow \quad \mathbb{P}^{\mathrm{reach}}(c_{init}) \leq \delta$

Completeness: $\mathbb{P}^{\mathrm{reach}}(c_{init}) \leq \delta \quad \Rightarrow \quad \exists(\mathrm{RepSM})$

Soundness/completeness

Ranking supermartingale

Known — Soundness: $\exists(\mathrm{RankSM}) \Rightarrow \mathbb{E}^{\mathrm{steps}}(c_{init}) < \infty$

$(\Rightarrow \mathbb{P}^{\mathrm{reach}}(c_{init}) = 1)$

Partly known — Completeness: $\mathbb{E}^{\mathrm{steps}}(c_{init}) < \infty \Rightarrow \exists(\mathrm{RankSM})$

Nonnegative repulsing supermartingale

Partly known — Soundness: $\exists(\mathrm{RepSM}) \Rightarrow \mathbb{P}^{\mathrm{reach}}(c_{init}) \leq \delta$

Not known — Completeness: $\mathbb{P}^{\mathrm{reach}}(c_{init}) \leq \delta \Rightarrow \exists(\mathrm{RepSM})$

Soundness/completeness

For certain endofunctions Φ and Ψ,

$\mathbb{E}^{\mathrm{steps}} = \mu\Phi$ and $\mathbb{P}^{\mathrm{reach}} = \mu\Psi$

Soundness/completeness

Our theorem

$$\mathbb{E}^{\mathrm{steps}} = \mu\Phi$$

The lattice $(\mathcal{F}, \sqsubseteq)$

$\mathcal{F}$ ... the set of all (measurable) functions

$$f : L \times \mathbb{R}^V \to [0, \infty]$$

$\sqsubseteq$ ... $\quad f \sqsubseteq g \Leftrightarrow \forall s. f(s) \leq g(s)$

Soundness/completeness

Our theorem

$$\mathbb{E}^{\mathrm{steps}} = \mu\Phi$$

The lattice $(\mathcal{F}, \sqsubseteq)$

$\mathcal{F}$ … the set of all (measurable) functions

$$f : L \times \mathbb{R}^V \to [0, \infty]$$

$\sqsubseteq$ … $\quad f \sqsubseteq g \;\Leftrightarrow\; \forall s.\, f(s) \leq g(s)$

Soundness

$$\frac{f \text{ is a RankSM}}{\mathbb{E}^{\mathrm{steps}} \sqsubseteq f}$$

Soundness/completeness

Our theorem

$$\mathbb{E}^{\mathrm{steps}} = \mu\Phi$$

The lattice $(\mathcal{F}, \sqsubseteq)$

$\mathcal{F}$ ... the set of all (measurable) functions

$$f: L \times \mathbb{R}^V \to [0, \infty]$$

$\sqsubseteq$ ... $\quad f \sqsubseteq g \Leftrightarrow \forall s. f(s) \le g(s)$

Soundness

$$\frac{f \text{ is a RankSM}}{\mathbb{E}^{\mathrm{steps}} \sqsubseteq f} \quad \Leftrightarrow \quad \frac{\Phi f \sqsubseteq f}{\mu\Phi \sqsubseteq f}$$

Soundness/completeness

Our theorem

$$\mathbb{E}^{\text{steps}} = \mu\Phi$$

The lattice $(\mathcal{F}, \sqsubseteq)$

$\mathcal{F}$ … the set of all (measurable) functions

$$f : L \times \mathbb{R}^V \to [0, \infty]$$

$\sqsubseteq$ … $\quad f \sqsubseteq g \Leftrightarrow \forall s.\, f(s) \leq g(s)$

Soundness

$$\frac{f \text{ is a RankSM}}{\mathbb{E}^{\text{steps}} \sqsubseteq f} \quad \begin{matrix} \Leftrightarrow \\ \Leftrightarrow \end{matrix} \quad \boxed{\frac{\Phi f \sqsubseteq f}{\mu\Phi \sqsubseteq f}}$$

Knaster-Tarski theorem

Soundness/completeness

Our theorem

$$\mathbb{E}^{\text{steps}} = \mu\Phi$$

The lattice $(\mathcal{F}, \sqsubseteq)$

$\mathcal{F}$ ... the set of all (measurable) functions

$$f : L \times \mathbb{R}^V \to [0, \infty]$$

$\sqsubseteq$ ... $\quad f \sqsubseteq g \Leftrightarrow \forall s. f(s) \leq g(s)$

Soundness

$$\frac{f \text{ is a RankSM}}{\mathbb{E}^{\text{steps}} \sqsubseteq f} \quad \begin{matrix} \Leftrightarrow \\ \Leftrightarrow \end{matrix} \quad \boxed{\frac{\Phi f \sqsubseteq f}{\mu\Phi \sqsubseteq f}}$$

Knaster-Tarski theorem

Completeness

$$\Phi\mathbb{E}^{\text{steps}} \sqsubseteq \mathbb{E}^{\text{steps}}$$

Soundness/completeness

Our theorem

$$\mathbb{P}^{\mathrm{reach}} = \mu\Psi$$

The lattice $(\mathcal{F}, \sqsubseteq)$

$\mathcal{F}$ ... the set of all (measurable) functions

$$f: L \times \mathbb{R}^V \to [0,1]$$

$\sqsubseteq$ ... $\quad f \sqsubseteq g \Leftrightarrow \forall s. f(s) \leq g(s)$

