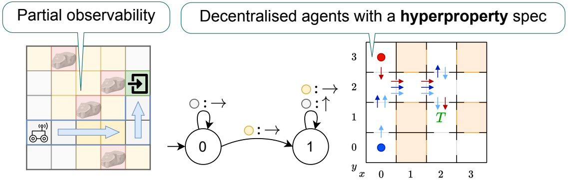
Controller Synthesis under Model Uncertainty and Structural Constraints

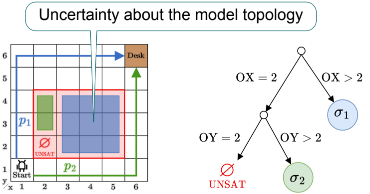
Milan Češka

BRNO UNIVERSITY OF TECHNOLOGY
FACULTY OF INFORMATION TECHNOLOGY

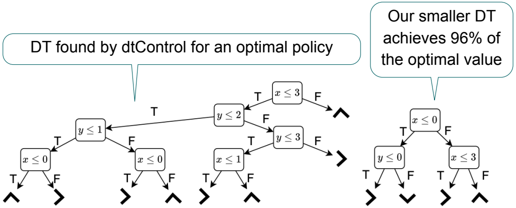
Applications & Recent results



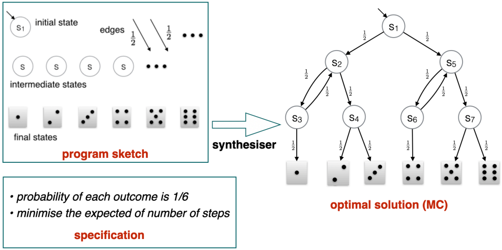
Synthesis of finite state controllers for POMDPs (UAI'22, CAV'23) and for decentralised planning (AAMAS'25)



Synthesis of policy trees for multiple-environment MDPs (ATVA'24)

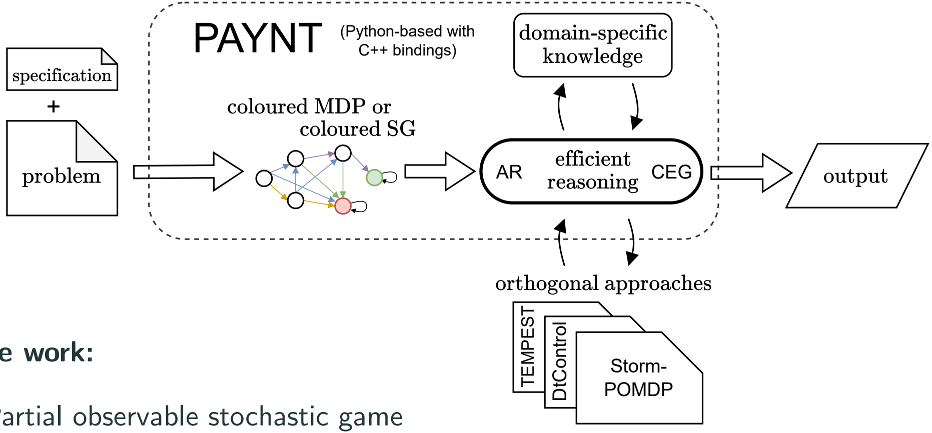


Synthesis of small almost optimal decision trees for MDPs (CAV'25)



Synthesis of finite-state probabilisitic programs from sketches (TACAS'21, JAIR'25)

Available via our synthesis framework PAYNT [CAV'21, JAIR'25]

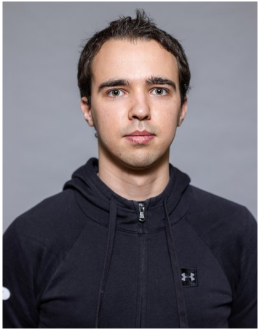


Future work:

- Partial observable stochastic game
- Roboust MDPs and POMDPs
- Robust monitoring/shielding for safe RL

Team

Roman Andriushchenko



Synthesis, Robustness

Filip Macák

POMDPs, POSGs

David Hudák

Safe RL

Cooperation with Sebastian Junges, Joost-Pieter Katoen, and Nils Jansen

Policies Grow on Trees: Model Checking Families of MDPs

Roman Andriushchenko Milan Češka Sebastian Junges **Filip Macák**

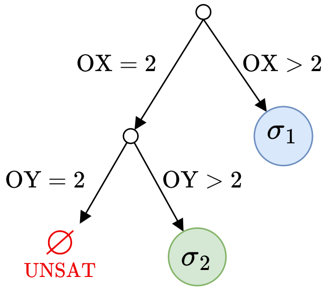
Distinguished Paper at ATVA'24

Motivation

Previous work: exploring families of discrete-time Markov chains (DTMCs)

Increased interest in robustness in nondeterministic models

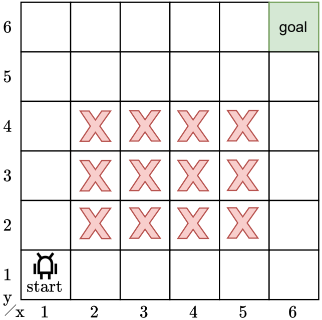
- obtain strategy which works well in multiple environments
- what if we don't know the exact model?



