Small Decision Trees for MDPs with Deductive Synthesis

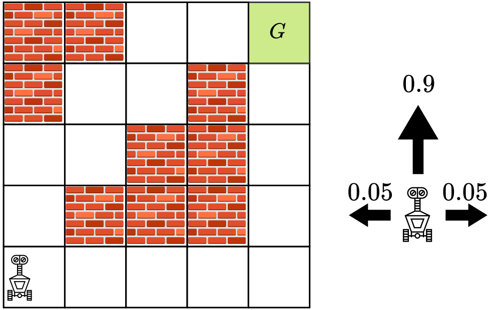
Roman Andriuschenko[1] Milan Češka[1] Sebastian Junges[2] **Filip Macák[1]**

[1]Brno University of Technology, Brno, Czech Republic
[2]Radboud University, Nijmegen, the Netherlands

Motivation I.

Markov Decision Process (MDP)

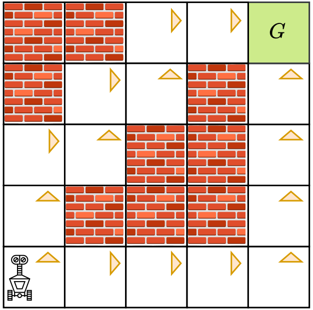
- $M = (S, Act, s_0, P)$
- important model for sequential decision making



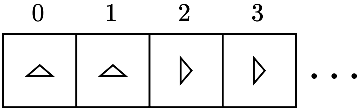
Motivation I.

Markov Decision Process (MDP)

- $M = (S, Act, s_0, P)$
- important model for sequential decision making



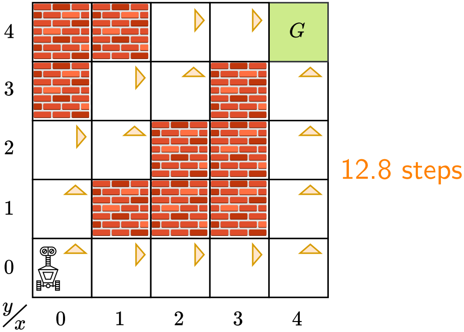
12.8 steps



Motivation I.

Markov Decision Process (MDP)

- $M = (S, Act, s_0, P)$
- important model for sequential decision making

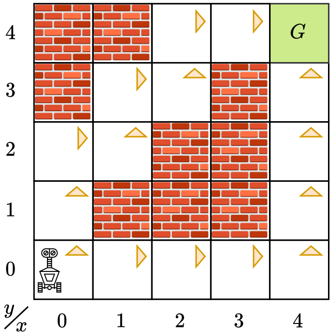


The states usually correspond to some assignment of variables $\mathcal{V}$

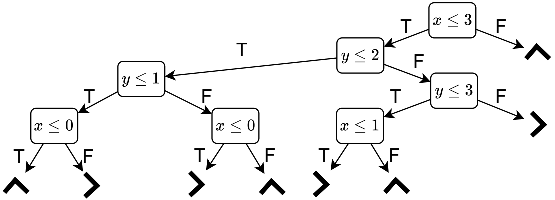
Motivation I.

Markov Decision Process (MDP)

- $M = (S, Act, s_0, P)$
- important model for sequential decision making



12.8 steps

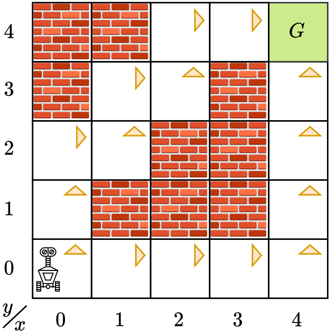


We want a concise policy representation e.g. **Decision Tree (DT)**

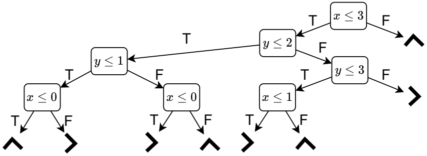
Motivation II.

