SCALING PROBABILISTICALLY SAFE LEARNING TO ROBOTICS

Scott Niekum

Assistant Professor, Department of Computer Science
The University of Texas at Austin

TEXAS
The University of Texas at Austin

PeARL
Personal Autonomous Robotics Lab









Safety and Correctness in Robotics

What does it mean for a learning agent to be "safe"?

- **Formal safety:** A self-driving car that will provably never crash if some model holds
- **Risk-sensitive safety:** A stock market agent with bounded value-at-risk
- **Robust safety:** An image classifier resistant to data poisoning or adversarial examples
- **Monotonic safety:** An RL-based advertising policy that always improves with high probability
- **Safe exploration:** A walking robot that can explore new gaits without falling over

More complete taxonomy: D. Amodei, C. Olah, J. Steinhardt, P. Christiano, J. Schulman, and D. Mané. "Concrete problems in AI safety."

A proposed definition of safety:

Safety = Correctness + Confidence

Correctness: Meeting or exceeding a measure of performance

Confidence: A (probabilistic) guarantee of correctness

A spectrum of safety

Guaranteed ⟷ Probabilistic ⟷ Approximate

Require perfect models

Verification / synthesis

[Kress-Gazit et. al 2009]
[Raman et. al 2015]

Sample inefficient

PAC-MDP methods

[Singh et. al 2002]
[Fu and Topcu 2014]

Concentration inequalities

[Thomas et. al 2015]
[Bottou et. al 2013]
[Abbeel and Ng 2004]
[Syed and Schapire 2008]

No guarantees

KL-divergence constraints

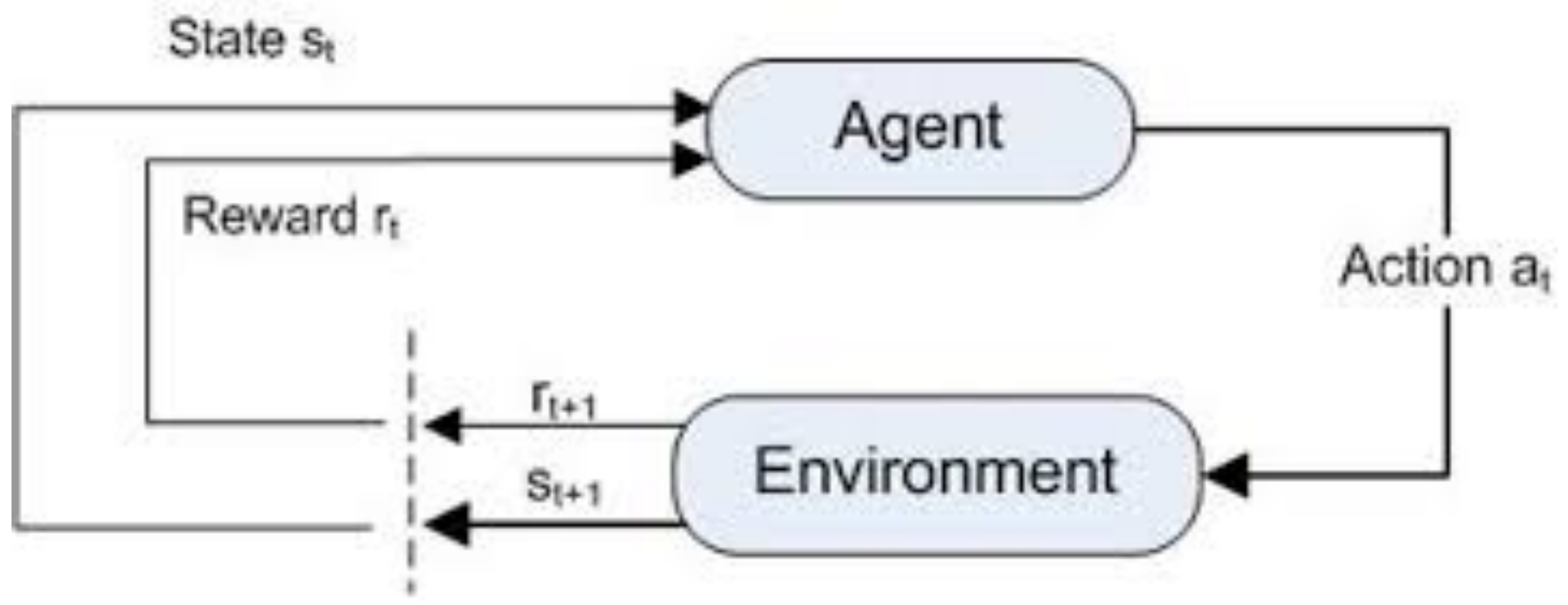
[Schulman et. al 2015]
[Schulman et. al 2017]
[Peters et. al 2010]

Address bad assumptions!

Part 1: Safe reinforcement learning

Part 2: Safe imitation learning

Background



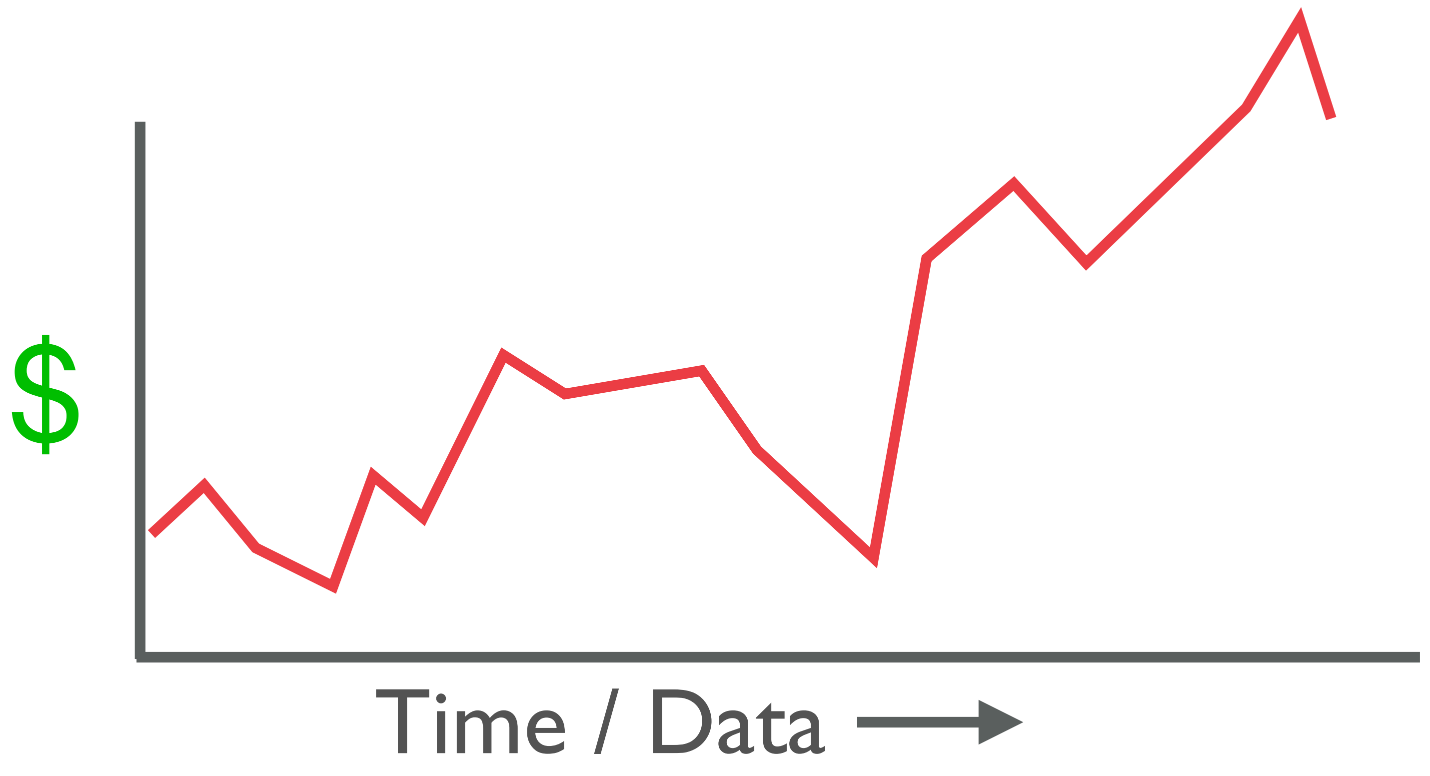
- Finite-horizon MDP.
- Agent selects actions with a *stochastic* policy, π.
- The policy and environment determine a distribution over trajectories, $H : S_0, A_0, R_0, S_1, A_1, R_1, ..., S_L, A_L, R_L$

Safe off-policy evaluation (OPE):

Determine a **probabilistic lower bound** on **expected performance** of a policy, given data generated by a **different policy**

Safe policy improvement (PI):

Ensure that expected performance **improves monotonically** at every learning step with **high confidence**



Policy Evaluation

Policy performance:

$$V(\pi) = \mathbb{E}\left[\sum_{t=0}^{L} \gamma^t R_t \middle| H \sim \pi\right]$$

Given a target policy, π_e, estimate $V(\pi_e)$

- Let $\pi_e \equiv \pi_{\boldsymbol{\theta}_e}$

Monte Carlo Policy Evaluation

Given a dataset $\mathcal{D}$ of trajectories where $\forall H \in \mathcal{D}$, $H \sim \pi_e$:

$$\mathrm{MC}(\mathcal{D}) := \frac{1}{|\mathcal{D}|} \sum_{H_i \in \mathcal{D}} \sum_{t=0}^{L} \gamma^t R_t^{(i)}$$

Importance Sampling Policy Evaluation[1]

Given a dataset $\mathcal{D}$ of trajectories where $\forall H_i \in \mathcal{D}$, H_i is sampled from a behavior policy π_i:

$$\text{IS}(\mathcal{D}) := \frac{1}{|\mathcal{D}|} \sum_{H_i \in \mathcal{D}} \underbrace{\prod_{t=0}^{L} \frac{\pi_e(A_t|S_t)}{\pi_i(A_t|S_t)}}_{\text{re-weighting factor}} \sum_{t=0}^{L} \gamma^t R_t^{(i)}$$

For convenience:

$$\text{IS}(H, \pi) := \prod_{t=0}^{L} \frac{\pi_e(A_t|S_t)}{\pi(A_t|S_t)} \sum_{t=0}^{L} \gamma^t R_t$$

[1] Precup, Sutton, and Singh (2000)

Confidence Intervals for Off-Policy Evaluation

Given:

- Trajectories generated by a *behavior* policy, π_b, $\{H, \pi_b\} \in \mathcal{D}$.
- An *evaluation* policy, π_e.
- $\delta \in [0, 1]$ is a confidence level.

Determine a lower bound $\hat{V}_{\text{lb}}(\pi_e, \mathcal{D})$ such that $V(\pi_e) \geq \hat{V}_{\text{lb}}(\pi_e, \mathcal{D})$ with probability $1 - \delta$.

Concentration Inequalities

Chernoff-Hoeffding Inequality

- Probabilistic bound on how a random variable deviates from its expectation
- No distributional assumptions
- With probability at least 1-δ:

$$\mu \geq \frac{1}{n}\sum_{i=1}^{n} X_i - b\sqrt{\frac{\log(1/\delta)}{2n}}$$

- Can use with importance sampled returns to bound value of a policy from off-policy samples
- Significantly tighter bounds exist under certain conditions (Thomas et. al 2015)

Sample (in)efficiency (Thomas et. al 2015)

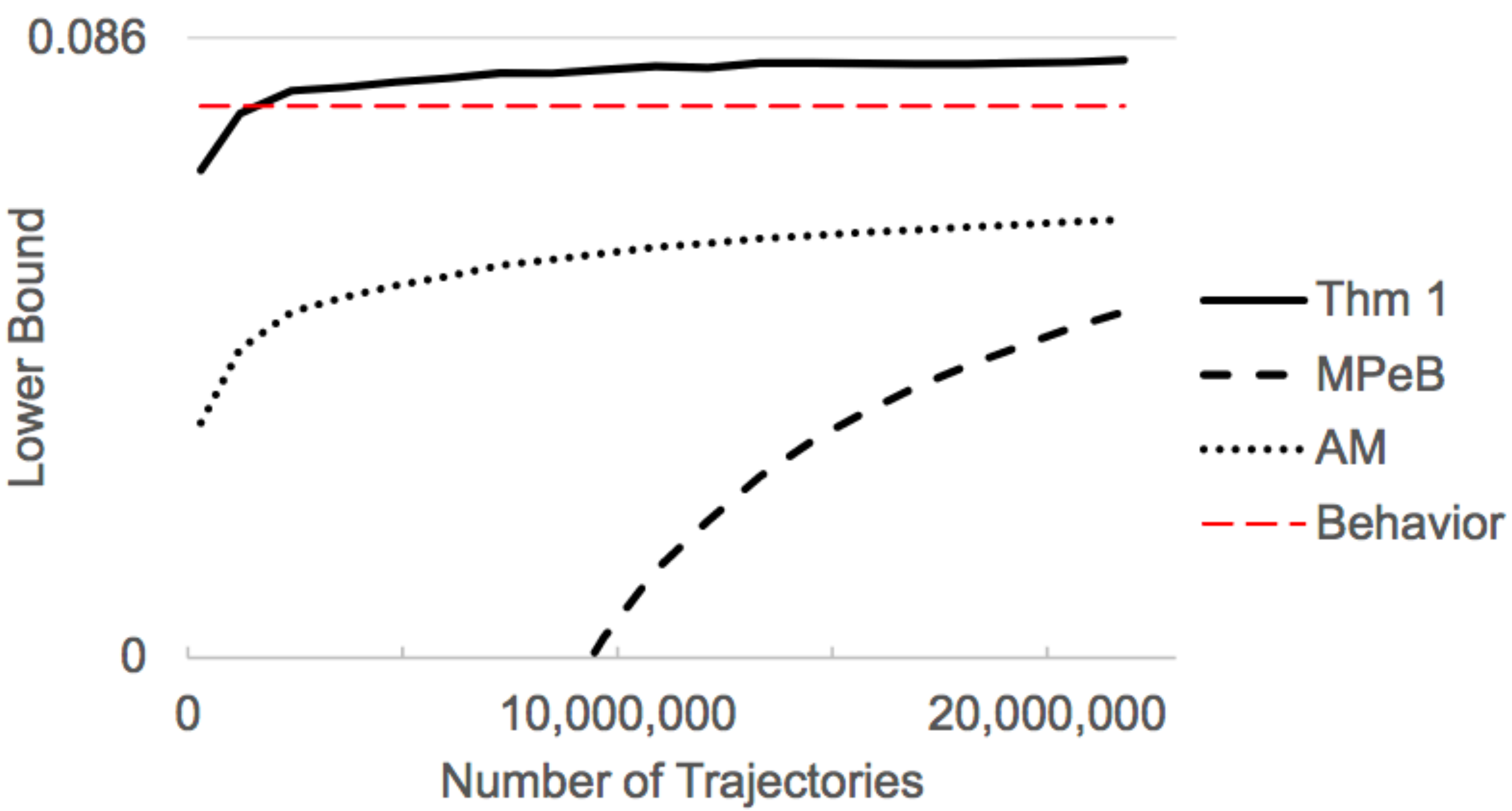


Figure 3: 95% confidence lower bound (unnormalized) on $\rho(\theta)$ using trajectories generated using the simulator described in the text. The behavior policy's true expected re-

Bad assumption #1:

"When performing policy evaluation, it is better to collect on-policy data than off-policy data"

J.P. Hanna, P.S. Thomas, P. Stone, and S. Niekum.
[Data-Efficient Policy Evaluation Through Behavior Policy Search]().
Proceedings of the 34th International Conference on Machine Learning (ICML), August 2017.

Optimal Behavior Policy

Claim: There exists an optimal behavior policy, $\pi_{b^\star}$, if all returns are positive and transitions are deterministic:

Optimal Behavior Policy

Claim: There exists an optimal behavior policy, $\pi_{b^\star}$, if all returns are positive and transitions are deterministic:

$$V(\pi_e) = g(H) \prod_{t=0}^{L} \frac{\pi_e(A_t|S_t)}{\pi_{b^\star}(A_t|S_t)}$$

$$\prod_{t=0}^{L} \pi_{b^\star}(A_t|S_t) = \frac{g(H)}{V(\pi_e)} \prod_{t=0}^{L} \pi_e(A_t|S_t)$$

$$w_{\pi_{b^\star}}(H) = \frac{g(H)}{V(\pi_e)} w_{\pi_e}(H)$$

Zero mean squared error with a single trajectory! Such a policy provably exists as a mixture over time-dependent deterministic policies (i.e. weighted trajectories).

Optimal Behavior Policy

Unfortunately, the optimal behavior policy is unknown in practice.

$$\prod_{t=0}^{L} \pi_{b^\star}(A_t|S_t) = \frac{g(H)}{V(\pi_e)} \prod_{t=0}^{L} \pi_e(A_t|S_t)$$

- Requires $V(\pi_e)$ be known!
- Requires the reward function be known.
- Requires deterministic transitions.

