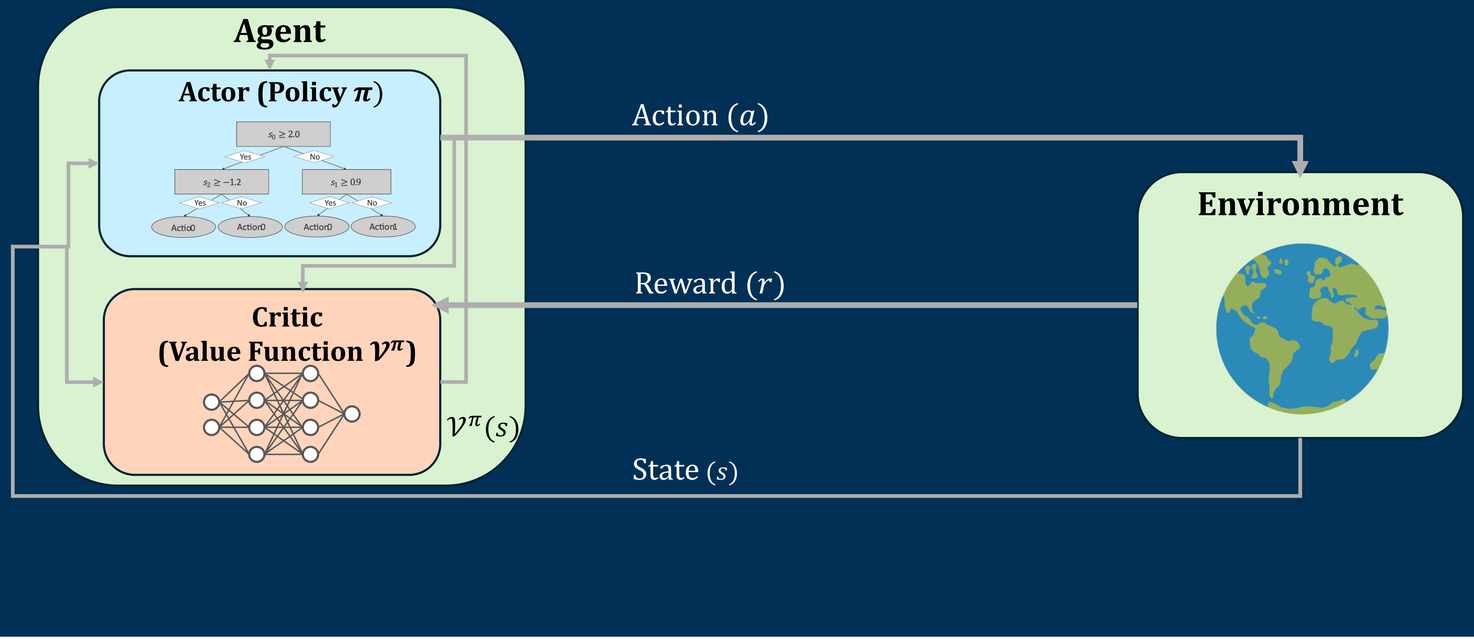
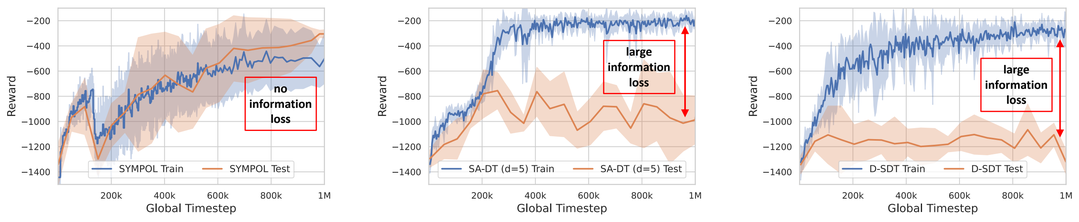
We can learn interpretable **Decision Trees** with **Policy Gradients**!