We want a concise policy representation!

- small decision tree might not exist for the optimal policy



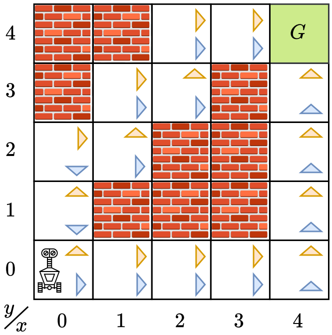
12.8 steps



Motivation II.

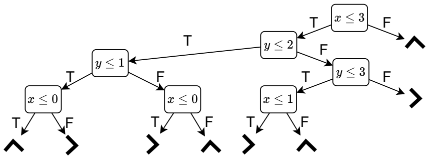
We want a concise policy representation!

- small decision tree might not exist for the optimal policy



12.8 steps

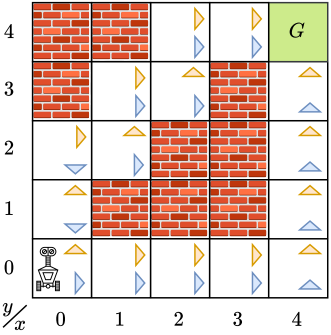
13.3 steps



Motivation II.

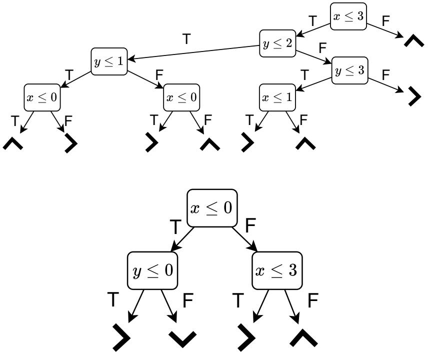
We want a concise policy representation!

- small decision tree might not exist for the optimal policy



12.8 steps

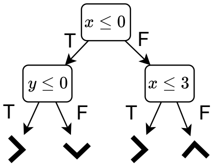
13.3 steps



Problem Formulation

DT $\mathcal{T} = (T, \gamma, \delta)$

- $T = (n_0, N, L, r, l)$ is a binary tree
- $\gamma : N \to \Psi_{\mathcal{V}}$ assigns state predicate to inner nodes
- $\delta : L \to Act$ assigns action to leaf nodes



Consider MDP M with some reachability property

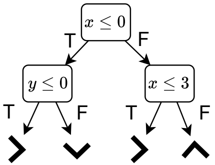
Bounded-depth synthesis problem

given an MDP M and a bound k, find a DT $\mathcal{T}$ of depth up to k with maximum value

Problem Formulation

DT $\mathcal{T} = (T, \gamma, \delta)$

- $T = (n_0, N, L, r, l)$ is a binary tree
- $\gamma : N \to \Psi_{\mathcal{V}}$ assigns state predicate to inner nodes
- $\delta : L \to Act$ assigns action to leaf nodes



Consider MDP M with some reachability property

Bounded-depth synthesis problem

given an MDP M and a bound k, find a DT $\mathcal{T}$ of depth up to k with maximum value

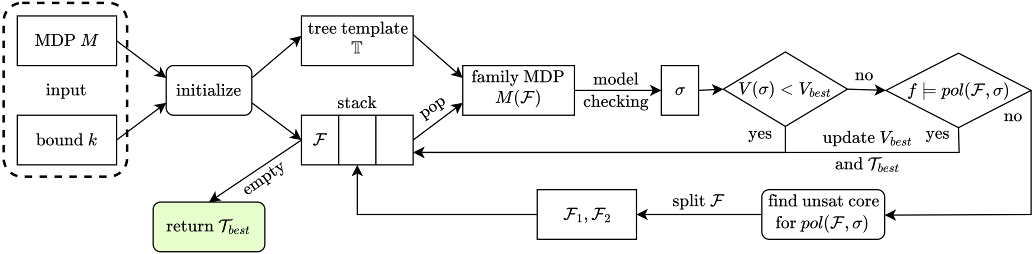
Policy mapping problem

- given a precomputed policy σ on MDP M, find DT $\mathcal{T}$ that implements σ
- σ typically obtained from a model checker

