Parametric Markov Models

Sebastian Junges, Joost-Pieter Katoen

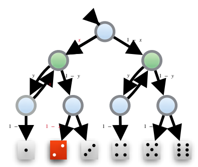
Take-home messages

- parametric Markov chains (pMCs): Markov chains with unknown probabilities
- What questions to ask about them (do we control the values of the parameters?)
- Supports different applications in control and in parameter tuning
- The idea can also be applied for families of Markov chains

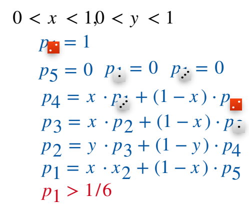
Sebastian Junges & Joost-Pieter Katoen

Overview

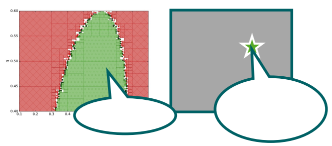
Concepts



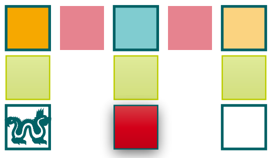
Encoding



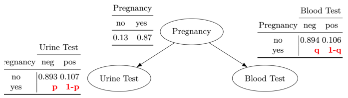
Methods



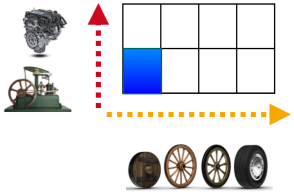
POMDPs



Parametric BNs



Product lines

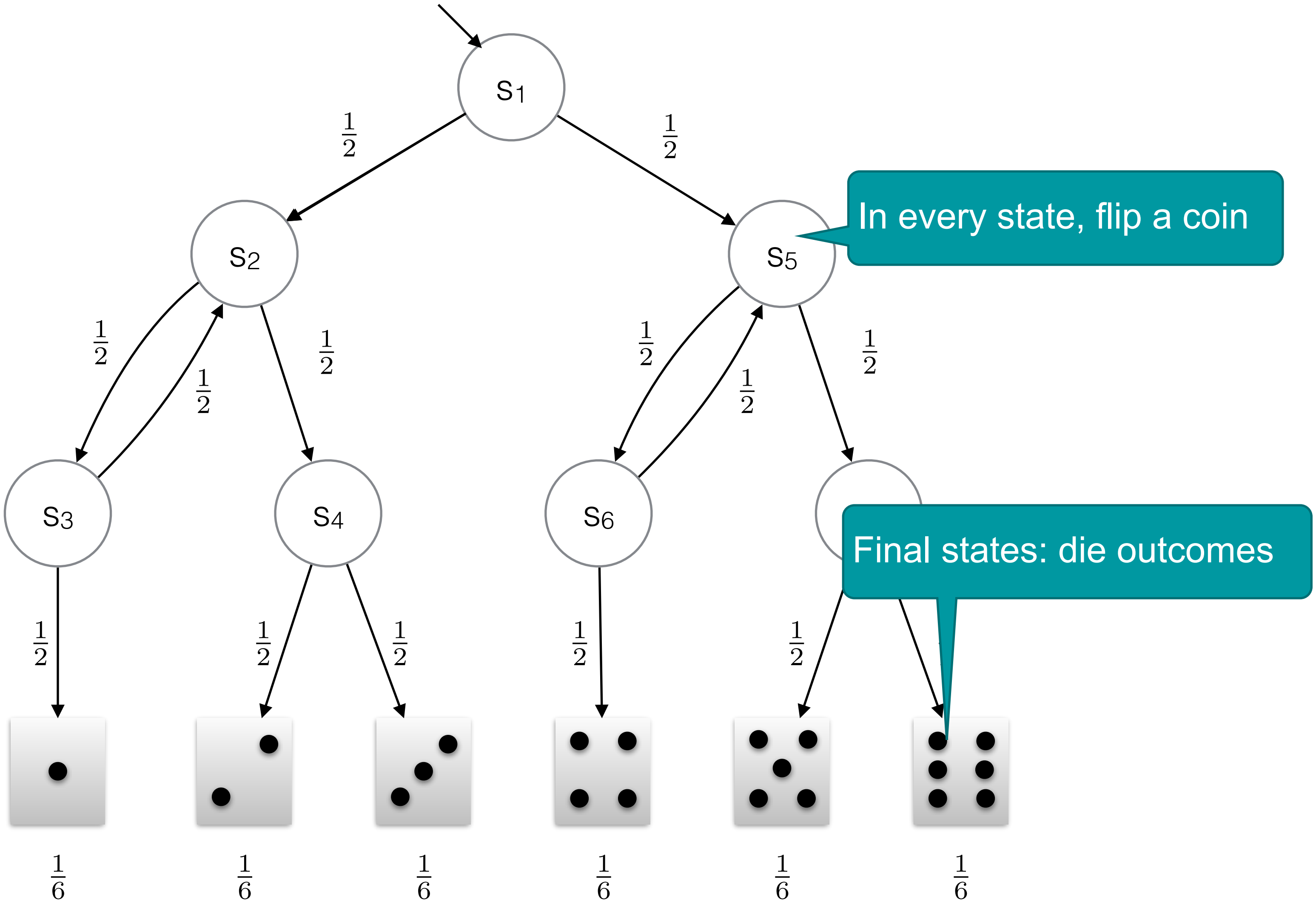


Sebastian Junges & Joost-Pieter Katoen

appliedAI institute for europe
Radboud University
RWTH AACHEN UNIVERSITY

Markov chains

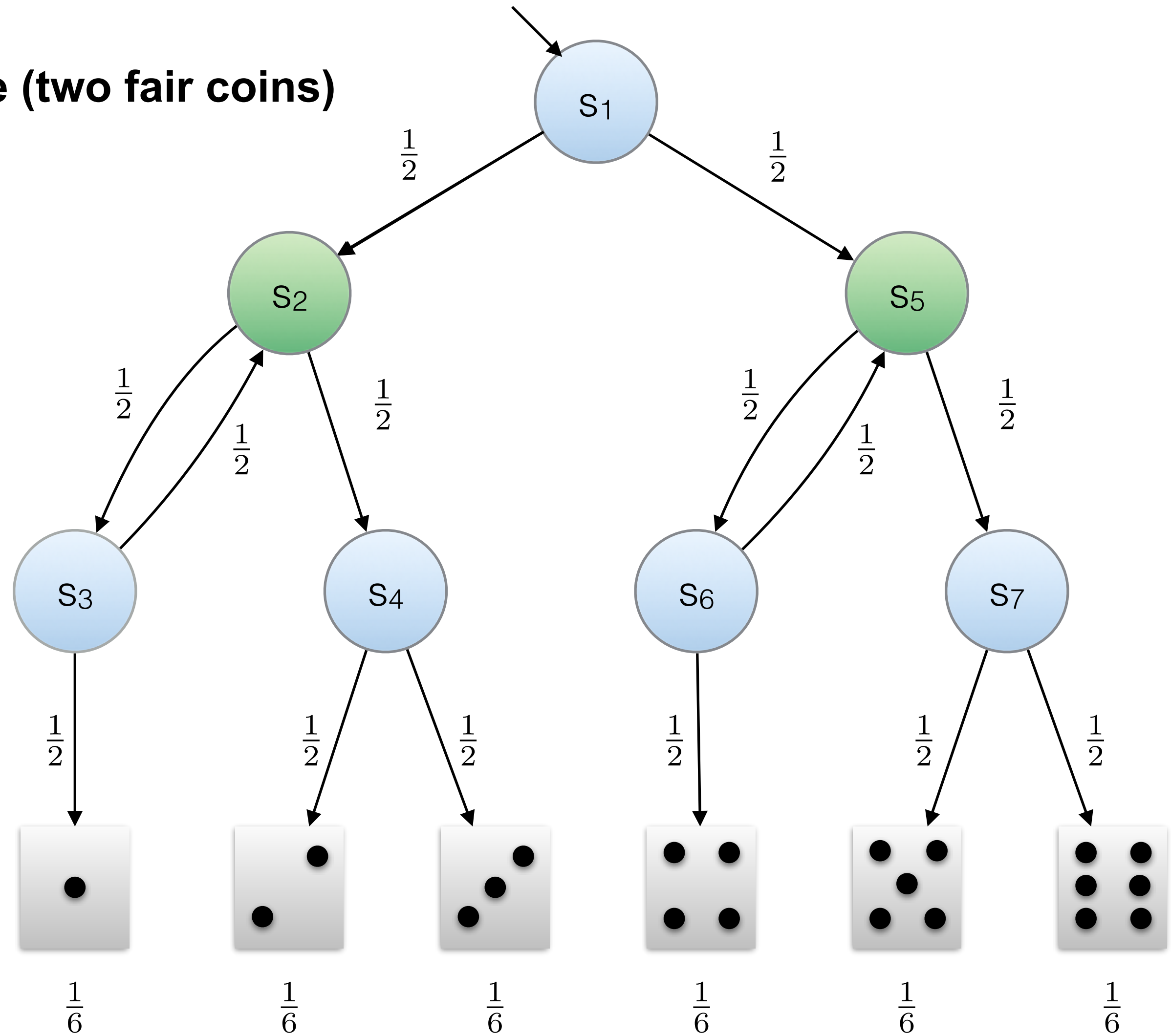
Knuth-Yao die



Sebastian Junges & Joost-Pieter Katoen
Probabilistic Verification Training
AI Transferlab 2023

Markov chains

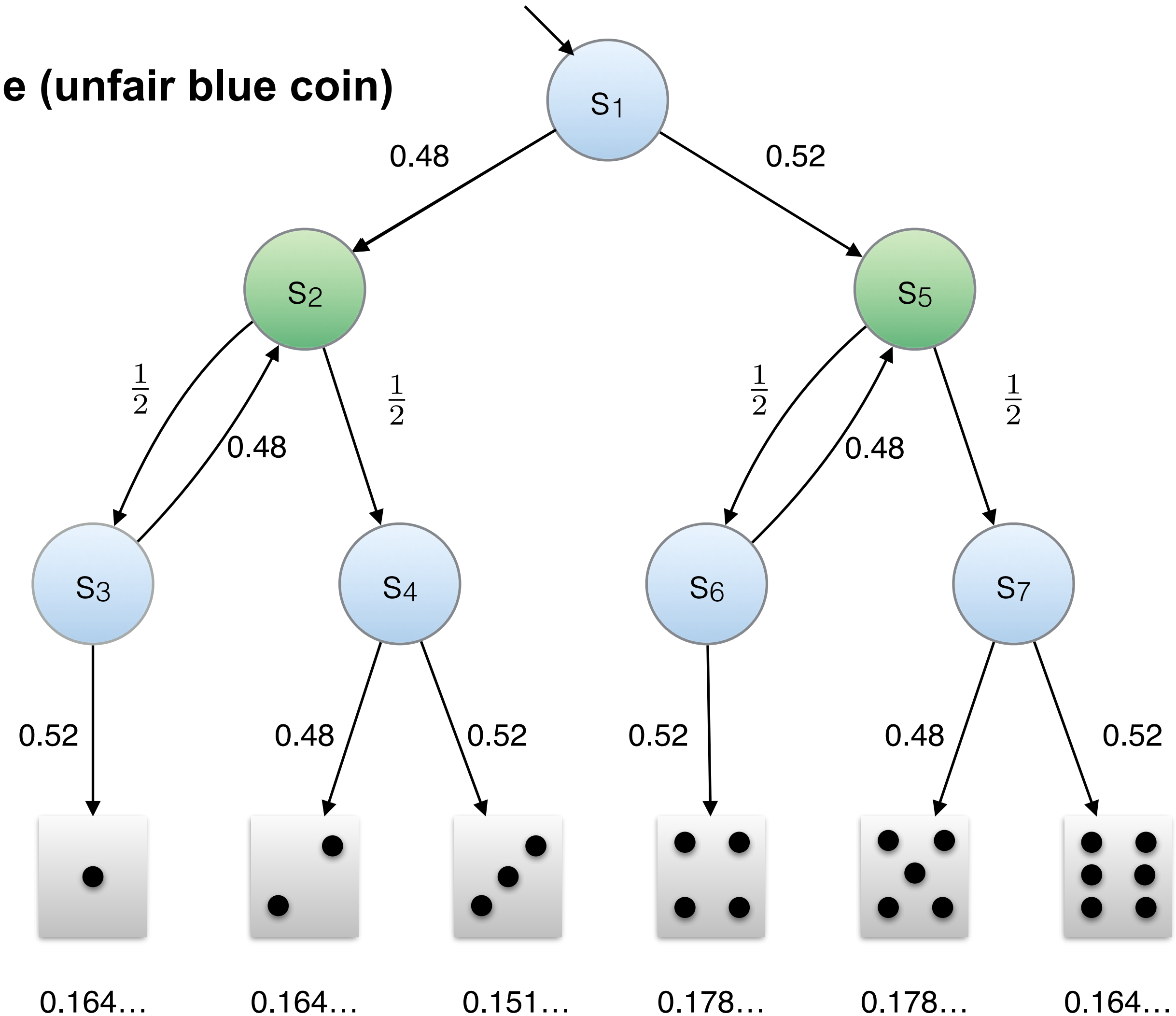
Knuth-Yao die (two fair coins)



Sebastian Junges & Joost-Pieter Katoen
Probabilistic Verification Training
AI Transferlab 2023

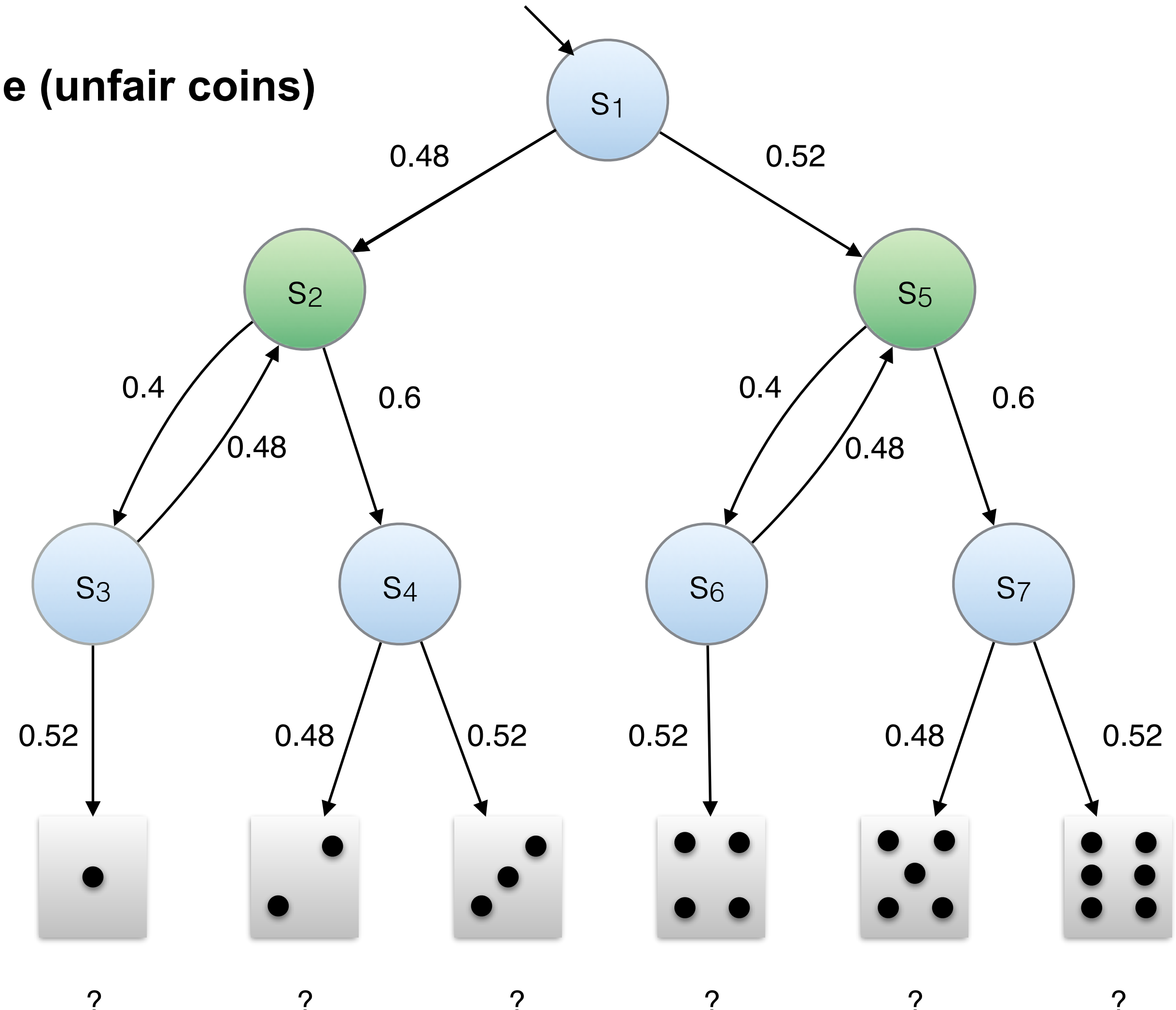
Markov chains

Knuth-Yao die (unfair blue coin)



Markov chains

Knuth-Yao die (unfair coins)



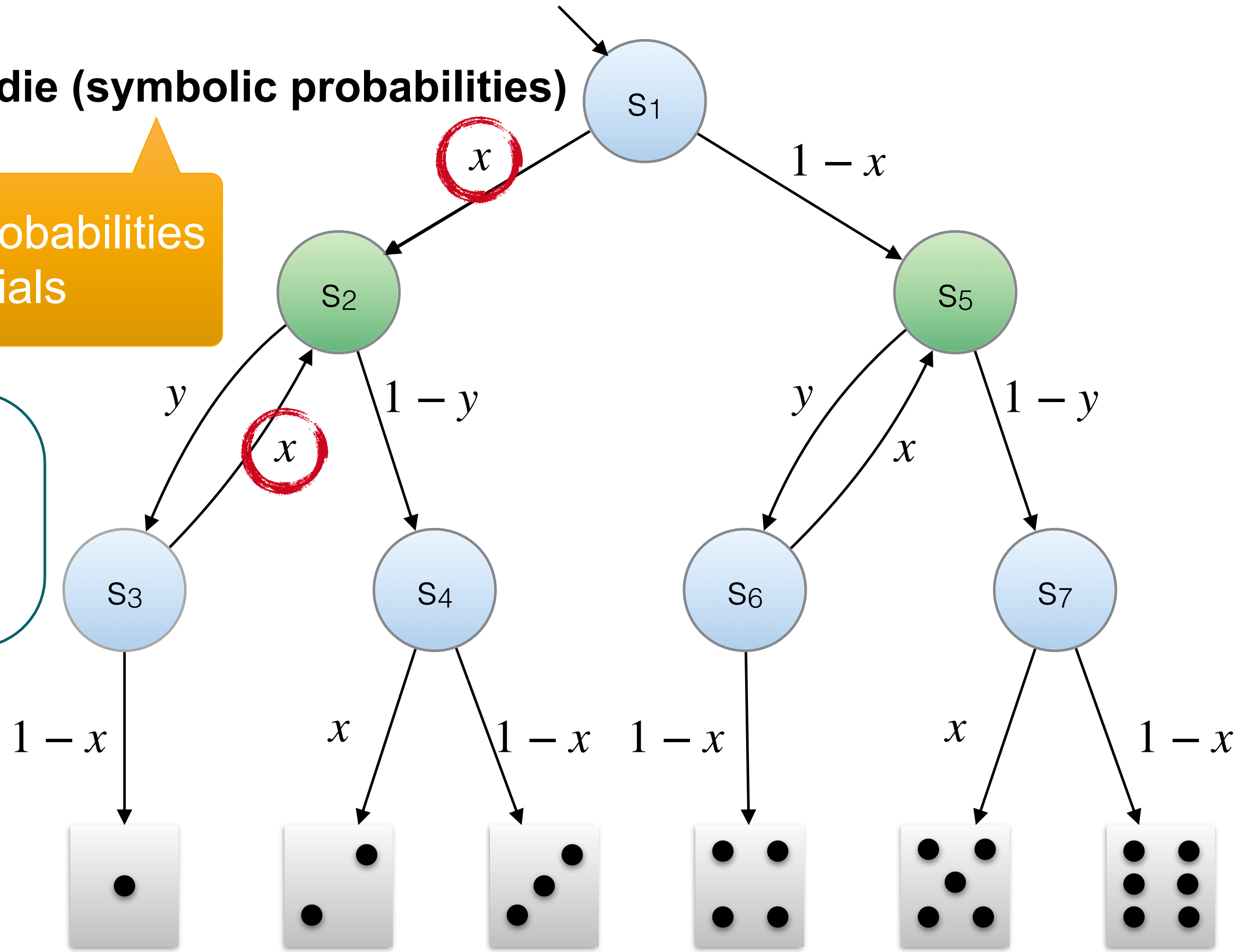
Sebastian Junges & Joost-Pieter Katoen
Probabilistic Verification Training
AI Transferlab 2023