SYMPOL: Symbolic Tree-Based On-Policy Reinforcement Learning

Interpretable Decision Tree Policies without Information Loss



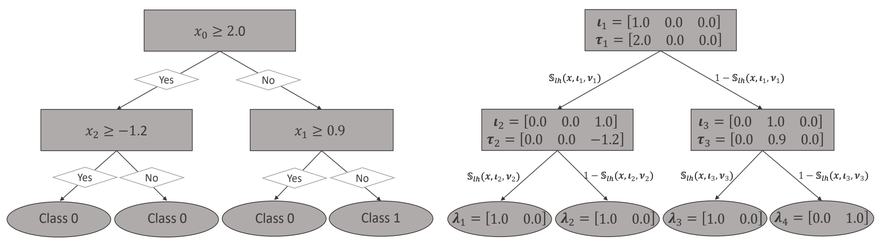
(a) SYMPOL (ours) (b) State-Action DT (c) Discretized Soft DT

Figure 1: **Information Loss in Tree-Based Reinforcement Learning on Pendulum.** Existing methods for symbolic, tree-based RL (see Figure 1b and 1c) suffer from severe information loss when converting the differentiable policy (high train reward) into the symbolic policy (low test reward). Using SYMPOL (Figure 1a), we can directly optimize the symbolic policy with PPO and therefore have no information loss during the application (high train and test reward).

GradTree: Gradient-Based Decision Trees

Dense DT Representation

- Relaxing the split indices and split thresholds
  →Allow reasonable optimization with policy gradients



(a) Vanilla DT Representation (b) Dense DT Representation

Straight-Through Operator for non-differentiable operations

(1) Hardmax function to enforce one-hot encoded split vectors → **univariate, axis-aligned DTs**

(2) Discretization of the split function (round the sigmoid output) → **hard splits**

SYMPOL: Symbolic Tree-Based On-Policy RL

Actor-Critic architecture

- Interpretable DT actor
- Full-complexity critic

→ capture complexity without sacrificing interpretability

Weight Decay

- Favor dynamic adjustments of tree architecture

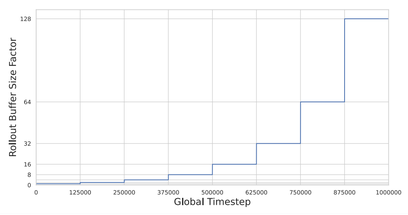
Exploration Stability

- dynamic rollout buffer size

→ Exploration in early

→ stability in later iterations

$$n_t = n_{\text{init}} \times 2^{\left\lfloor \frac{(t+1)\times i}{1+t_{\text{total}}} \right\rfloor - 1}$$
$$\text{with} \quad i = 1 + \log_2\left(\frac{n_{\text{init}}}{n_{\text{final}}}\right)$$



Gradient Stability

- dynamic batch size

→ fast convergence early

→ gradient stability later on

SYMPOL learns accurate DT policies

- SYMPOL is consistently among the best interpretable models
- Significantly higher rewards on several Environments (LL and PD-C)
- SYMPOL is competitive to full-complexity models on most environments

	CP	AB	LL	MC-C	PD-C
SYMPOL (ours)	**500**	**- 80**	**- 57**	**94**	**- 323**
D-SDT	128	-205	-221	-10	-1343
SA-DT (d=5)	446	-97	-197	**97**	-1251
SA-DT (d=8)	476	**- 75**	-150	**96**	- 854
MLP	500	- 72	241	95	- 191
SDT	500	- 77	-124	- 4	- 310

DT policies offer a good inductive bias for categorical environments

- DTs are not well-suited for modeling physical relationships

  → DTs are best suited for categorical environments
  - due to their effective use of axis-aligned splits
- SYMPOL achieves comparable or superior results to full-complexity model on categorical environments

	E-R	DK	LG-5	LG-7	DS
SYMPOL (ours)	**0.964**	**0.959**	**0.951**	**0.953**	0.939
D-SDT	0.662	0.654	0.262	0.381	0.932
SA-DT (d=5)	0.583	**0.958**	**0.951**	0.458	**0.952**
SA-DT (d=8)	0.845	**0.961**	**0.951**	0.799	**0.954**
MLP	0.963	0.963	0.951	0.760	0.951
SDT	0.966	0.959	0.839	0.953	0.954

SYMPOL does not exhibit information loss

- Existing methods for learning DT policies usually involve postprocessing to obtain the interpretable model.