Family of MDPs

Family $\{M_i\}_{i \in \mathcal{I}}$ of MDPs = MDP with parameters

- parameters affect MDP topology
- $i \in \mathcal{I}$ is a parameter assignment, $|\mathcal{I}| < \infty$
- choice of parameter assignment $i \in \mathcal{I}$ represents uncontrollable nondeterminism (adversary, environment)
- choice of action $\alpha \in Act$ represents controllable nondeterminism



- parameters: $OX = \{2, 3, 4, 5\}$ and $OY = \{2, 3, 4\}$

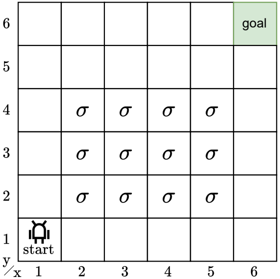
Robustness problem

input: family $\{M_i\}_{i \in \mathcal{I}}$ of MDPs

input: PCTL reachability property $P(\mathrm{F}\ T) \bowtie \lambda$

output: robust controller σ s.t. $\forall i \in \mathcal{I}\colon P(M_i^\sigma \models \mathrm{F}\ T) \bowtie \lambda$

- requires non-memoryless controllers
- related to solving POMDPs

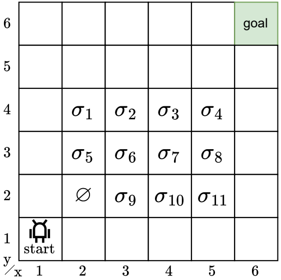


Problem statement

input: family $\{M_i\}_{i \in \mathcal{I}}$ of MDPs

input: PCTL reachability property $P(\mathrm{F}\ T) \bowtie \lambda$

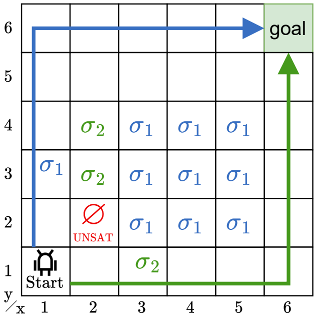
output: for each parameter assignment $i \in \mathcal{I}$ a controller σ_i s.t. $P\left(M_i^{\sigma_i} \models \mathrm{F}\ T\right) \bowtie \lambda$ (if such σ_i exists)

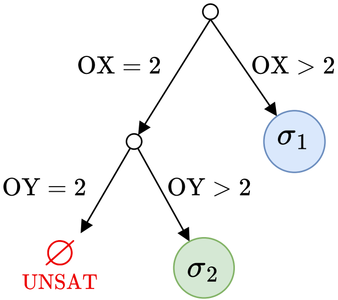


Problem statement

Additional requirement: produce a decision tree of controllers

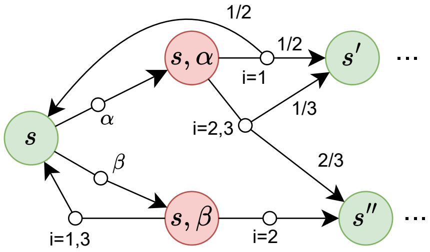
- nodes of the tree reason about a single parameter
- leaves of the tree (describing sub-families) contain controllers (or $\varnothing$)
- space-efficient, fast lookup, more understandable for engineers





Stochastic game abstraction

- Player 1 picks an action, Player 2 picks a parameter assignment



the above is an over-approximation since Player 2 is too powerful:

- Player 2 can pick parameter assignments inconsistently
  - consistent abstraction would mimic the all-in-one abstraction
- Player 2 acts second
  - this order avoids the abstraction blow-up

Robust policy heuristic

- assume a family $\mathcal{M}$ of MDPs and a specification $P(\mathrm{F}\ T) \geq 0.9$
- construct game abstraction $\mathcal{G}(\mathcal{M})$
- the following is a sufficient (but not necessary) condition for σ_1 to be a robust controller for $\mathcal{M}$:

$$\max_{\sigma_1} \min_{\sigma_2} P(\mathcal{G}(\mathcal{M})^{\sigma_1 \sigma_2} \models \mathrm{F}\ T) \geq 0.9$$

- if the above condition does *not* hold and σ_2 is consistent in its parameter assignment, then this assignment is unsatisfiable