Parametric Markov chains (pMCs)

Knuth-Yao die (symbolic probabilities)

Transition probabilities are polynomials

Unless mentioned otherwise: $\{x, 1-x\}$

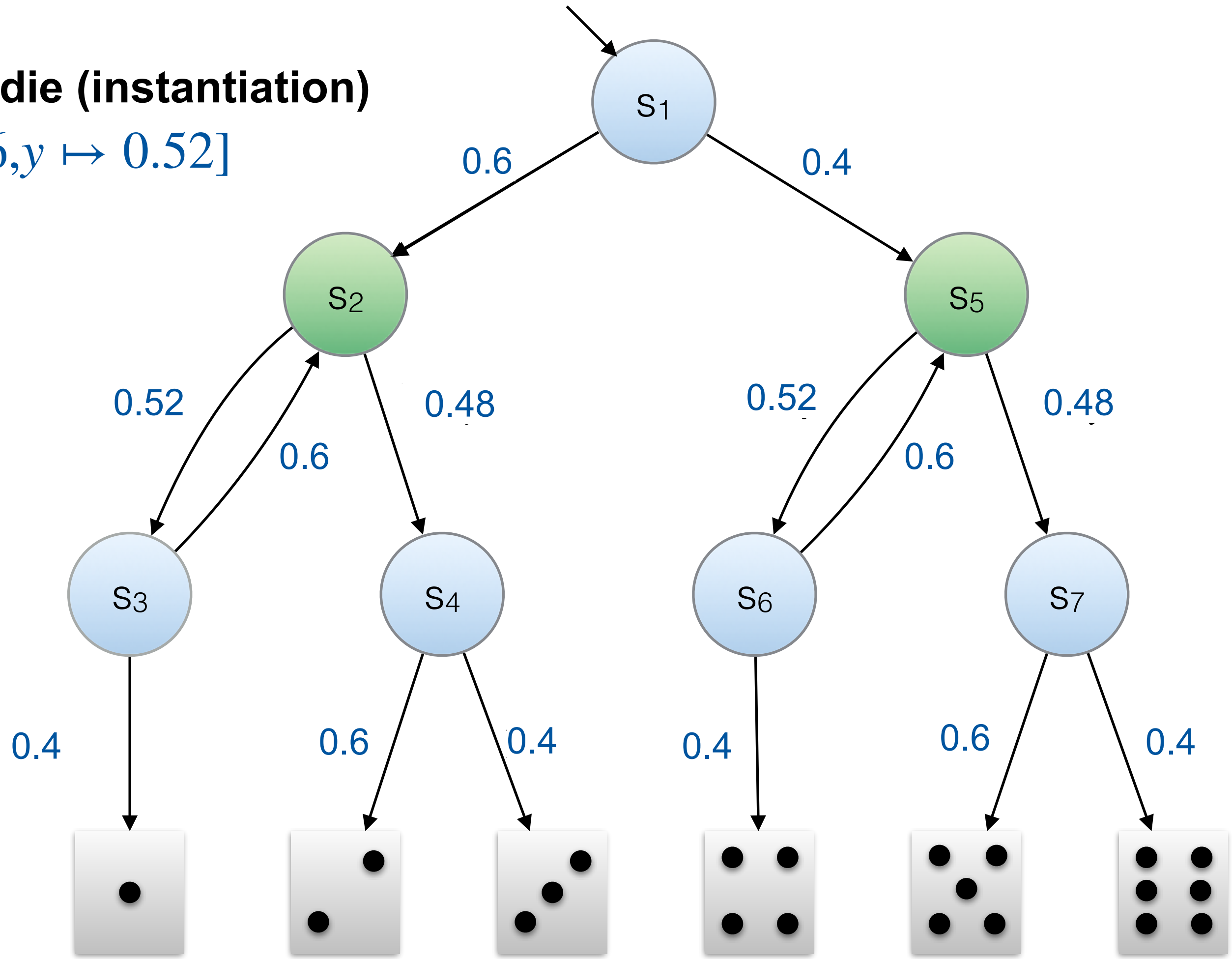


Sebastian Junges & Joost-Pieter Katoen
Probabilistic Verification Training
AI Transferlab 2023

Parametric Markov chains (pMCs)

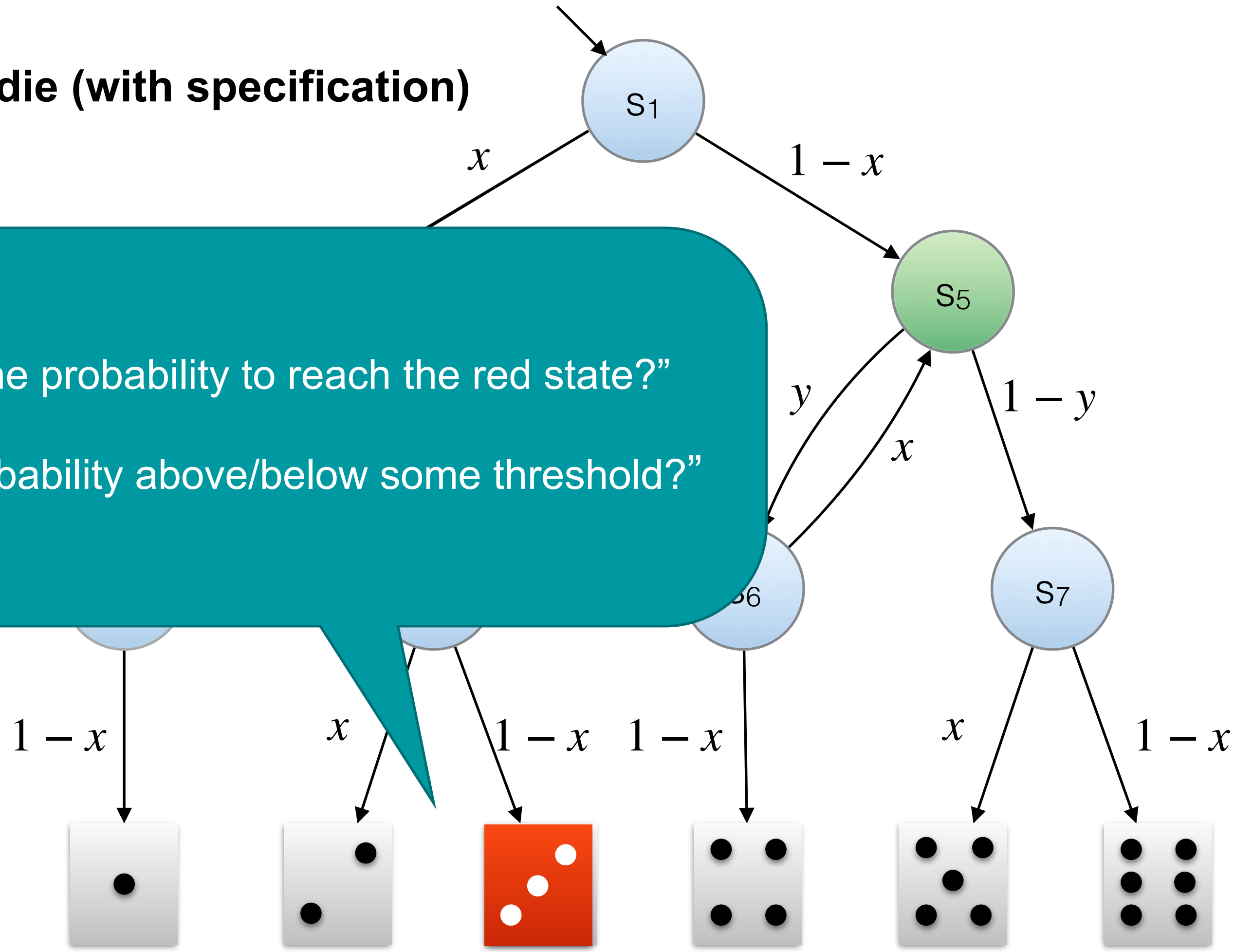
Knuth-Yao die (instantiation)

$\mathscr{M}[x \mapsto 0.6, y \mapsto 0.52]$



Parametric Markov chains (pMCs)

Knuth-Yao die (with specification)

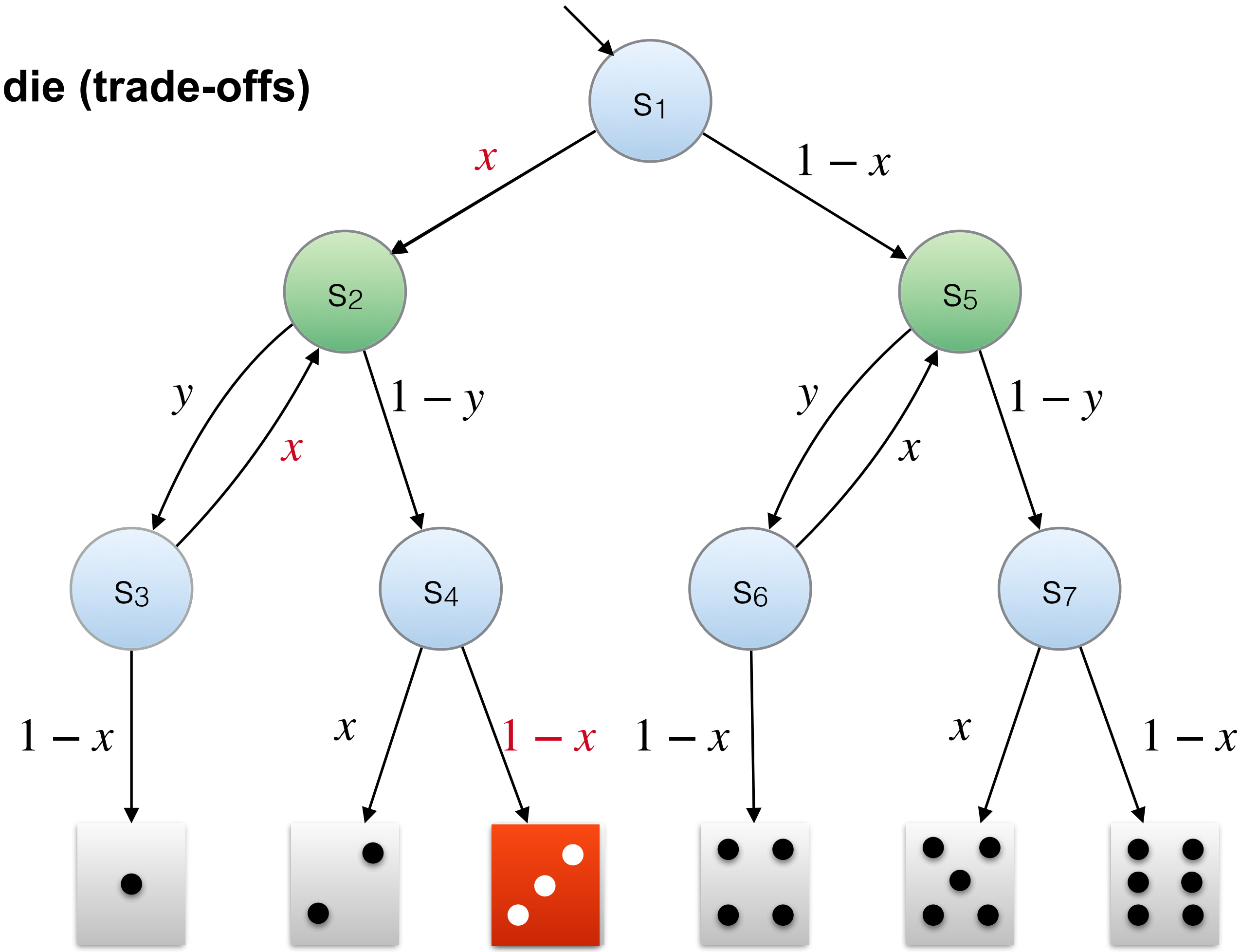


"What is the probability to reach the red state?"
or
"Is the probability above/below some threshold?"

Sebastian Junges & Joost-Pieter Katoen

Parametric Markov chains (pMCs)

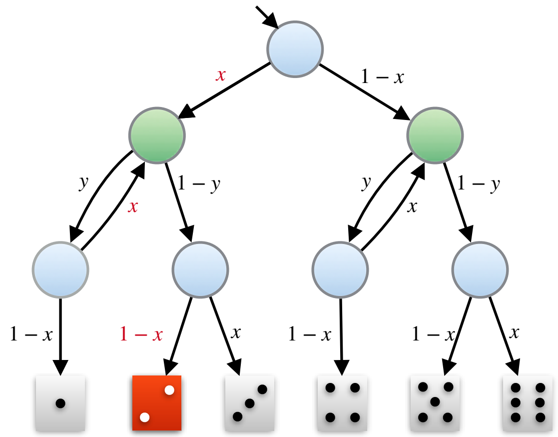
Knuth-Yao die (trade-offs)



Sebastian Junges & Joost-Pieter Katoen

appliedAI institute for europe
Radboud University
RWTH AACHEN UNIVERSITY

Problem statement: Parameter synthesis



Given: a parametric MC $\mathscr{M}$ with parameters $\mathbf{x}$

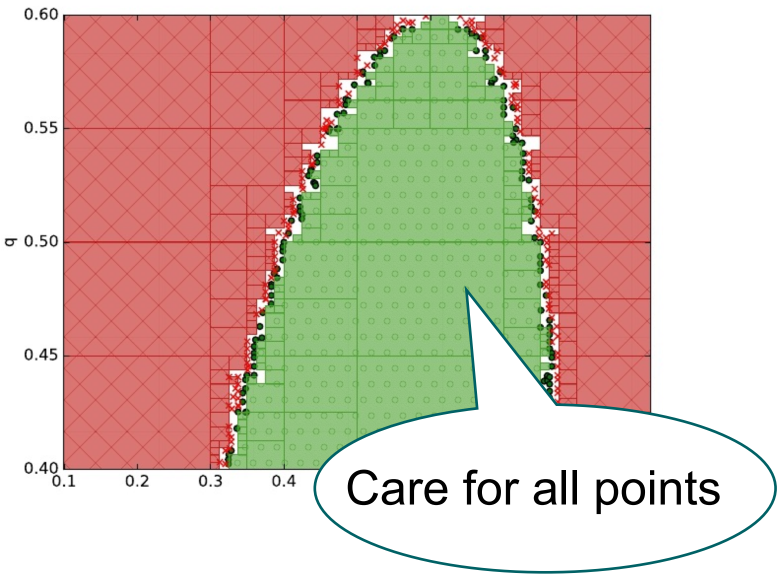
Find: $\text{val} : \mathbf{x} \rightarrow [0,1]$

such that: $\mathscr{M}[\text{val}] \models \varphi$, i.e., a red state is reached with probability at least/at most λ

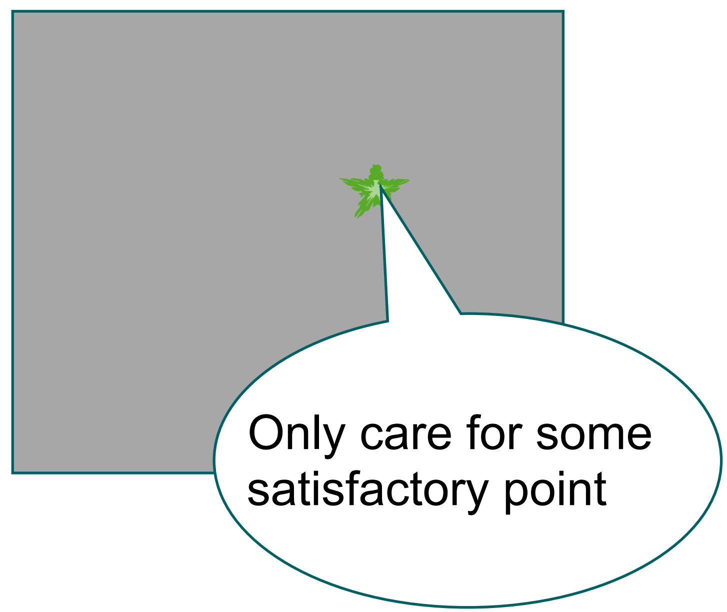
Parameter Synthesis

Various settings

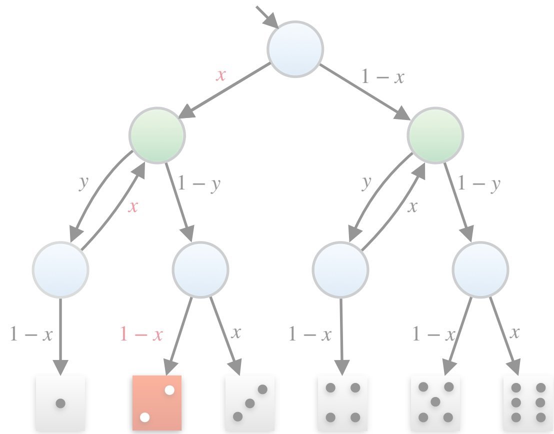
parameter space partitioning

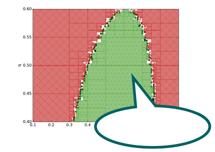


feasibility



Problem statement: Parameter synthesis



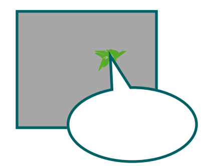


all/ many

Given:

Find: $\mathsf{val} : \mathbf{x} \to [0,1]$

some



a parametric MC $\mathscr{M}$
with parameters $\mathbf{x}$

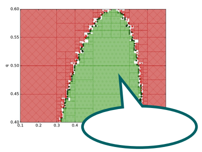
such that: $\mathscr{M}[\mathsf{val}] \models \varphi$, i.e., a red state is reached with probability at least/at most λ

Sebastian Junges & Joost-Pieter Katoen

Two types of motivation

Uncontrollable parameters

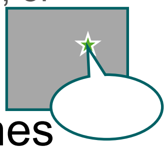
- “robustness”
  probabilities in environment are only estimates