Behavior Policy Gradient

Key Idea: Adapt the behavior policy parameters, $\boldsymbol{\theta}$, with gradient descent on the mean squared error of importance-sampling.

$$\boldsymbol{\theta}_{i+1} = \boldsymbol{\theta}_i - \alpha \frac{\partial}{\partial \boldsymbol{\theta}} \mathrm{MSE}[\mathrm{IS}(H_i, \boldsymbol{\theta})]$$

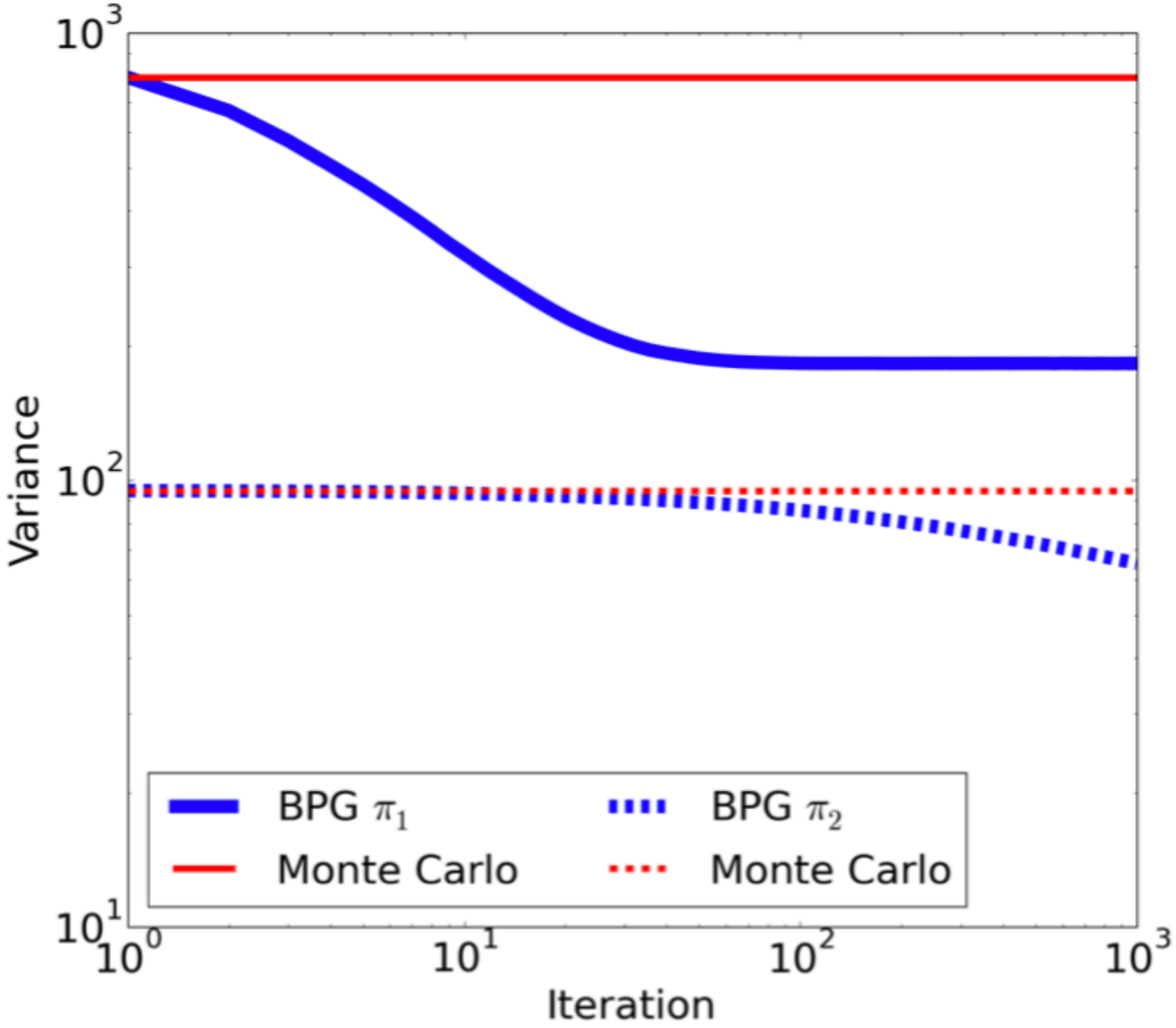
- $\mathrm{MSE}[IS(H, \boldsymbol{\theta})]$ is **not** computable.
- $\frac{\partial}{\partial \boldsymbol{\theta}} \mathrm{MSE}[IS(H, \boldsymbol{\theta})]$ is computable.

Behavior Policy Gradient Theorem

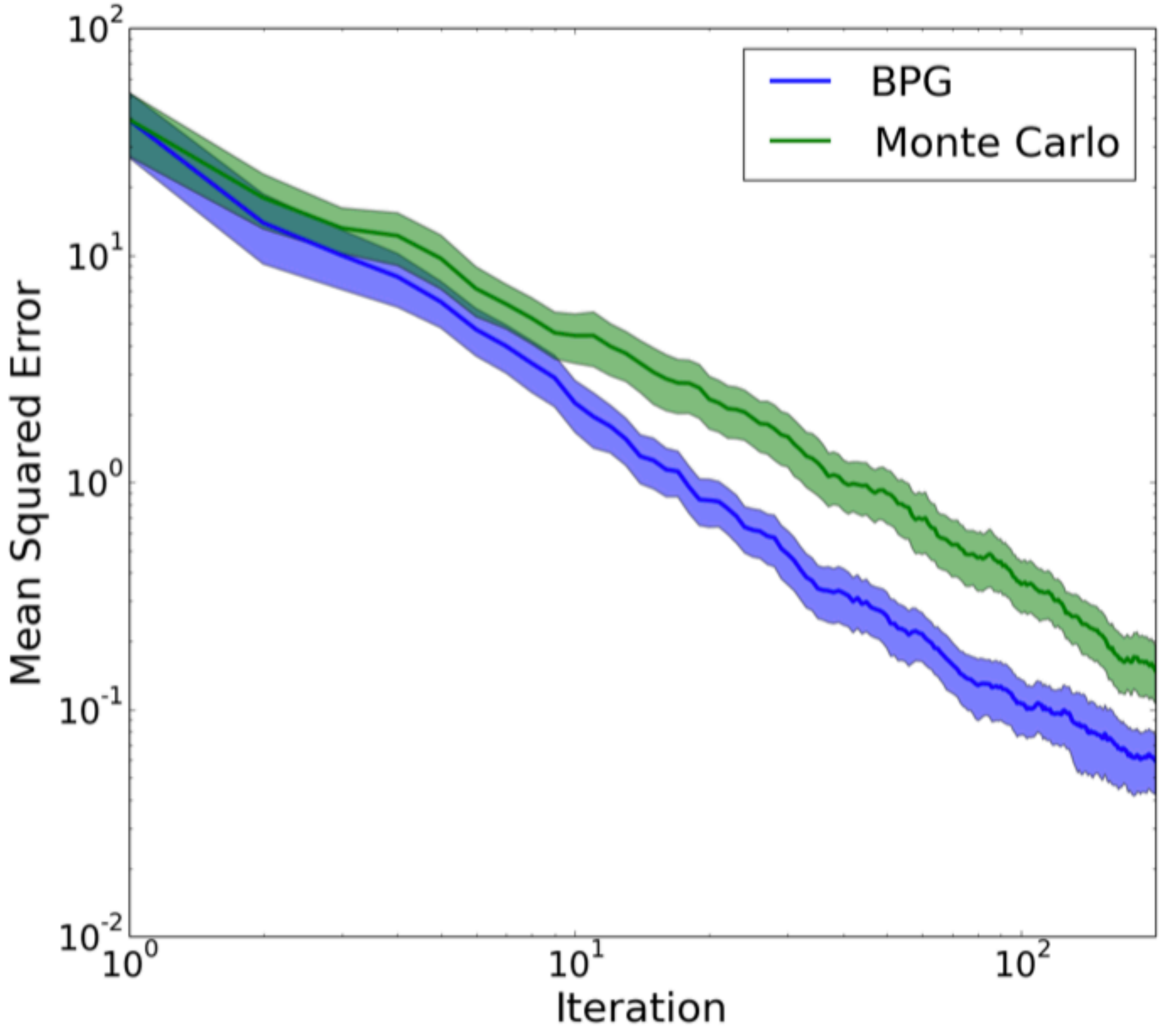
Theorem

$$\frac{\partial}{\partial \boldsymbol{\theta}} \mathrm{MSE}(\mathrm{IS}(H, \boldsymbol{\theta})) = \mathbf{E}_{\pi_{\boldsymbol{\theta}}}\left[- \mathrm{IS}(H, \boldsymbol{\theta})^2 \sum_{t=0}^{L} \frac{\partial}{\partial \boldsymbol{\theta}} \log\left(\pi_{\boldsymbol{\theta}}(A_t | S_t)\right)\right]$$

Variance reduction



Improved sample efficiency

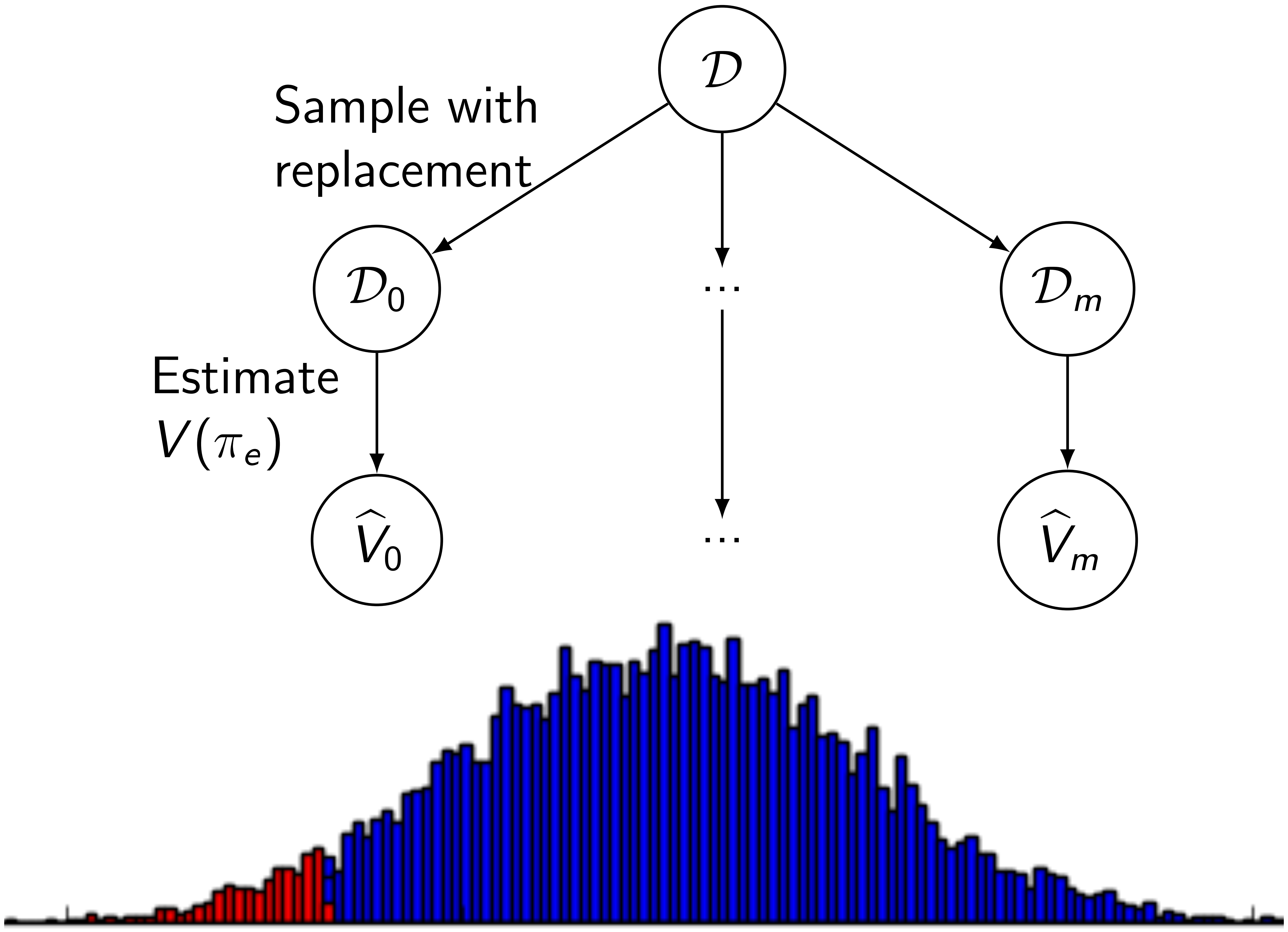


Cartpole Swing-up

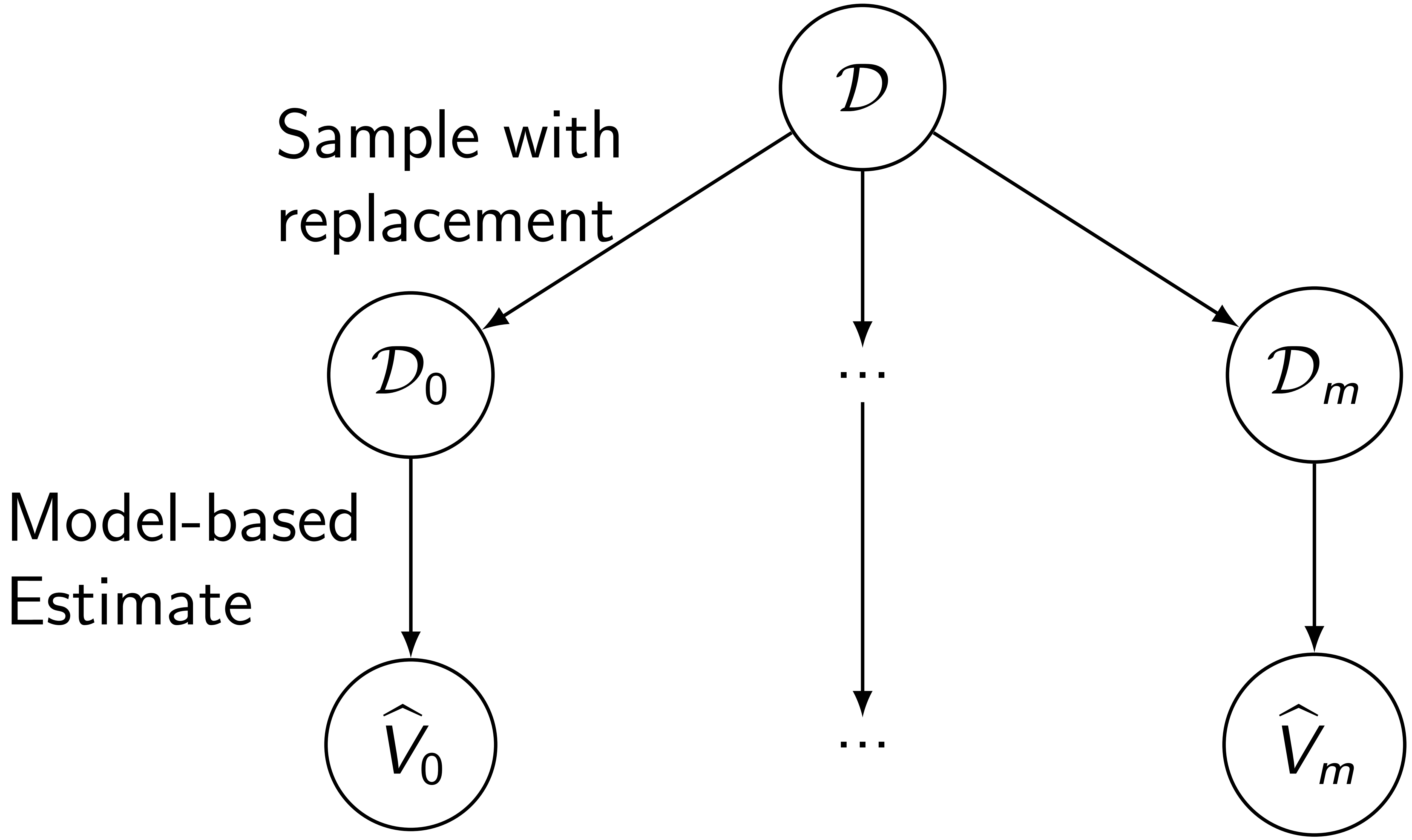
Better, but not good enough.

- Are “semi-safe”, consistent methods good enough? (e.g. bootstrapping)
- Why only use model-free methods?

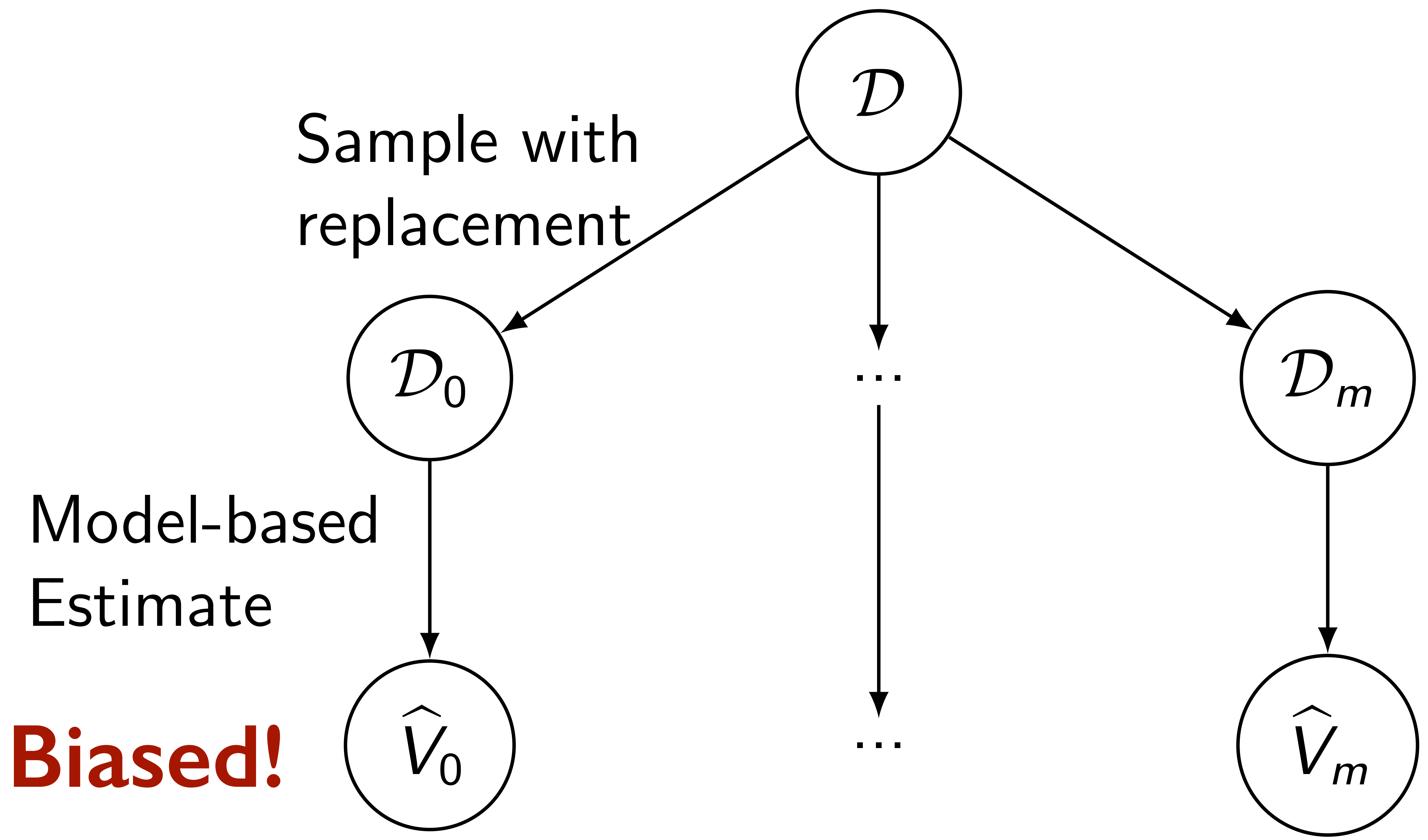
Bootstrap Confidence Intervals



Model-Based Bootstrap



Model-Based Bootstrap



Bad assumption #2:

"Biased models lead to biased estimators"

J.P. Hanna, P. Stone, and S. Niekum.
Bootstrapping with Models: Confidence Intervals for Off-Policy Evaluation.
International Conference on Autonomous Agents and Multiagent Systems (AAMAS), May 2017.

