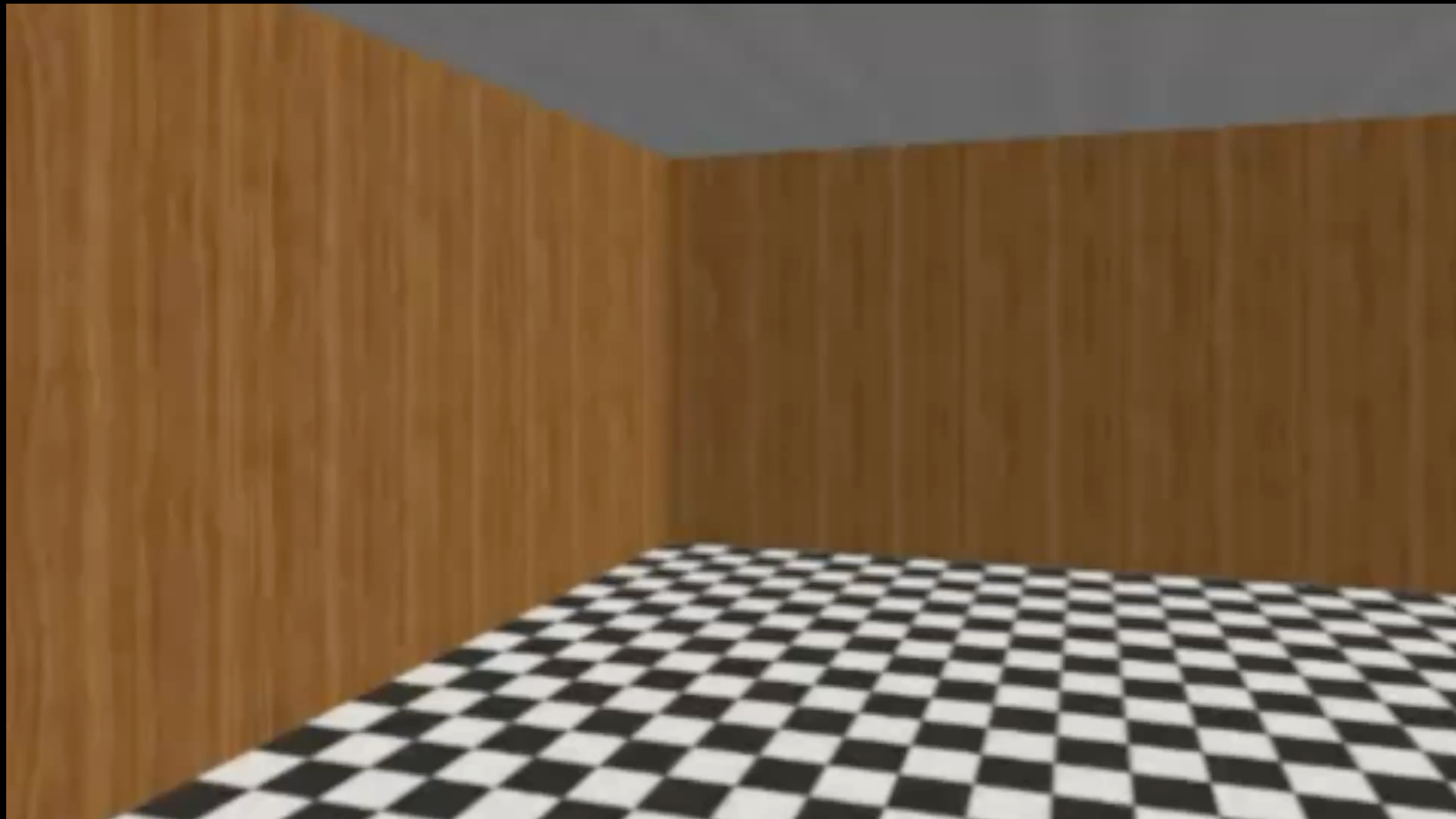
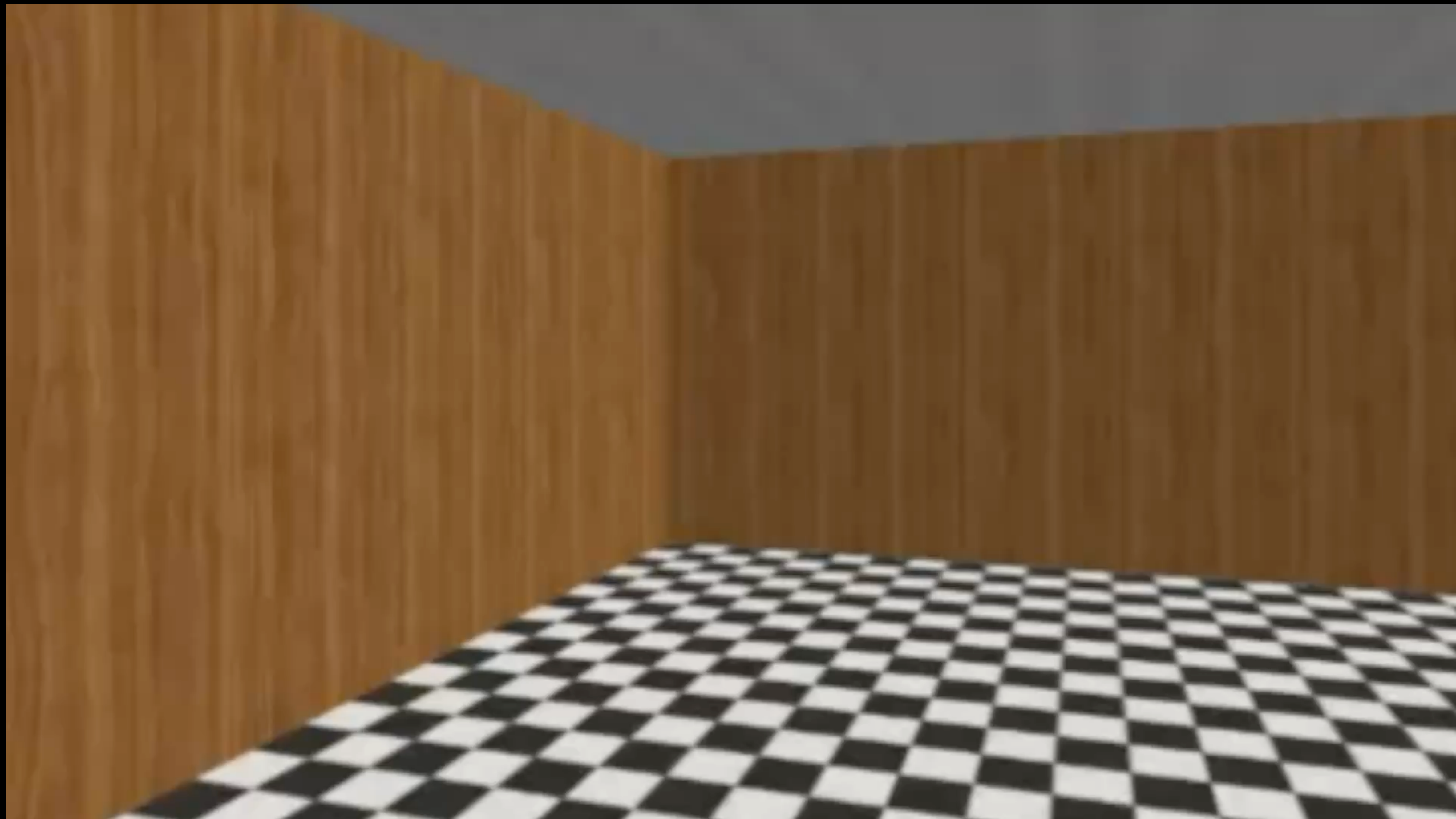
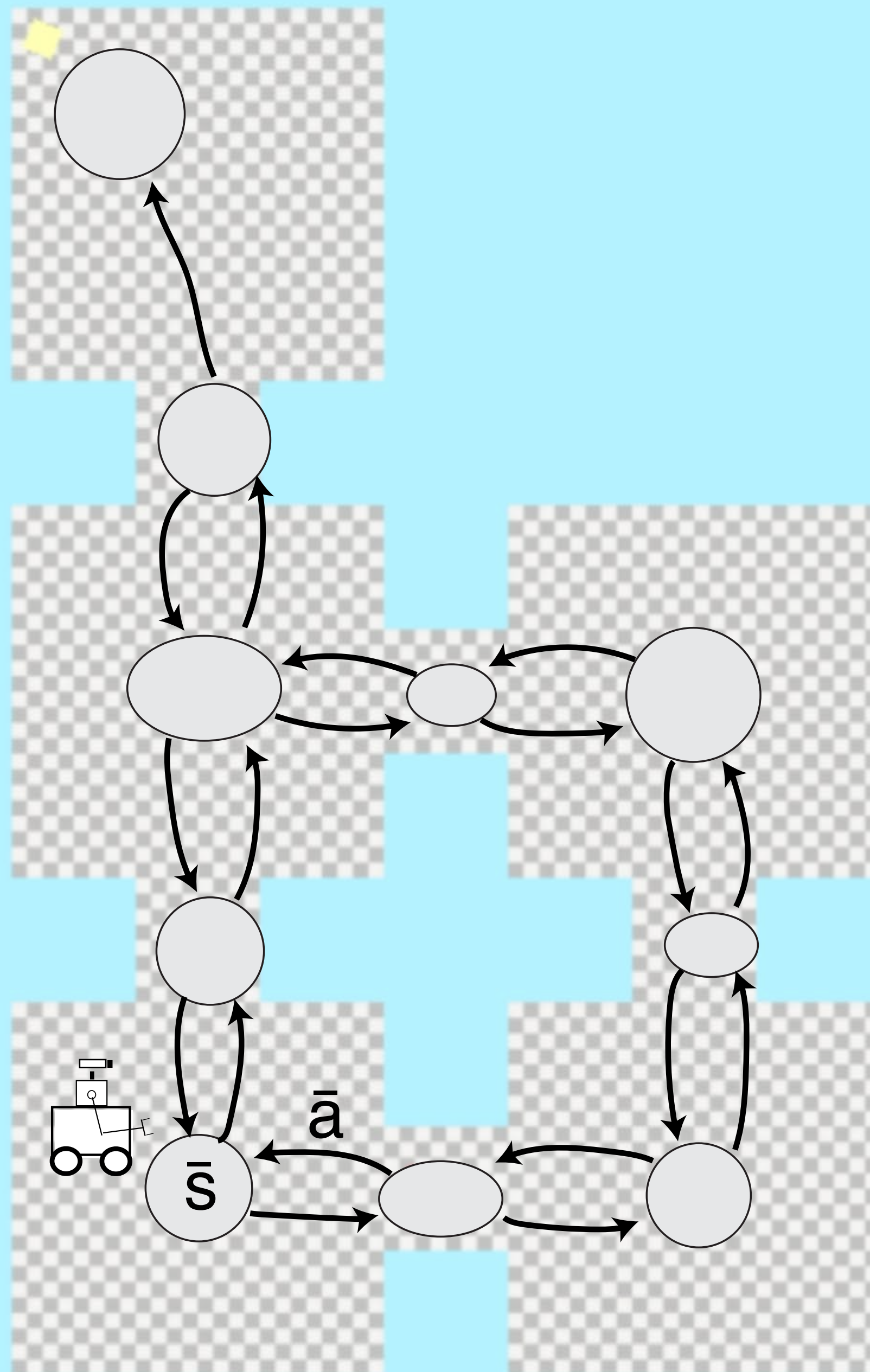