  → mismatch between the optimized and interpreted policy
- SYMPOL directly optimizes a DT on-policy

  → learned policy remains consistent from training to inference

	Cohen's D ↓
SYMPOL (ours)	**-0.019**
SA-DT (d=5)	3.449
SA-DT (d=8)	2.527
D-SDT	3.126
MLP	**0.306**
SDT	**0.040**

DT policies learned with SYMPOL are small and interpretable

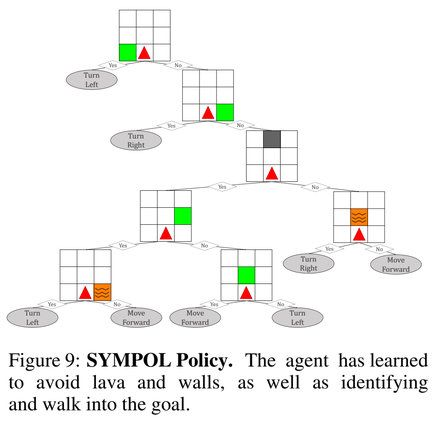


Figure 9: **SYMPOL Policy.** The agent has learned to avoid lava and walls, as well as identifying and walk into the goal.

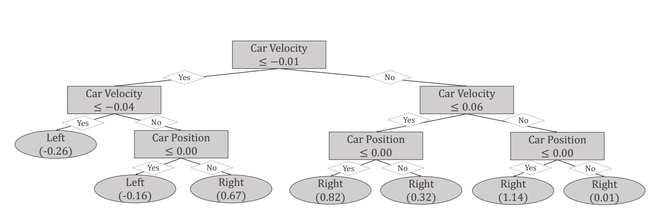


Figure 4: **SYMPOL Policy for MountainCar.** The main rule encoded by this tree is that the car should accelerate to the left, if its velocity is negative and to the right if it is positive, which essentially increases the speed of the car over time, making it possible to reach the goal at the top of the hill. The magnitude of the acceleration is mainly determined by the current position, reducing the cost of the actions.

Technical contributions are relevant to performance

- Each component substantially contributes to the overall performance

  → supports intuitive justifications for our modifications

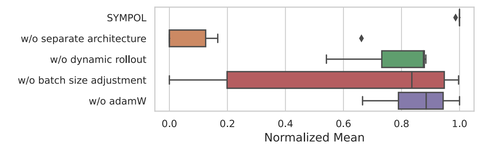
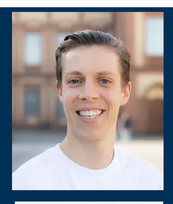


Figure 5: **Ablation Study**. We report the mean normalized performance over all control environments.

Sascha Marton
sascha.marton@uni-mannheim.de
University of Mannheim



Tim Grams
tim.nico.grams@uni-mannheim.de
University of Mannheim

Florian Vogt
forian.vogt@uni-mannheim.de
University of Mannheim

Dr. Christian Bartelt
christian.bartelt@uni-mannheim.de
University of Mannheim

Jun.-Prof. Dr. Stefan Lüdtke
stefan.luedtke@uni-rostock.de
Institute for Visual and Analytic Computing

Prof. Dr. Heiner Stuckenschmidt
heiner.stuckenschmidt@uni-mannheim.de
University of Mannheim

https://github.com/s-marton/SYMPOL