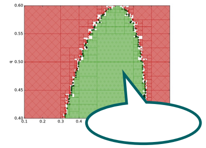
- “effectiveness”
  existence of scenarios that justify redundancy

Controllable parameters

- Randomised algorithms
  to break symmetry in distributed protocols, or
  to maximise entropy



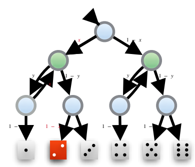
- System configuration, product lines
  E.g, use of higher quality components, or
  use of additional redundancy



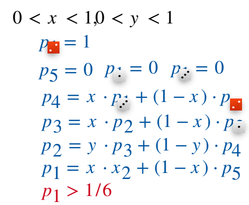
- Small strategies
  for partially observable MDPs

Overview

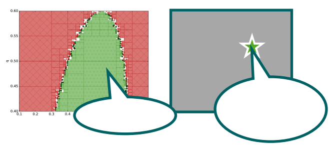
Concepts



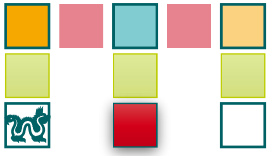
Encoding



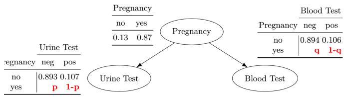
Methods



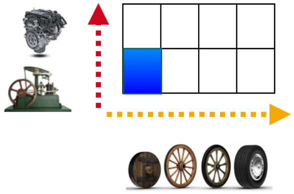
POMDPs



Parametric BNs



Product lines

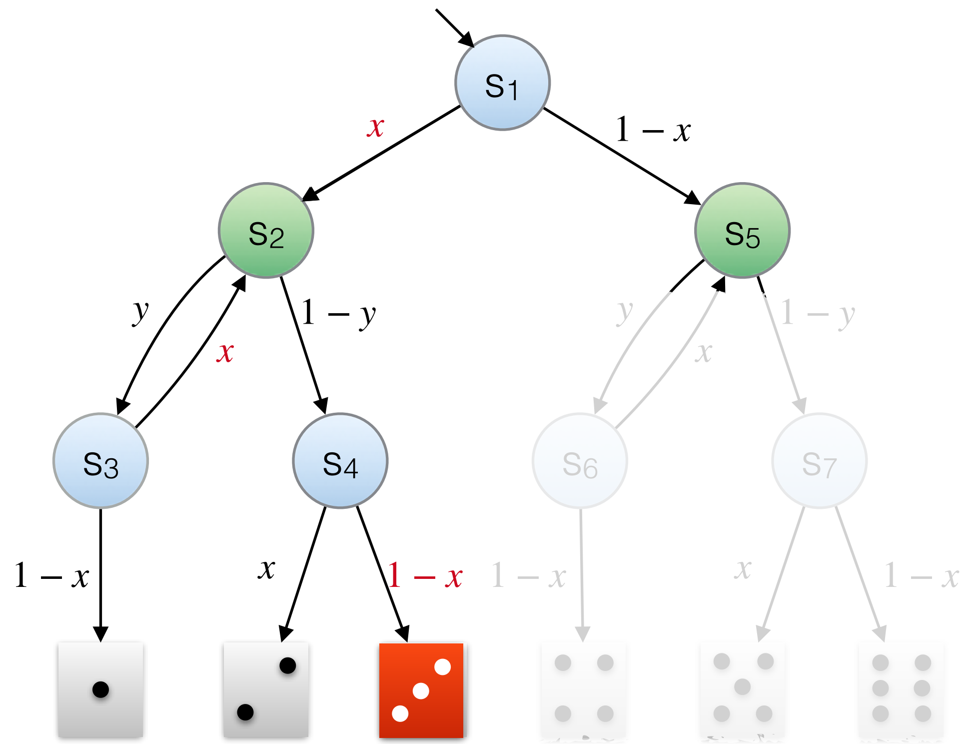


Sebastian Junges & Joost-Pieter Katoen
Probabilistic Verification Training
AI Transferlab 2023

Encoding feasibility in Existential Theory of the Reals (ETR)

Does a valuation exist s.t. a red state is reached with probability is more than 1/6?

yes, iff the constraints are satisfiable



$$\exists p_i \exists x, y :$$

$$0 < x < 1,\ 0 < y < 1$$

$$p_{⚂} = 1$$

$$p_5 = 0 \quad p_{⚀} = 0 \quad p_{⚁} = 0$$

$$p_4 = x \cdot p_{⚁} + (1 - x) \cdot p_{⚂}$$

$$p_3 = x \cdot p_2 + (1 - x) \cdot p_{⚀}$$

$$p_2 = y \cdot p_3 + (1 - y) \cdot p_4$$

$$p_1 = x \cdot x_2 + (1 - x) \cdot p_5$$

$$p_1 > 1/6$$

Sebastian Junges & Joost-Pieter Katoen
Probabilistic Verification Training
AI Transferlab 2023

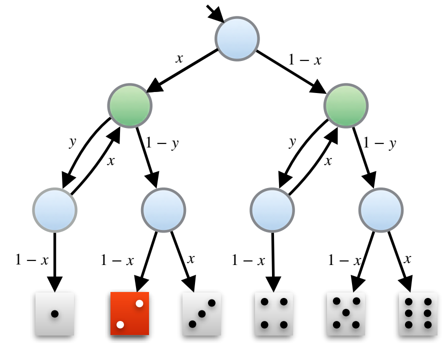
Efficiency?

Solving systems of polynomials — in general — is exponential in number of variables

states + parameters

Eliminating state-variables to get a Solution Function

Results in a rational function $f(\mathbf{x})$ over the parameters $\mathbf{x}$



$$x \cdot (1-x) \cdot \frac{1-y}{1-xy}$$

State elimination (as in NFAs) or Gaussian elimination w/ polynomials

[Daws'04]
[Hahn et al.'11]
[Delgado et al.'11]
[Jansen et al.'14]
[CAV'2015]
[Hutschenreiter et al.'17]
[INFOCOMP'20]

For a pMC with k parameters, n states and linear polynomials as probabilities:

- The rational function can be exponential in k (even for acyclic pMCs)
- For any fixed k, the computation can be done in polynomial time in n

Result of state elimination

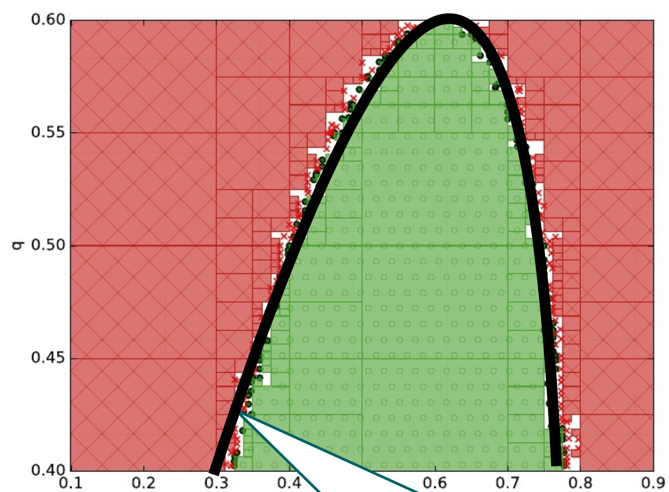
41 States - 138 Transitions - 2 Parameters

The numerator has 408 terms,
The denominator is the product of 48 linear polynomials

5 seconds

Exact Partitioning

Split R into $R_+ = \{ \text{val} \in R \mid \mathscr{M}[\text{val}] \models \varphi \}$ and $R\_ = \{ \text{val} \in R \mid \mathscr{M}[\text{val}v] \not\models \varphi \}$



This curve is the solution function

Sebastian Junges & Joost-Pieter Katoen

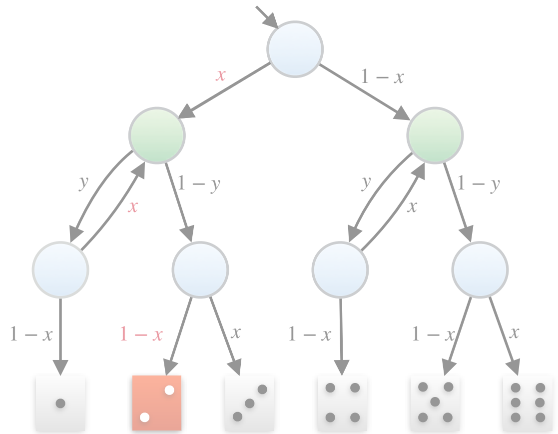
Efficiency?

exponential in parameters

Solving polynomial inequality — in general — is exponential in number of variables

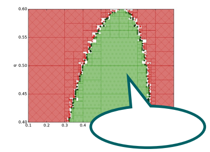
parameters

Problem statement: Parameter synthesis



Given:

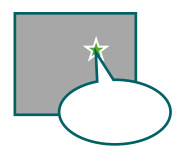
a parametric MDP $\mathscr{M}$
with parameters $\mathbf{x}$



all/
many

Find: $\text{val} : \mathbf{x} \rightarrow [0,1]$

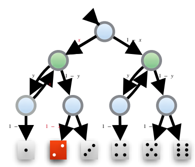
some



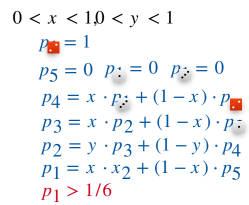
such that: $\mathscr{M}_\sigma[\text{val}] \models \varphi$, i.e., a red state is reached with probability at least/at most λ

Overview

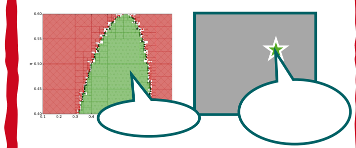
Concepts



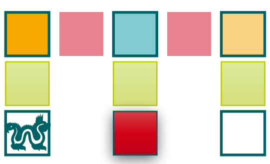
Encoding



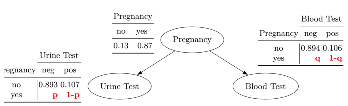
Methods



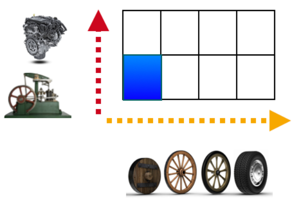
POMDPs



Parametric BNs



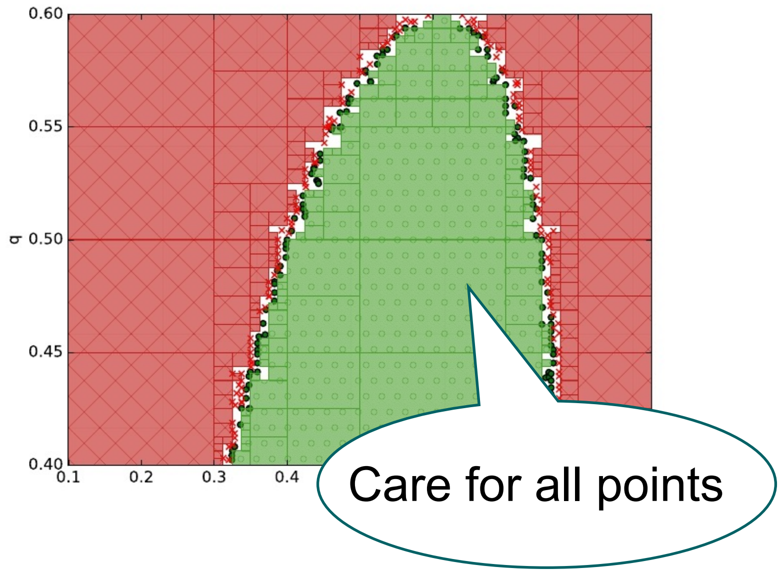
Product lines

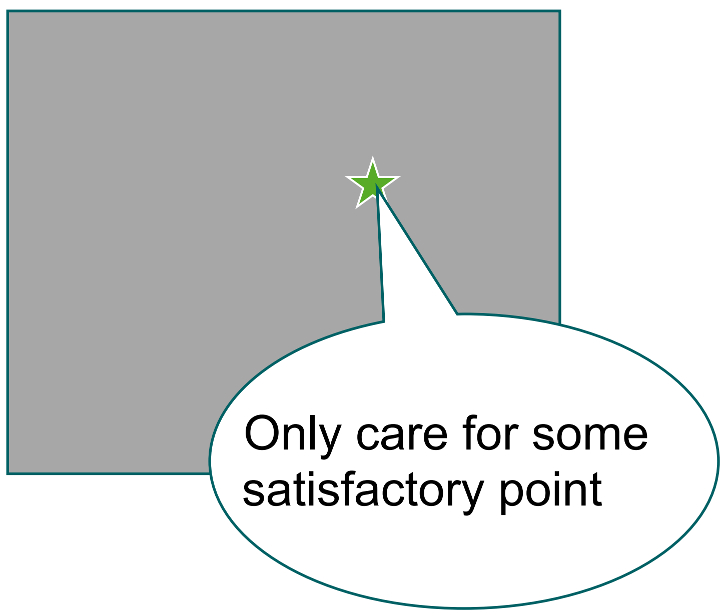


Sebastian Junges & Joost-Pieter Katoen
Probabilistic Verification Training
AI Transferlab 2023

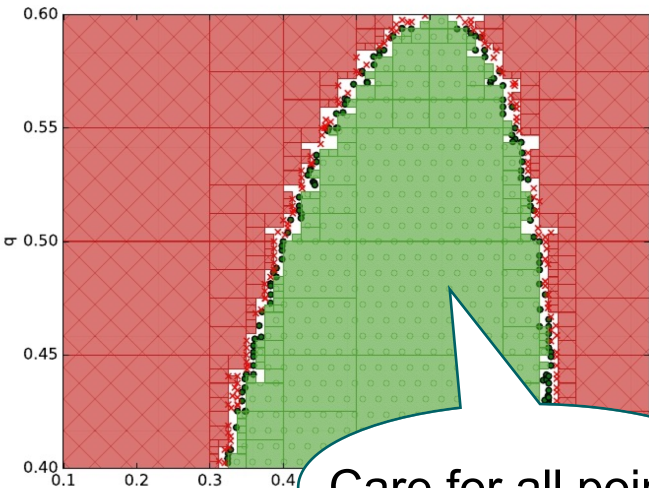
Practical parameter synthesis

Two settings





Practical Parameter Synthesis



Care for all points

Only care for some satisfactory point

Several variants of encoding via SMT solvers [CAV15]

Parameter lifting: [ATVA16, TACAS21] abstraction-refinement

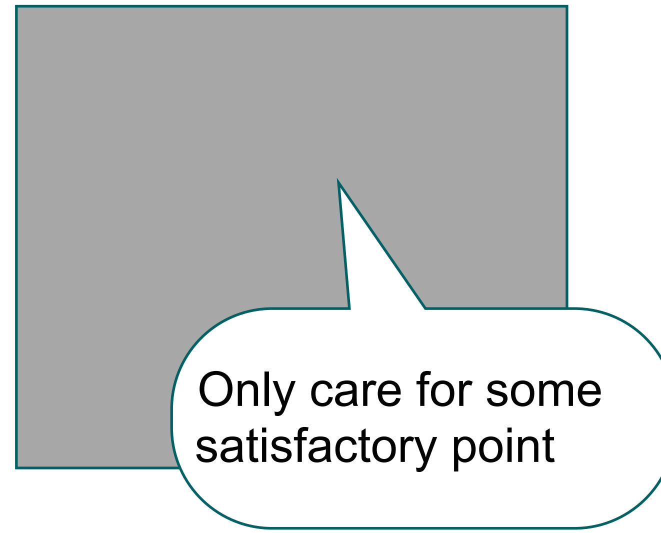
surveyed in [FMSD24]

Sampling based methods: particle swarm or gradient descent [VMCAI'22, Chen et al.'14]

Iterative convex optimisation schemes [TACAS'17, ATVA'18, TAC'22]

Sebastian Junges & Joost-Pieter Katoen
Probabilistic Verification Training
AI Transferlab 2023

Practical Parameter Synthesis



Several variants of encoding via SMT solvers [CAV15]

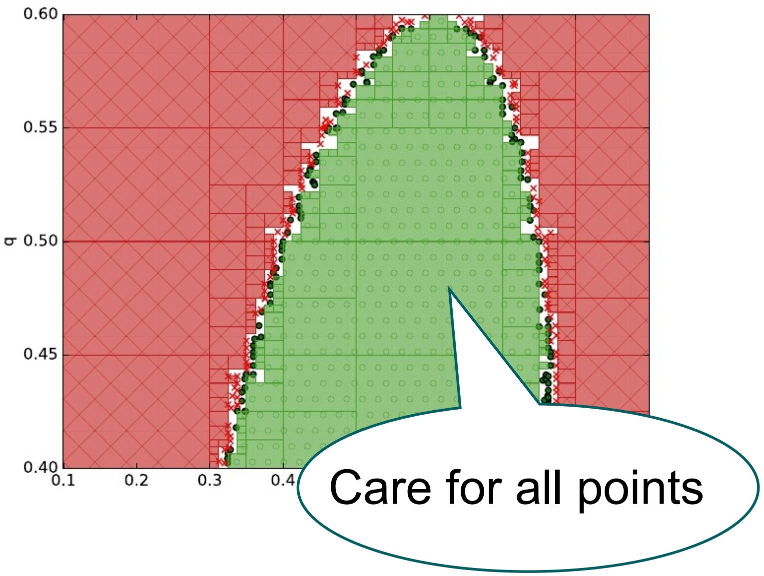
Parameter lifting: [ATVA16, TACAS21] abstraction-refinement

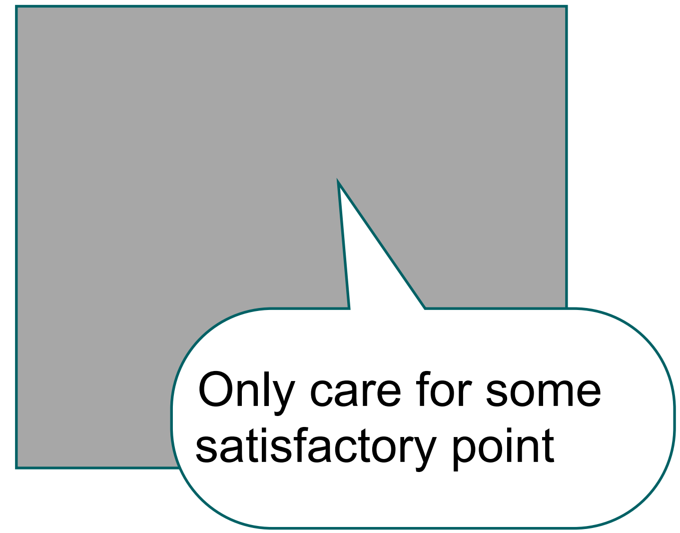
surveyed in [FMSD24]

Sampling based methods: particle swarm or gradient descent [VMCAI'22, Chen et al.'14]

Iterative convex optimisation schemes [TACAS'17, ATVA'18, TAC'22]

Practical Parameter Synthesis





Several variants of encoding via SMT solvers [CAV15]