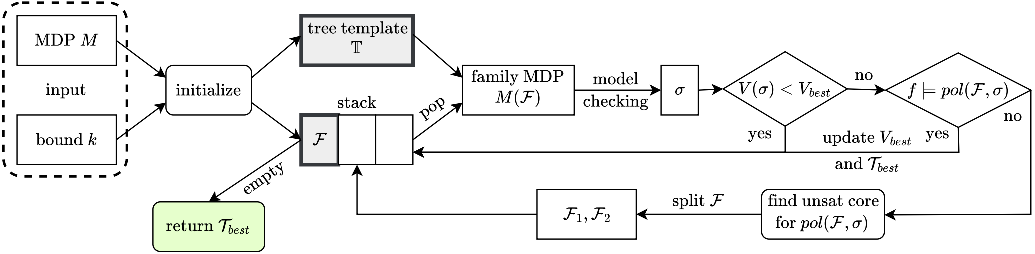
Contributions

1. We propose dtPAYNT, an abstraction-refinement loop with SMT-based subroutine that iteratively searches for DTs with maximal value among all DTs up to a given depth
2. We show that dtPAYNT outperforms previous state-of-the-art for the bounded-depth synthesis problem
3. In contrast to policy mapping approaches, dtPAYNT is able to effectively control the trade-off between size and value of the DT
4. dtPAYNT can be effectively used to reduce large DTs
5. We prove NP-hardness of the bounded-depth synthesis problem

Overview of Our Approach



Overview of Our Approach



Templates and Parametrization Set

Tree Template $\mathbb{T} = (T, \Gamma, \Delta)$

- T is a binary tree
- $\Gamma : N \to 2^{\Psi_{\mathcal{V}}}$ assigns set of state predicates to inner nodes
- $\Delta : L \to 2^{Act}$ assigns set of actions to leaf nodes

Set of variables $\chi^{\mathbb{T}} = \{\mathcal{D}_n, \mathcal{B}_n \mid n \in N\} \cup \{A_n \mid n \in L\}$

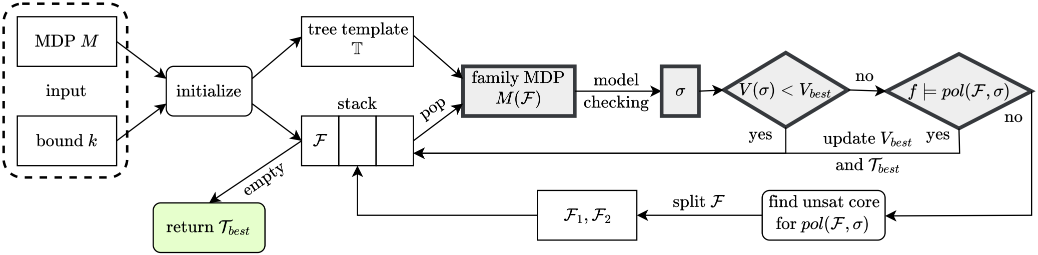
Parametrization set $\mathcal{F} : \chi^{\mathbb{T}} \to 2^{\mathbb{Z}}$

- $\mathcal{F}(\mathcal{D}_n) \subseteq \{1, 2, \ldots, |\mathcal{V}|\}$ for all $n \in N$
- $\mathcal{F}(\mathcal{B}_n) \subseteq \mathbb{Z}$ for all $n \in N$
- $\mathcal{F}(\mathcal{A}_n) \subseteq \{1, 2, \ldots, |Act|\}$ for all $n \in L$