Skill-Driven Neurosymbolic State Abstractions

Alper Ahmetoglu*, Steven James, Cameron Allen,
Sam Lobel, David Abel, George Konidaris







```
pos: (-0.38, 0.00, 3.95)
angle: 348
steps: 397
```

Semantics of Abstract States

Semantics of Abstract States

Bellman Equation—the bedrock of almost all value-based RL methods.

$$V(s) = \sum_{\bar{a}} \pi(\bar{a} \mid s) \mathbb{E}_{s'} [R(s, \bar{a}, s') + \gamma^{\tau} V(s')]$$

Semantics of Abstract States

Bellman Equation—the bedrock of almost all value-based RL methods.

$$V(s) = \sum_{\bar{a}} \pi(\bar{a} \mid s) \mathbb{E}_{s'} \left[R(s, \bar{a}, s') + \gamma^\tau V(s') \right]$$

$$\approx \sum_{\bar{a}} \pi(\bar{a} \mid s) \left(\mathbb{E}_{s'} \left[R(s, \bar{a}, s') \right] + \gamma^{\mathbb{E}_{s'}[\tau]} \mathbb{E}_{s'} \left[V(s') \right] \right)$$

(Abel et al., 2019)

Semantics of Abstract States

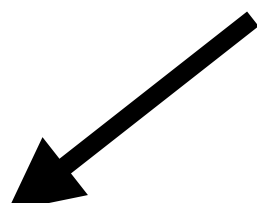
$$\approx \sum_{\bar{a}} \pi(\bar{a} \mid s) \left(\mathbb{E}_{s'} [R(s, \bar{a}, s')] + \gamma^{\mathbb{E}_{s'}[\tau]} \mathbb{E}_{s'} [V(s')] \right)$$

Semantics of Abstract States

$$\approx \sum_{\bar{a}} \pi(\bar{a} \mid s) \left(\mathbb{E}_{s'} [R(s, \bar{a}, s')] + \gamma^{\mathbb{E}_{s'}[\tau]} \mathbb{E}_{s'} [V(s')] \right)$$

$\mathbb{E}_{s'}[R(s, \bar{a}, s')]$


$\mathbb{E}_{s'}[\tau]$


$\mathbb{E}_{s'}[V(s')]$


$$V(s) \approx \sum_{\bar{a}} \pi(\bar{a} \mid s) \left[R(s, \bar{a}, \bar{s}') + \gamma^{\bar{\tau}} V(\bar{s}') \right]$$

Semantics of Abstract States

$$\approx \sum_{\bar{a}} \pi(\bar{a} \mid s) \left(\mathbb{E}_{s'} [R(s, \bar{a}, s')] + \gamma^{\mathbb{E}_{s'}[\tau]} \mathbb{E}_{s'} [V(s')] \right)$$

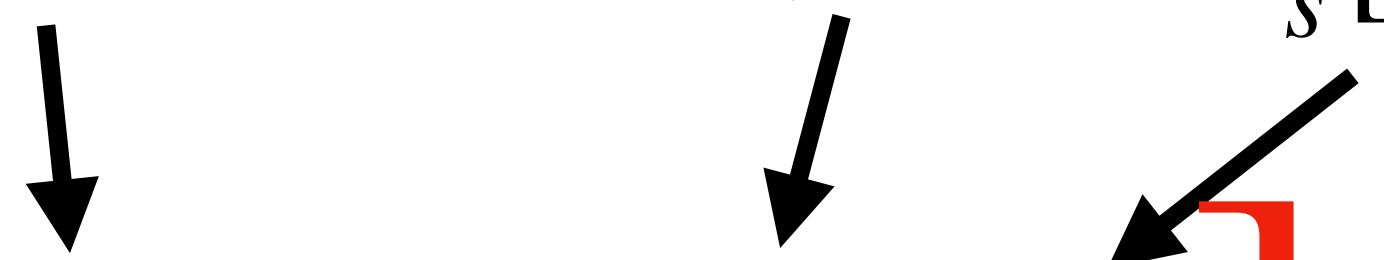
$$V(s) \approx \sum_{\bar{a}} \pi(\bar{a} \mid s) \left[R(s, \bar{a}, \bar{s}') + \gamma^{\bar{\tau}} V(\bar{s}') \right]$$

$\mathbb{E}_{s'}[R(s, \bar{a}, s')] \quad \mathbb{E}_{s'}[\tau] \quad \mathbb{E}_{s'}[V(s')]$

Semantics of Abstract States

$$\approx \sum_{\bar{a}} \pi(\bar{a} \mid s) \left(\mathbb{E}_{s'} [R(s, \bar{a}, s')] + \gamma^{\mathbb{E}_{s'}[\tau]} \mathbb{E}_{s'} [V(s')] \right)$$

$$V(s) \approx \mathbb{E}_s \left[\sum_{\bar{a}} \pi(\bar{a} \mid s) \left[R(s, \bar{a}, \bar{s}') + \gamma^{\bar{\tau}} V(\bar{s}') \right] \right]$$

$\mathbb{E}_{s'}[R(s, \bar{a}, s')] \quad \mathbb{E}_{s'}[\tau] \quad \mathbb{E}_{s'}[V(s')]$


Semantics of Abstract States

$$\approx \sum_{\bar{a}} \pi(\bar{a} \mid s) \left(\mathbb{E}_{s'} [R(s, \bar{a}, s')] + \gamma^{\mathbb{E}_{s'}[\tau]} \mathbb{E}_{s'} [V(s')] \right)$$

$$V(s) \approx \mathbb{E}_s \left[\sum_{\bar{a}} \pi(\bar{a} \mid s) \left[R(s, \bar{a}, \bar{s}') + \gamma^{\bar{\tau}} V(\bar{s}') \right] \right]$$

Diagram illustrating the semantic approximation of the value function $V(s)$ using abstract states. The expression shows the expectation over the concrete state s of the sum over abstract states $\bar{a}$ of the probability $\pi(\bar{a} \mid s)$ multiplied by the expected reward and discounted value in the abstract state space.

Annotations with arrows pointing to the corresponding terms in the abstract state expression:

- $\mathbb{E}_{s'}[R(s, \bar{a}, s')]$ points to $R(s, \bar{a}, \bar{s}')$
- $\mathbb{E}_{s'}[\tau]$ points to $\gamma^{\bar{\tau}}$
- $\mathbb{E}_{s'}[V(s')]$ points to $V(\bar{s}')$

Semantics of Abstract States

$$\approx \sum_{\bar{a}} \pi(\bar{a} \mid s) \left(\mathbb{E}_{s'} [R(s, \bar{a}, s')] + \gamma^{\mathbb{E}_{s'}[\tau]} \mathbb{E}_{s'} [V(s')] \right)$$

$$V(s) \approx \mathbb{E}_s \left[\sum_{\bar{a}} \pi(\bar{a} \mid s) \left[\overset{\mathbb{E}_{s'}[R(s, \bar{a}, s')]}{\downarrow} R(s, \bar{a}, \bar{s}') + \gamma^{\overset{\mathbb{E}_{s'}[\tau]}{\downarrow}} \overset{\mathbb{E}_{s'}[V(s')]}{\downarrow} V(\bar{s}') \right] \right]$$

$$V(\bar{s}) \approx \sum_{\bar{a}} \bar{\pi}(\bar{a} \mid \bar{s}) [R(\bar{s}, \bar{a}, \bar{s}') + \gamma^{\bar{\tau}} V(\bar{s}')]]$$

Semantics of Abstract States

$$\approx \sum_{\bar{a}} \pi(\bar{a} \mid s) \left(\mathbb{E}_{s'} [R(s, \bar{a}, s')] + \gamma^{\mathbb{E}_{s'}[\tau]} \mathbb{E}_{s'} [V(s')] \right)$$

$$V(s) \approx \mathbb{E}_s \left[\sum_{\bar{a}} \pi(\bar{a} \mid s) \left[\overset{\mathbb{E}_{s'}[R(s, \bar{a}, s')]}{\downarrow} R(s, \bar{a}, \bar{s}') + \gamma^{\overset{\mathbb{E}_{s'}[\tau]}{\downarrow}} \overset{\mathbb{E}_{s'}[V(s')]}{\downarrow} V(\bar{s}') \right] \right]$$