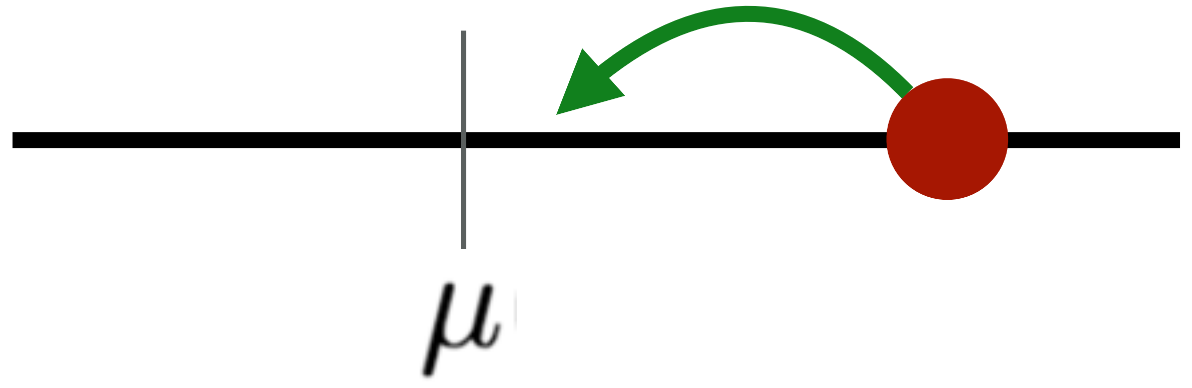
Doubly Robust Estimator

[Jiang and Li 2016; Thomas and Brunskill 2016]

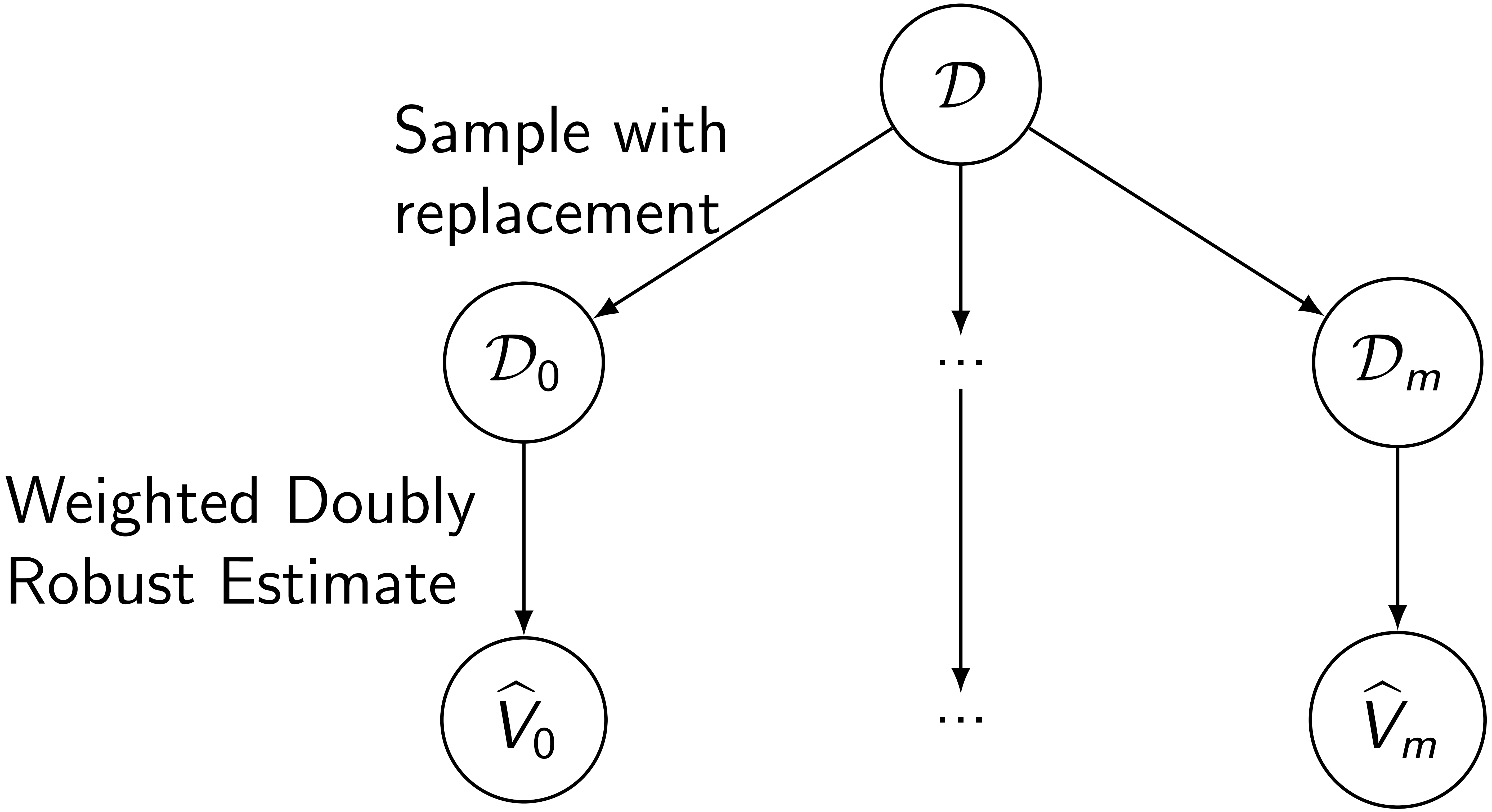
$$\mathrm{DR}(\mathcal{D}) := \underbrace{\mathrm{PDIS}(\mathcal{D})}_{\text{Unbiased estimator}} - \underbrace{\sum_{i=1}^{n}\sum_{t=0}^{L} w_t^i \hat{q}^{\pi_e}(S_t^i, A_t^i) - w_{t-1}^i \hat{v}^{\pi_e}(S_t^i)}_{\text{Zero in Expectation}}$$

Control variate

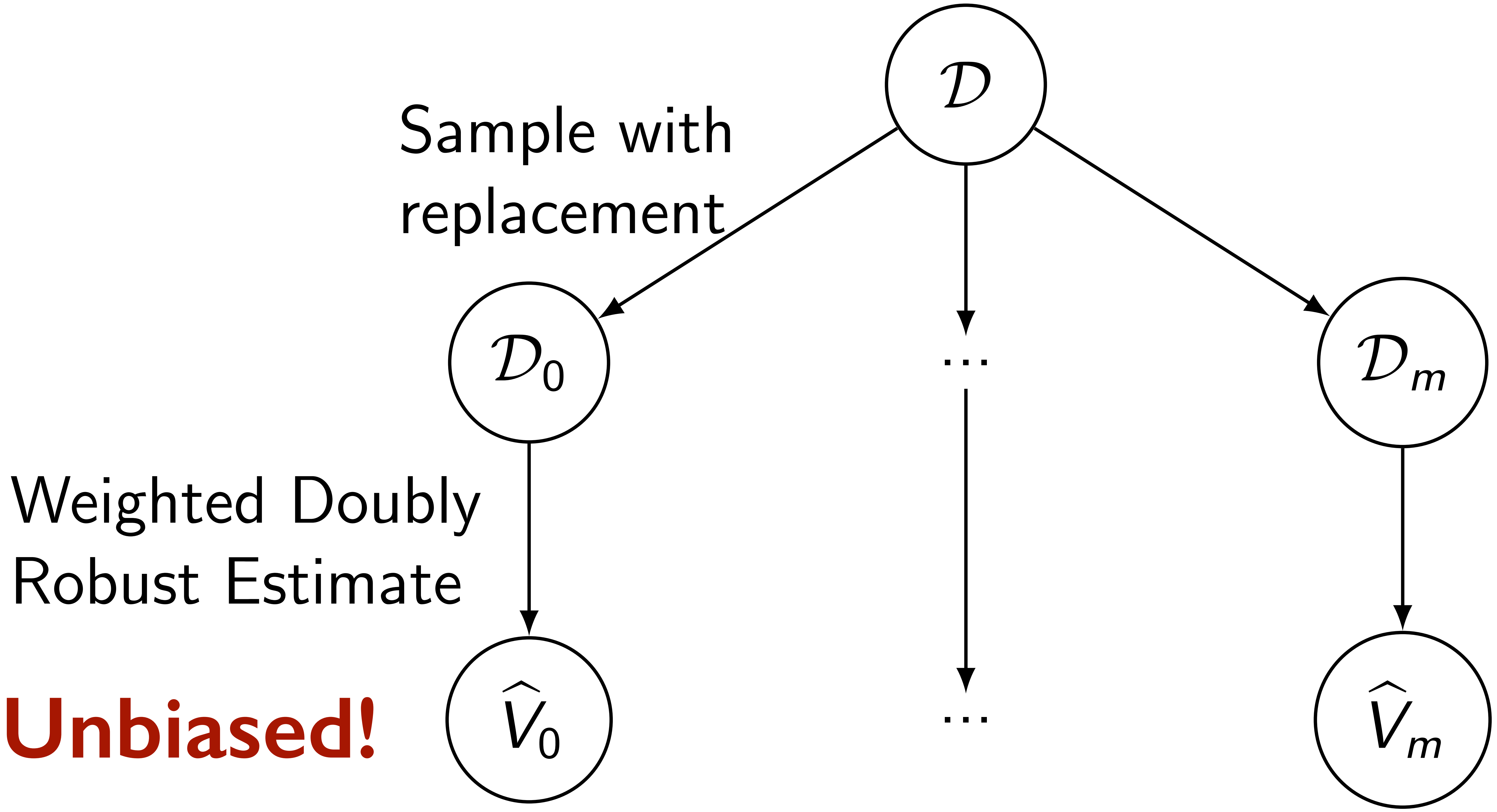
- $\hat{v}^{\pi}(S) := \mathbb{E}_{A\sim\pi, S'\sim\hat{P}(\cdot|S,A)}\left[r(S,A) + \hat{v}(S')\right]$
  - State value function.
- $\hat{q}^{\pi}(S,A) := r(S,A) + \mathbb{E}_{S'\sim P(\cdot|S,A)}\left[\hat{v}(S')\right]$
  - State-action value function.
- w_t is the importance weight of the first t time-steps.



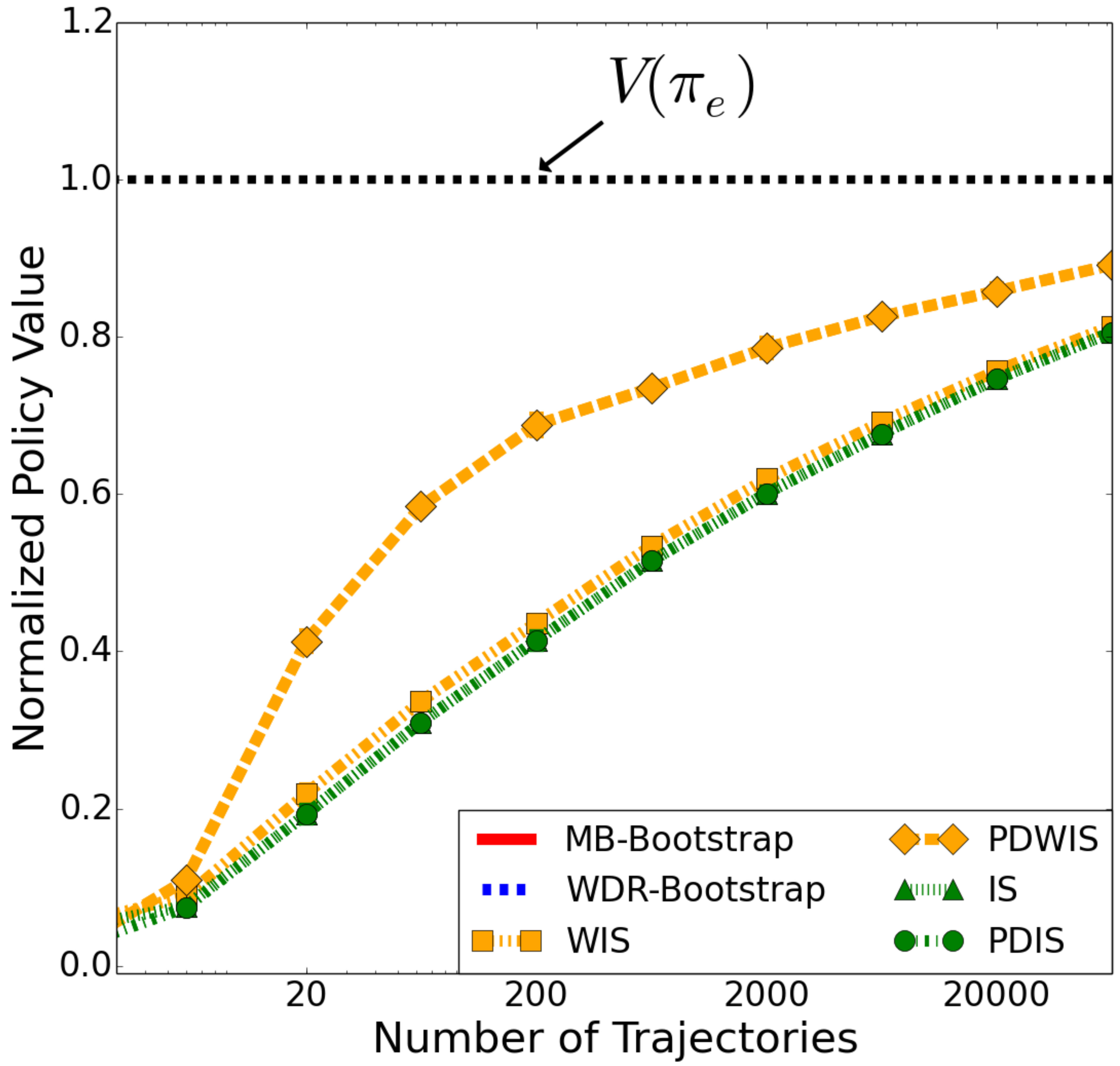
Weighted Doubly Robust Bootstrap



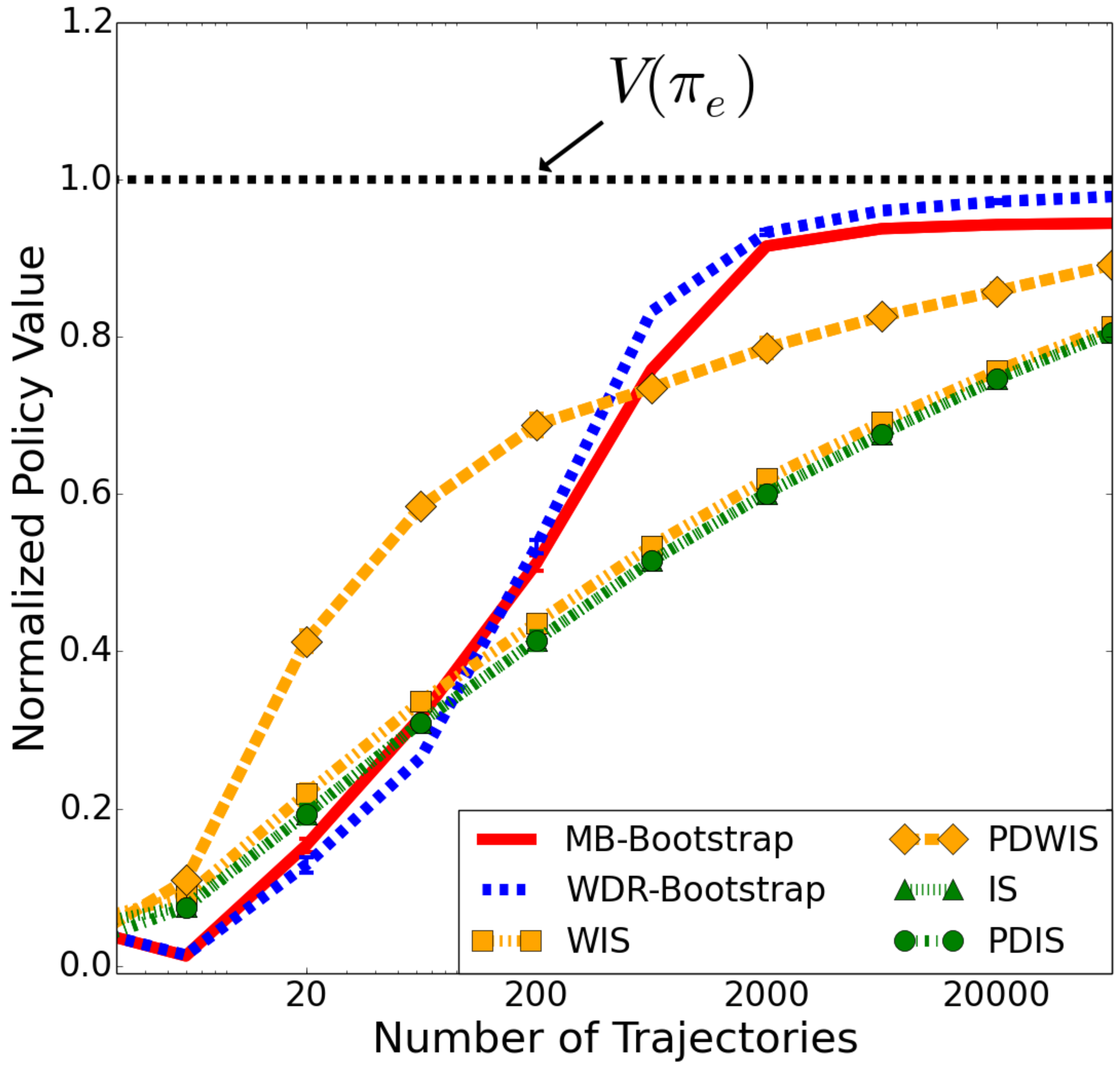
Weighted Doubly Robust Bootstrap



Mountain Car Results



Mountain Car Results



Similar ideas apply to **safe policy improvement:**

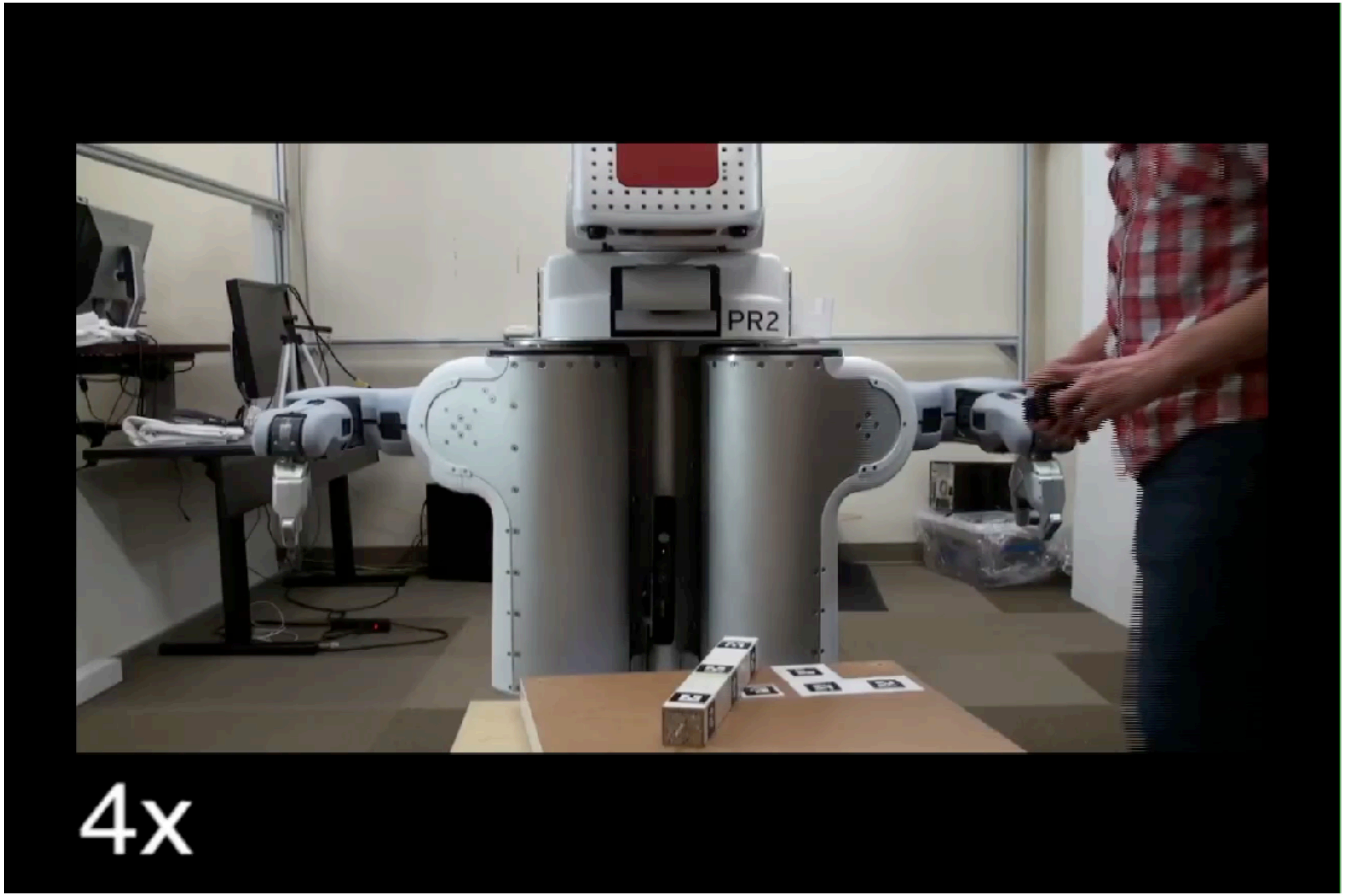
Loop:

1. Propose a policy (e.g. via an unsafe RL step)
2. Perform safe policy evaluation
3. Accept or reject

Part 1: Safe reinforcement learning

Part 2: Safe imitation learning

Imitation learning

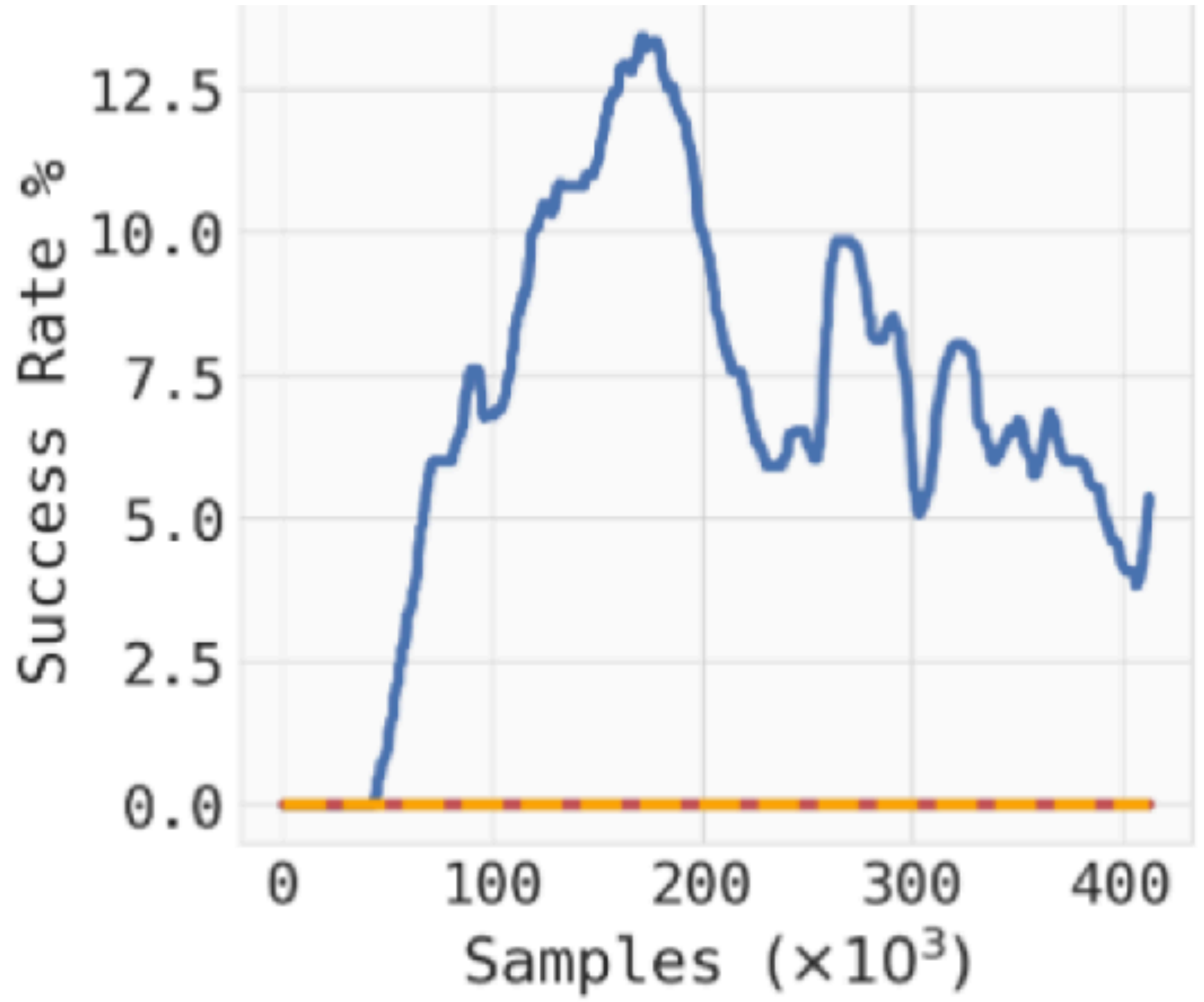


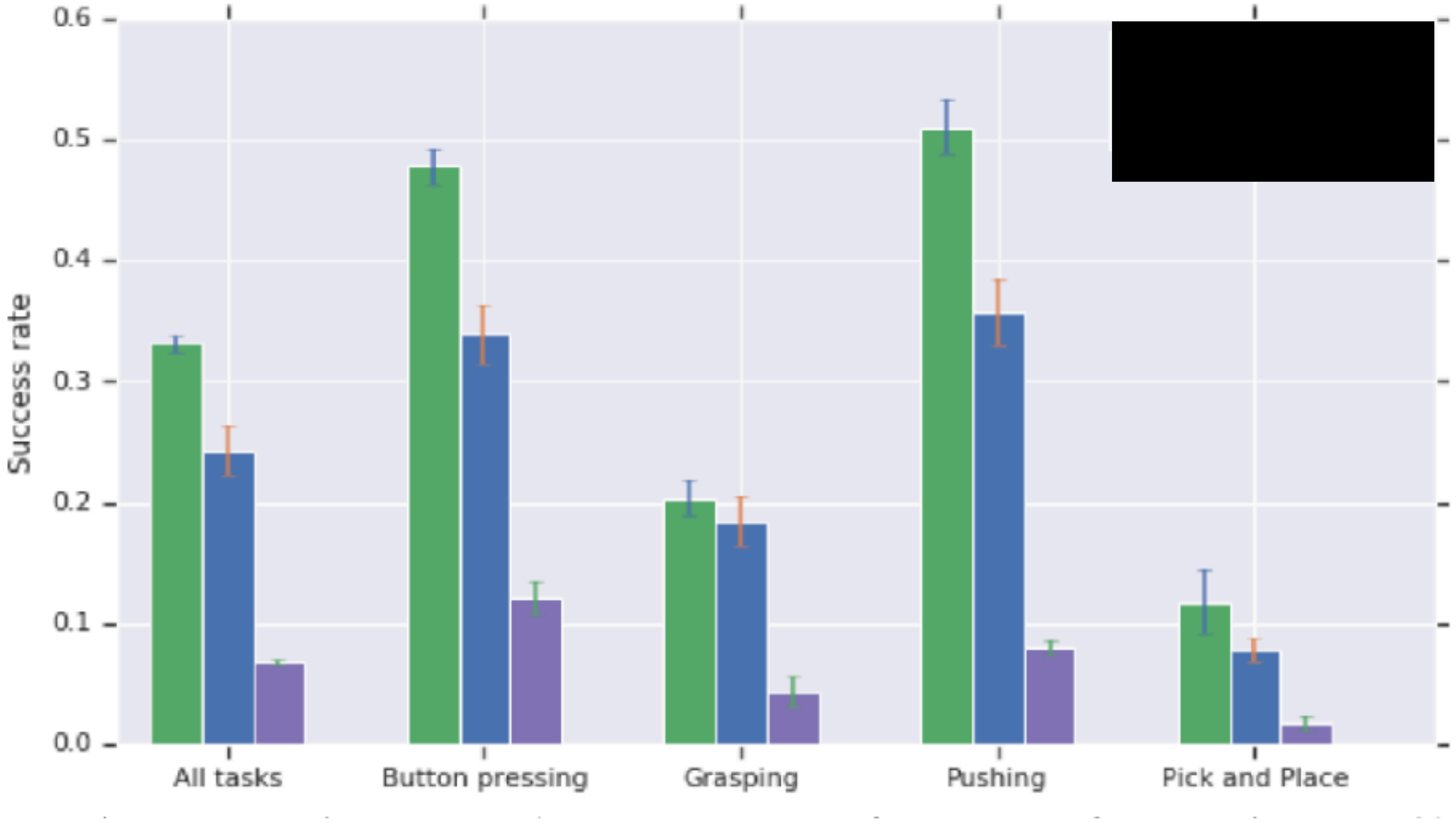
Never trust a robot video!

	ASM	FSA-basic	FSA-split	FSA-IC
Straight	Fail	Fail	1	0
	Fail	Fail	1	2
	Fail	Fail	2	2
	Fail	Fail	1	2
Far away	Fail	Fail	Fail	1
	Fail	Fail	Fail	1
	Fail	Fail	Fail	Fail
Difficult angle	Fail	Fail	2	1
	Fail	Fail	3	1
	Fail	Fail	3	2
Successes / Avg assists	0 / –	0 / –	7 / 1.857	9 / 1.333

Video

Surely things are better in 2021?



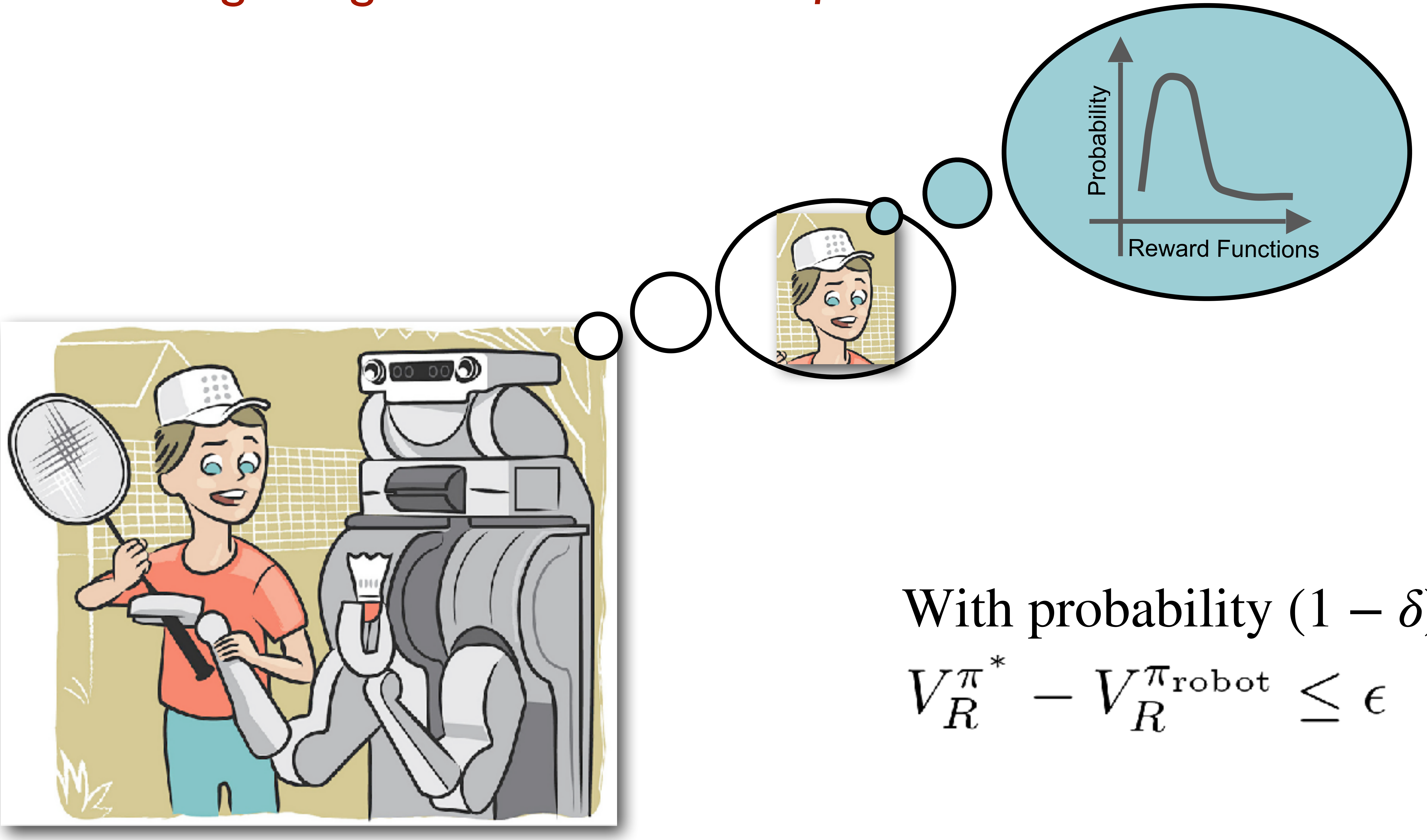


	[redacted]		
Success Rate (%)	**21.7**	8.8	7.6
Average Step Completion (of 4)	**2.4 ± 1.13**	2.2 ± 0.95	1.78 ± 1.0

Three sample robot imitation learning papers published 2019-2021

Value Aligned Imitation Learning (VAIL):

Upper bound the **policy loss** of the robot vs. human demonstrator with **high confidence**, *without knowing the ground-truth reward function.*



With probability $(1-\delta)$:

$$V_R^{\pi^*} - V_R^{\pi_{\text{robot}}} \leq \epsilon$$

Background: Inverse reinforcement learning
(Abbeel and Ng 2004)

Given: demonstrations $\tau_1 \ldots \tau_N$

Assume: reward function is linear in features of state: $R(s) = w \cdot \phi(s)$

Objective: find w such that $\tau_1 \ldots \tau_N$ are optimal under w

Value of a policy: $V_w^\pi = w \cdot \mu(\pi)$

where $\mu(\pi) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t \phi(s_t) | \pi\right]$ "Feature expectations"

Standard approach: find a reward function whose optimal policy matches expert's feature expectations

If expert's feature expectations are matched, then expected return is also identical

Background: performance bounds for IRL

(Abbeel and Ng 2004, Syed and Schapire 2008)

Theorem 2. *(Syed and Schapire 2008) To obtain a policy $\hat{\pi}$ such that with probability $(1 - \delta)$*

$$\epsilon \geq |V^{\hat{\pi}}(R^*) - V^{\pi^*}(R^*)| \tag{26}$$

it suffices to have

$$m \geq \frac{2}{(\frac{\epsilon}{3}(1-\gamma))^2} \log \frac{2k}{\delta}. \tag{27}$$

Worst-case bound that assumes an adversarial reward function

Standard assumption:

Worst-case reasoning is the best we can do if we don’t know the ground-truth reward function

It is much more efficient to consider the likelihood of reward functions when assessing risk

Existing tools that we'll need

(1) An IRL algorithm to obtain a **posterior distribution** over reward functions

(2) A metric for **measuring risk** with respect to distributions of outcomes

Background: Bayesian Inverse Reinforcement Learning

[Ramachandran and Amir 2007]

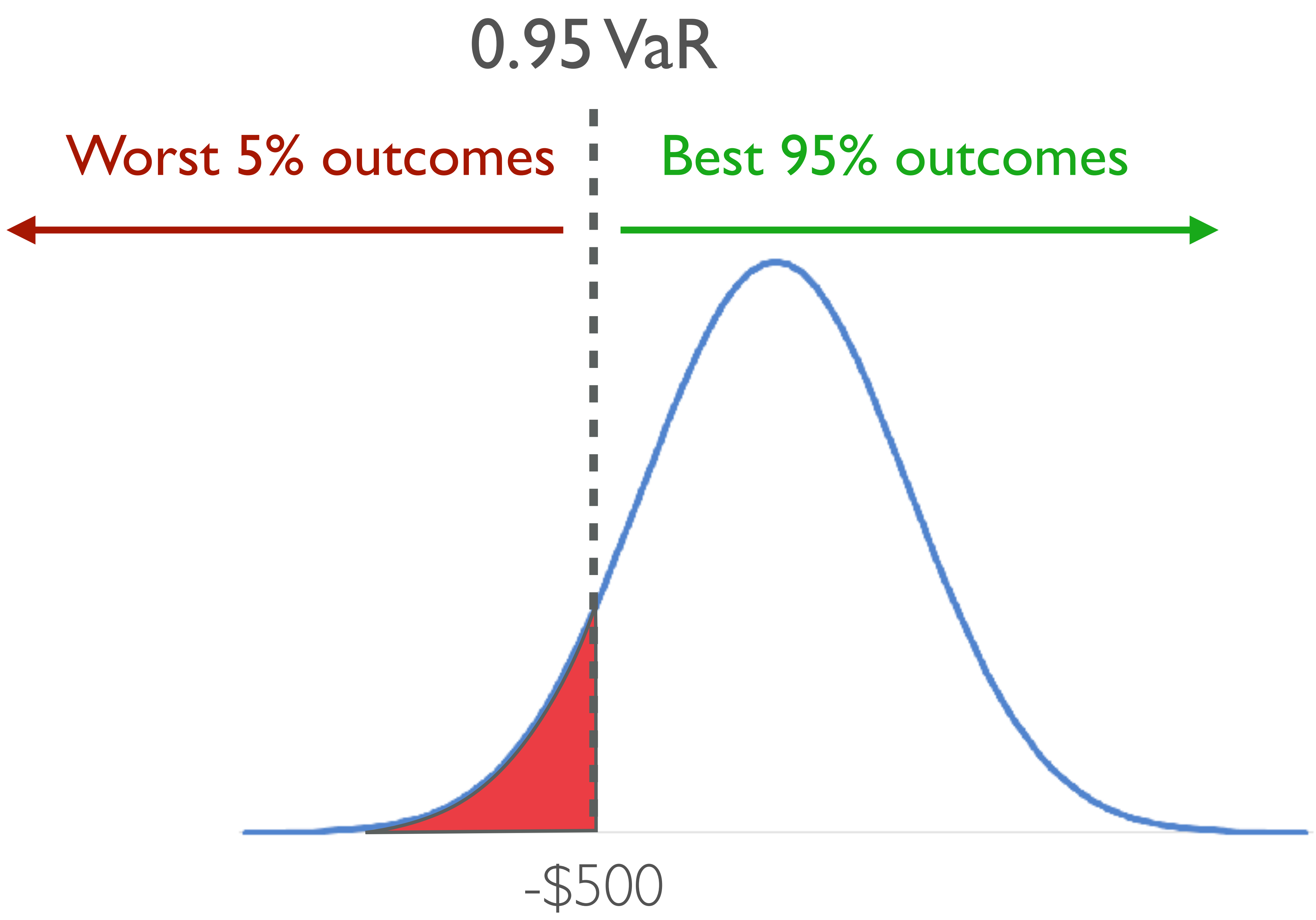
- Use MCMC to sample from posterior:

$$P(R|D) \propto P(D|R)P(R)$$

- Assume demonstrations follow softmax policy with temperature c:

$$P(D|R) = \prod_{(s,a)\in D} \frac{e^{cQ^*(s,a,R)}}{\sum_{b\in A} e^{cQ^*(s,b,R)}}$$