Soundness

$$\frac{f \text{ is a RepSM}}{\mathbb{P}^{\mathrm{reach}} \sqsubseteq f} \quad \begin{matrix}\Leftrightarrow \\ \Leftrightarrow\end{matrix} \quad \frac{\Psi f \sqsubseteq f}{\mu\Psi \sqsubseteq f}$$

Knaster-Tarski theorem

Completeness

$$\Psi\mathbb{P}^{\mathrm{reach}} \sqsubseteq \mathbb{P}^{\mathrm{reach}}$$

Our contributions

Comprehensive account of martingale-based approximation methods via fixed point argument

Soundness/completeness for uncountable-states MDPs, under angelic/demonic nondeterminism

Implementation and experiments

Soundness/completeness for martingale methods

Approximation method	It certifies	Soundness	Completeness
Additive ranking Supermartingale (Chakarov-Sankaranarayanan, CAV'13 etc.)	$\mathbb{E}^{\mathrm{steps}} < \infty$ $(\mathbb{P}^{\mathrm{reach}} = 1)$	Yes (MDP, continuous variable)	Yes (MDP, discrete variable)
Nonnegative repulsing supermartingale (Steinhardt+, IJRR'12 etc.)	$\mathbb{P}^{\mathrm{reach}} \leq \delta$	Yes (Markov Chain)	-
γ-scaled submartingale (Urabe+, LICS'17)	$\mathbb{P}^{\mathrm{reach}} \geq \delta$	Yes (Markov Chain)	-
ε-decreasing repulsing supermartingale (Chatterjee+, POPL'17)	$\mathbb{P}^{\mathrm{reach}} \leq \delta$	Yes (MDP, continuous variable, linearity assumpt.)	-

Soundness/completeness for martingale methods

Approximation method	It certifies	Soundness	Completeness
Additive ranking Supermartingale (Chakarov-Sankaranarayanan, CAV'13 etc.)	$\mathbb{E}^{\text{steps}} < \infty$ ($\mathbb{P}^{\text{reach}} = 1$)	Yes (MDP, continuous variable)	Yes (MDP, continuous variable)
Nonnegative repulsing supermartingale (Steinhardt+, IJRR'12 etc.)	$\mathbb{P}^{\text{reach}} \leq \delta$	Yes (MDP, continuous variable)	
γ-scaled submartingale (Urabe+, LICS'17)	$\mathbb{P}^{\text{reach}} \geq \delta$	Yes (MDP, continuous variable)	-
ε-decreasing repulsing supermartingale (Chatterjee+, POPL'17)	$\mathbb{P}^{\text{reach}} \leq \delta$	Yes (MDP, continuous variable, linearity assumpt.)	No

Our contributions

Comprehensive account of martingale-based approximation methods via fixed point argument

Soundness/completeness for uncountable-states MDPs, under angelic/demonic nondeterminism

Implementation and experiments

Implementation and experiments

		Prog. I (linear)		Prog. II (deg.-2 poly.)		Prog. II (deg.-3 poly.)	
	param.	time (s)	bound	time (s)	bound	time (s)	bound
(a-1)	$p_1 = 0.2$ $p_2 = 0.4$	0.021	≤ 0.825	530.298	≤ 0.6552	572.393	≤ 0.6555
	$p_1 = 0.8$ $p_2 = 0.1$	0.024	≤ 1	526.519	≤ 1.0	561.327	≤ 1.0

Table 1. Bounds by U-NNRepSupM

①

	true reachability probability	U-NNRepSupM	1-RepSupM
(c-1)	$\frac{(0.4/0.6)^5-(0.4/0.6)^{10}}{1-(0.4/0.6)^{10}} \approx 0.116$	0.505	< 1
(c-2)	0.5	0.5	—
(c-3)	$\int_0^1 (\frac{0.25}{0.75})^{\lceil \log_2(1/x) \rceil} dx \approx 0.2$	0.5	—
(c-4)	$(\frac{0.25}{0.75})^1 \approx 0.333$	—	< 1

②

Table 3. Probabilistic bounds given by U-NNRepSupM and ε-RepSupM

		Prog. III (linear)	
	param.	time (s)	bound
(a-1)	$p_1 = 0.2$ $p_2 = 0.4$	0.026	≥ 0
	$p_1 = 0.8$ $p_2 = 0.1$	0.022	≥ 0.751
(a-2)	$M_1 = -1$ $M_2 = 2$	0.033	≥ 0
	$M_1 = -2$ $M_2 = 1$	0.033	≥ 0.767
(a-3)	$M_1 = -1$ $M_2 = 2$	0.028	≥ 0
	$M_1 = -2$ $M_2 = 1$	0.040	≥ 0.801
(b)	$c = 0.1$ $p = 0.5$	0.056	≥ 0
	$c = 0.1$ $p = 0.1$	0.054	≥ 0.148

①

Table 2. Bounds by L-γ-SclSubM with $\gamma = 0.999$

- Implemented template-based synthesis algorithms
- Nontrivial bounds are found (①)
- Observed comparative advantage of nonnegative RepSM over ε-decreasing RepSM (②)

Summary

- Martingale can evaluate reachability of probabilistic programs in various ways
- We gave a comprehensive account of martingale-based approximation methods via fixed point argument
- We proved soundness/completeness of several methods for uncountable-states MDPs, which extends known results
- We demonstrated implementation and experiments