Abstract Bellman

$$V(\bar{s}) \approx \sum_{\bar{a}} \bar{\pi}(\bar{a} \mid \bar{s}) [R(\bar{s}, \bar{a}, \bar{s}') + \gamma^{\bar{\tau}} V(\bar{s}')]]$$

Satisfying the Markov Property

Satisfying the Markov Property

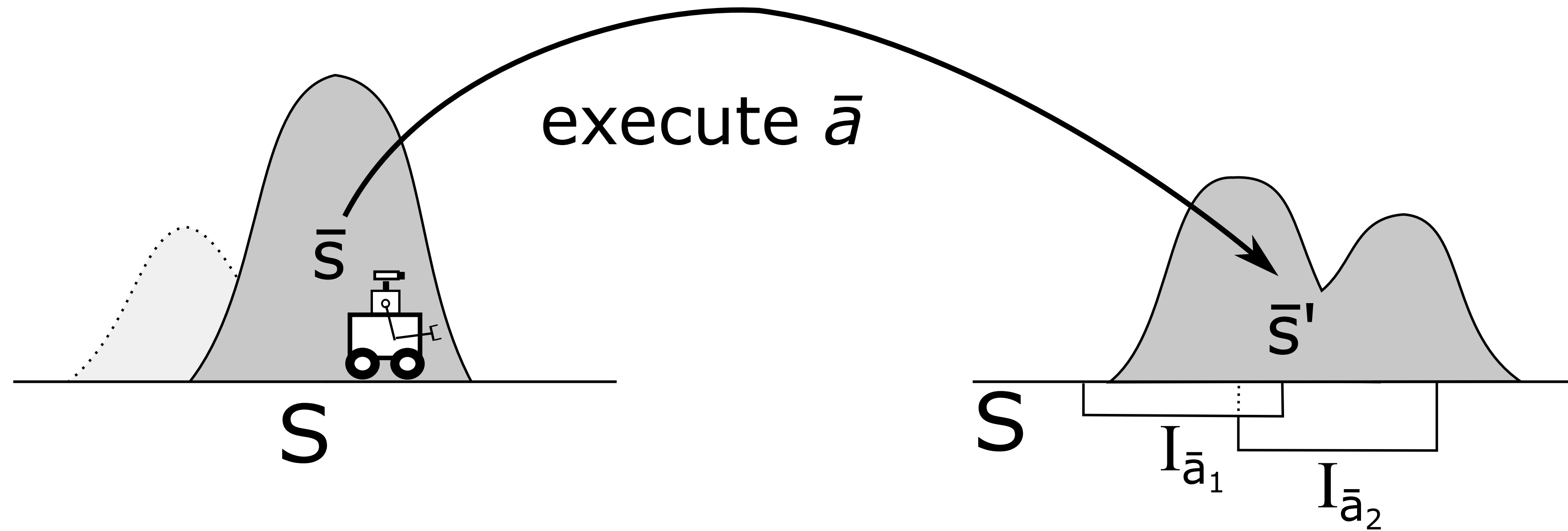
1. Transition only a function of $s, \bar{a}, s'$
2. Reward only a function of $s, \bar{a}, s'$
3. Available actions only a function s .

Satisfying the Markov Property

1. Transition only a function of $s, \bar{a}, s'$ ✓
2. Reward only a function of $s, \bar{a}, s'$ ✓
3. Available actions only a function s .

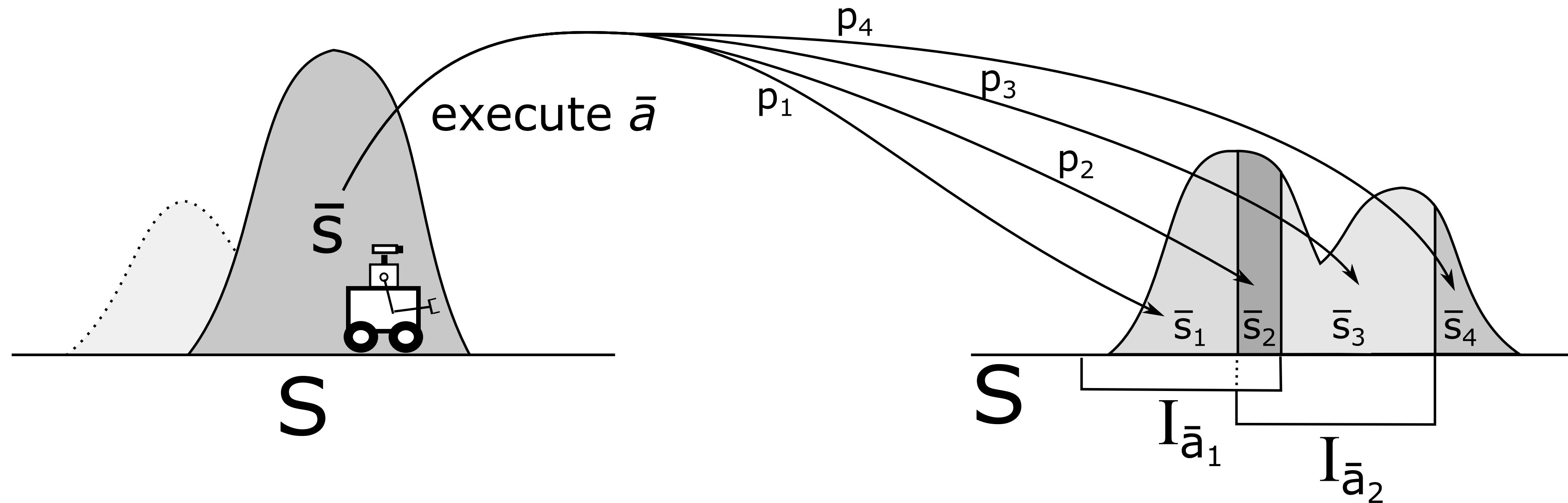
Satisfying the Markov Property

1. Transition only a function of $s, \bar{a}, s'$ ✓
2. Reward only a function of $s, \bar{a}, s'$ ✓
3. Available actions only a function s .



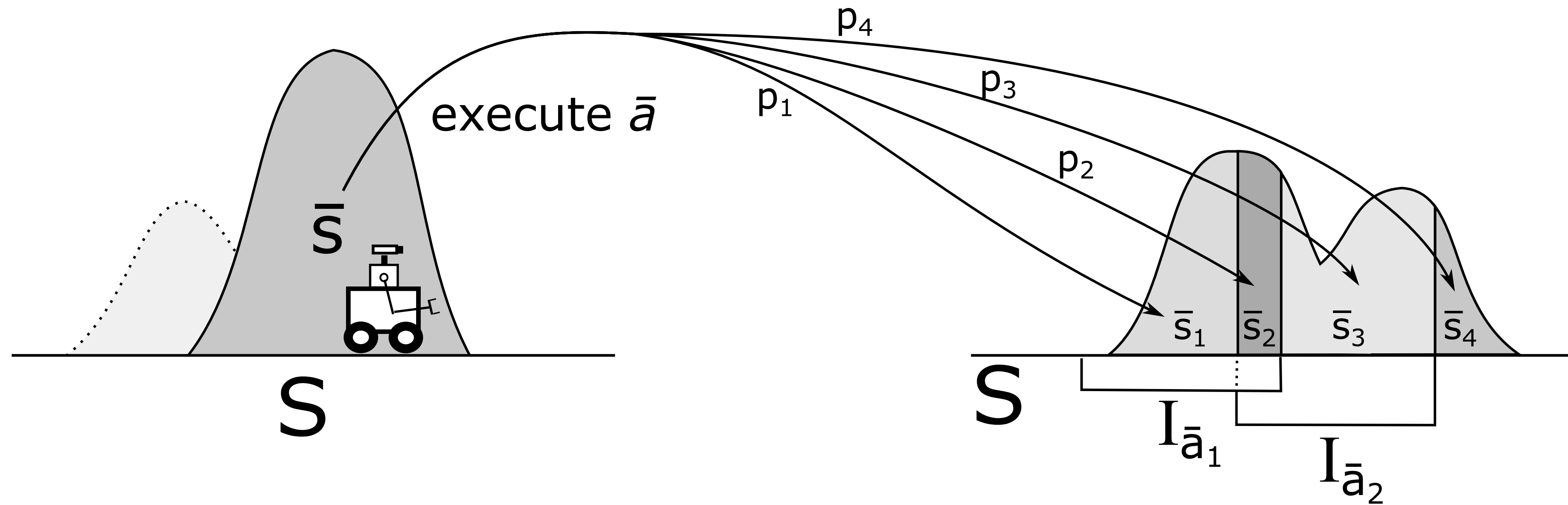
Satisfying the Markov Property

1. Transition only a function of $s, \bar{a}, s'$ ✓
2. Reward only a function of $s, \bar{a}, s'$ ✓
3. Available actions only a function s .