Template $\mathbb{T}$ and parametrization set $\mathcal{F}$ represent a set of DTs $\mathbb{T}(\mathcal{F})$

A single parametrization $f \in \mathcal{F}$ corresponds to one DT $\mathbb{T}(f)$

Overview of Our Approach



Family MDP and Implementability

Family MDP $M(\mathcal{F})$ is a sub-MDP of M, where action α is enabled in state s only if there exists at least one DT in the set $\mathbb{T}(\mathcal{F})$ which induces a policy that picked α in s

We can model check $M(\mathcal{F})$ to obtain optimal policy σ and **if $V(\sigma)$ is worse than running optimum we discard the current parametrization set**

Given a policy σ, is there a DT in $\mathbb{T}(\mathcal{F})$ that implements σ?

To answer this question we use an SMT encoding over the theory of quantifier-free linear integer arithmetic (LIA)

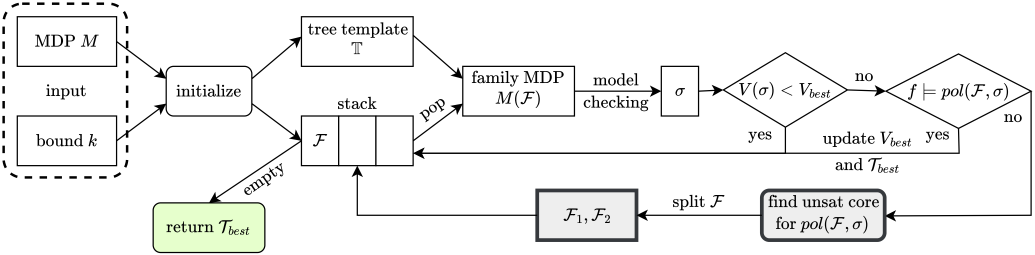
$$\mathsf{pol}(\mathcal{F},\sigma) \coloneqq \quad \mathsf{dom}(\mathcal{F}) \wedge \bigwedge_{s\in S,\sigma(s)\neq\perp} \mathsf{act}_{s,\sigma(s)}$$

$$\mathsf{act}_{s,\alpha} \coloneqq \quad \bigwedge_{n\in L} \mathsf{sel}_{s,n} \rightarrow \mathsf{act}_{s,\alpha,n}$$

$$\mathsf{dom}(\mathcal{F}) \coloneqq \bigwedge\nolimits_{x\in\mathcal{X}^{\mathbb{T}}} x \in \mathcal{F}(x)$$

$$\mathsf{sel}_{s,n} \coloneqq \quad \bigwedge\nolimits_{i=0}^{k-1} \bigwedge\nolimits_{j=1}^{|\mathcal{V}|} \quad \big(j = \mathcal{D}_{n_i}\big) \rightarrow \big(s(v_j) \bowtie_i^n \mathcal{B}_{n_i}\big)$$

Overview of Our Approach



Abstraction Refinement and Unsat Cores

If $pol(\mathcal{F}, \sigma)$ is unsat, we want to examine why

Unsat core $UC = (\chi^{UC} \subseteq \chi^{\mathbb{T}}, SP^{UC} \subseteq S \times L)$ where:

$$\mathrm{pol}^{\mathrm{UC}}(\mathcal{F}, \sigma) \coloneqq \bigwedge_{x \in \mathcal{X}^{\mathrm{UC}}} x \in \mathcal{F}(x) \wedge \bigwedge_{s,n \in SP^{\mathrm{UC}}} \mathrm{act}_{s,\sigma(s),n}$$