Background: α-value at risk

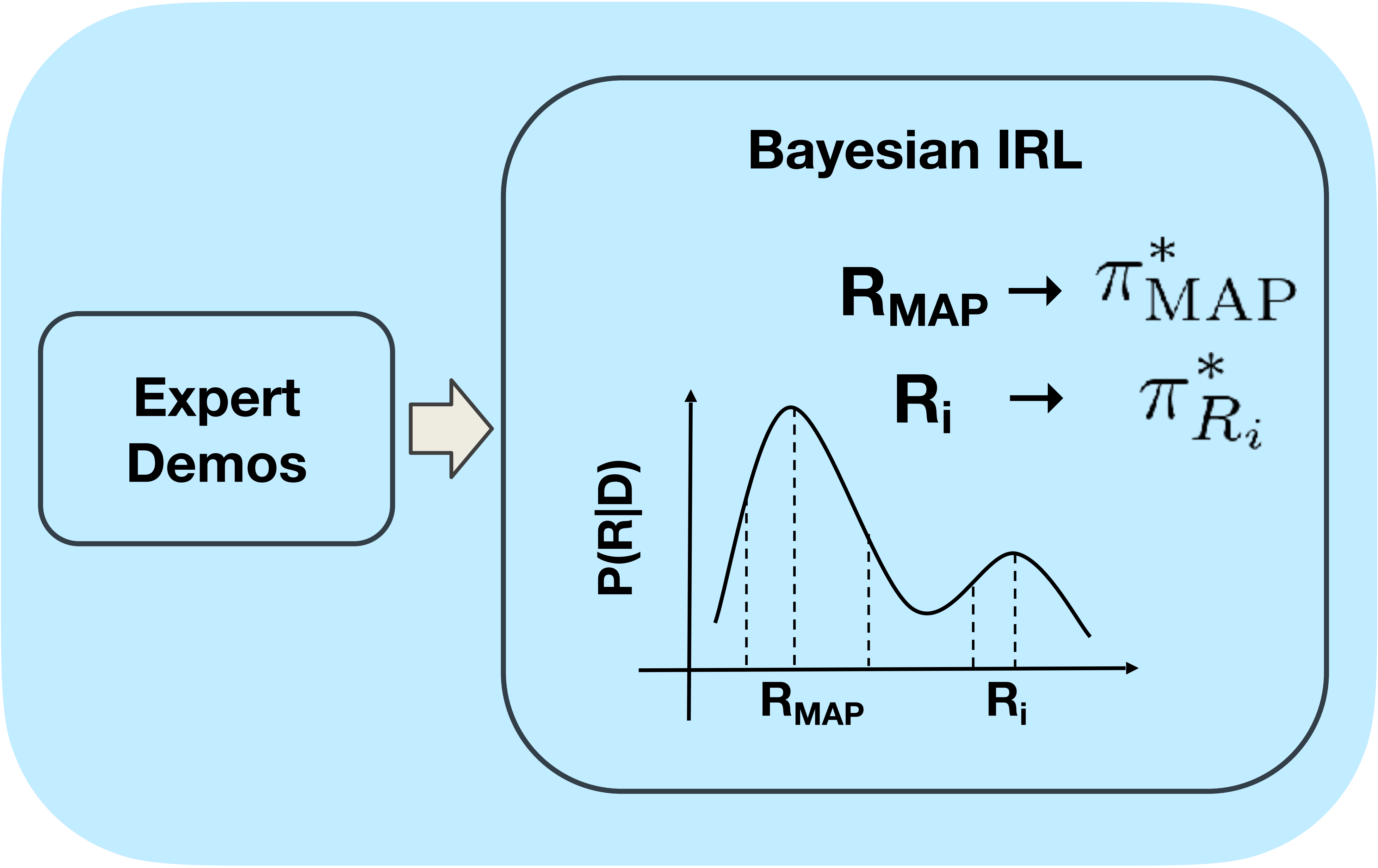


+ Single-sided confidence bound

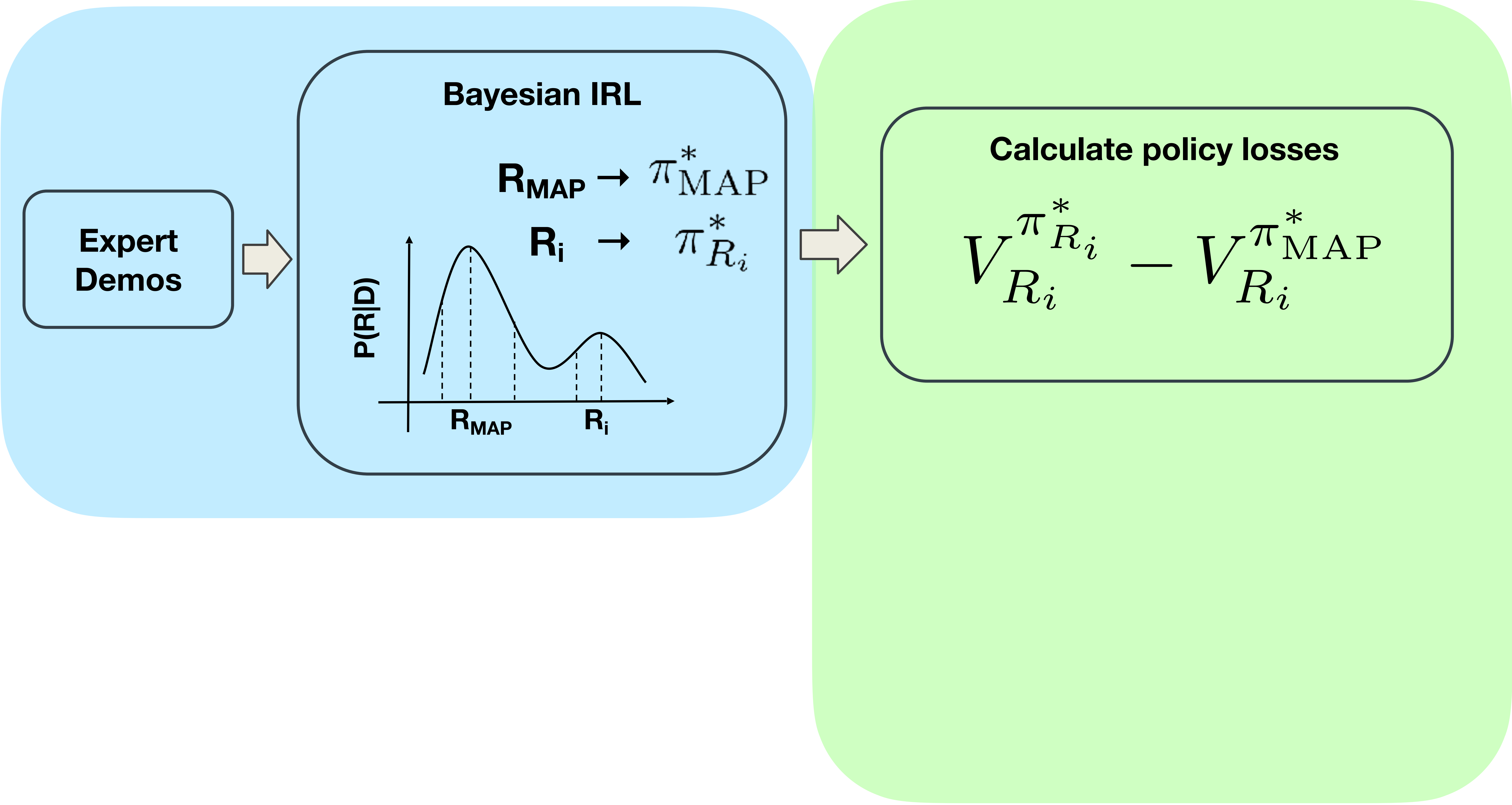
"With high confidence, you won't lose more than $500 more than 95% of the time when using this investing strategy"

Data efficient, intractable VAIL

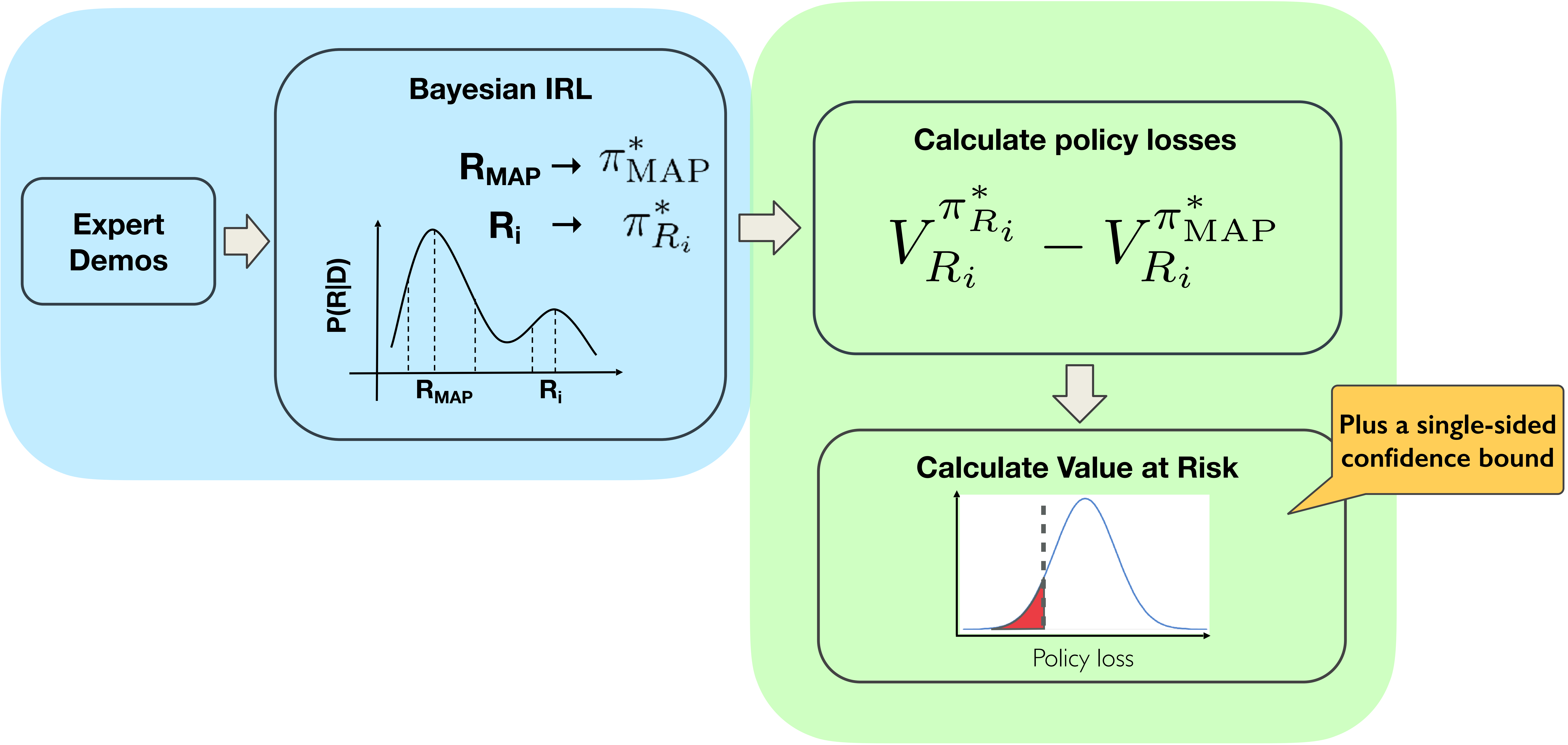
Data efficient, intractable VAIL



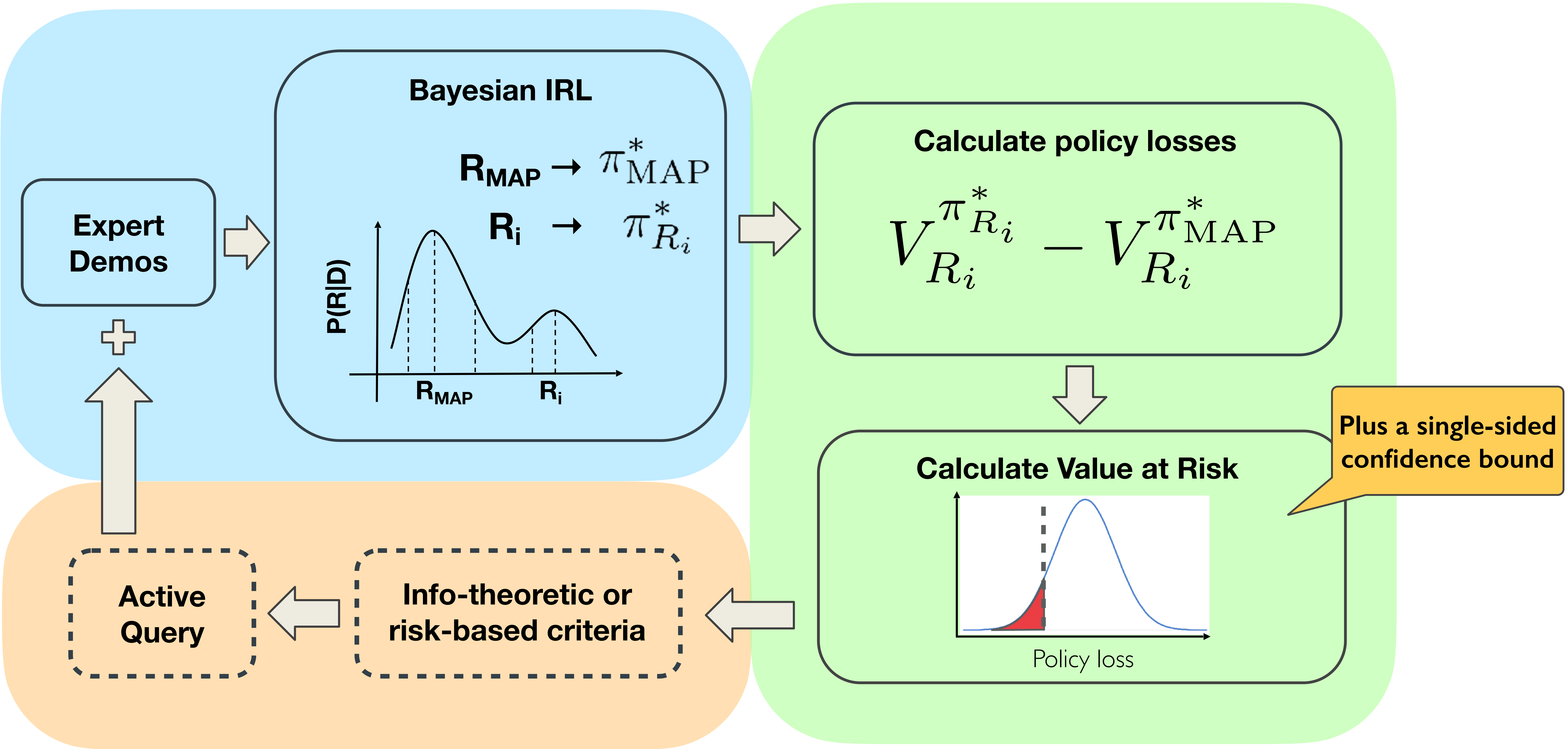
Data efficient, intractable VAIL



Data efficient, intractable VAIL



Data efficient, intractable VAIL



D.S. Brown, Y. Cui, and S. Niekum.
Risk-Aware Active Reinforcement Learning.
Conference on Robot Learning (CoRL), October 2018.

Y. Cui and S. Niekum.
Active Reward Learning from Critiques.
International Conference on Robotics and Automation (ICRA), May 2018.