Satisfying the Markov Property

1. Transition only a function of $s, \bar{a}, s'$ ✓
2. Reward only a function of $s, \bar{a}, s'$ ✓
3. Available actions only a function s . ✓



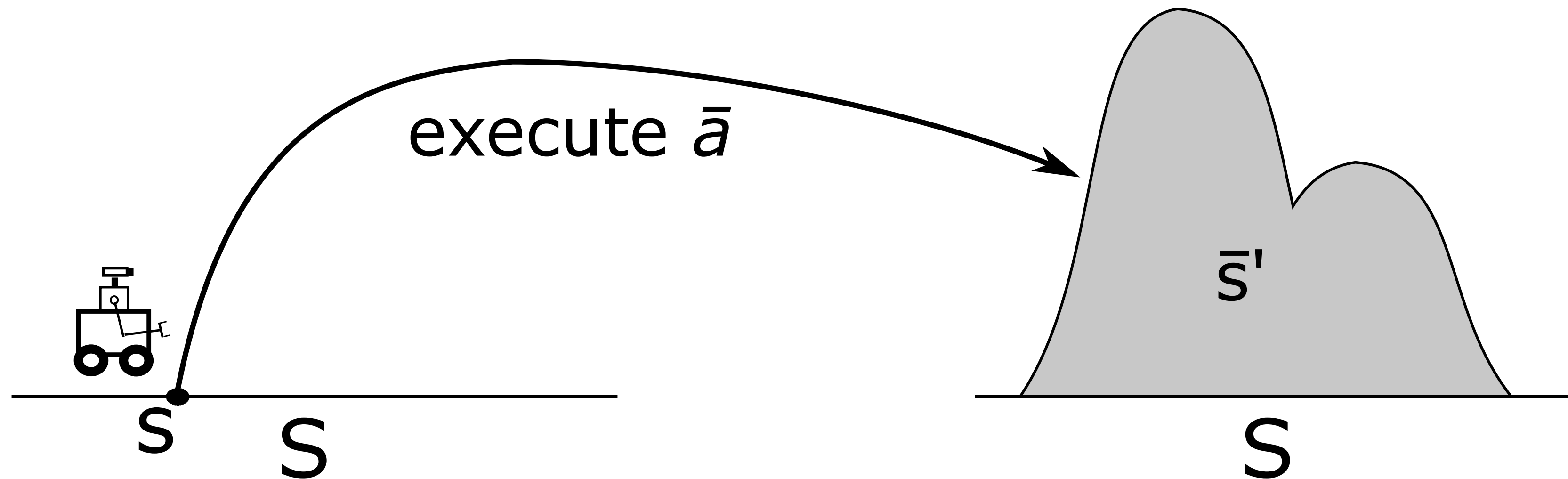
Model-Preserving Abstract MDP

Model-Preserving Abstract MDP

$$\int_{s'} |T(s' \mid s, \bar{a}) - T(s' \mid \bar{s}, \bar{a})| ds' \leq \epsilon_T \quad \text{and} \quad |R(s, \bar{a}) - R(\bar{s}, \bar{a})| \leq \epsilon_R$$
$$\implies |Q^\pi(s, \bar{a}) - Q^\pi(\bar{s}, \bar{a})| \leq \frac{\epsilon_R + \gamma V_{\text{Max}} \epsilon_T}{1 - \gamma}$$

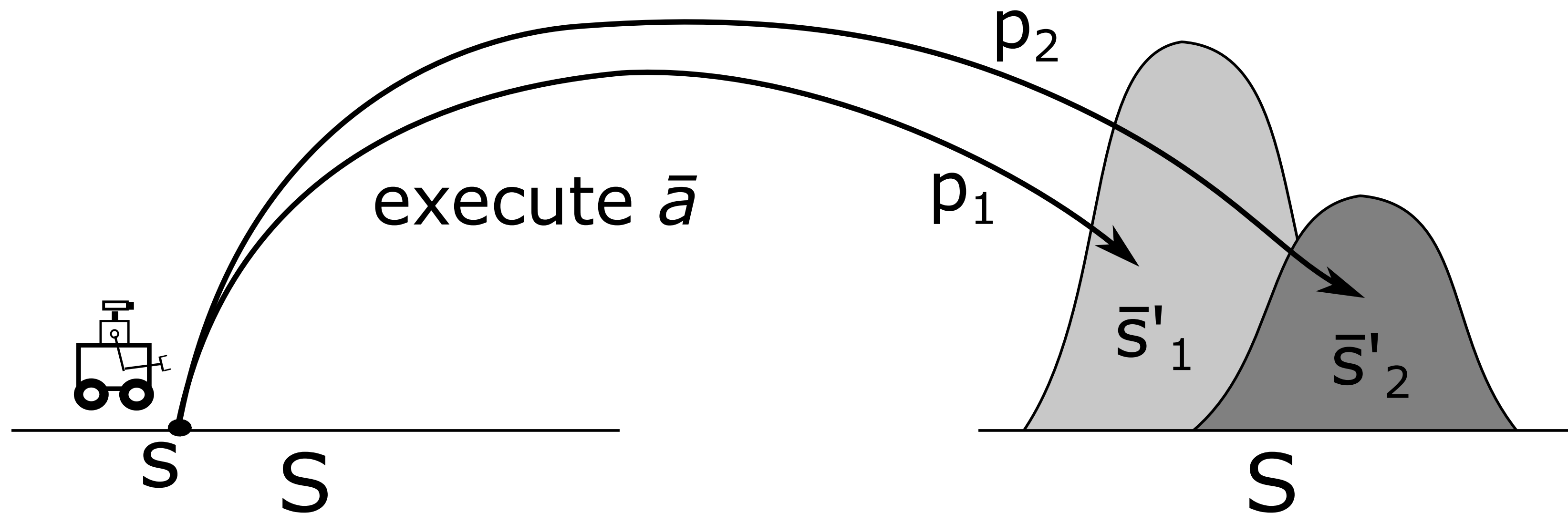
Model-Preserving Abstract MDP

$$\int_{s'} |T(s' | s, \bar{a}) - T(s' | \bar{s}, \bar{a})| ds' \leq \epsilon_T \quad \text{and} \quad |R(s, \bar{a}) - R(\bar{s}, \bar{a})| \leq \epsilon_R$$
$$\Rightarrow |Q^\pi(s, \bar{a}) - Q^\pi(\bar{s}, \bar{a})| \leq \frac{\epsilon_R + \gamma V \text{Max} \epsilon_T}{1 - \gamma}$$



Model-Preserving Abstract MDP

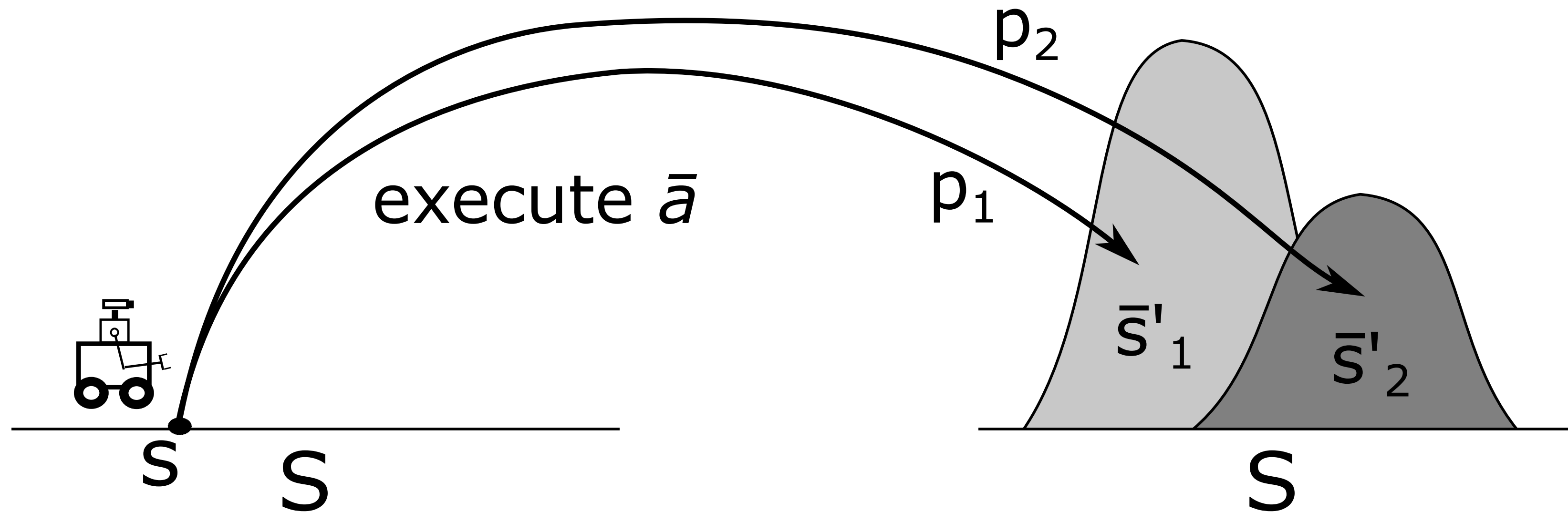
$$\int_{s'} |T(s' | s, \bar{a}) - T(s' | \bar{s}, \bar{a})| ds' \leq \epsilon_T \quad \text{and} \quad |R(s, \bar{a}) - R(\bar{s}, \bar{a})| \leq \epsilon_R$$
$$\Rightarrow |Q^\pi(s, \bar{a}) - Q^\pi(\bar{s}, \bar{a})| \leq \frac{\epsilon_R + \gamma V \text{Max} \epsilon_T}{1 - \gamma}$$



Model-Preserving Abstract MDP

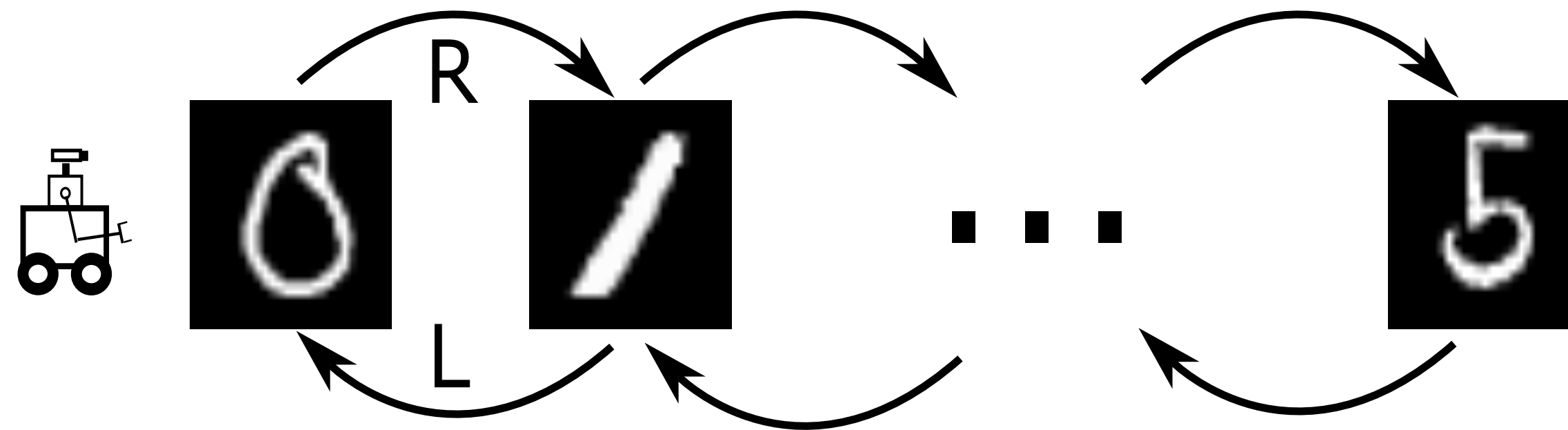
$$\int_{s'} |T(s' | s, \bar{a}) - T(s' | \bar{s}, \bar{a})| ds' \leq \epsilon_T \quad \text{and} \quad |R(s, \bar{a}) - R(\bar{s}, \bar{a})| \leq \epsilon_R$$

$$\Rightarrow |Q^\pi(s, \bar{a}) - Q^\pi(\bar{s}, \bar{a})| \leq \frac{\epsilon_R + \gamma V \text{Max} \epsilon_T}{1 - \gamma}$$