Proving unsatisfiability heuristic

- assume a family $\mathcal{M}$ of MDPs and a specification $P(\mathrm{F}\ T) \geq 0.9$
- the following is a sufficient (but not necessary) condition for no MDP in $\mathcal{M}$ being satisfiable:

$$\max_{\sigma_1} \max_{\sigma_2} P(\mathcal{G}(\mathcal{M})^{\sigma_1 \sigma_2} \models \mathrm{F}\ T) < 0.9$$

- such "game" abstraction is simply an MDP
- if the above condition does *not* hold and σ_2 is consistent in its parameter assignment, then this assignment is satisfiable

Abstraction refinement step: if neither of the tests was successful, we split family $\mathcal{M}$ into smaller subfamilies based on the controller (σ_1, σ_2) for the game abstraction $\mathcal{G}(\mathcal{M})$

- if σ_2 is not consistent i.e. in parameter X, we split wrt. X to disallow such an inconsistency in the subfamilies
- if σ_2 is consistent, representing some satisfiable assignment i, we try to separate i (and other assignments in which σ_2 is consistent) into a smaller subfamily

Proposed Algorithm

Algorithm 1 Policy tree synthesis

Input: family $\mathcal{M} = \{M_i\}_{i \in \mathcal{I}}$ of MDPs, PCTL property φ

Output: policy tree for $\mathcal{M}$ wrt. φ

function $\textsc{buildTree}(\mathcal{M}, \varphi)$

    $\sigma \leftarrow$ try to find a robust controller for $\mathcal{M}$ wrt. φ

    if succeeded **then**

        return $\textsc{LeafNode}(\mathcal{M}, \sigma)$

    try to prove that no $M_i \in \mathcal{M}$ can satisfy φ

    if succeeded **then**

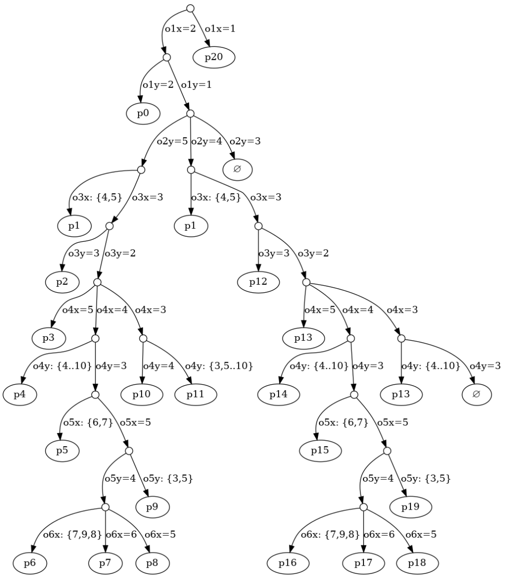
        return $\textsc{LeafNode}(\mathcal{M}, \emptyset)$

    $\mathcal{M}', \mathcal{M}'' \leftarrow$ split($\mathcal{M}$)

    return $\textsc{InnerNode}(\mathcal{M}, \textsc{buildTree}(\mathcal{M}', \varphi), \textsc{buildTree}(\mathcal{M}'', \varphi))$

Main idea: given a family of MDPs, try to find a robust controller or try to prove that no satisfying MDP exists, split the family if a conclusive result was not obtained

Decision tree example



Experimental results

model	model info			our approach		speedup wrt.	
	$\lvert S_{\mathcal{M}}\rvert$	$\lvert\mathcal{M}\rvert$	SAT %	P/SAT %	time	1-by-1	all-in-1
dodge-2	2e5	3e4	100	0.1	**122**	8	1.1
dodge-3	2e5	9e7	100	<0.01	**1445**	†1764	MO
dpm-10-b	9e3	1e5	22	0.02	**74**	21	TO
obs-8-6	5e2	5e4	90	0.6	**6**	4	1.5
obs-10-6	8e2	3e6	98	<0.01	**5**	412	MO
obs-10-9	1e3	4e8	100	<0.01	**259**	†1661	MO
rov-1000	2e4	4e6	99	0.03	**1402**	†65	TO
uav-work	9e3	2e6	99	<0.01	**113**	55	TO
virus	2e3	7e4	83	0.9	50	**0.8**	TO
rocks-6-4	3e3	7e3	100	34	102	0.2	**0.1**

Conclusion

Main contributions:

1. We contribute a scalable approach to policy synthesis for sets of MDPs
2. The key technique is a game-based abstraction with abstraction refinement
3. The resulting algorithm finds policies for millions of MDPs and provides a compact representation of them

Future work:

- Investigate the robustness problem further
- Incorporate the compact representation of policies (e.g. as decision trees)
- Extend the framework to families of POMDPs

Thank you for attention!