Results: efficiency (no active learning)

	Number of demonstrations					Average Accuracy
	1	5	9	···	23,146	
0.95-VaR EVD Bound	**0.9372**	**0.2532**	**0.1328**		-	0.98
0.99-VaR EVD Bound	1.1428	0.2937	0.1535			1.0
EVD Bound (Syed and Schapire 2008)	142.59	63.77	47.53		0.9372	1.0

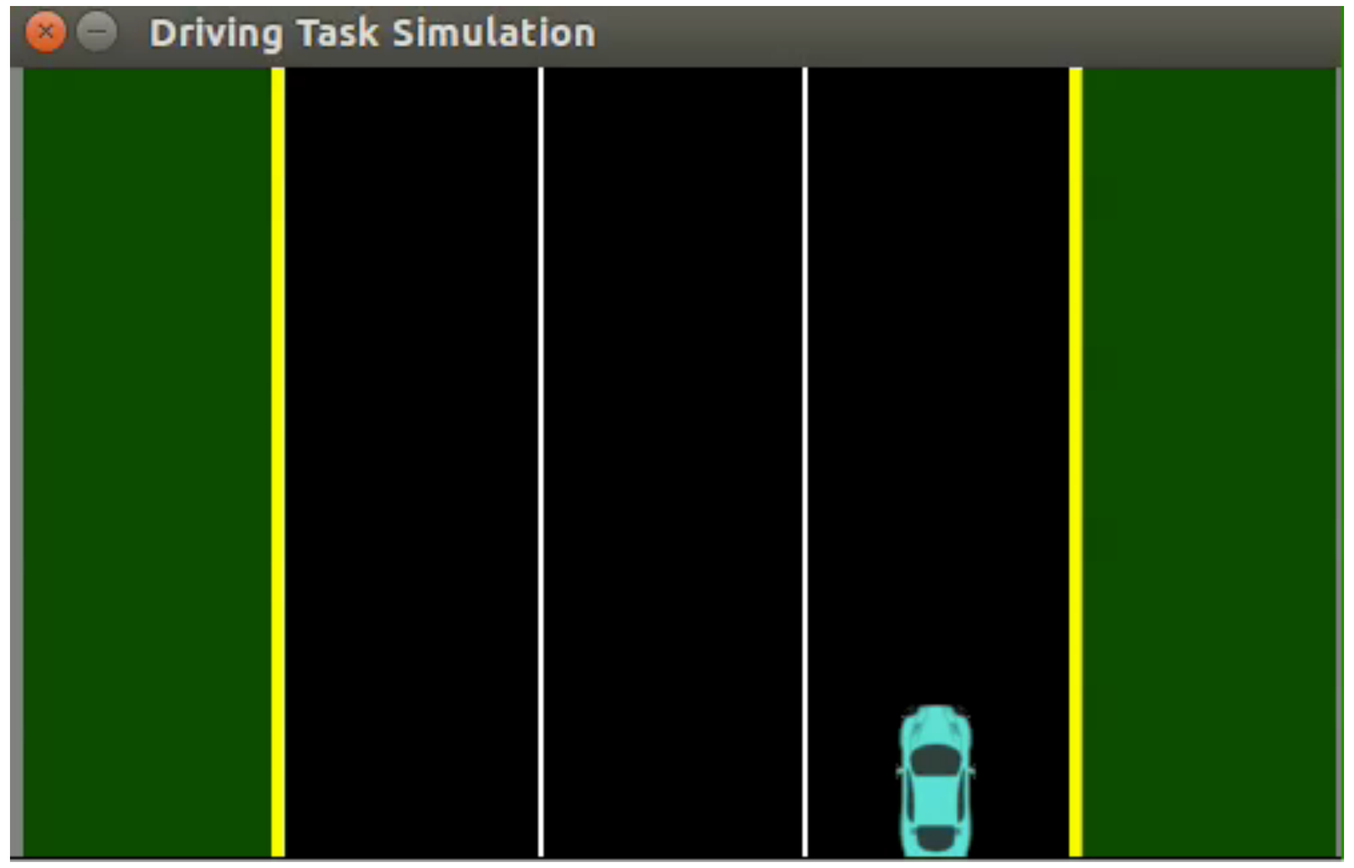
Table 1: Comparison of 95% confidence α-VaR bounds with a 95% confidence Hoeffding-style bound (Syed and Schapire 2008). Both bounds use the Projection algorithm (Abbeel and Ng 2004) to obtain the evaluation policy. Results are averaged over 200 random navigation tasks.

Four orders of magnitude more data efficient!

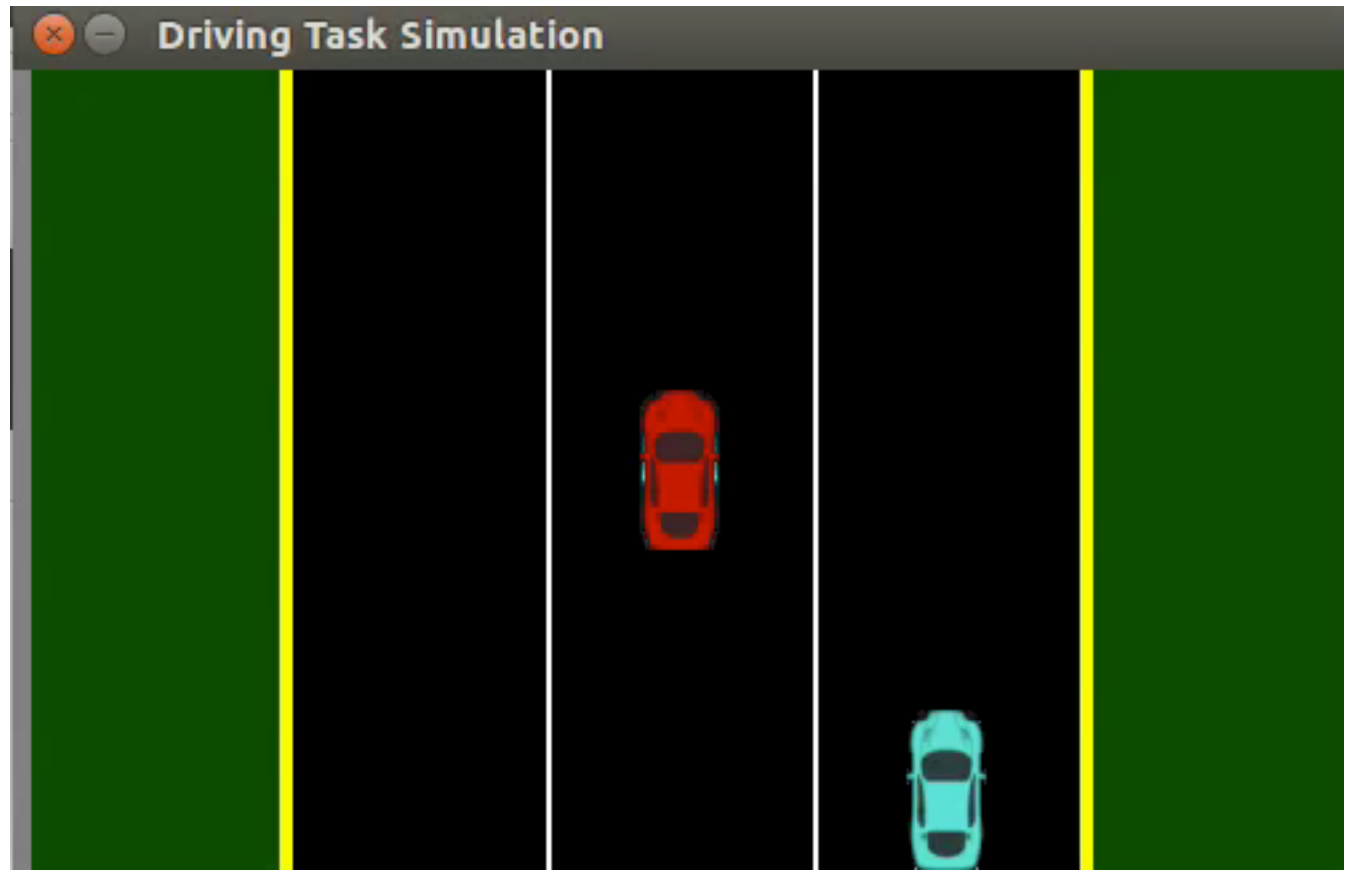
… but computationally intractable

D.S. Brown and S. Niekum.
Efficient Probabilistic Performance Bounds for Inverse Reinforcement Learning.
AAAI Conference on Artificial Intelligence, February 2018.

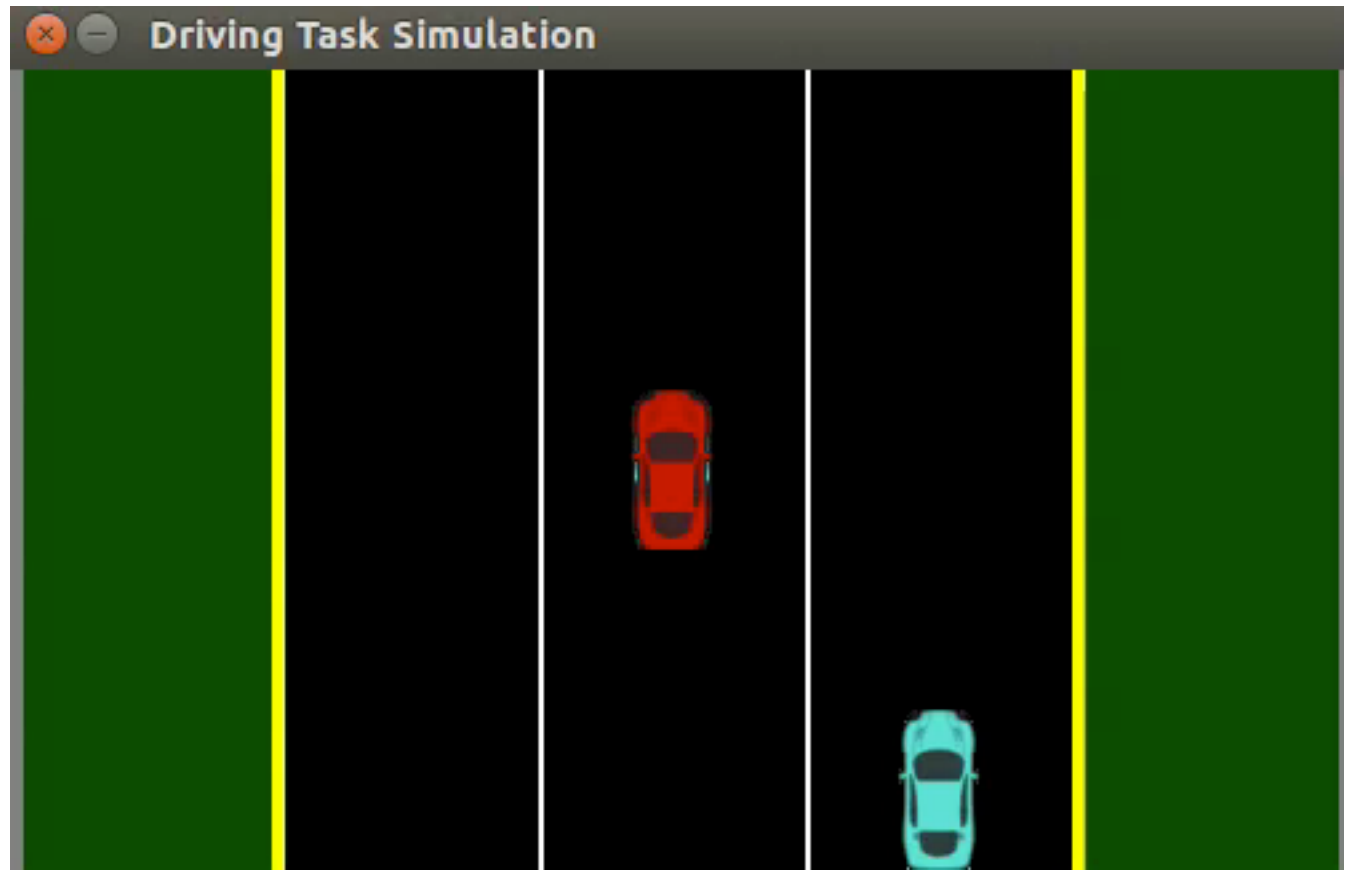
Risk-sensitive preferences



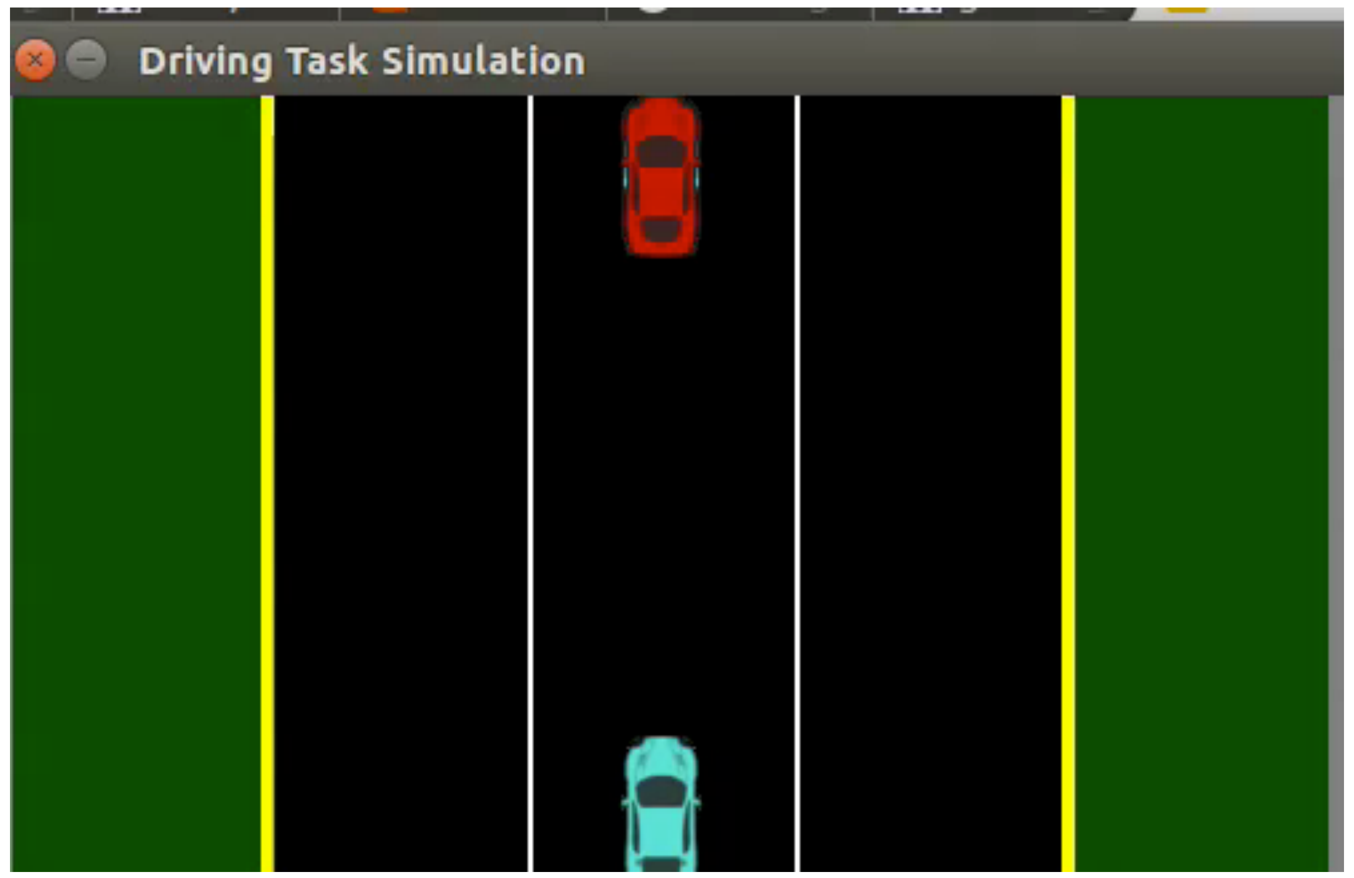
Demonstration: avoids cars, no lane pref