is unsatisfiable

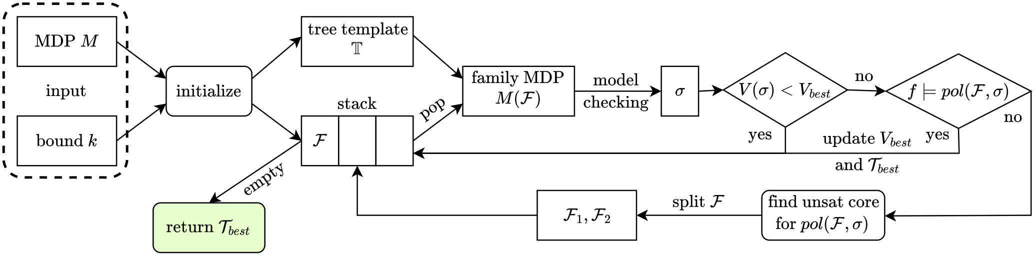
Given a UC we find a critical subset of state $S^{UC} = \{s \in S \mid \exists n : (s, n) \in SP^{UC}\}$

We call f_1, f_2 Harmonizing parameterizations if $\exists!\ x \in \chi^{\mathbb{T}} : f_1(x) \neq f_2(x)$ and for all state $s \in S^{UC}$ it holds $\sigma(s) \in \{\sigma_{\mathbb{T}(f_1)}(s), \sigma_{\mathbb{T}(f_2)}(s)\}$

We split the current parametrization set $\mathcal{F}$ in variable x to two sets $\mathcal{F}_1, \mathcal{F}_2$ and push into the stack

This algorithm is sound and complete

Overview of Our Approach



Iterative Algorithm

Find a DT $\mathcal{T}$ of depth up to k with maximum value

Given k we can construct a template that adheres to the depth constraint and use the recursive DT construction algorithm

Even for modest values of k it is infeasible to explore all possible DTs and finding good DTs early can accelerate the synthesis, therefore, we iteratively explore 0-DTs, 1-DTs, 2-DTs... up to k with some timeout

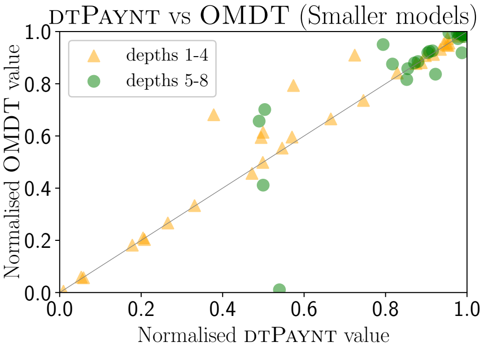
We reuse results from previous iterations (depths)

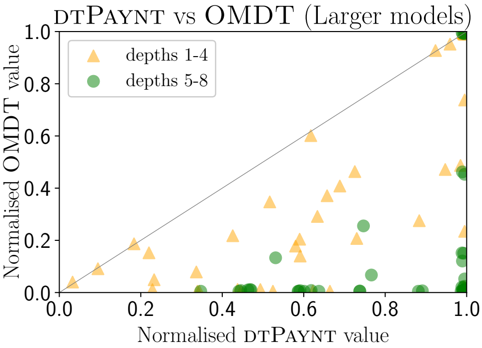
Finally we apply trivial DT post-processing

This proposed algorithm is called dtPAYNT and is implemented in the tool PAYNT, we use Storm for MDP model checking and Z3 as SMT solver

Experimental Evaluation I.