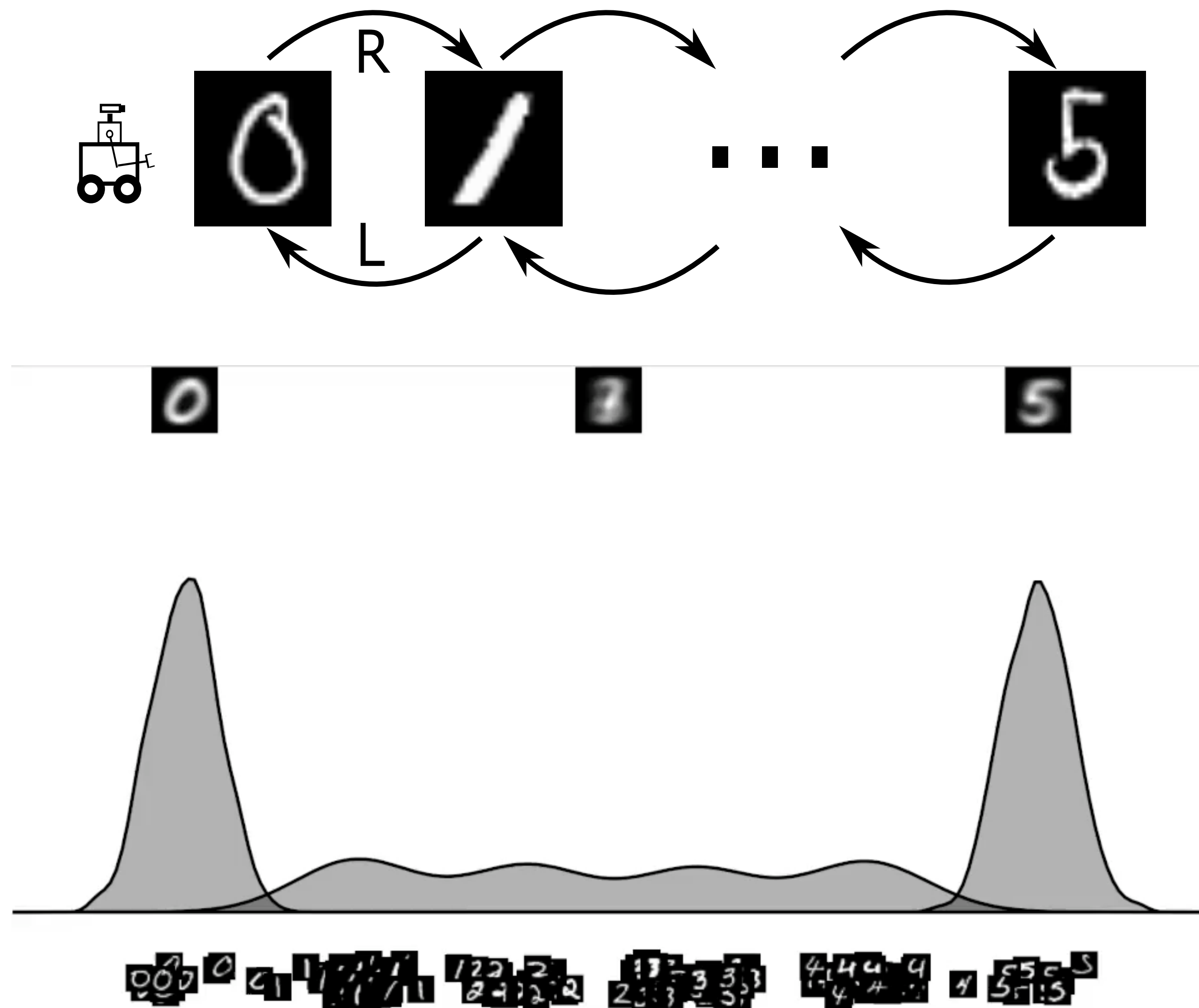


$$V(\bar{s}) \approx \sum_{\bar{a}} \pi(\bar{a} | \bar{s}) [R(\bar{s}, \bar{a}, \bar{s}') + \gamma^{\bar{\tau}} V(\bar{s}')] \rightarrow V(\bar{s}) \approx \sum_{\bar{a}} \pi(\bar{a} | \bar{s}) \sum_i p_i [R(\bar{s}, \bar{a}, \bar{s}'_i) + \gamma^{\bar{\tau}_i} V(\bar{s}'_i)]$$