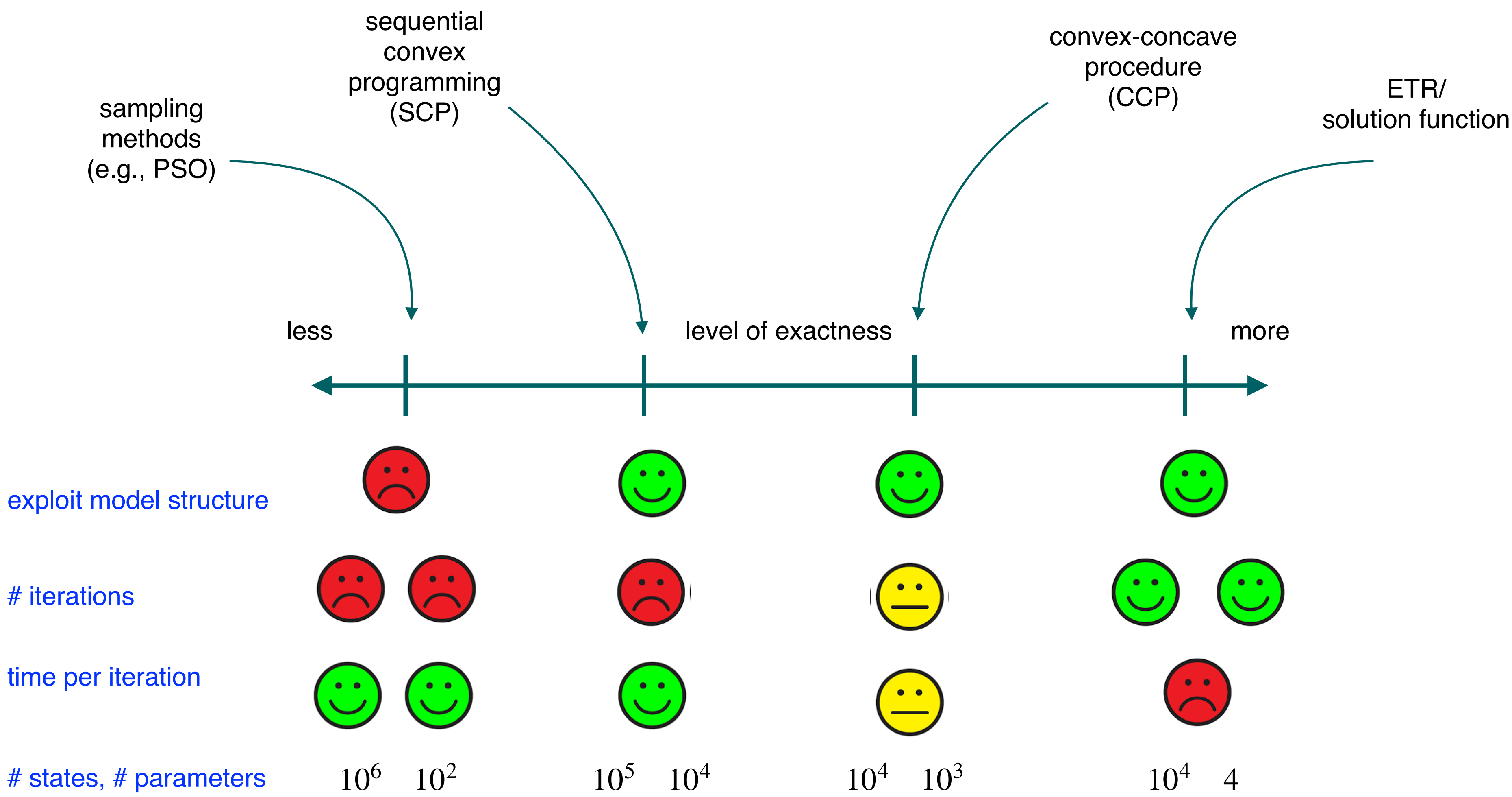
Parameter lifting: [ATVA16, TACAS21] abstraction-refinement

surveyed in [FMSD24]

Sampling based methods: particle swarm or gradient descent [VMCAI'22, Chen et al.'14]

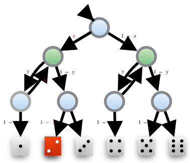
Iterative convex optimisation schemes [TACAS'17, ATVA'18, TAC'22]

Practical Approaches to Feasibility



Overview

Concepts



Encoding

$0 < x < 1, 0 < y < 1$

$p_{⚁} = 1$

$p_5 = 0 \quad p_{⚀} = 0 \quad p_{⚂} = 0$

$p_4 = x \cdot p_{⚂} + (1 - x) \cdot p_{⚁}$

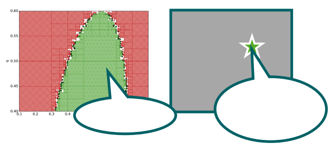
$p_3 = x \cdot p_2 + (1 - x) \cdot p_{⚀}$

$p_2 = y \cdot p_3 + (1 - y) \cdot p_4$

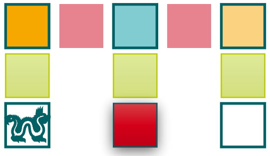
$p_1 = x \cdot x_2 + (1 - x) \cdot p_5$

$p_1 > 1/6$

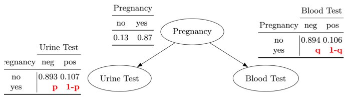
Methods



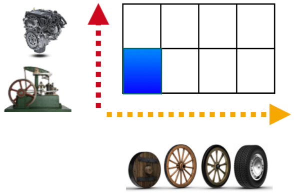
POMDPs



Parametric BNs



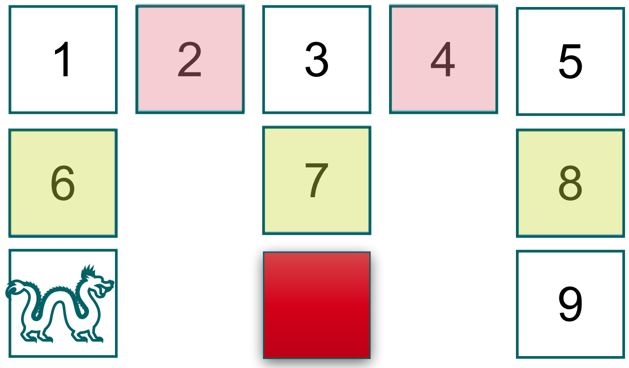
Product lines

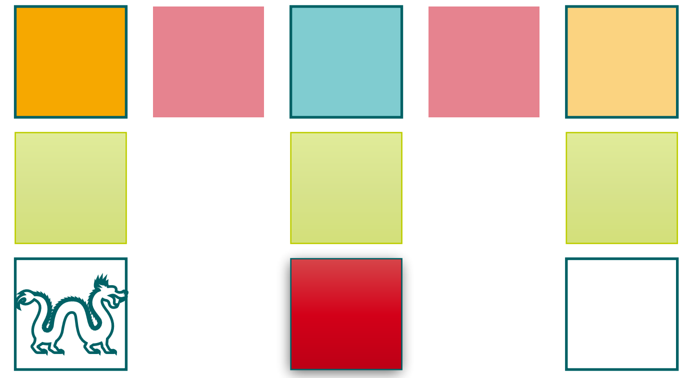


Sebastian Junges & Joost-Pieter Katoen
Probabilistic Verification Training
AI Transferlab 2023

Randomisation and memory

POMDP: Reach red state without visiting the dragon.





same observations:
- {2,4}
- {6,7,8}

Start in 1 or 5:
Positional policy has to randomise in {2,4}

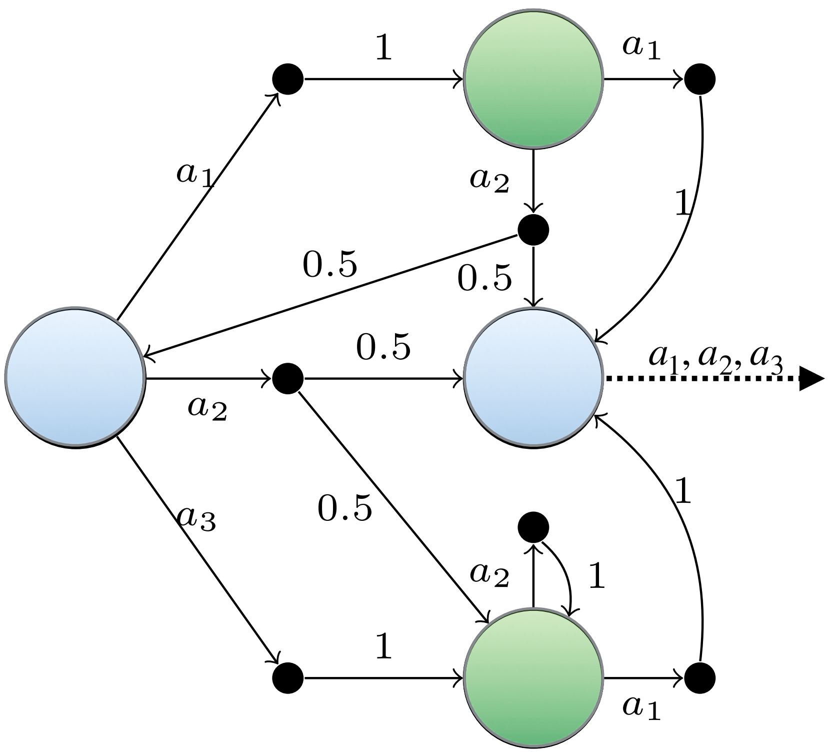
Start in 6 or 7:
no positional policy
store whether we have been in 3

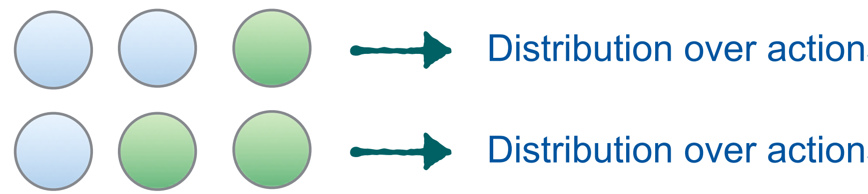
Sebastian Junges & Joost-Pieter Katoen

appliedAI institute for europe
Radboud University
RWTH AACHEN UNIVERSITY

POMDPs

MDPs with ‘observable colours’

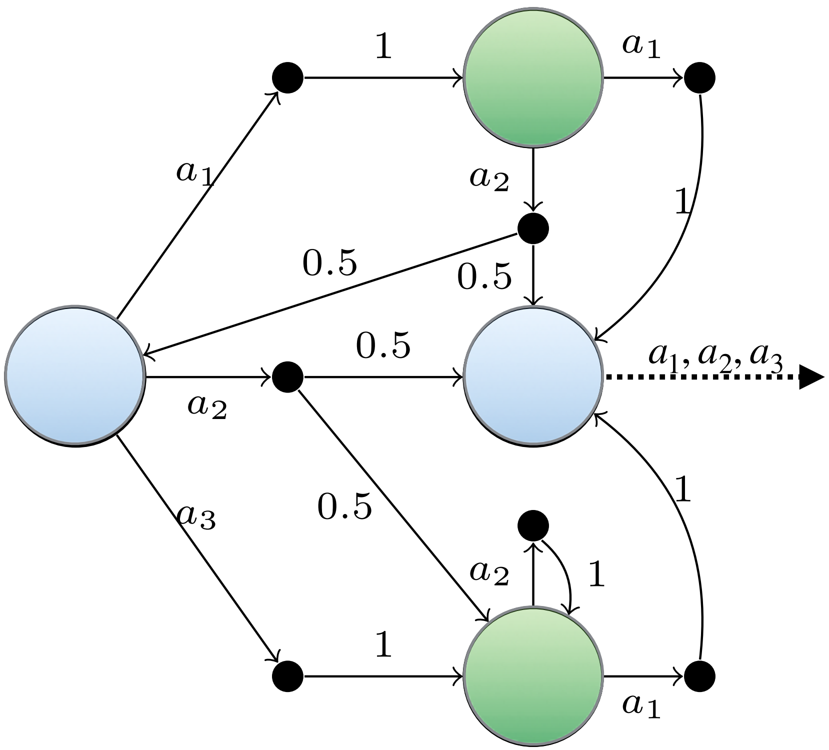




Given **any** POMDP
is there an observation-based policy s.t.
the probability reaching > λ

Sebastian Junges & Joost-Pieter Katoen
Probabilistic Verification Training
AI Transferlab 2023

Partially observable MDPs (POMDPs)



For this talk:
POMDP = MDP with coloured states

Given **any** POMDP
is there a positional policy s.t. the
probability reaching ● $> \lambda$

POMDP
Positional policy:
colours to distributions over actions

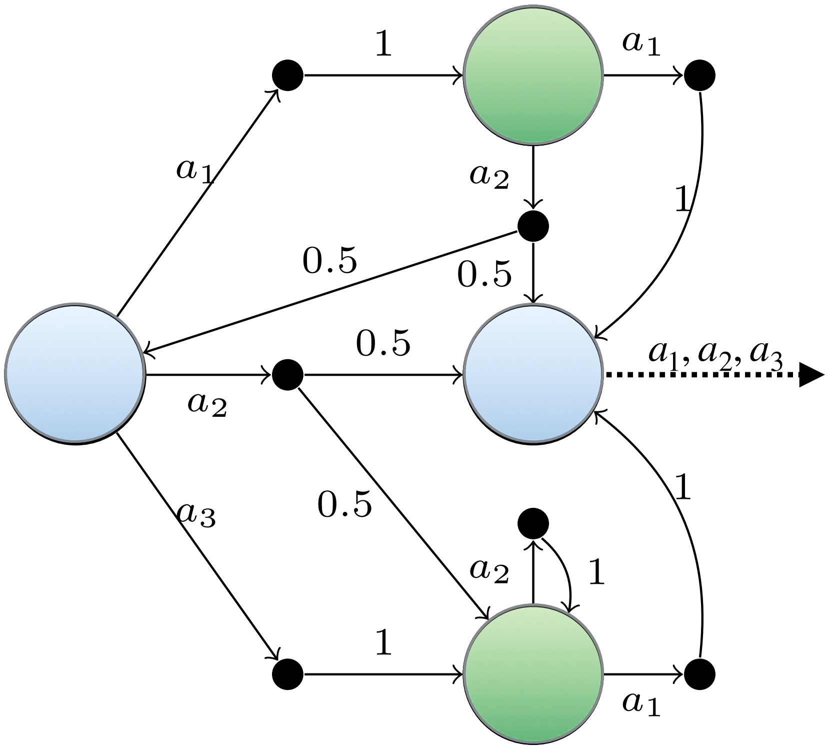
Sebastian Junges & Joost-Pieter Katoen
Probabilistic Verification Training
AI Transferlab 2023

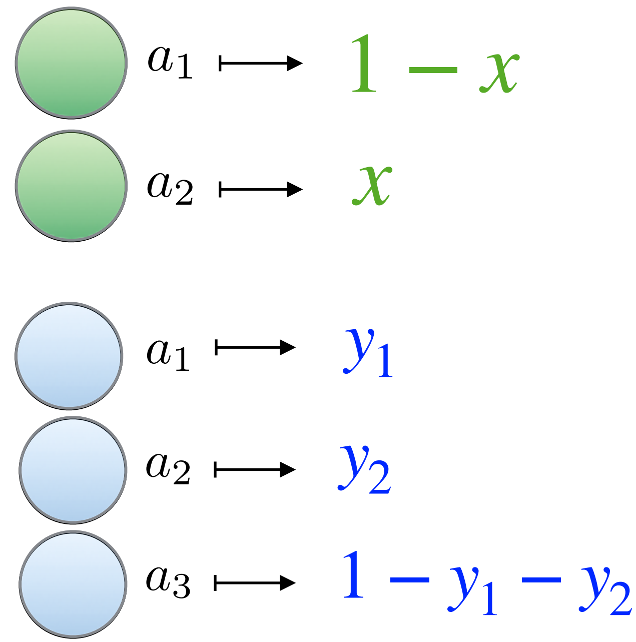
POMDPs with memoryless strategies

[UAI'18]

Maps observations to distributions over actions

maps observation/action pairs to probabilities





Strategy is uniquely described by values for x, y_1, y_2

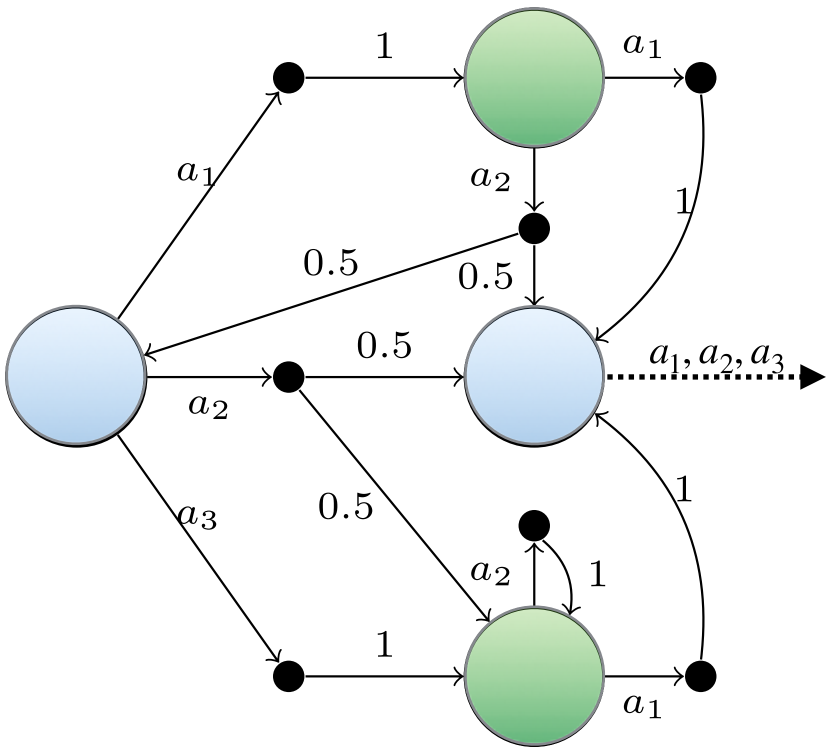
appliedAI institute for europe

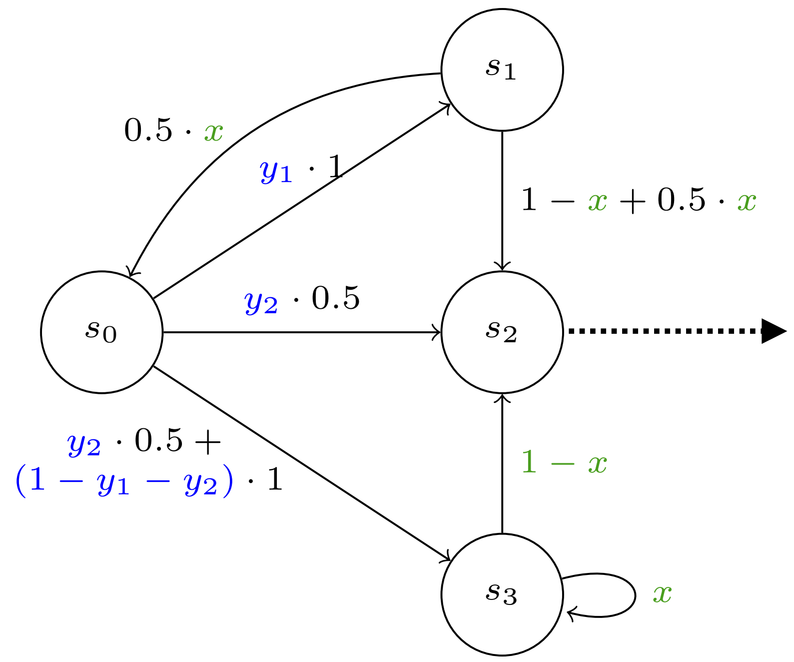
Radboud University

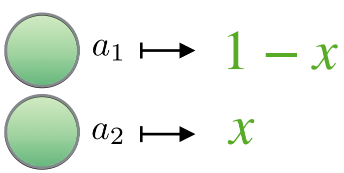
RWTH AACHEN UNIVERSITY

Applying the memoryless strategy

Induced Markov Chain with unknown probabilities







POMDPs and parametric Markov chains

[UAI'18]