Avoids cars, but prefers right lane

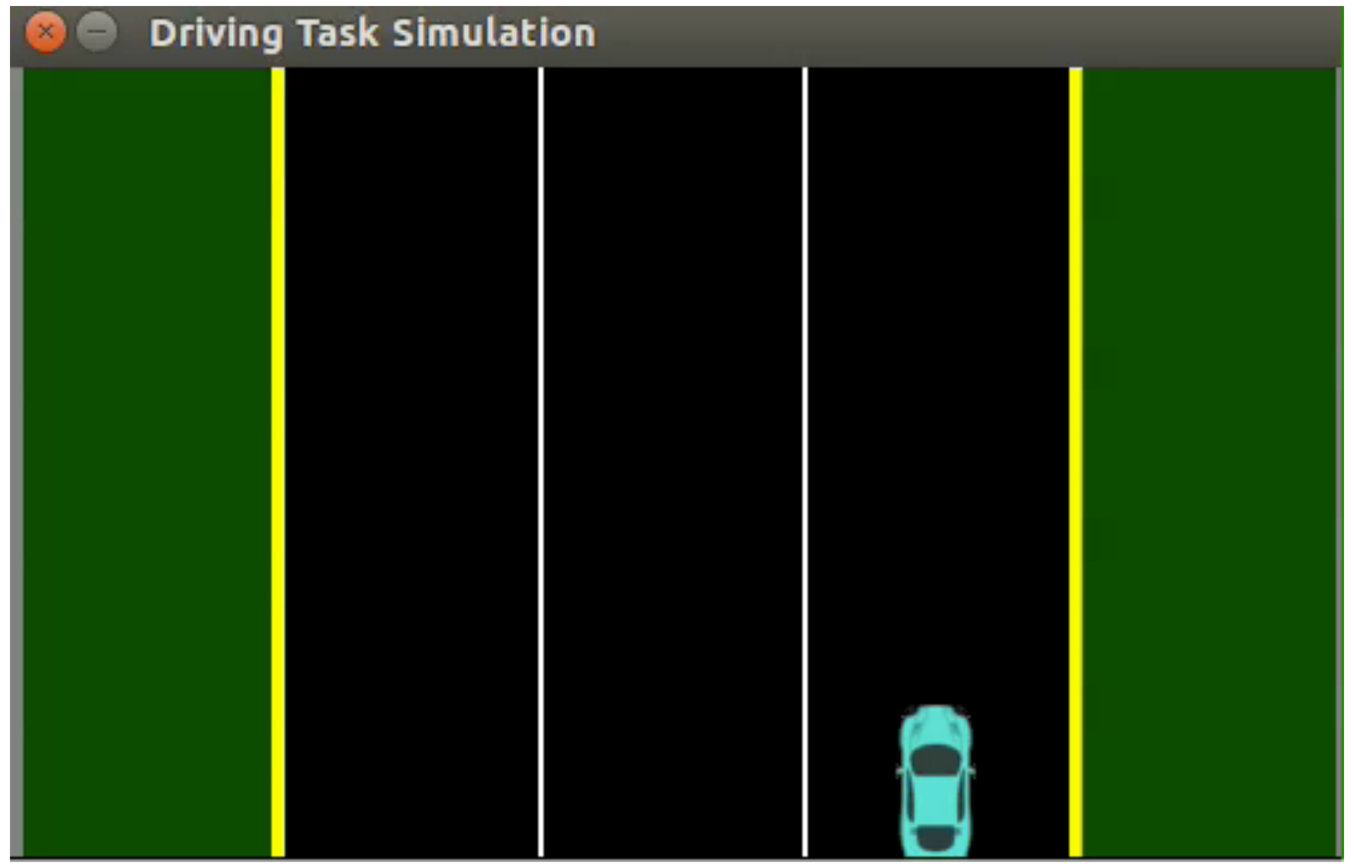


Stays on road, but ignores other cars



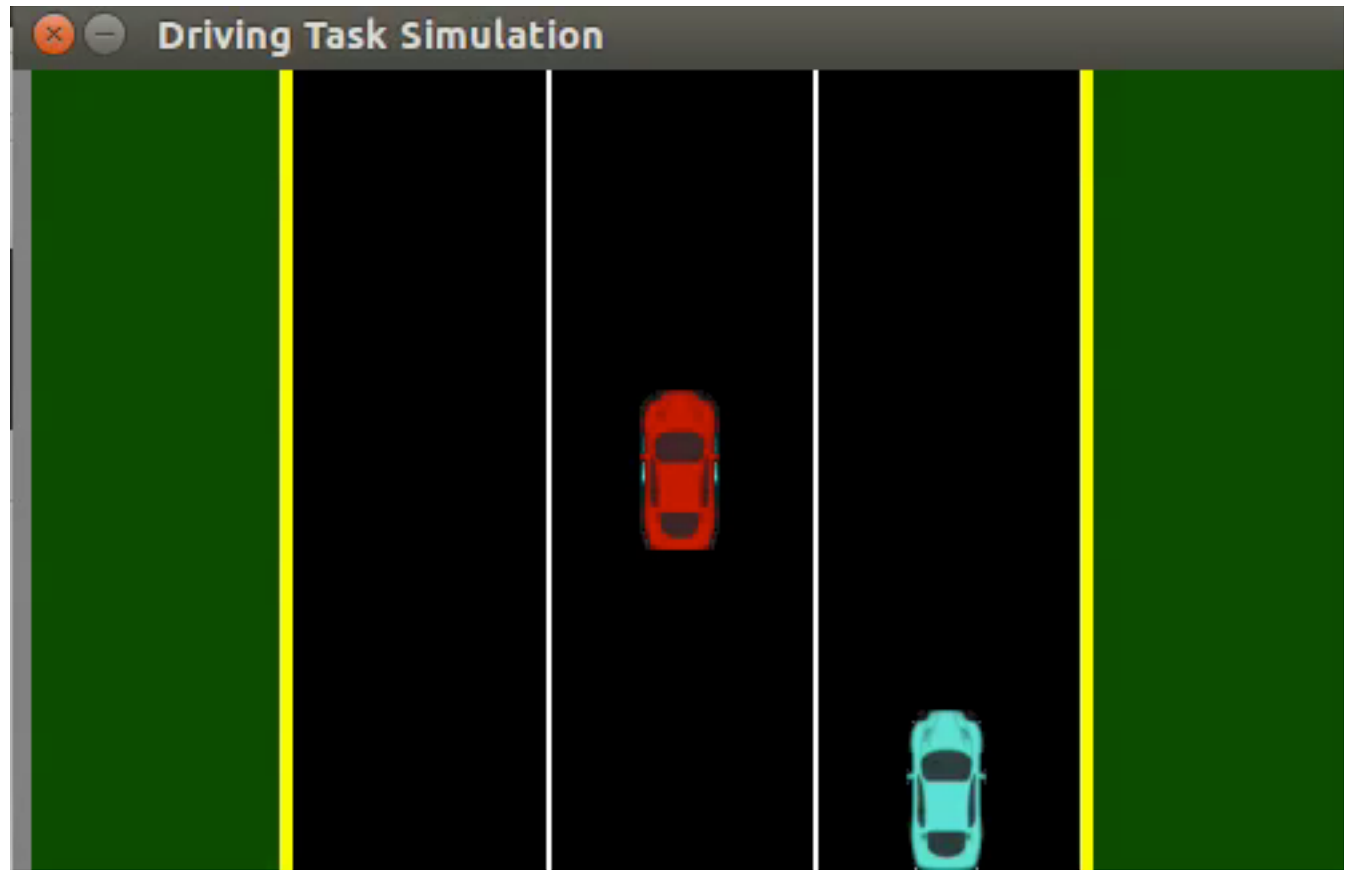
Seeks collisions

Risk-sensitive preferences (feature count-based)



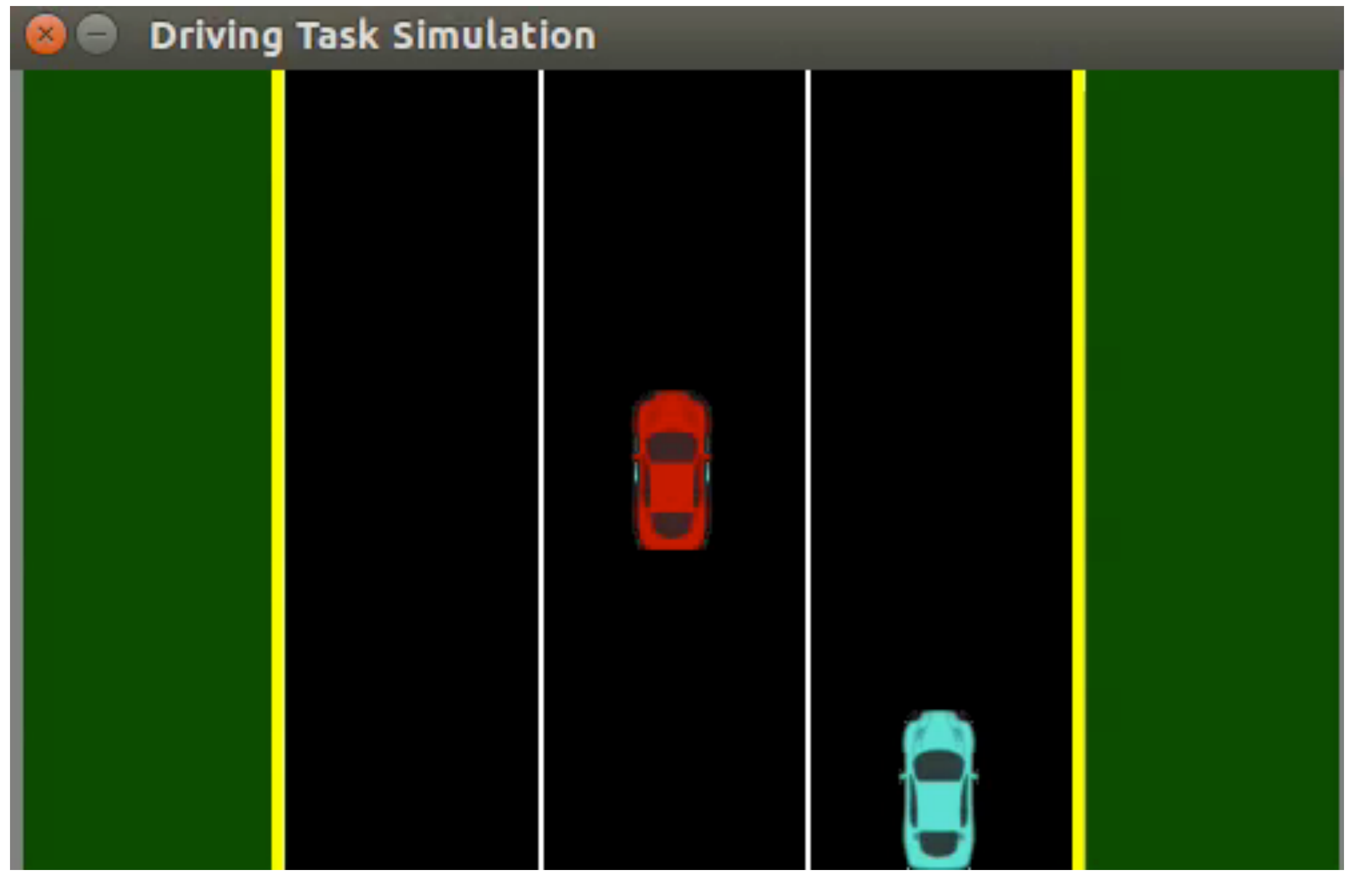
Demonstration: avoids cars, no lane pref

3



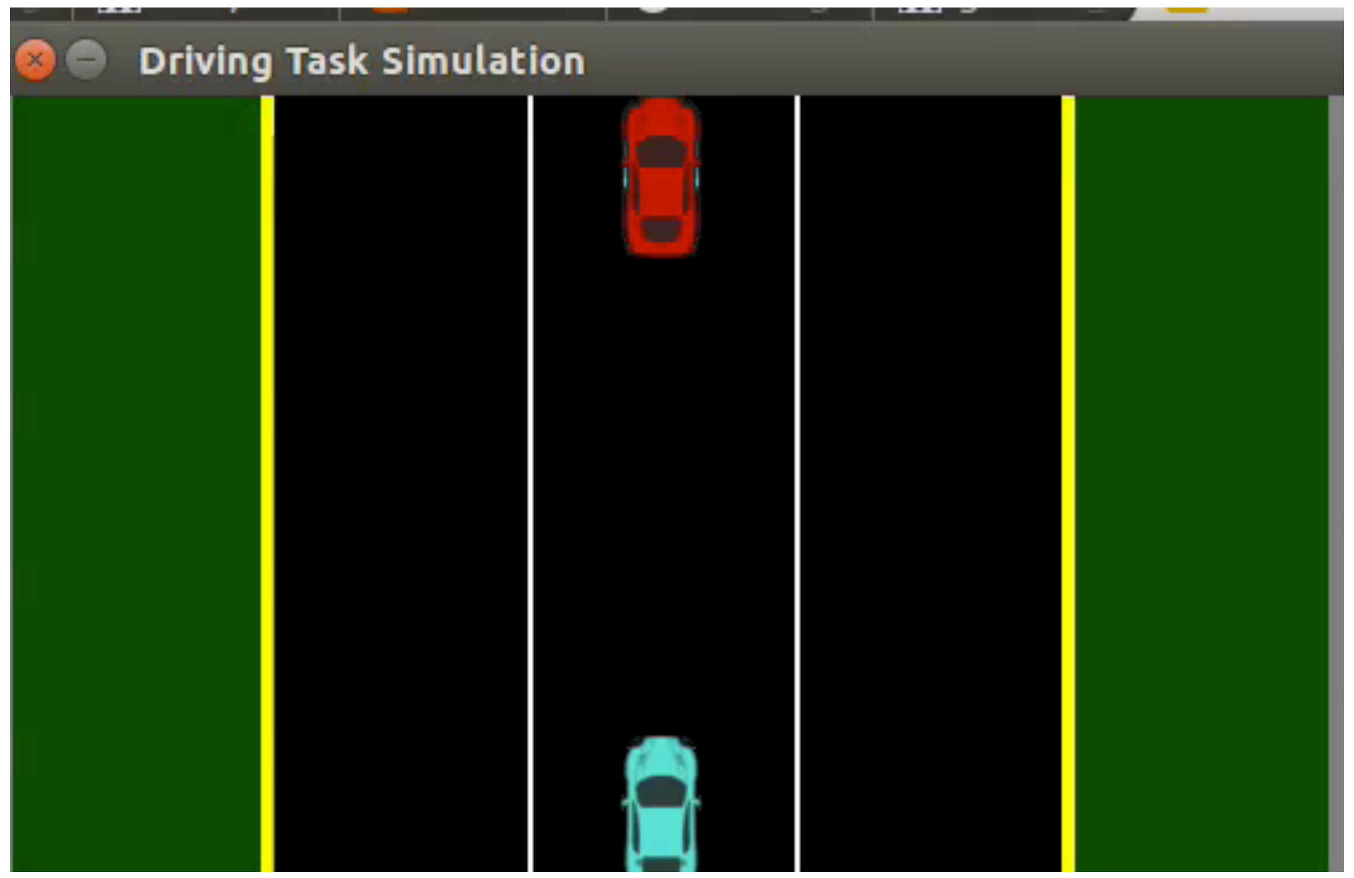
Avoids cars, but prefers right lane

1



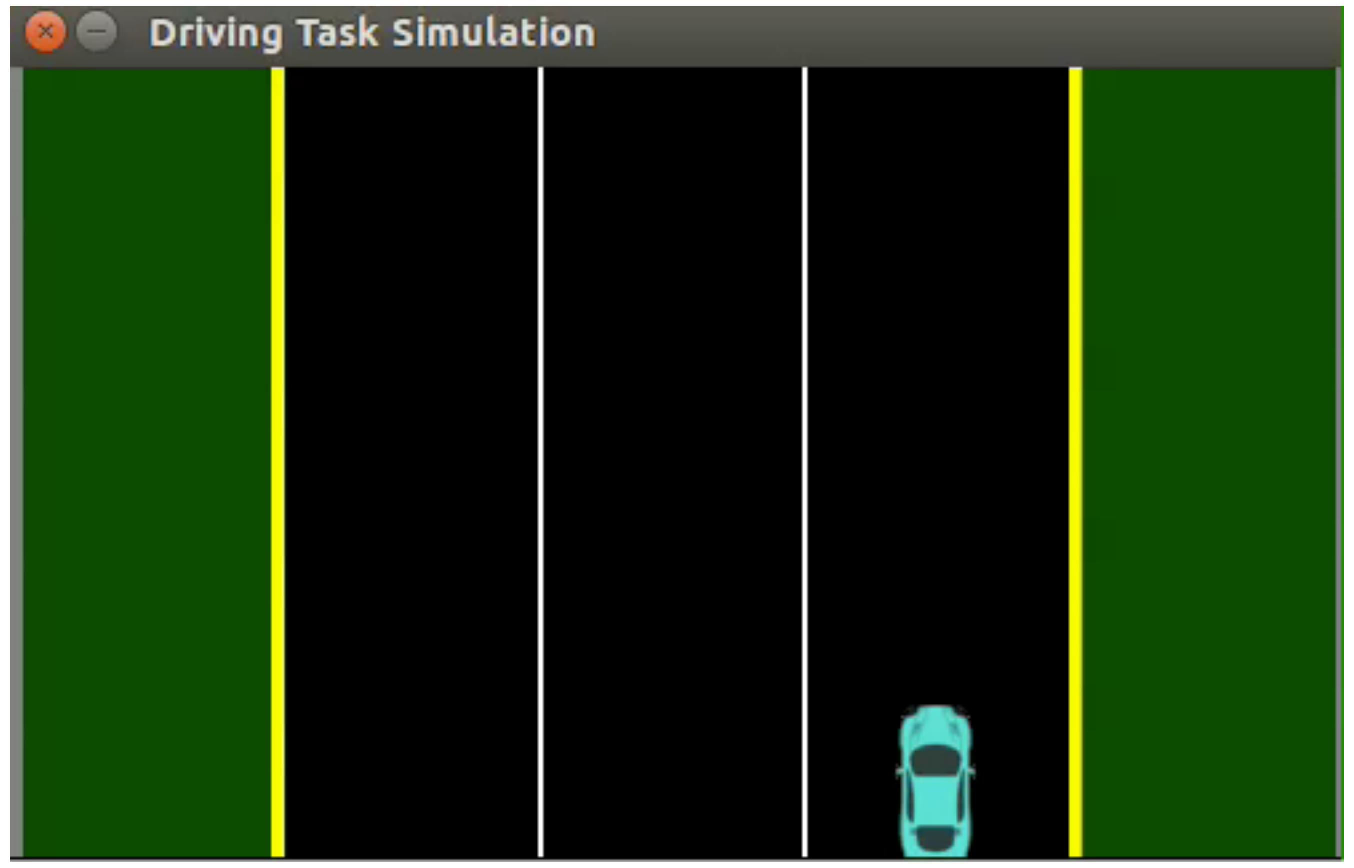
Stays on road, but ignores other cars

2



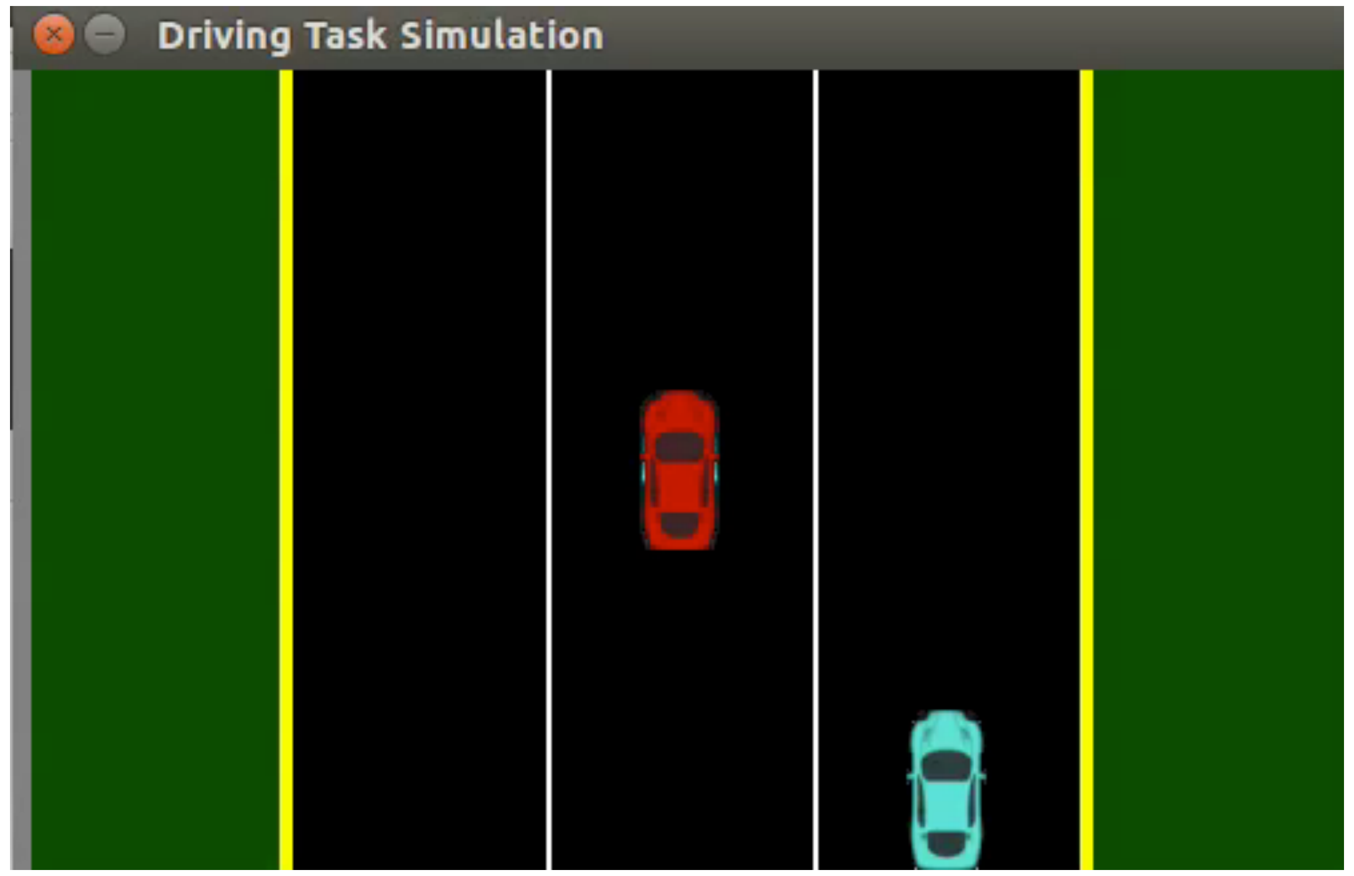
Seeks collisions

Risk-sensitive preferences (our approach)