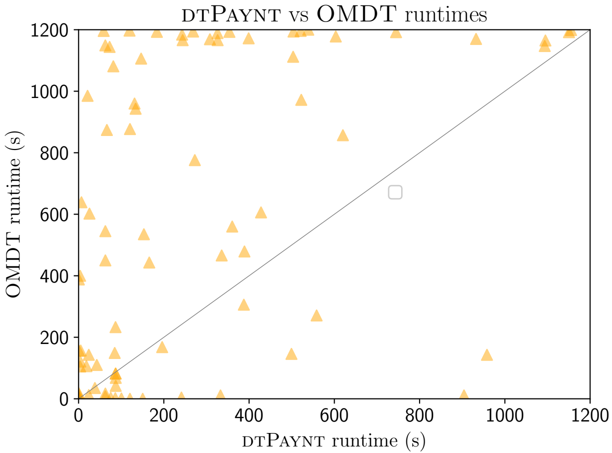
Comparison with OMDT - DT value





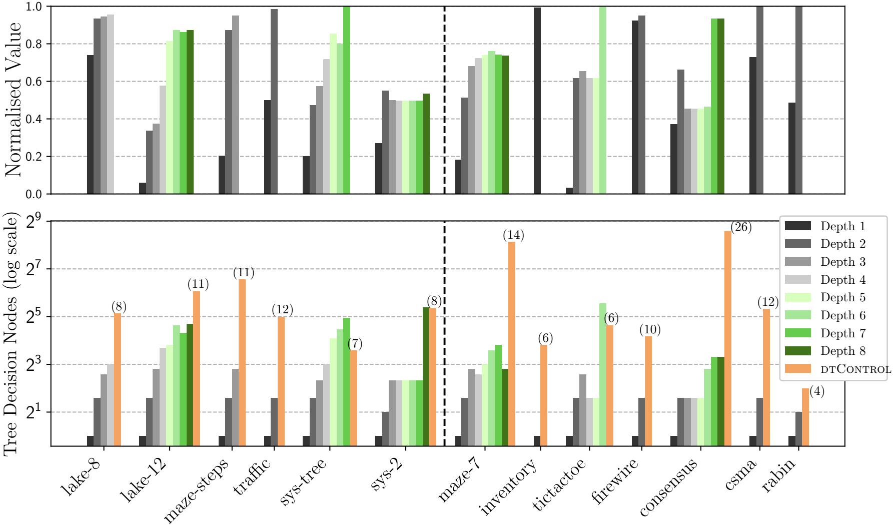
Experimental Evaluation I.

Comparison with OMDT - Runtime



Experimental Evaluation II.

Comparison with dtControl - size and value trade-off



Experimental Evaluation III.

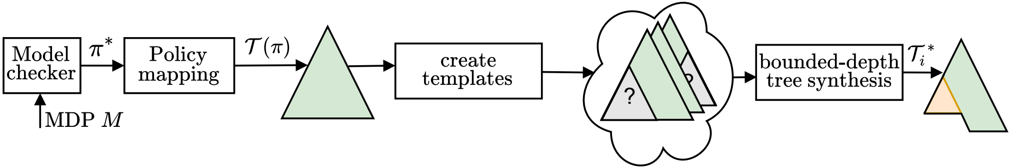
Scalability and the policy mapping problem

model	$\lvert S \rvert$	choices	dtPAYNT			dtControl		
			value	nodes	time	$\lvert\sigma^*_{rel}\rvert$	nodes	time
ij-20	1M	10M	1	0	547s	624k	393k	210s
pnueli-zuck-5	308k	1.7M	1	0	103s	2395	1258	1s
firewire-f-36	212k	479k	1	14	531s	376	12	1s
pacman-30	853k	1.1M	0.76	6	1360s	673	144	1s
firewire-t-3-600	1.1M	1.5M	0.85	8	3135s	1147	12	1s

Use Case

dtPAYNT can be used for DT minimization

- For the model *csma* with 1.5M states, dtControl finds a DT with 236 nodes and depth 19
- After 30 minutes and 24 successfully minimized subtrees, dtPAYNT finds a DT with 22 nodes and depth 6 (more than 90% decrease) while retaining the performance of the policy at 99.5%



Check out: Andriushchenko et al. Symbiotic Local Search for Small Decision Tree Policies in MDPs. Accepted to UAI'25.

Conclusion

We proposed dtPAYNT a novel algorithm for synthesizing small DTs in MDPs

We showcased state-of-the-art performance on bounded-depth synthesis problem, advantages in size/value trade-off compared to policy mapping techniques and potential use case of the algorithm

Future work:

- exploit counter-examples in the DT synthesis
- represent the policy only for some relevant subset of states

Artifact

Preprint