Finite-state memory can be supported using a simple reduction

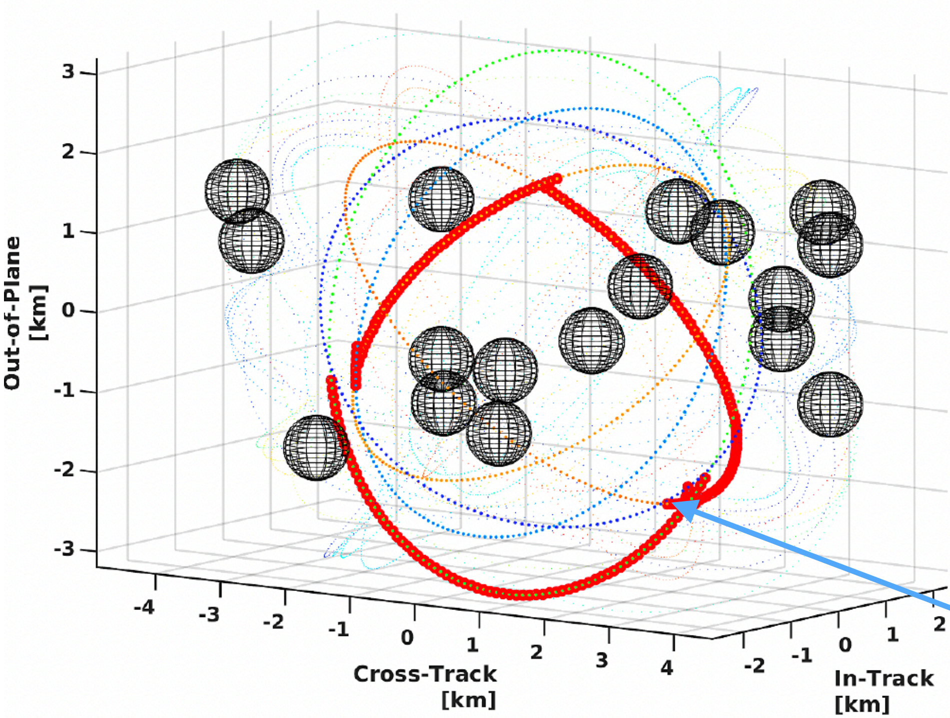
Given **any** POMDP
is there a memoryless policy s.t.
the probability reaching ● > λ

Given **any** pMC
is there a parameter valuation s.t.
the probability reaching ● > λ

Parameter synthesis yields **new competitive methods** for POMDPs

Empirical Results

[IEEE TAC 2022]



start

memoryless

Spec	States	Trans.	Par.	SCP t	SCP iter
$\mathbb{P}_{\geq 0.5}$	6265	17436	231	8	6
$\mathbb{P}_{\geq 0.9}$	6265	17436	231	14	12
$\mathbb{P}_{\geq 0.95}$	6265	17436	231	TO	–
$\mathbb{P}_{\geq 0.95}$	31325	156924	2555	146	10
$\mathbb{P}_{\geq 0.995}$	31325	156924	2555	239	18
$\mathbb{P}_{\geq 0.995}$	217561	615433	2248	386	4
$\mathbb{P}_{\geq 0.995}$	217561	615433	5337	336	4
$\mathbb{P}_{\geq 0.995}$	217561	615433	10042	370	4

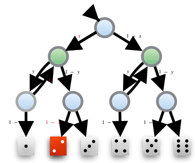
1440, 3600, 7200 observations

Trajectory for a finite-memory policy with memory size five

50% reduction in trajectory length and cost

Overview

Concepts



Encoding

$0 < x < 1, 0 < y < 1$

$p_{⚁} = 1$

$p_5 = 0 \quad p_{⚀} = 0 \quad p_{⚂} = 0$

$p_4 = x \cdot p_{⚂} + (1 - x) \cdot p_{⚁}$

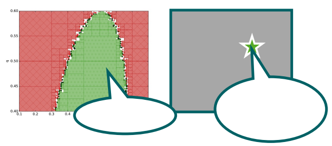
$p_3 = x \cdot p_2 + (1 - x) \cdot p_{⚀}$

$p_2 = y \cdot p_3 + (1 - y) \cdot p_4$

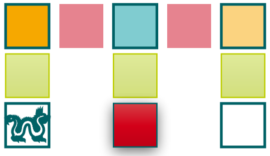
$p_1 = x \cdot x_2 + (1 - x) \cdot p_5$

$p_1 > 1/6$

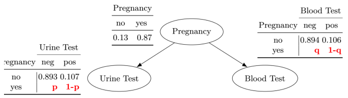
Methods



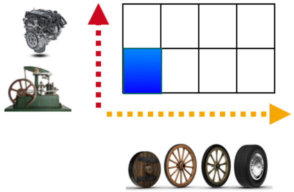
POMDPs



Parametric BNs



Product lines

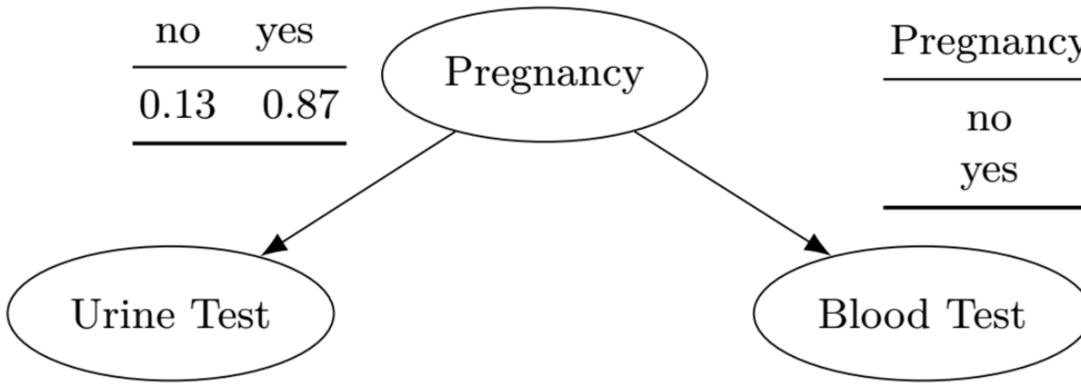


Sebastian Junges & Joost-Pieter Katoen
Probabilistic Verification Training
AI Transferlab 2023

Parametric Bayesian Networks

[Salmani & Katoen, 2022]

Pregnancy	
no	yes
0.13	0.87

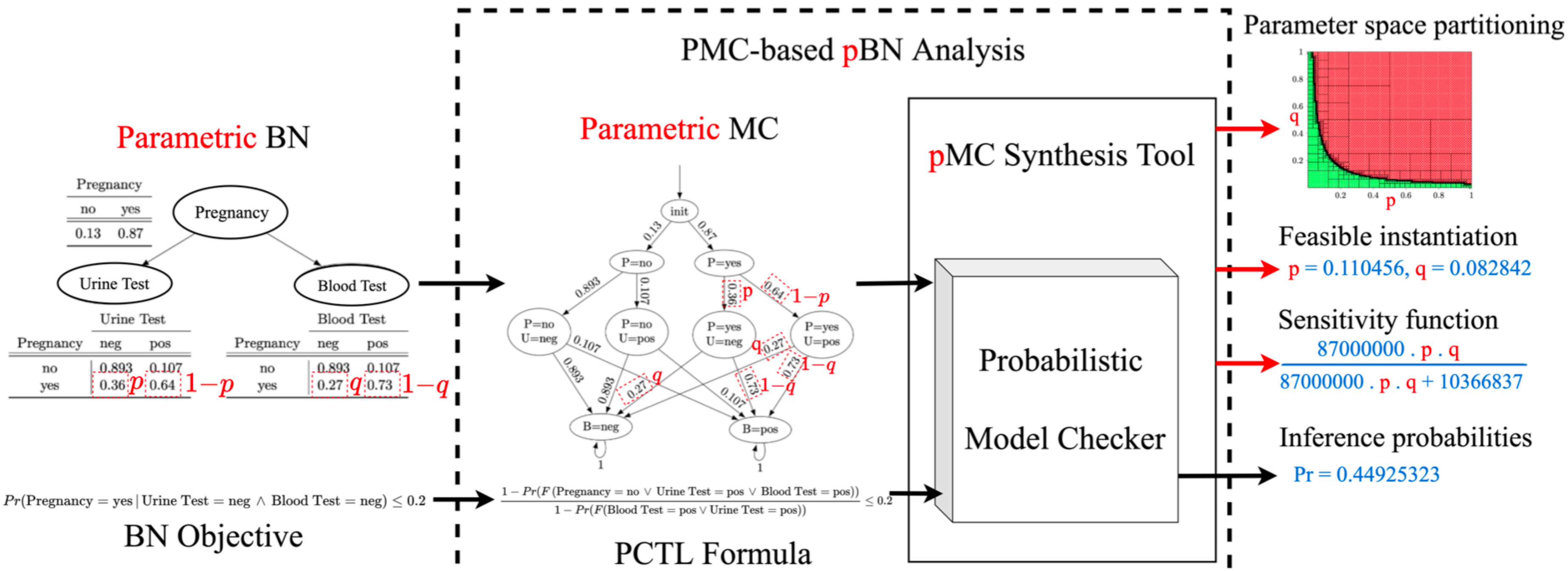


	Urine Test	
Pregnancy	neg	pos
no	0.893	0.107
yes	**p**	**1-p**

	Blood Test	
Pregnancy	neg	pos
no	0.894	0.106
yes	**q**	**1-q**

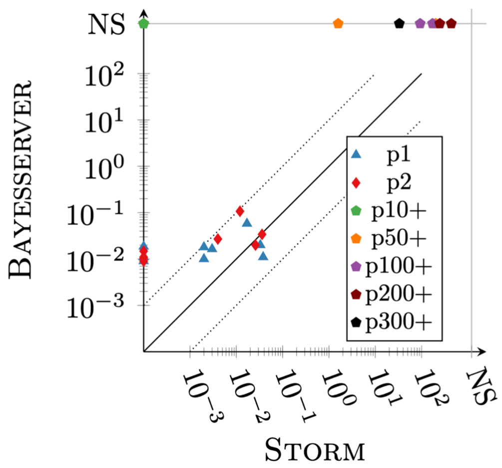
Approach

[Salmani & Katoen, 2022]

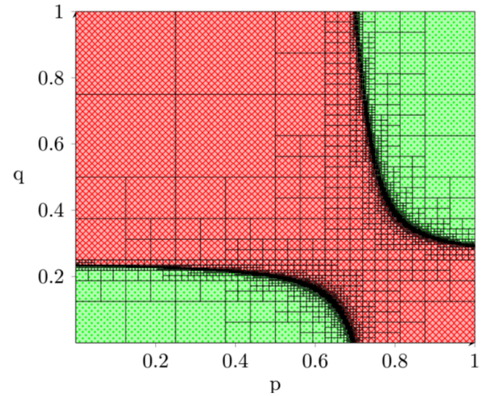


Some Results for pBNs

[Salmani & Katoen, 2022]



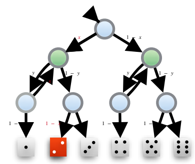
Fast sensitivity analysis with many parameters



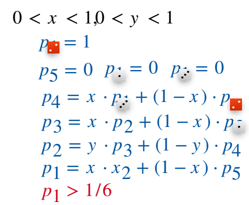
Parameter partitioning ("alarm" benchmark)

Overview

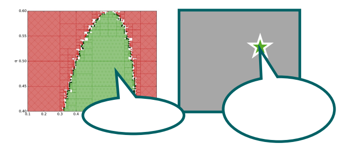
Concepts



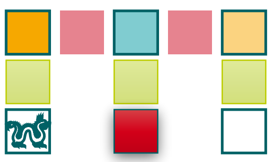
Encoding



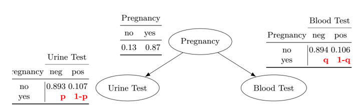
Methods



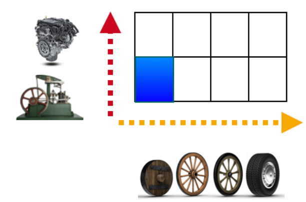
POMDPs



Parametric BNs

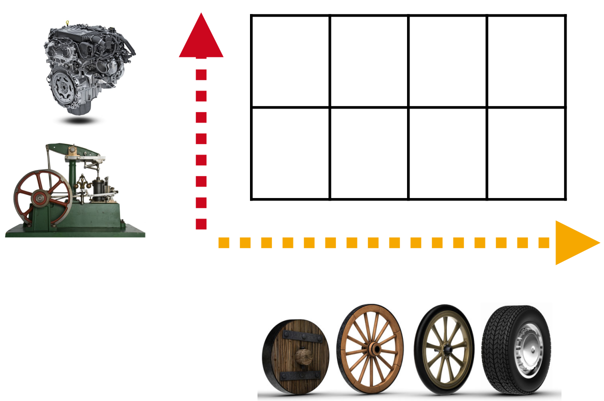


Product lines



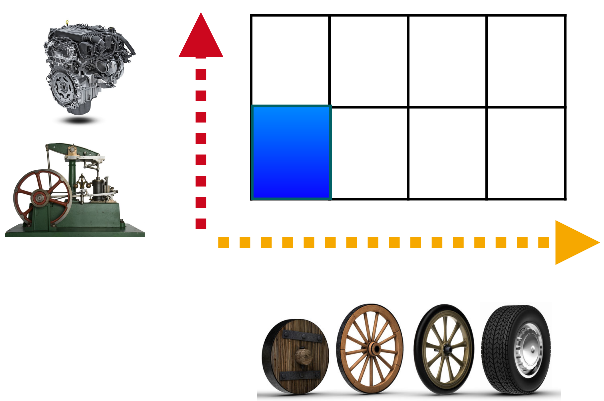
A discrete setting

Product line



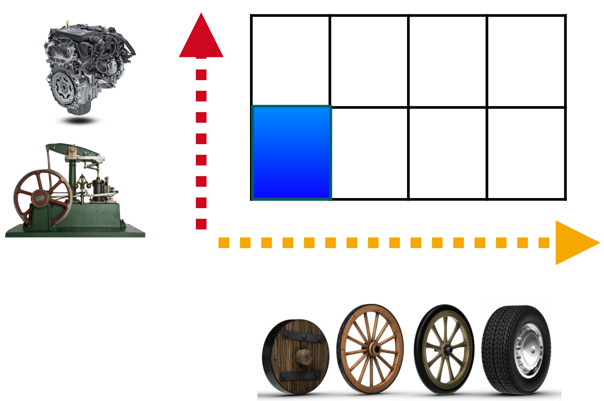
A discrete setting

Product line

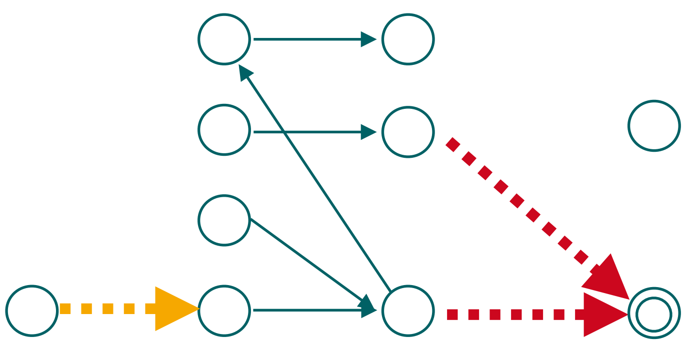


A discrete setting

Product line

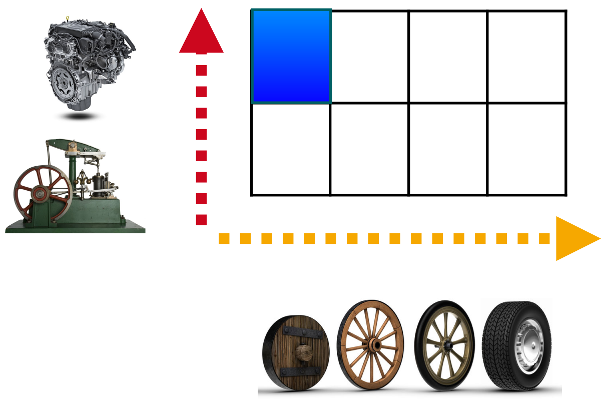


Markov chain to evaluate reliability:

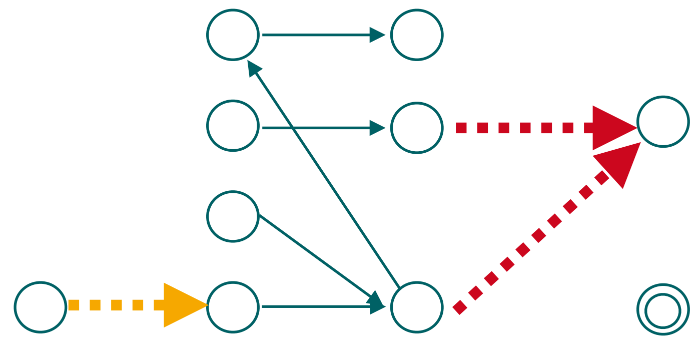


A discrete setting

Product line



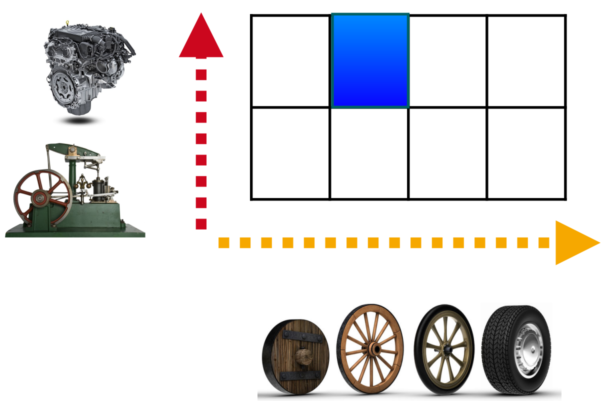
Markov chain to evaluate reliability:



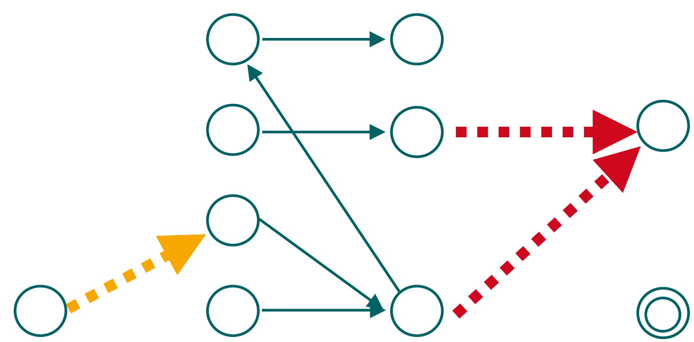
Sebastian Junges & Joost-Pieter Katoen
Probabilistic Verification Training
AI Transferlab 2023

A discrete setting

Product line



Markov chain to evaluate reliability:

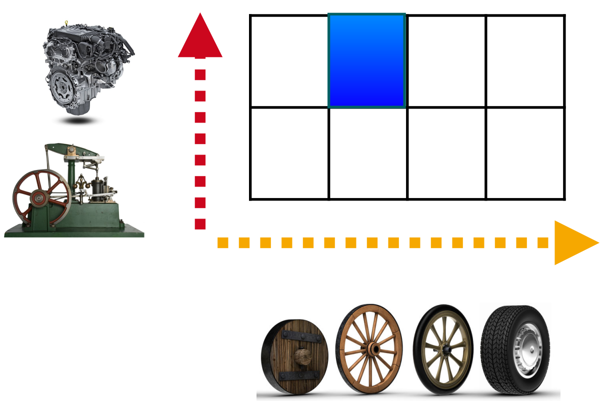


Sebastian Junges & Joost-Pieter Katoen
Probabilistic Verification Training
AI Transferlab 2023

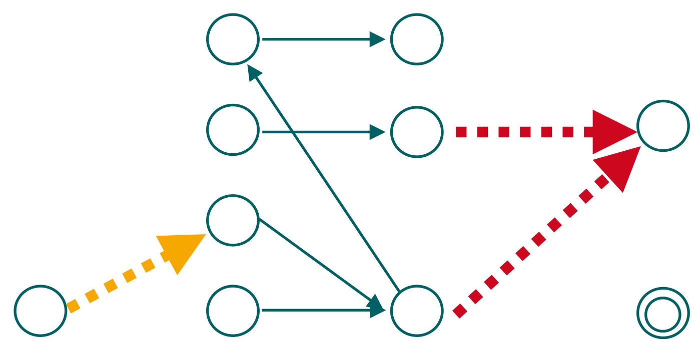
appliedAI institute for europe
Radboud University
RWTH AACHEN UNIVERSITY

A discrete setting

Product line



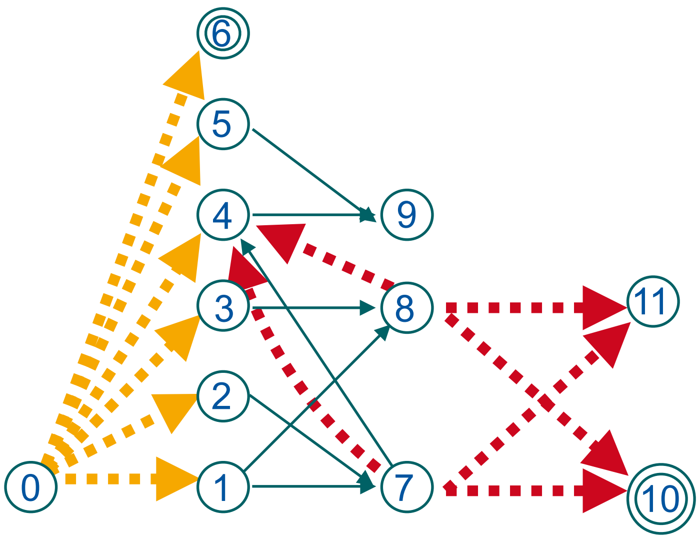
Markov chain to evaluate reliability:



Families (simplified, informal)

- **States {0,…..,11}, initial state 0**

- Parameters (subsets of states)
  - ★ **H** from **{1,2,3,4,5,6}**
  - ★ **K** from **{4,10,11}**
- Transitions:
  - ★ from **state 0**
    - ✤ with probability **1.0** to state **H**
  - ★ from **state 1**
    - ✤ 0.5: to state 7
    - ✤ 0.5: to state 8
  - ★ …
  - ★ from **state 7**
    - ✤ 0.2: to state 4
    - ✤ 0.8: to state **K**
  - ★ from **state 8**
    - ✤ 1.0: to state **K**



Families (simplified, informal)

- **States {0,…..,11}, initial state 0**

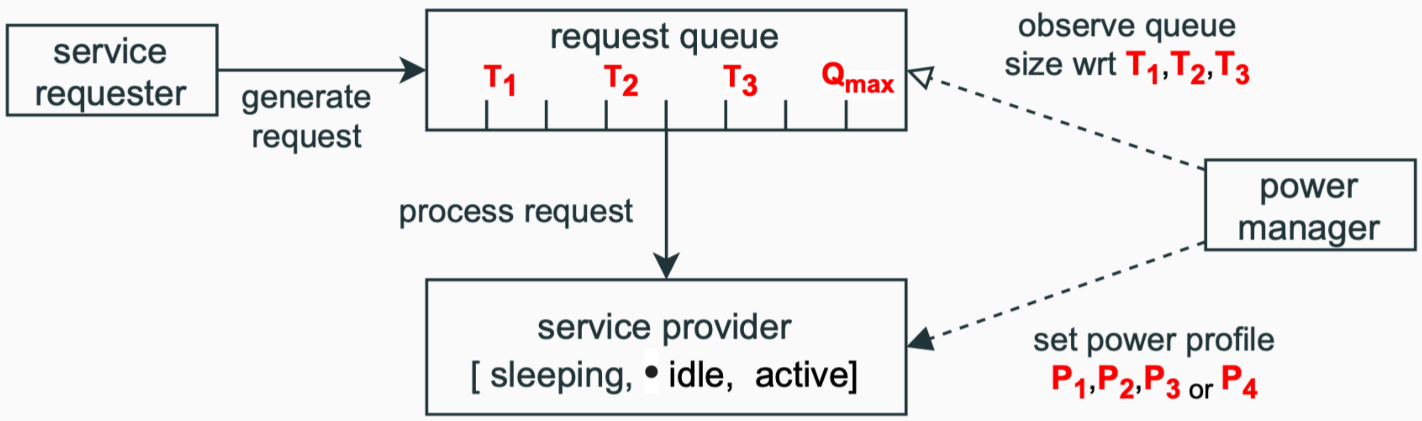
- Parameters (subsets of states)
  - ★ **H** from **{1,2,3,4,5,6}**
  - ★ **K** from **{4,10,11}**
- Transitions:
  - ★ from **state 0**
    - ✤ with probability **1.0** to state **H**
  - ★ from **state 1**
    - ✤ 0.5: to state 7
    - ✤ 0.5: to state 8
  - ★ …
  - ★ from **state 7**
    - ✤ 0.2: to state 4
    - ✤ 0.8: to state **K**
  - ★ from **state 8**
    - ✤ 1.0: to state **K**

```
dtmc

const int H;
const int K;

module example
    s : [0..11] init 0;
    [] s=0 -> 1:   (s'=H);
    [] s=1 -> 0.5:(s'=7) + 0.5:(s'=8);
    //...
    [] s=7 -> 0.8:(s'=K) + 0.2:(s'=2);
    [] s=8 -> 1:   (s'=K);
    //...
endmodule
```

PAYNT example (1)

Details: see
https://github.com/randriu/synthesis

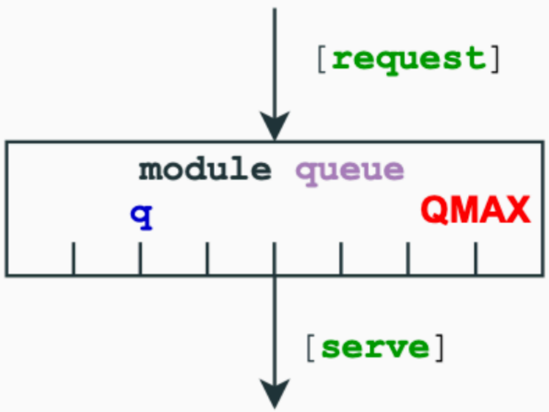
[CAV'21]



- specification:
  - expected number of lost requests must be at most 1
  - expected power consumption is minimal
- problem: how to choose $\mathbf{Q_{max}}, \mathbf{T_1}, \mathbf{T_2}, \mathbf{T_3}, \mathbf{P_1}, \mathbf{P_2}, \mathbf{P_3}, \mathbf{P_4}$ in order to satisfy the specification?

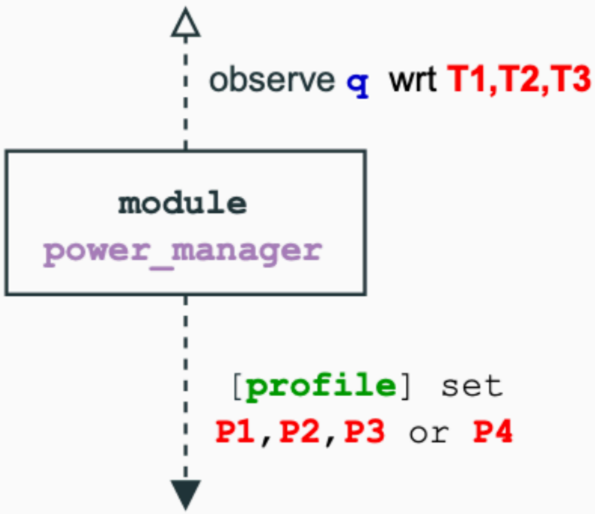
PAYNT example (2)


```
R{"lost"}<=1 [F finished];
R{"power"}min=? [F finished];
```




```
...

hole int QMAX in {1,2,3,4,5,6,7,8,9,10};

module queue
    q : [0..QMAX] init 0;
    lost : [0..1] init 0;

    [request] q < QMAX -> (q'=q+1) & (lost'=0);
    [request] q = QMAX -> (lost'=1);

    [serve] q > 0 -> (q'=q-1) & (lost'=0);
endmodule

...
```

PAYNT example (3)




```
...

hole double T1 in {0.0,0.1,0.2,0.3};
hole double T2 in {0.4,0.5,0.6};
hole double T3 in {0.7,0.8,0.9};

// 0 - sleep, 1 - idle, 2 - active
hole int P1 in {0,1,2};
hole int P2 in {0,1,2};
hole int P3 in {0,1,2};
hole int P4 in {0,1,2};

module power_manager
    pm : [0..2] init 0;

    [profile] q <= T1*QMAX -> (pm'=P1);
    [profile] q > T1*QMAX & q <= T2*QMAX -> (pm'=P2);
    [profile] q > T2*QMAX & q <= T3*QMAX -> (pm'=P3);
    [profile] q <= T3*QMAX -> (pm'=P4);
endmodule

...
```

Using PAYNT

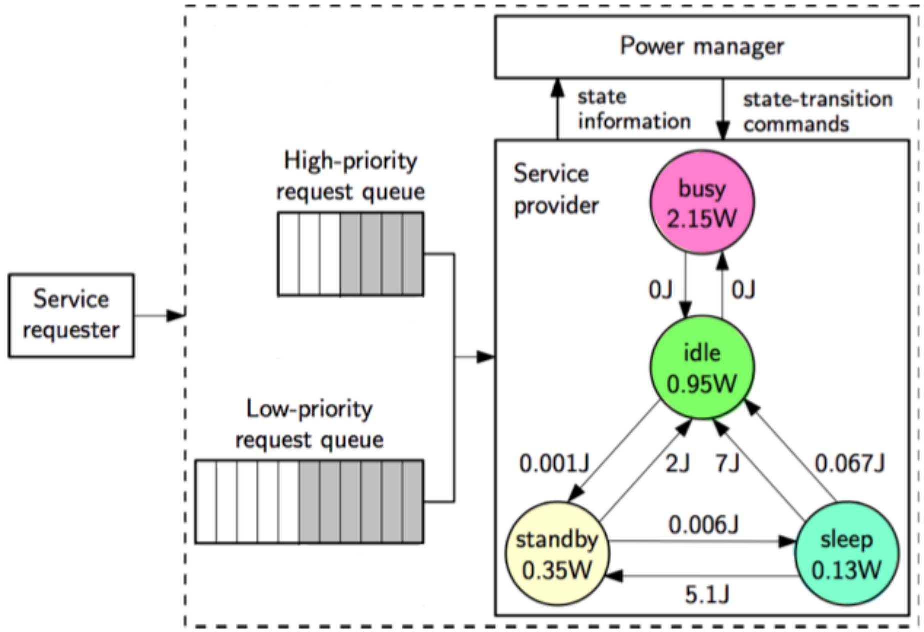
[CAV'21]

```
> python3 paynt/paynt.py --project sketch_path \
--properties sketch.properties
```

```
method: hybrid, synthesis time: 200.0 s
number of holes: 8, family size: 29160
super MDP size: 1502, average MDP size: 903, MDP
checks: 238, iterations: 125
average DTMC size: 234, DTMC checks: 26574,
iterations: 13287 optimal: 9100.064246

hole assignment:
P1=1,P2=2,P3=2,P4=2,T1=0.1,T2=0.4,T3=0.7,QMAX=5
```

Sketching Probabilistic Programs

[Benini *et al*, IEEE CAD 1999]

20 Parameters, 1 m  **tates quotient**

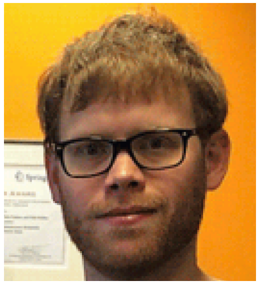
Challenge

- Synthesise guards and updates in DPM control program with 16 holes
- Specification = conjunction of expected #lost reqs and energy consumption

Results (16 parameters)

- Family size = 43,000,000 control programs of average size of 3,600 states
- Our approach: 9 hours; baseline: > 1 month

A Big Thanks to Our Parametric Co-Authors!



Tim Quatmann
(RWTH, D)

Matthias Volk
(Eindhoven, NL)

Nils Jansen
(Bochum, D)

Milan Ceska
(Brno, Cz)

Ufuk Topcu
(UT Austin US)

Guillermo Perez
(Antwerp, B)

Tobias Winkler
(RWTH, D)

Murat Cubuktepe
(UT Austin US)

Roman Andriusshenko, Erika Abraham, Harold Bruintjes, Florian Corzilius, Christel Baier, Linus Heck, Christian Hensel, Lisa Hutschenreiter, Joachim Klein, Filip Macak, Bahare Salmani, Jip Spel, Ralf Wimmer, Leonore Winterer, Bernd Becker

Want to know more?

Parameter Synthesis in Markov Models: A Gentle Survey*

Nils Jansen[1], Sebastian Junges[1] and Joost-Pieter Katoen[2]

[1] Radboud University, Nijmegen, The Netherlands
[2] RWTH Aachen University, Aachen, Germany

Abstract. This paper surveys the analysis of parametric Markov models whose transitions are labelled with functions over a finite set of parameters. These models are symbolic representations of uncountable many concrete probabilistic models, each obtained by instantiating the parameters. We consider various analysis problems for a given logical specification φ: do all parameter instantiations within a given region of parameter values satisfy φ?, which instantiations satisfy φ and which ones do not?, and how can all such instantiations be characterised, either exactly or approximately? We address theoretical complexity results and describe the main ideas underlying state-of-the-art algorithms that established an impressive leap over the last decade enabling the fully automated analysis of models with millions of states and thousands of parameters.

PAYNT: A Tool for Inductive Synthesis of Probabilistic Programs

Roman Andriushchenko[1], Milan Češka[1(✉)], Sebastian Junges[2], Joost-Pieter Katoen[3], and Šimon Stupinský[1]

[1] Brno University of Technology, Brno, Czech Republic
ceskam@fit.vutbr.cz
[2] University of California, Berkeley, USA
[3] RWTH Aachen University, Aachen, Germany

functional
Artifact
CAV
AEC
Evaluated
available
reusable

Abstract. This paper presents PAYNT, a tool to automatically synthesise probabilistic programs. PAYNT enables the synthesis of finite-state probabilistic programs from a program sketch representing a finite family of program candidates. A tight interaction between inductive oracle-guided methods with state-of-the-art probabilistic model checking is at the heart of PAYNT. These oracle-guided methods effectively reason about all possible candidates and synthesise programs that meet a given specification formulated as a conjunction of temporal logic constraints and possibly including an optimising objective. We demonstrate the performance and usefulness of PAYNT using several case studies from different application domains; e.g., we find the optimal randomized protocol for network stabilisation among 3M potential programs within minutes, whereas alternative approaches would need days to do so.

or get in touch!

Combining Model Checking with Deep RL

Sebastian Junges, Joost-Pieter Katoen

appliedAI institute for europe

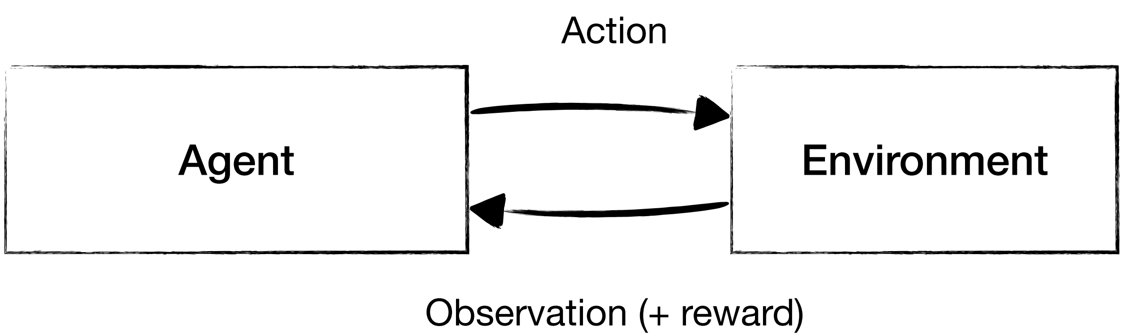
Radboud Universiteit

Software Modeling and Verification Chair

RWTH AACHEN UNIVERSITY

Shielding Reinforcement Learning

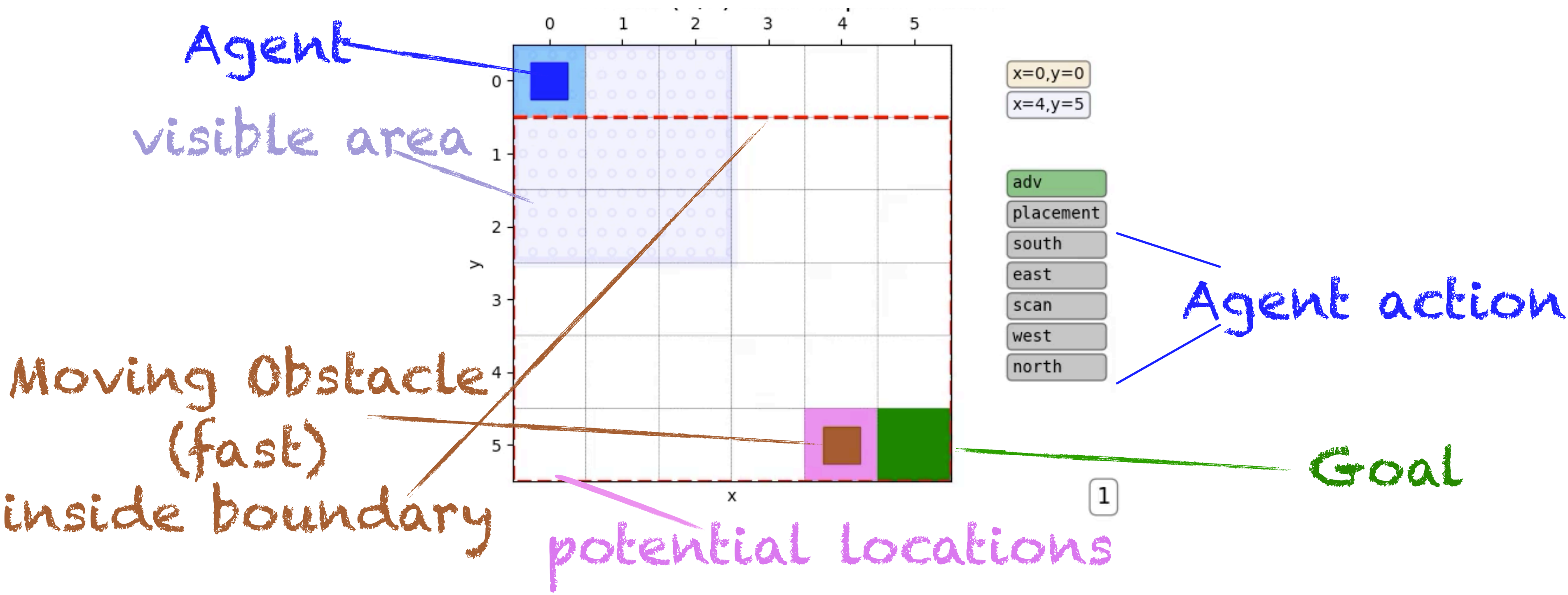
[Koenighofer et al, 2018]