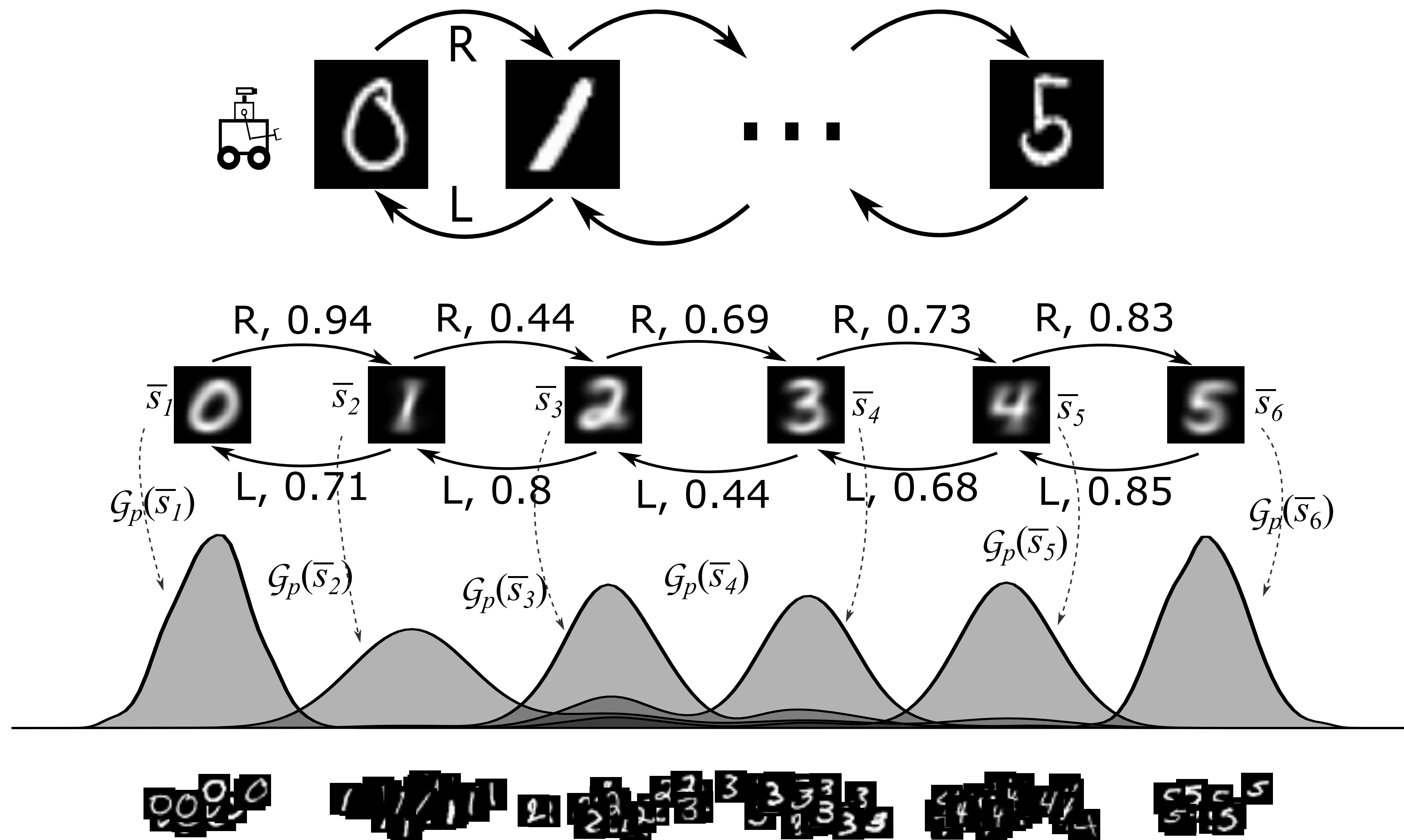
A Chainwalk Example



A Chainwalk Example



A Chainwalk Example



Action: To clockwise hallway



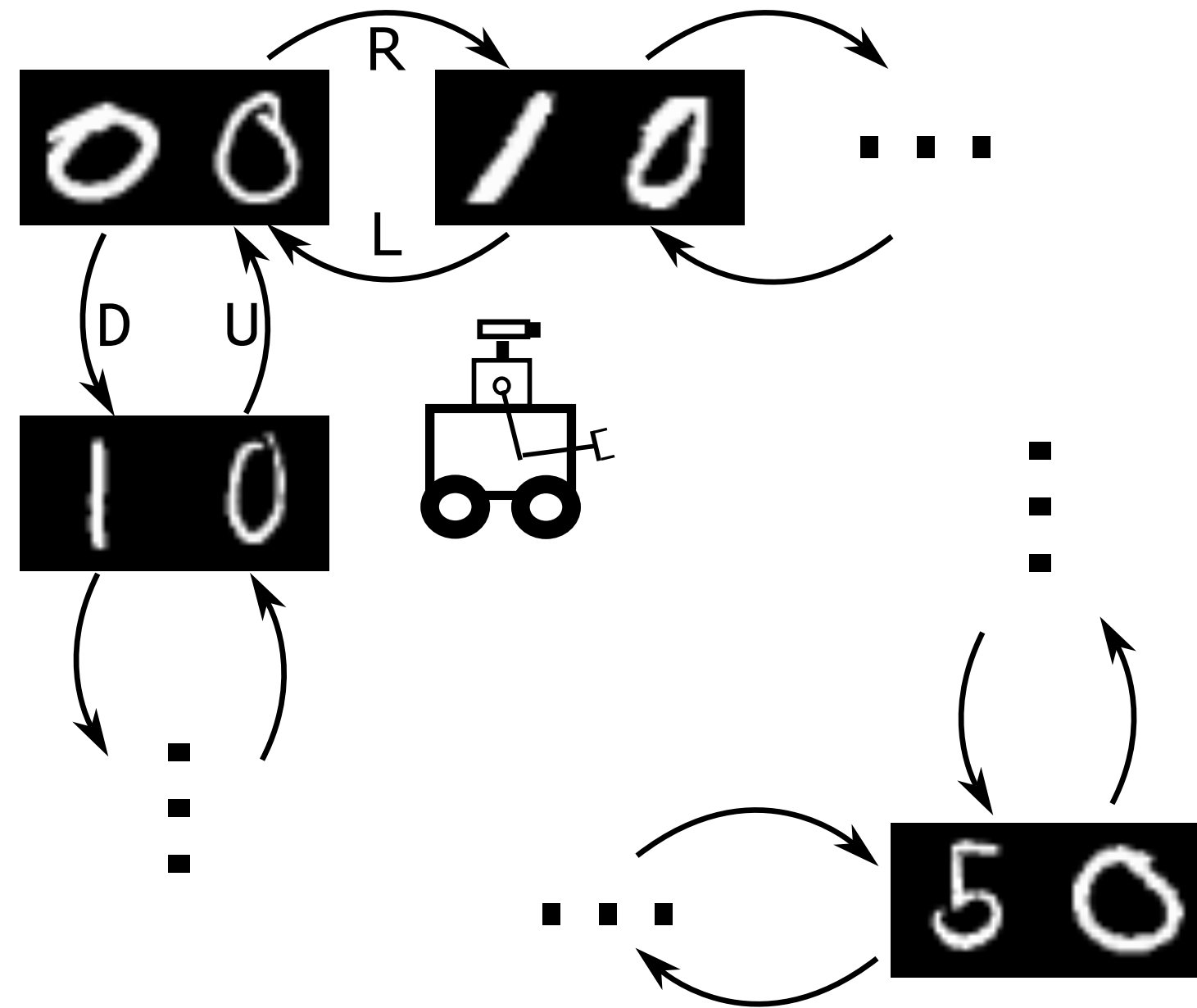
Action: To clockwise hallway



Factored Skills to Factored MDPs

Actions modify only some variables.

$$s = (s_1, s_2, s_3, s_4, s_5) \xrightarrow{\bar{a}} s' = (s_1, s_2, s_3, s'_4, s'_5)$$

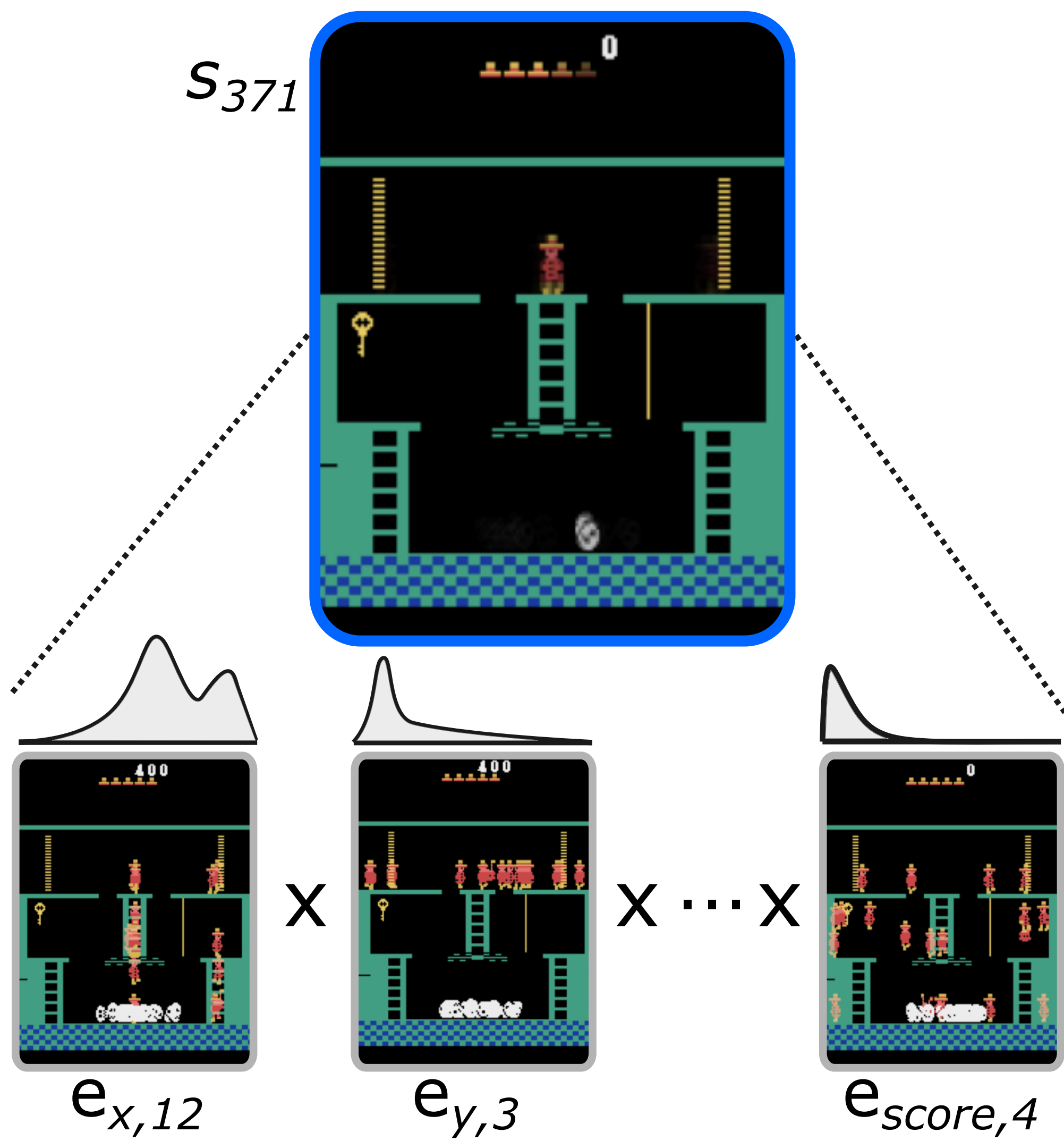


Action: Up



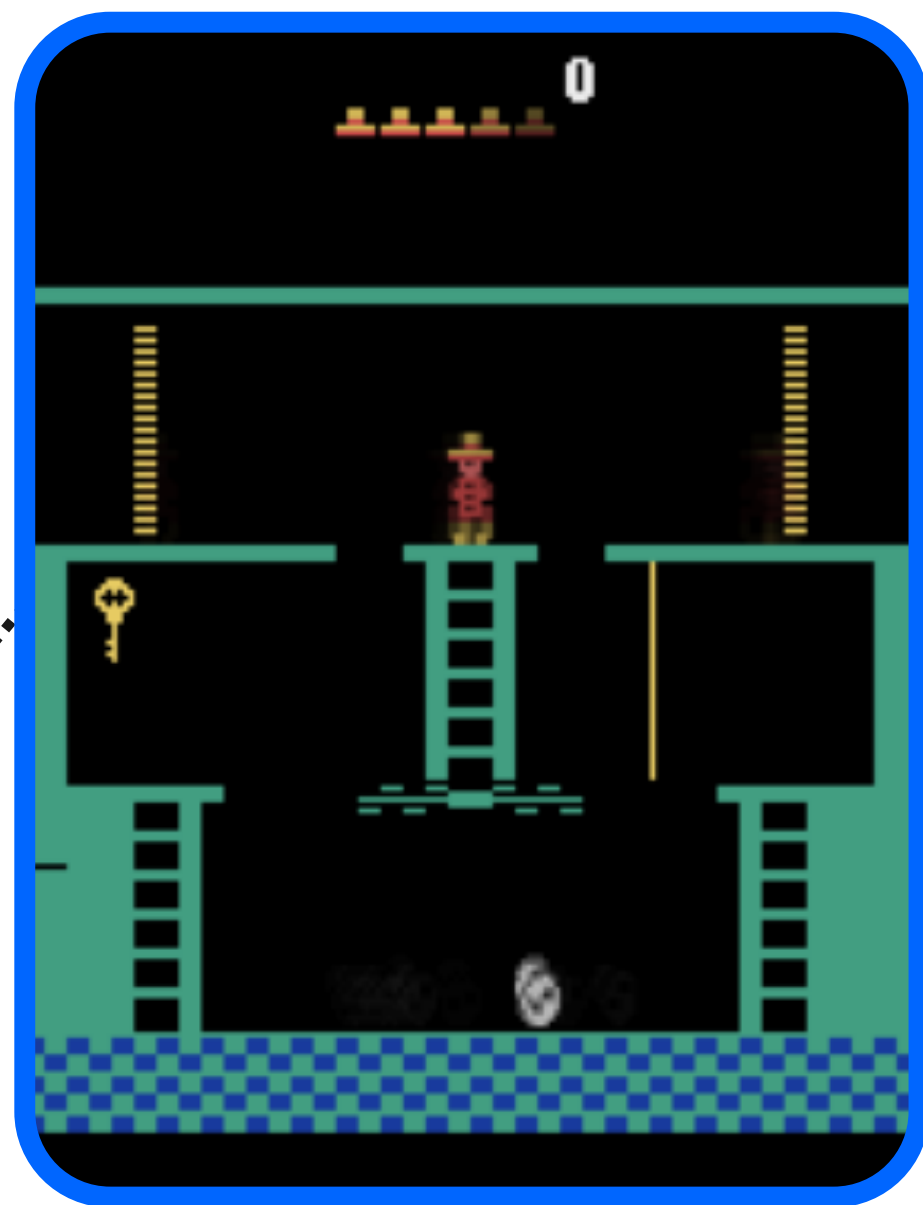
Action: Up



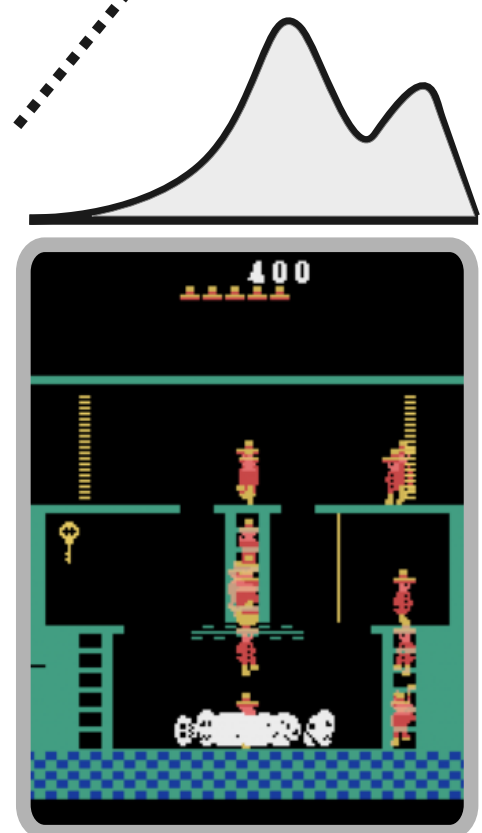
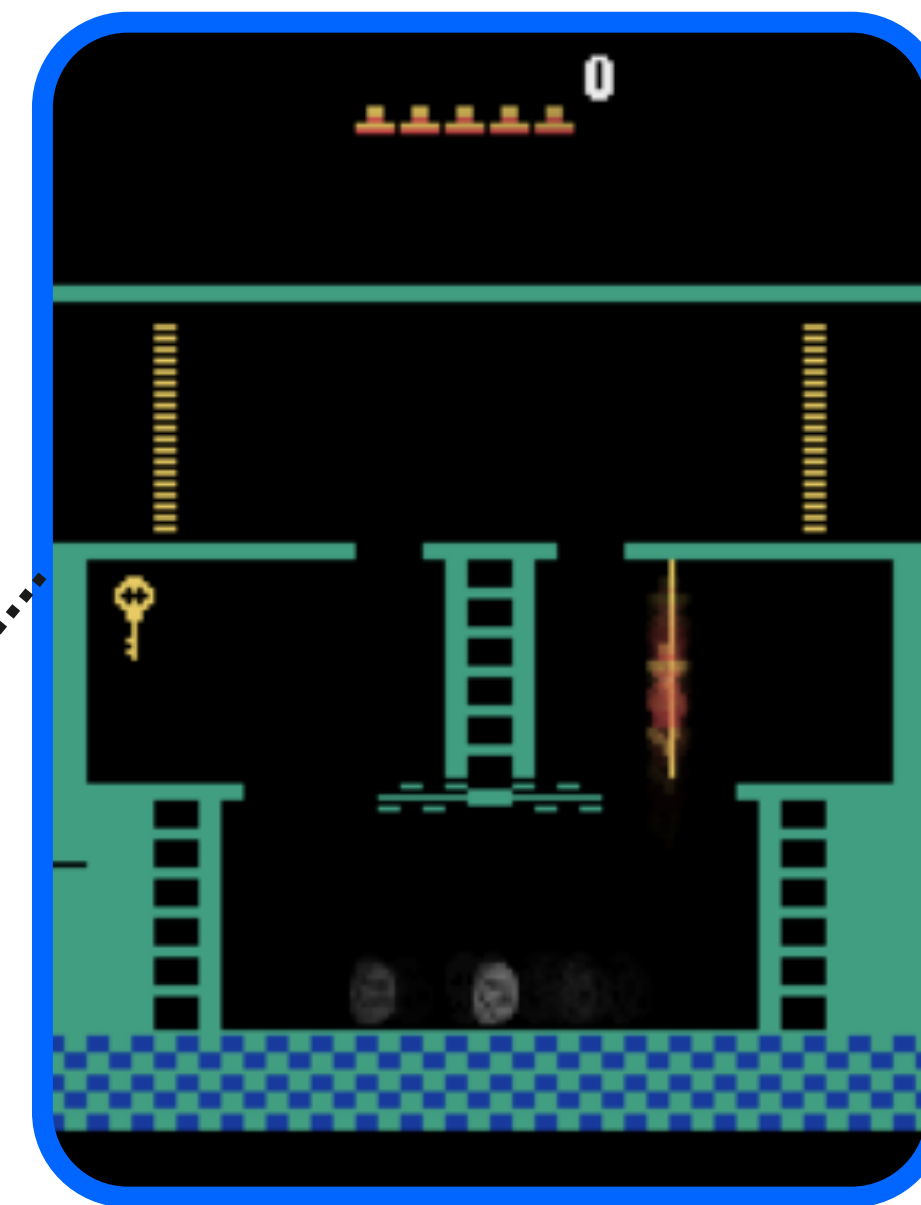