- Agent decides based on history of actions, observations and rewards
- Exploration: Must take actions to understand the environment
- Safety concerns: No reason why a dangerous action would not be taken
- Liveness: Often safe = not do anything. We want to have progress towards a goal

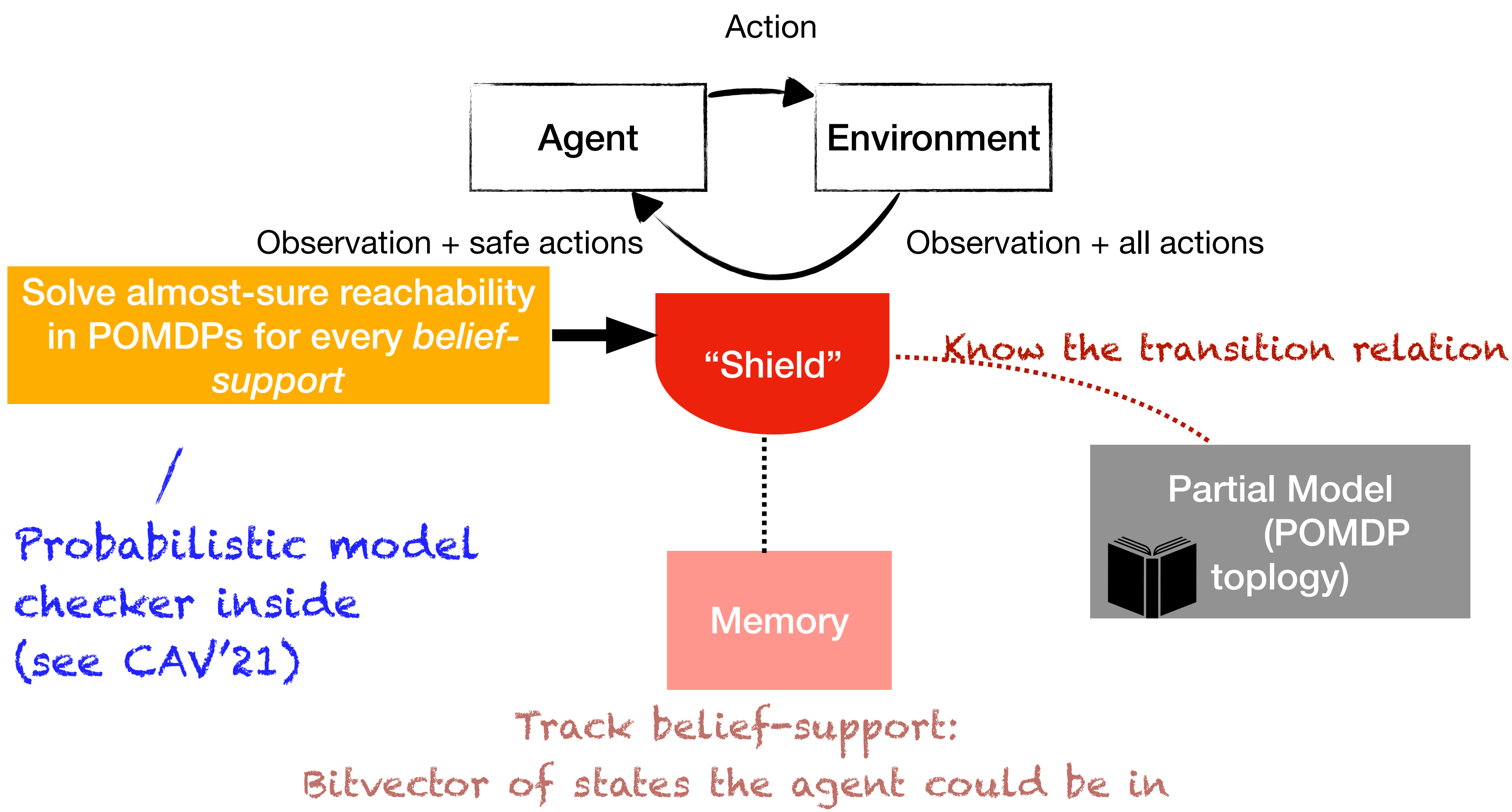
Shielding Exemplified

Illustration for the Avoid task



Shielding Reinforcement Learning

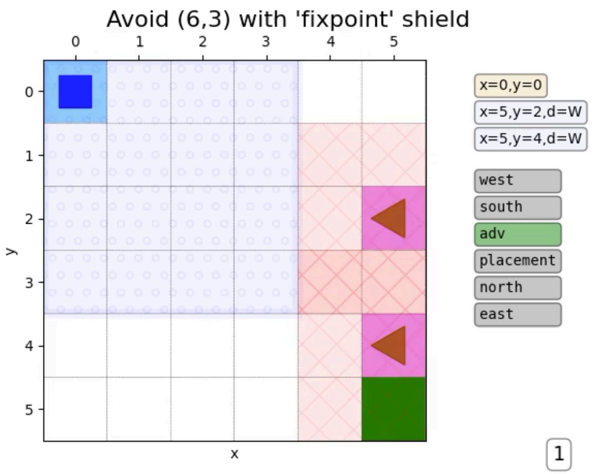
[AAAI’23]

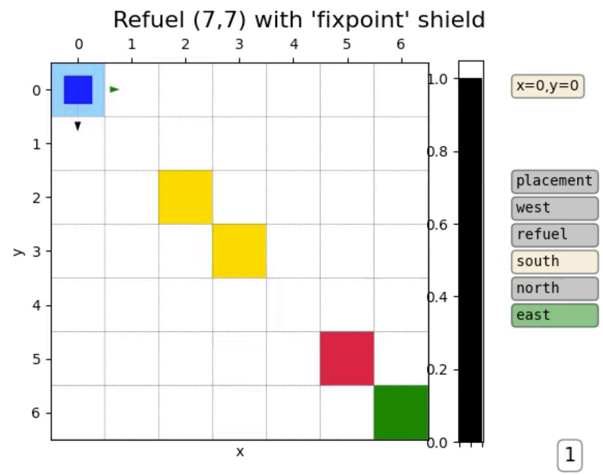


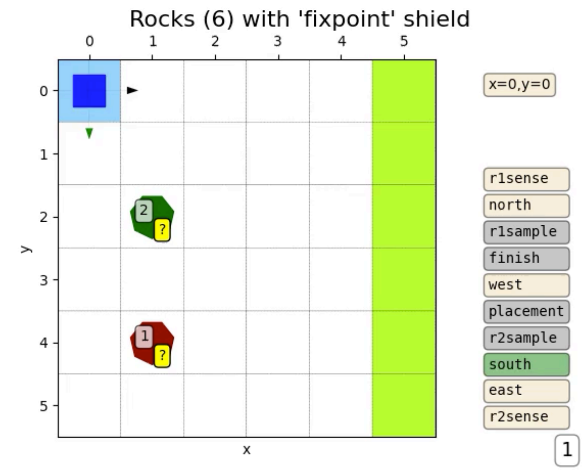
Sebastian Junges & Joost-Pieter Katoen
Probabilistic Verification Training
AI Transferlab 2023

Empirical evaluation

More examples



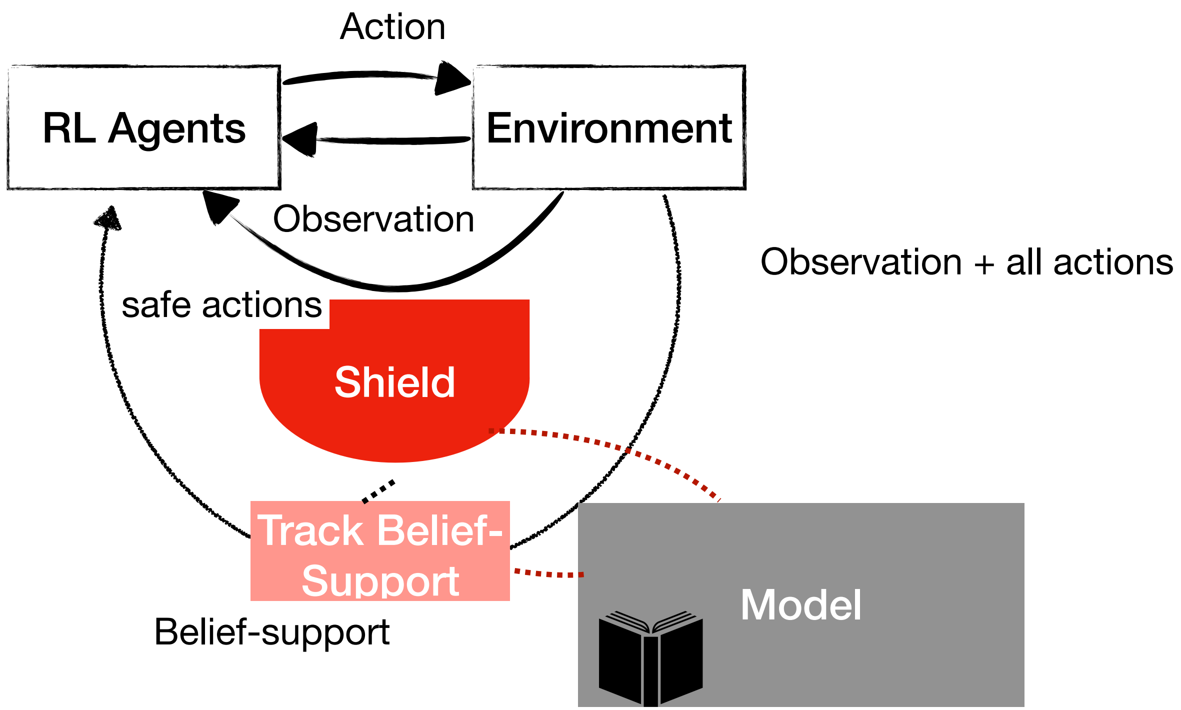




Sebastian Junges & Joost-Pieter Katoen
Probabilistic Verification Training
AI Transferlab 2023

What about combining with RL

tf-agents
(default parameters)
- Reinforce
- DDQN
- DRQN
- DQN
- SAC+lstm
- PPO

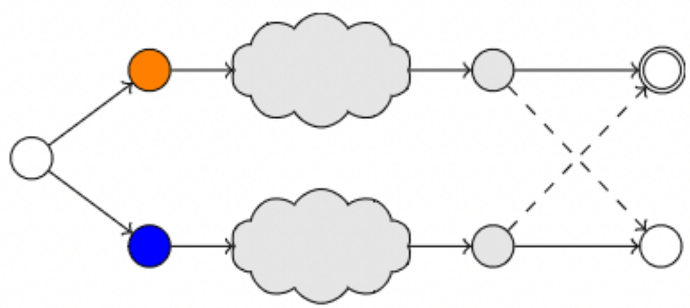


appliedAI institute for europe Radboud University RWTH AACHEN UNIVERSITY

Estimators Accelerate Learning

[AAAI'23]

To reach the target,
agent must observe and remember
whether light initially is orange or blue



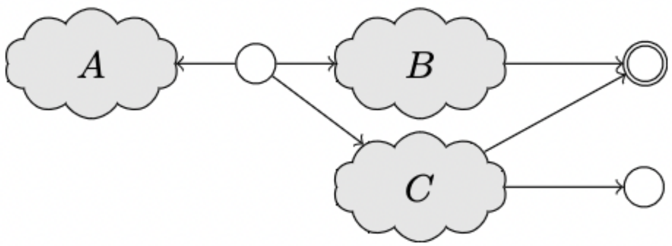
Observing the state estimation makes this trivial

Sebastian Junges & Joost-Pieter Katoen
Probabilistic Verification Training
AI Transferlab 2023

Shields Accelerate Learning

[AAAI'23]

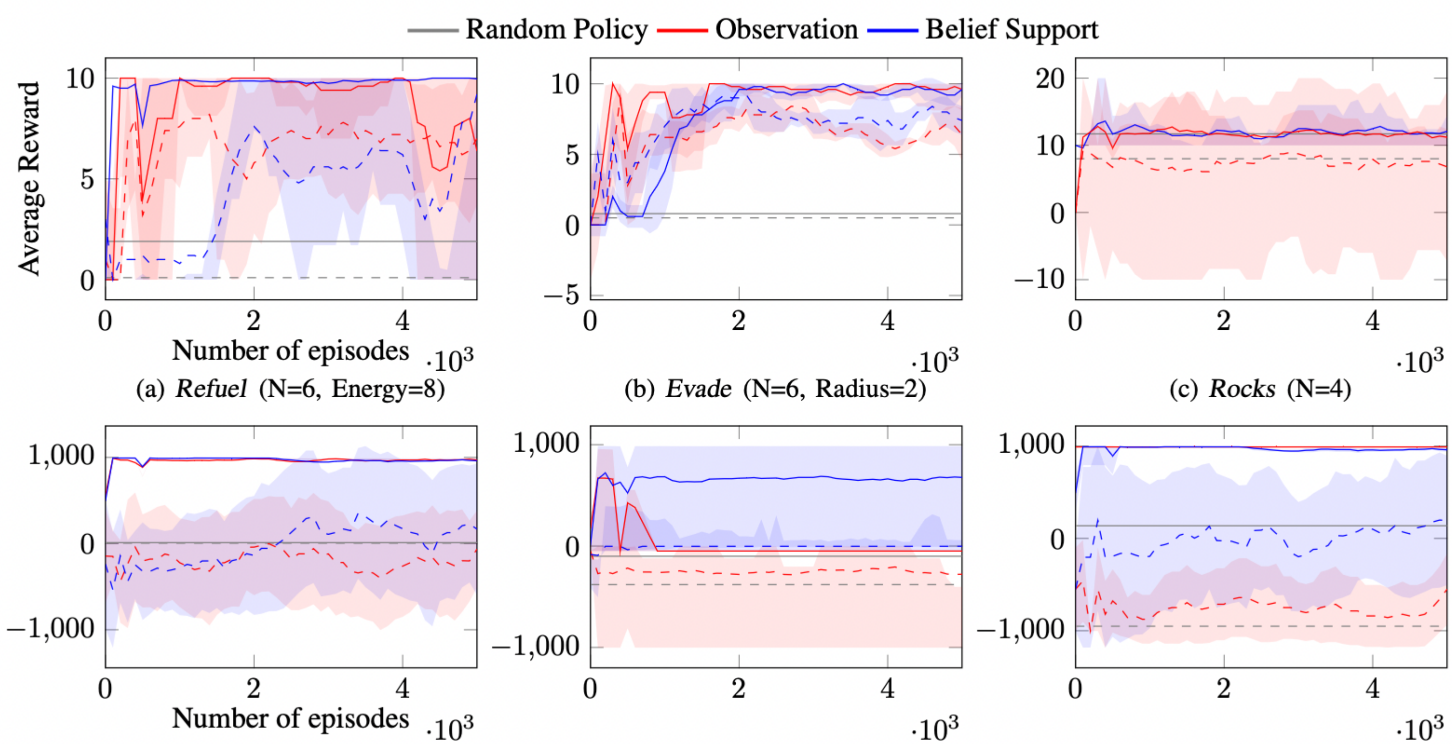
To always reach the target, agent must not visit states in C



The shield ensures that C is not visited, which accelerates the learning within B

Shields accelerate learning in sparse environments

[AAAI'23]



(a) *Refuel* (N=6, Energy=8)

(b) *Evade* (N=6, Radius=2)

(c) *Rocks* (N=4)

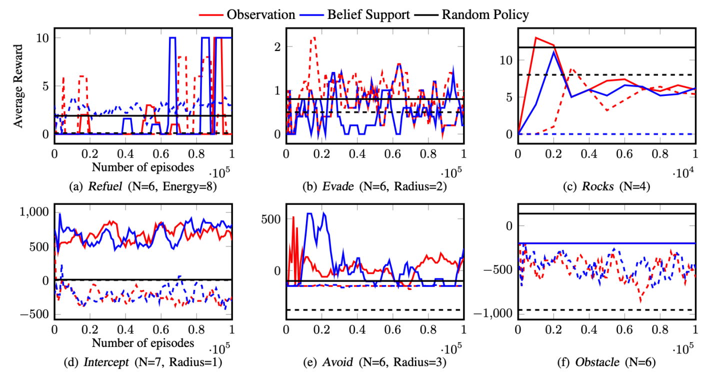
(d) *Intercept* (N=7, Radius=1)

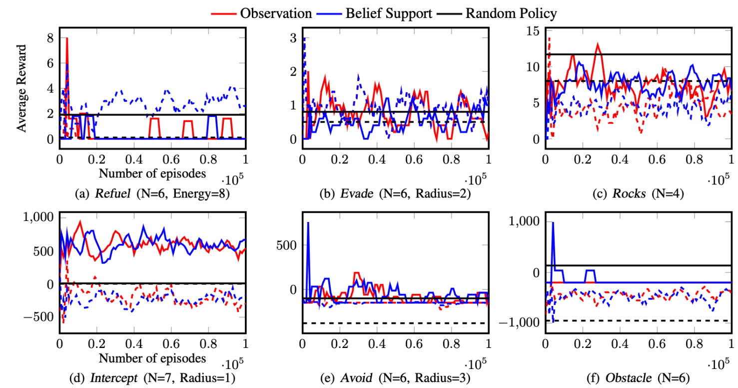
(e) *Avoid* (N=6, Radius=3)

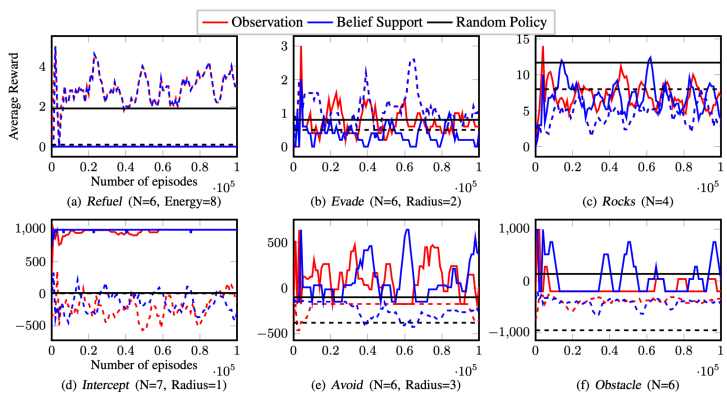
(f) *Obstacle* (N=6)