JumpRight

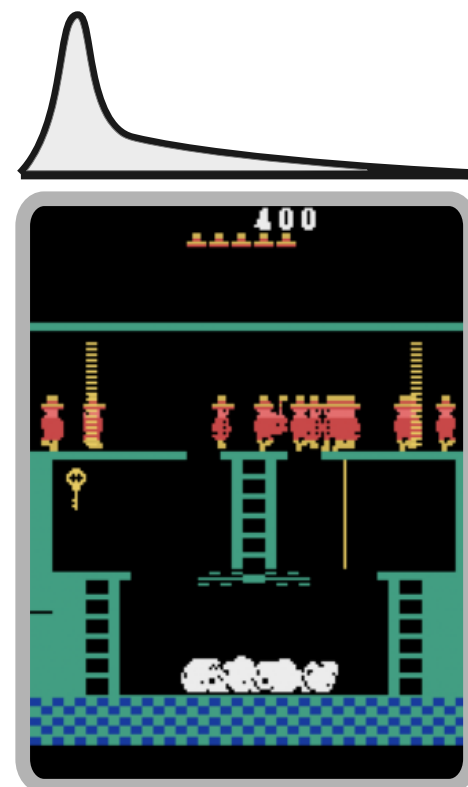
s_{371}



s_5

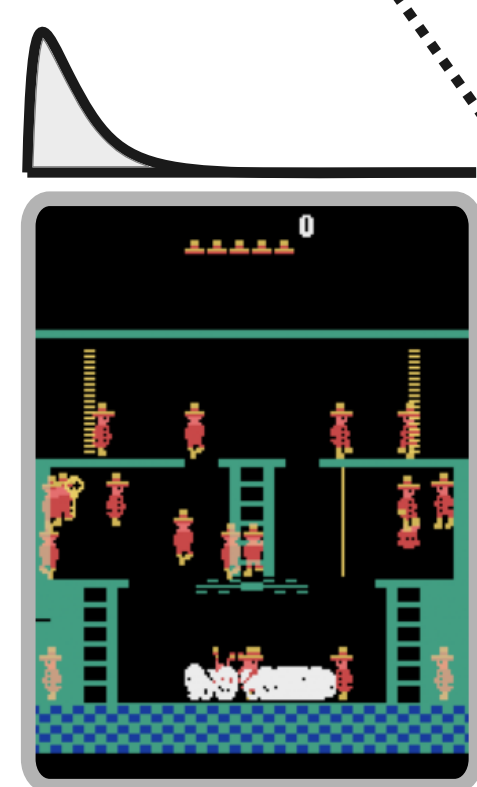


$\times$



$\times$

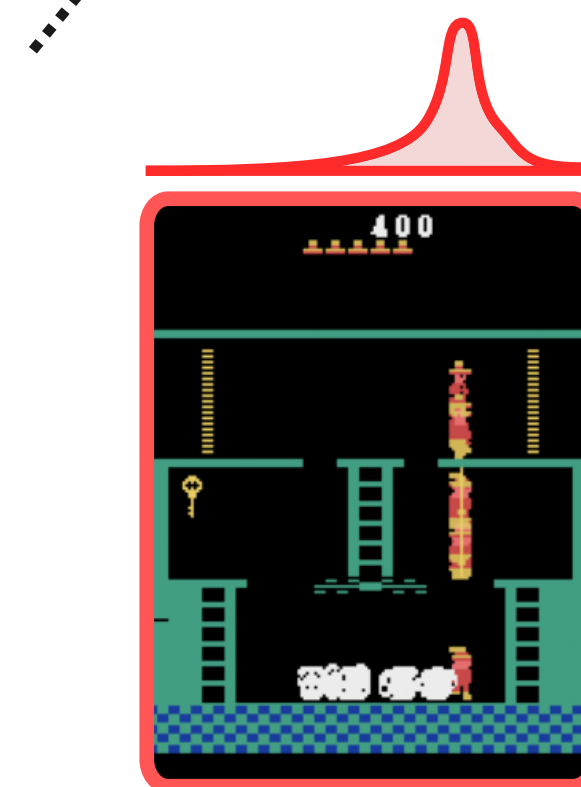
$\dots \times$



$e_{x,12}$

$e_{y,3}$

$e_{score,4}$

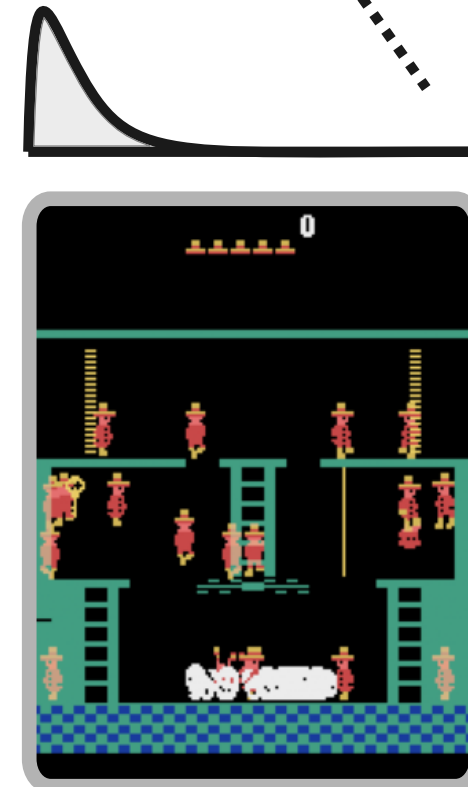


$\times$



$\times$

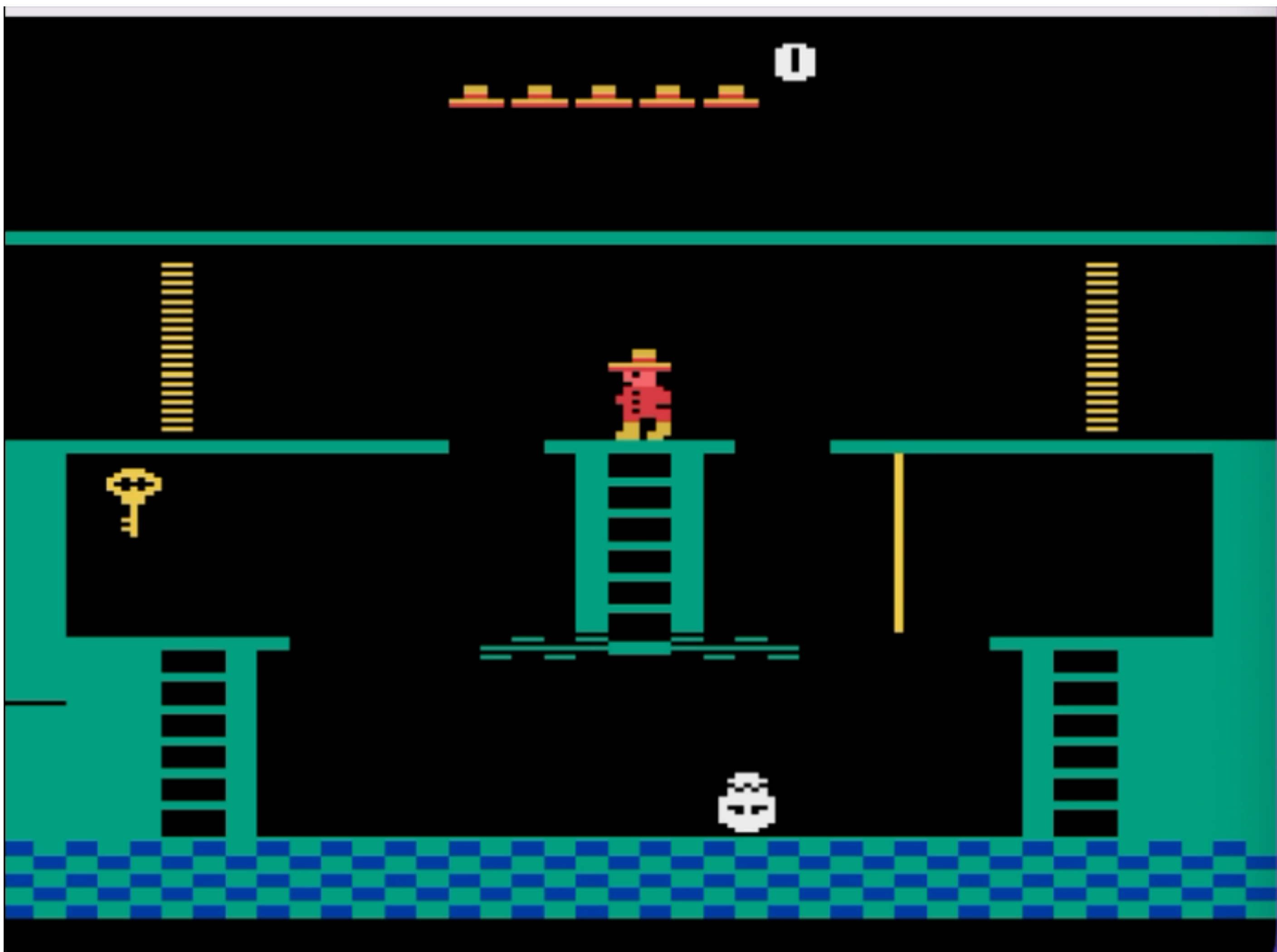
$\dots \times$



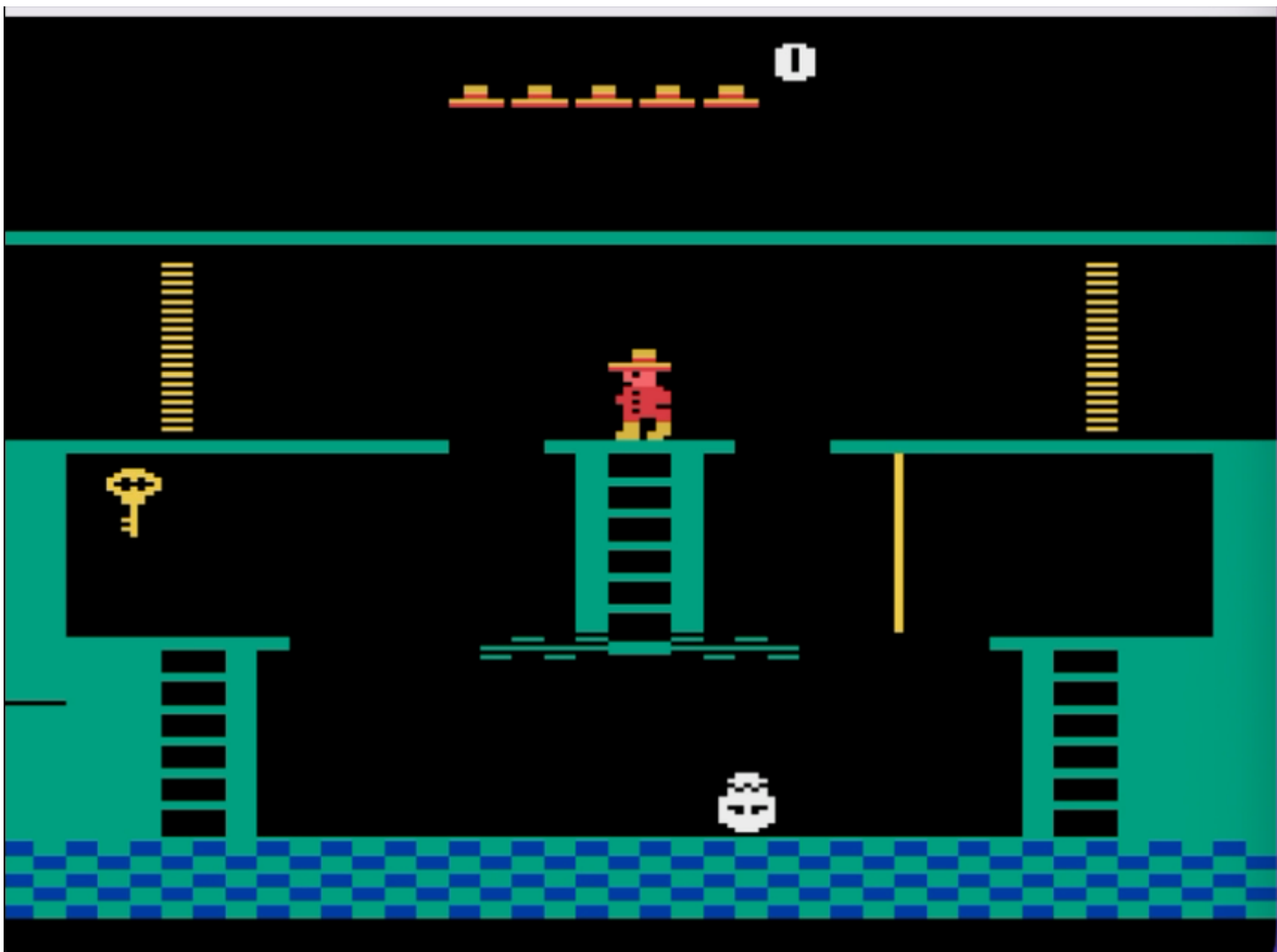
$e_{x,2}$

$e_{y,3}$

$e_{score,4}$



```
(s2s) → skills-to-hierarchies git:(incrementa  
1) × python run_monty.py  
A.L.E: Arcade Learning Environment (version 0.  
7.5+db37282)  
[Powered by Stella]  
Q(1304, JumpRightSkill)=301.42609  
□
```



```
(s2s) → skills-to-hierarchies git:(incrementa  
1) × python run_monty.py  
A.L.E: Arcade Learning Environment (version 0.  
7.5+db37282)  
[Powered by Stella]  
Q(1304, JumpRightSkill)=301.42609  
□
```

Conclusion

- A principled framework targeting a specific computation: Bellman.
- Provably (approximately) model-preserving.
- Plugged-in with a neural component to process high dimensions.