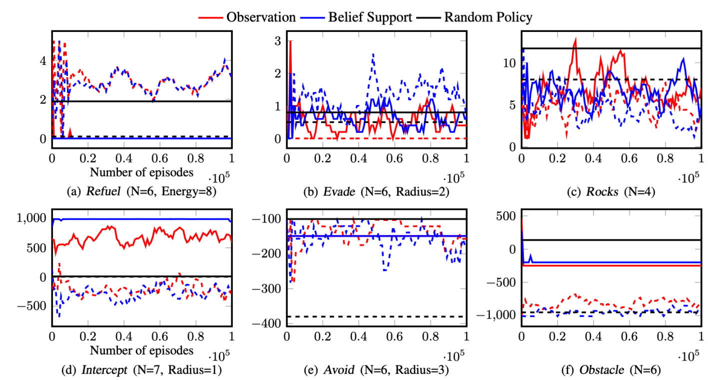
Sebastian Junges & Joost-Pieter Katoen
Probabilistic Verification Training
AI Transferlab 2023

More data…


Observation
Belief Support
Random Policy
Average Reward
Number of episodes
(a) *Refuel* (N=6, Energy=8)
(b) *Evade* (N=6, Radius=2)
(c) *Rocks* (N=4)
(d) *Intercept* (N=7, Radius=1)
(e) *Avoid* (N=6, Radius=3)
(f) *Obstacle* (N=6)

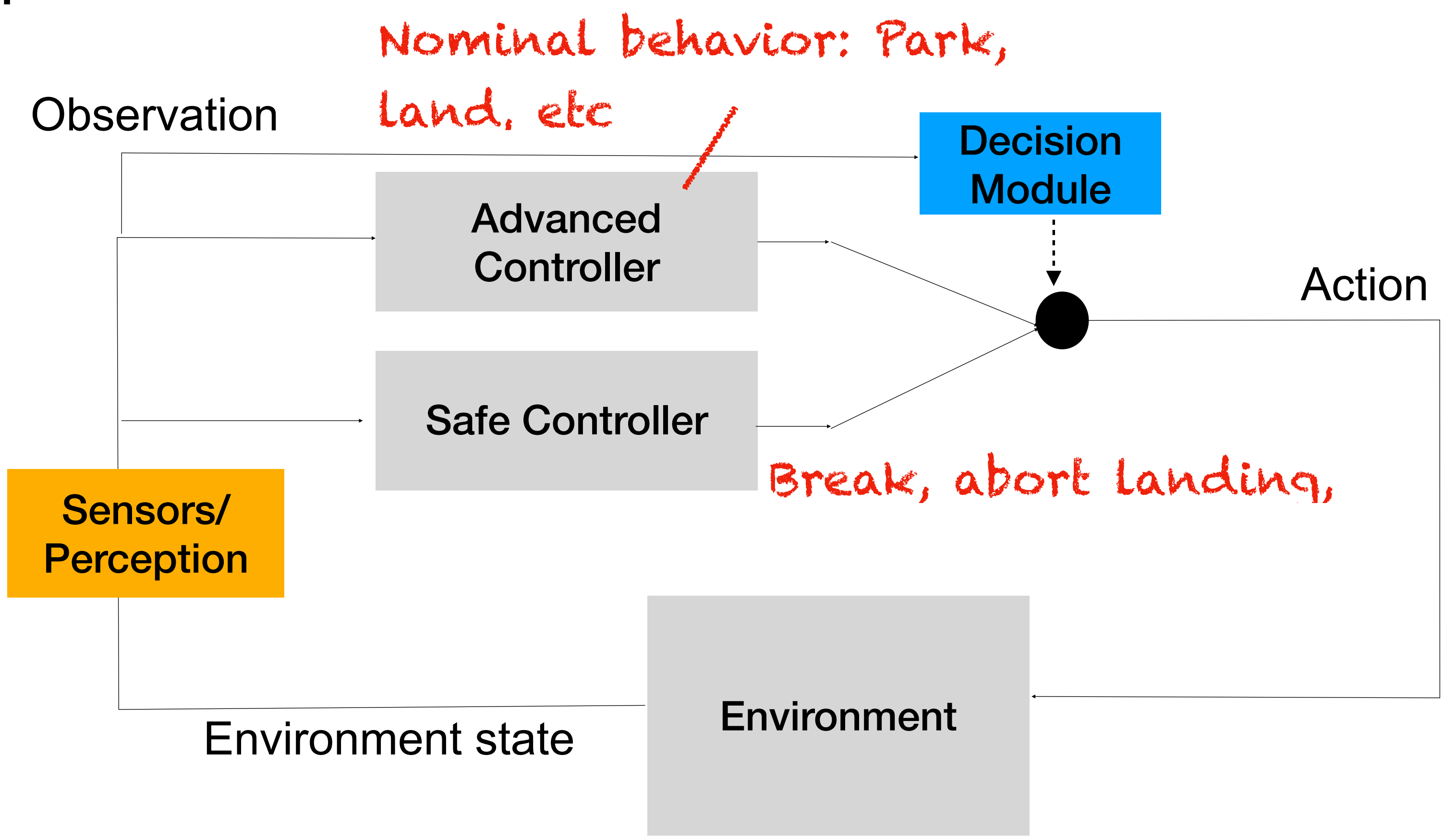

Observation
Belief Support
Random Policy
Average Reward
Number of episodes
(a) *Refuel* (N=6, Energy=8)
(b) *Evade* (N=6, Radius=2)
(c) *Rocks* (N=4)
(d) *Intercept* (N=7, Radius=1)
(e) *Avoid* (N=6, Radius=3)
(f) *Obstacle* (N=6)


Observation
Belief Support
Random Policy
Average Reward
Number of episodes
(a) *Refuel* (N=6, Energy=8)
(b) *Evade* (N=6, Radius=2)
(c) *Rocks* (N=4)
(d) *Intercept* (N=7, Radius=1)
(e) *Avoid* (N=6, Radius=3)
(f) *Obstacle* (N=6)


Observation
Belief Support
Random Policy
Number of episodes
(a) *Refuel* (N=6, Energy=8)
(b) *Evade* (N=6, Radius=2)
(c) *Rocks* (N=4)
(d) *Intercept* (N=7, Radius=1)
(e) *Avoid* (N=6, Radius=3)
(f) *Obstacle* (N=6)

Runtime Monitoring

Simplex architecture

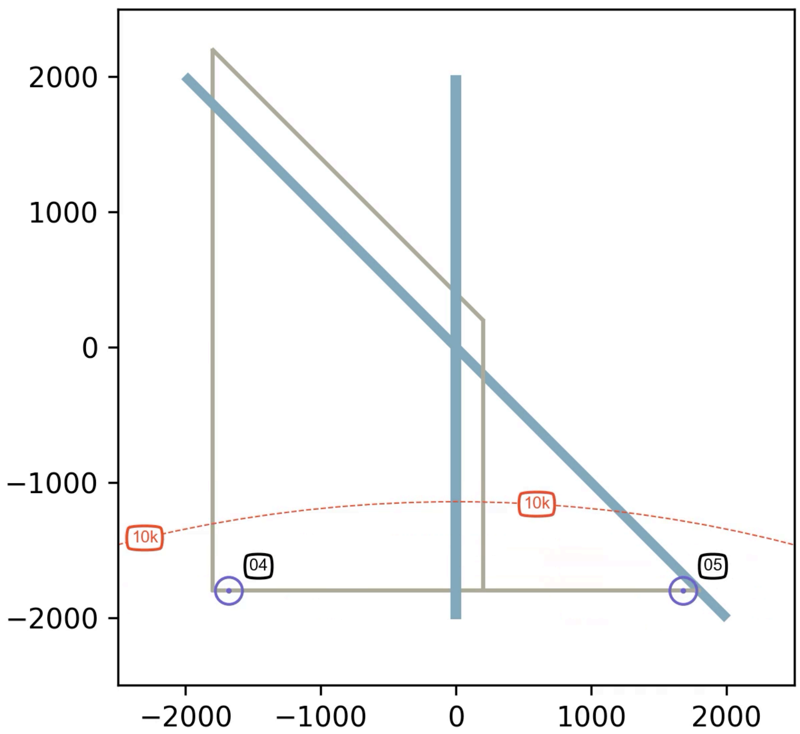


Sebastian Junges & Joost-Pieter Katoen

Outlook: monitoring your autonomous system

[CAV'21]

Berkeley AIRfield Collision Avoidance Monitor



- Aircraft (10k away) approaches airfield
- Ground vehicles move around based on stochastic model
- Decide:
  Continue landing or go-around?

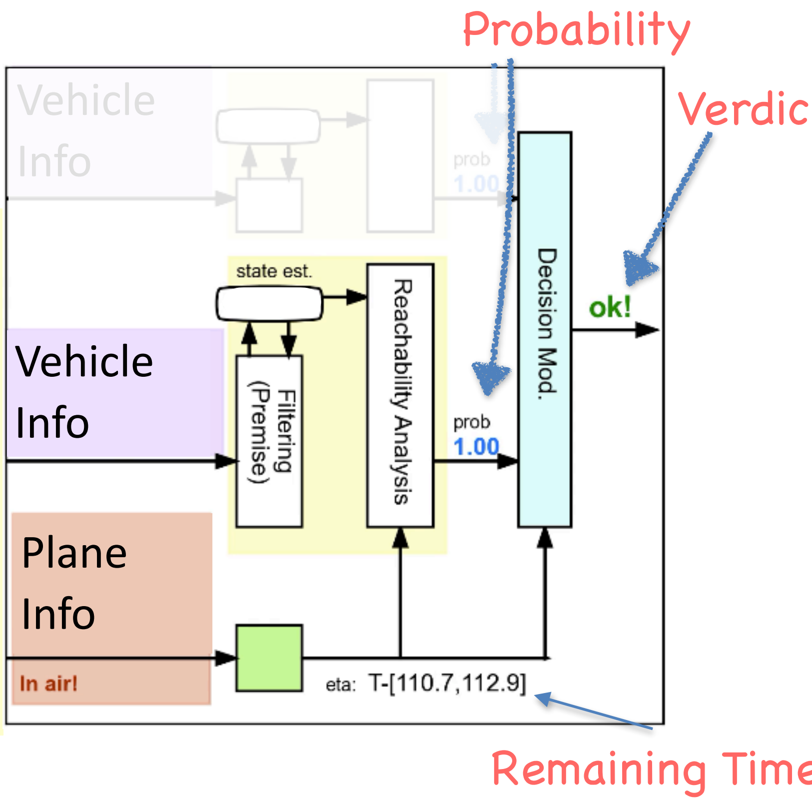
Goal: Decision Making on Top of Robust Reachability Analysis

[CAV'21]

Challenge: Partial observability, non-deterministic and probabilistic uncertainty

- Robust decision making on top of (worst-case) probability

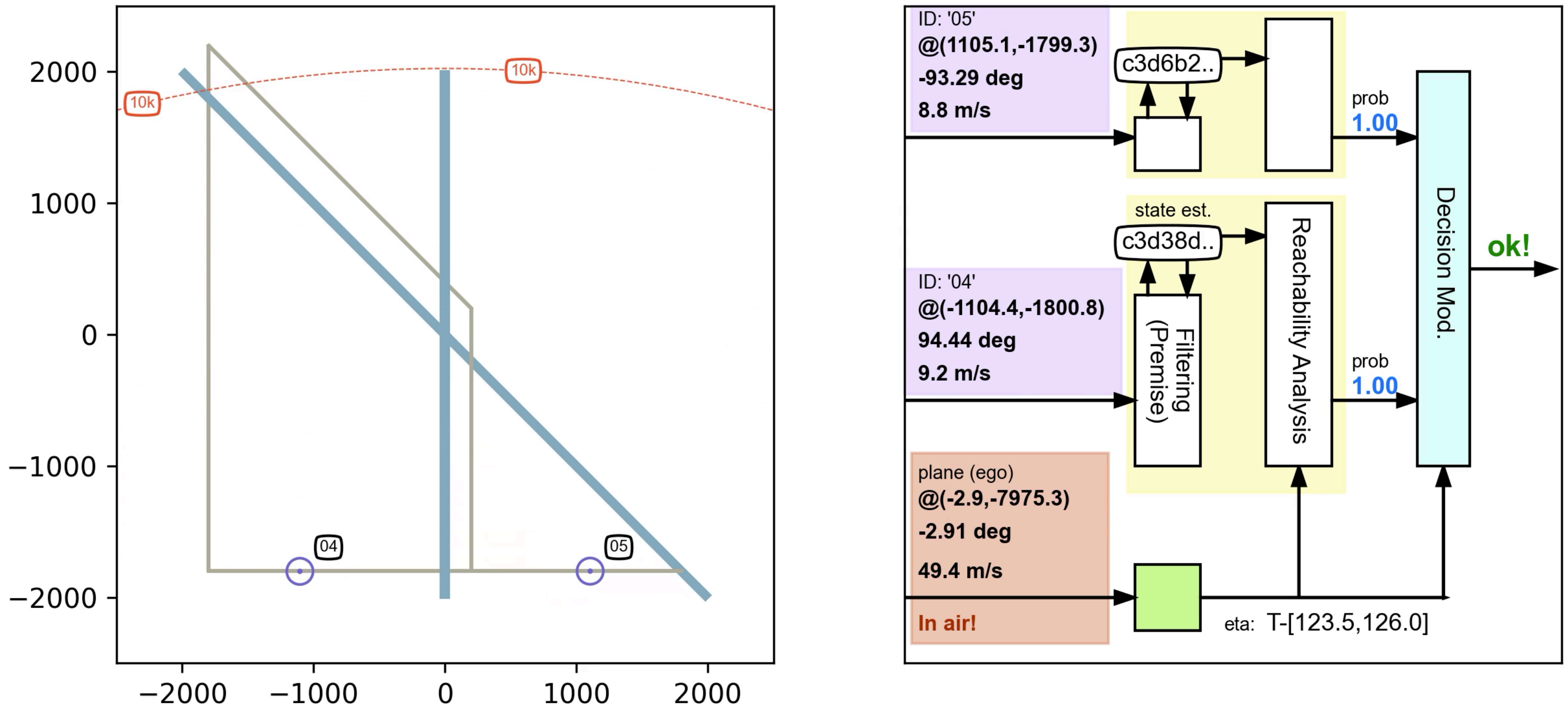
- Model-based analysis of probability:
  - Novel filtering approach
  - Analysis connects to model checkers
  - Monitor knows its operational domain



Demo

[CAV'21]

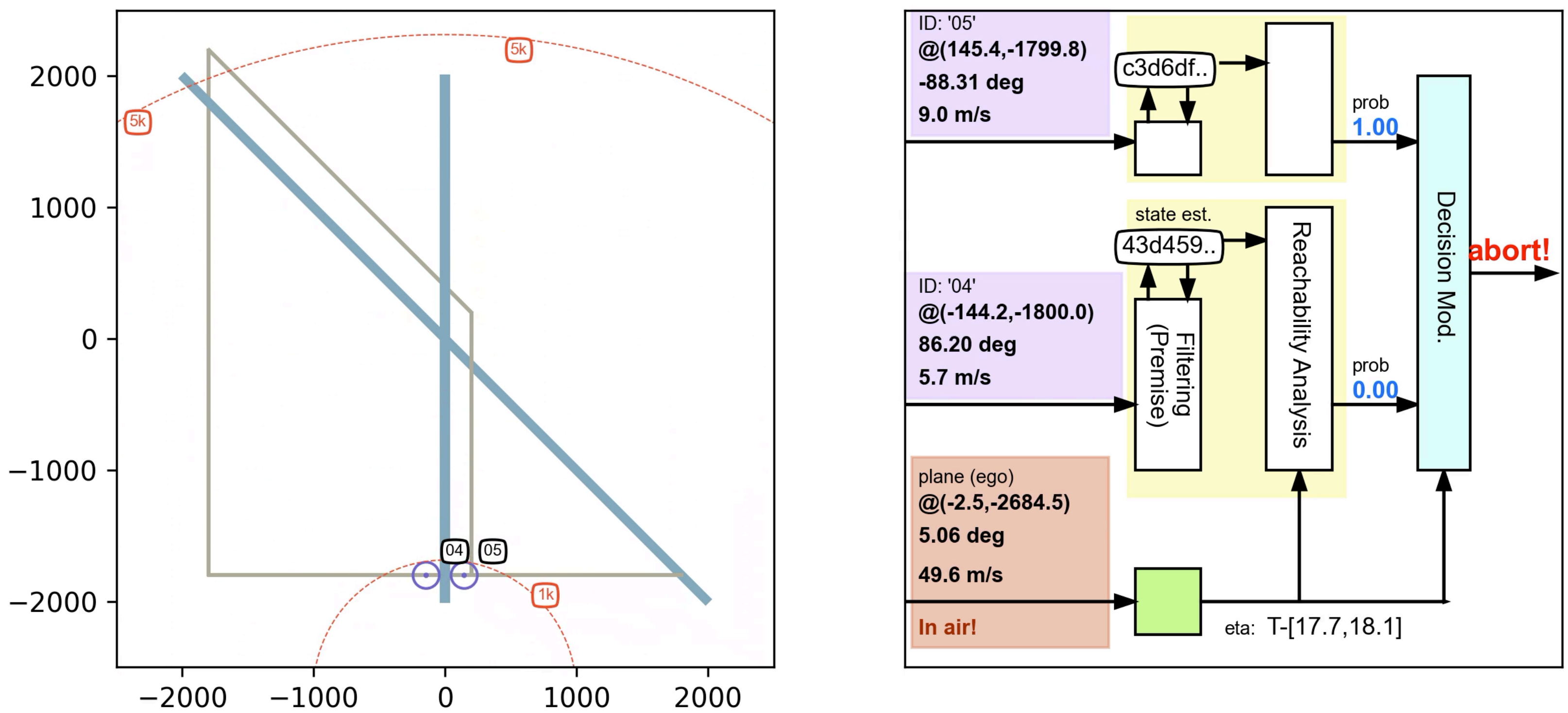
Berkeley AIRfield Collision Avoidance Monitor



Demo

[CAV'21]

Berkeley AIRfield Collision Avoidance Monitor



Literature

Mohammed Alshiekh, Roderick Bloem, Rüdiger Ehlers, Bettina Könighofer, Scott Niekum, Ufuk Topcu:
Safe Reinforcement Learning via Shielding. AAAI 2018: 2669-2678

Steven Carr, Nils Jansen, Sebastian Junges, Ufuk Topcu:
Safe Reinforcement Learning via Shielding under Partial Observability. AAAI 2023: 14748-14756

Sebastian Junges, Nils Jansen, Sanjit A. Seshia:
Enforcing Almost-Sure Reachability in POMDPs. CAV (2) 2021: 602-625

Sebastian Junges, Hazem Torfah, Sanjit A. Seshia:
Runtime Monitors for Markov Decision Processes. CAV (2) 2021: 553-576

Hazem Torfah, Sebastian Junges, Daniel J. Fremont, Sanjit A. Seshia:
Formal Analysis of AI-Based Autonomy: From Modeling to Runtime Assurance. RV 2021: 311-330

Epilogue

Sebastian Junges, Joost-Pieter Katoen

Wrap-Up

Recent Trends in Probabilistic Model Checking and Verification

- **Do not explicitly represent the state space, use the structure of the program**
- Verification of partially observable MDPs
- Verification of Multi-player MDPs / Stochastic Games
- Synthesis of robust policies
- Synthesis of small policy representations

Which features would help you? What case studies should we consider?

katoen@cs.rwth-aachen.de sjunges@cs.ru.nl