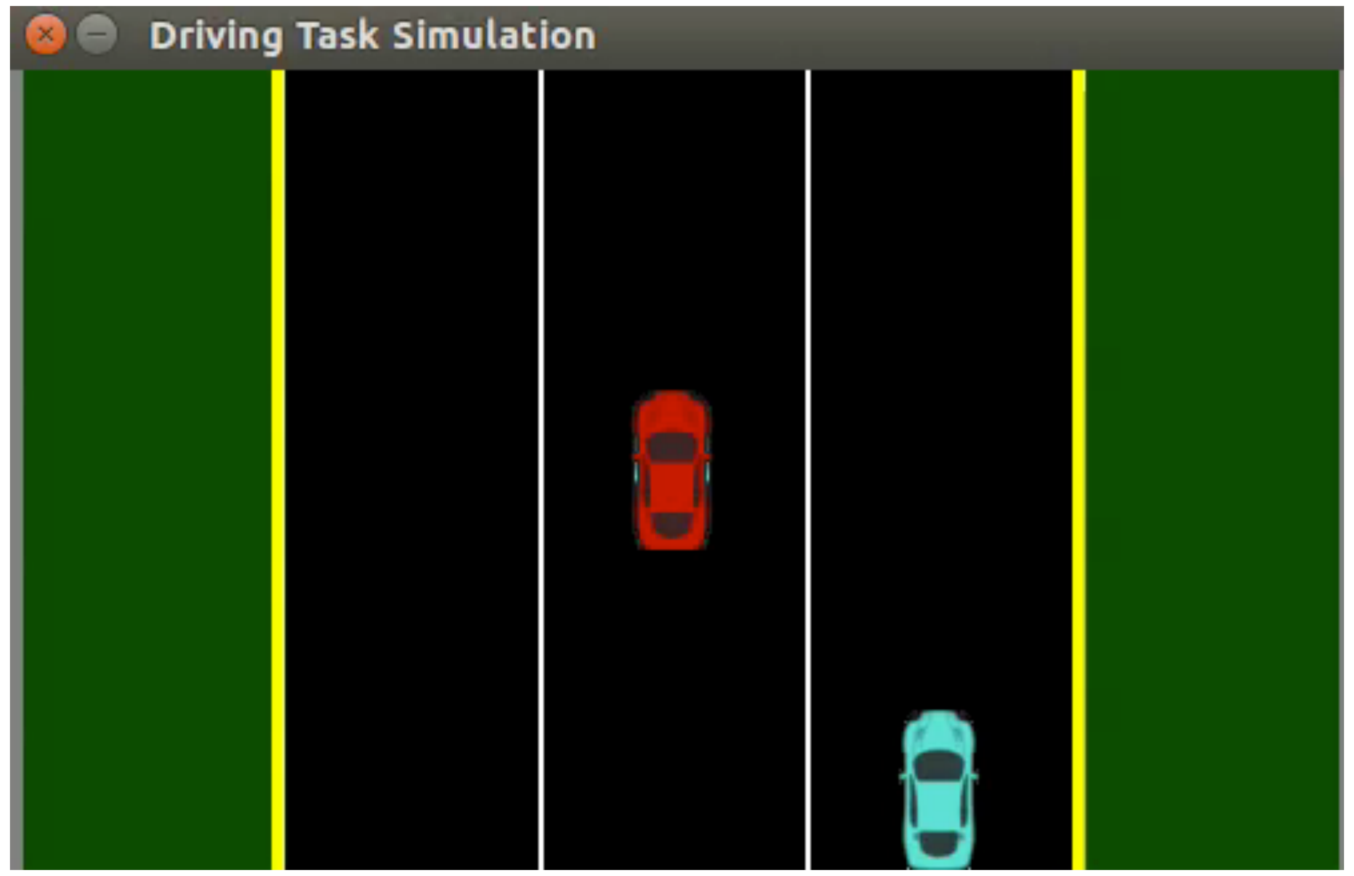
Demonstration: avoids cars, no lane pref

1



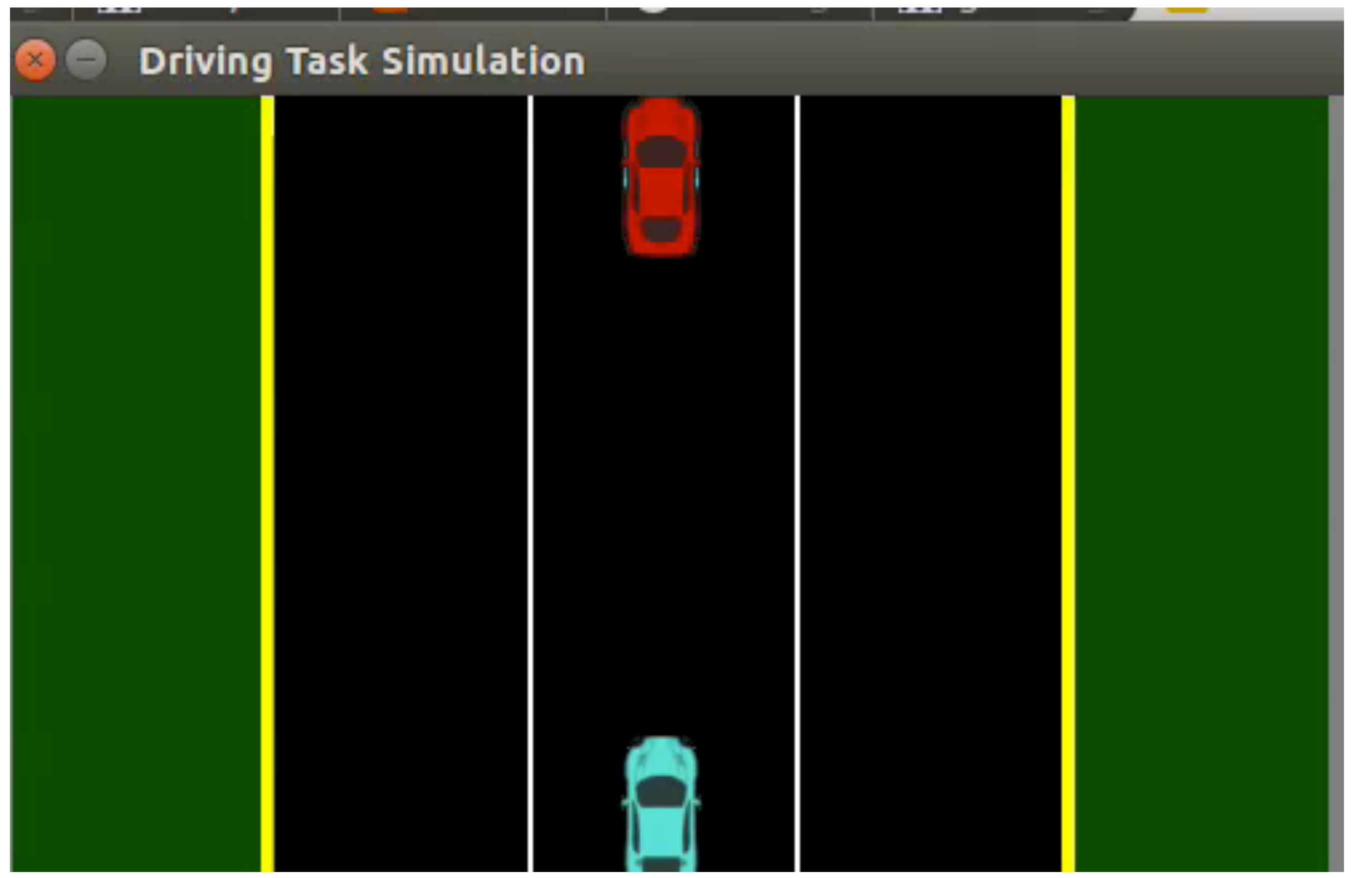
Avoids cars, but prefers right lane

2



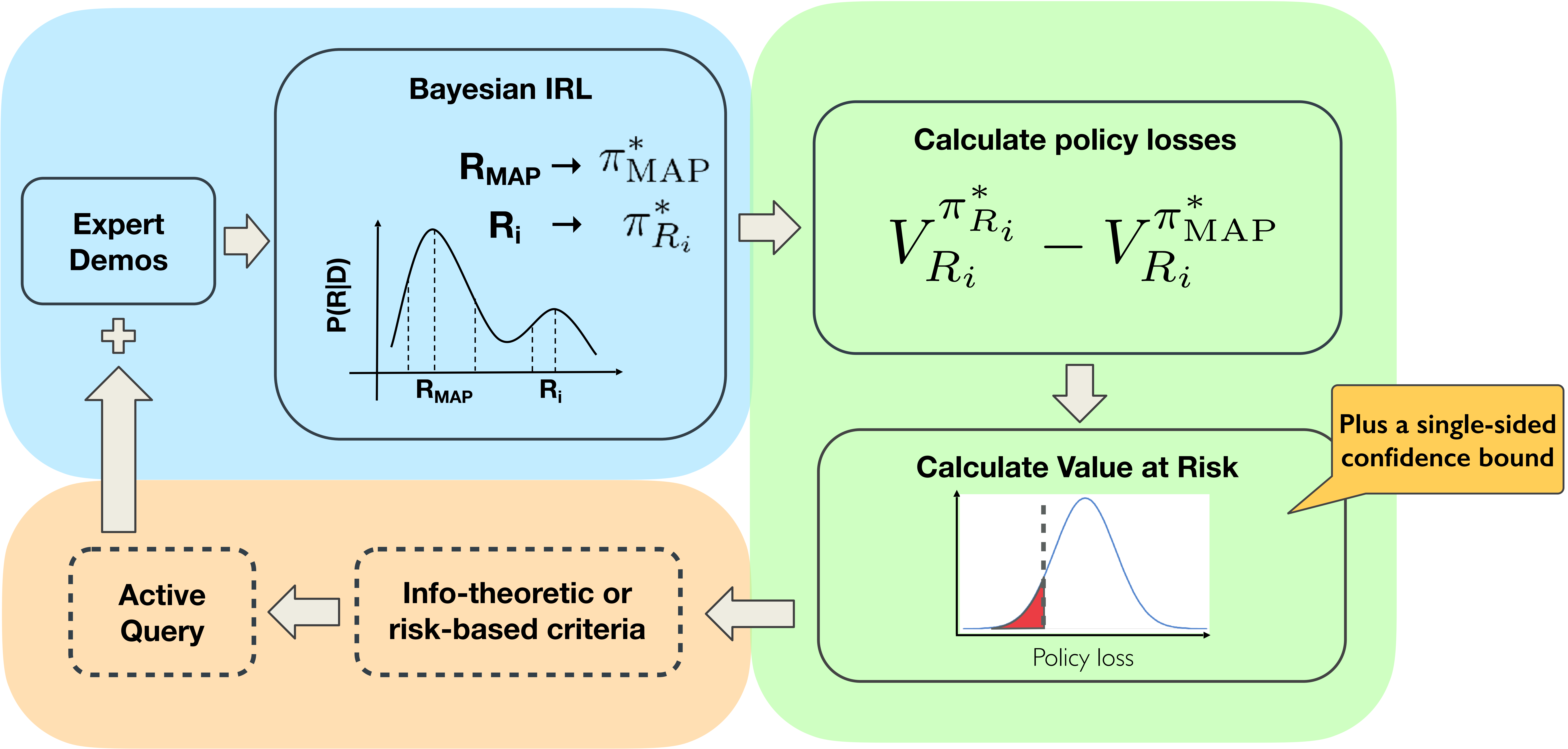
Stays on road, but ignores other cars

3



Seeks collisions

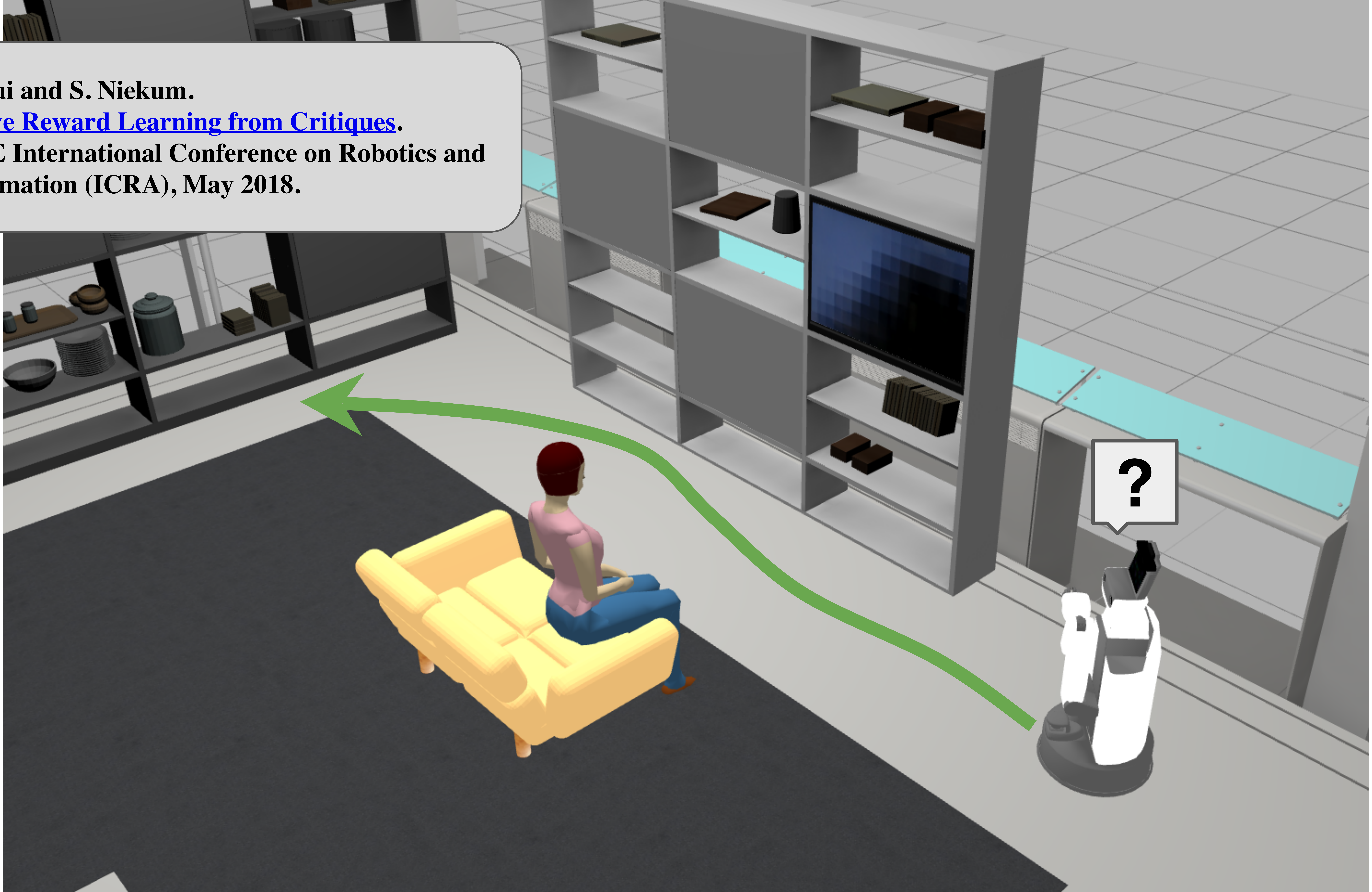
Data efficient, intractable VAIL



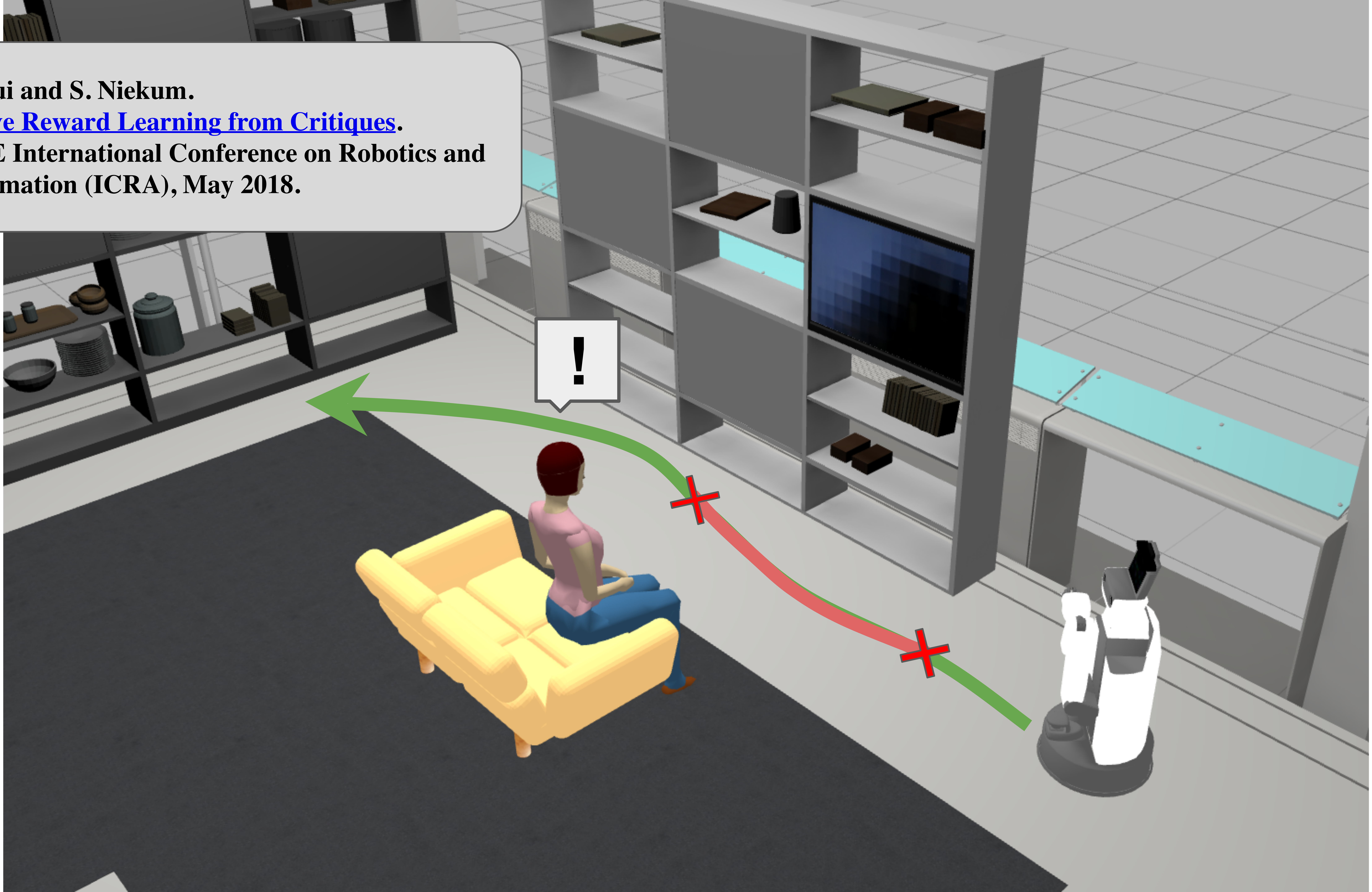
D.S. Brown, Y. Cui, and S. Niekum.
Risk-Aware Active Reinforcement Learning.
Conference on Robot Learning (CoRL), October 2018.

Y. Cui and S. Niekum.
Active Reward Learning from Critiques.
International Conference on Robotics and Automation (ICRA), May 2018.

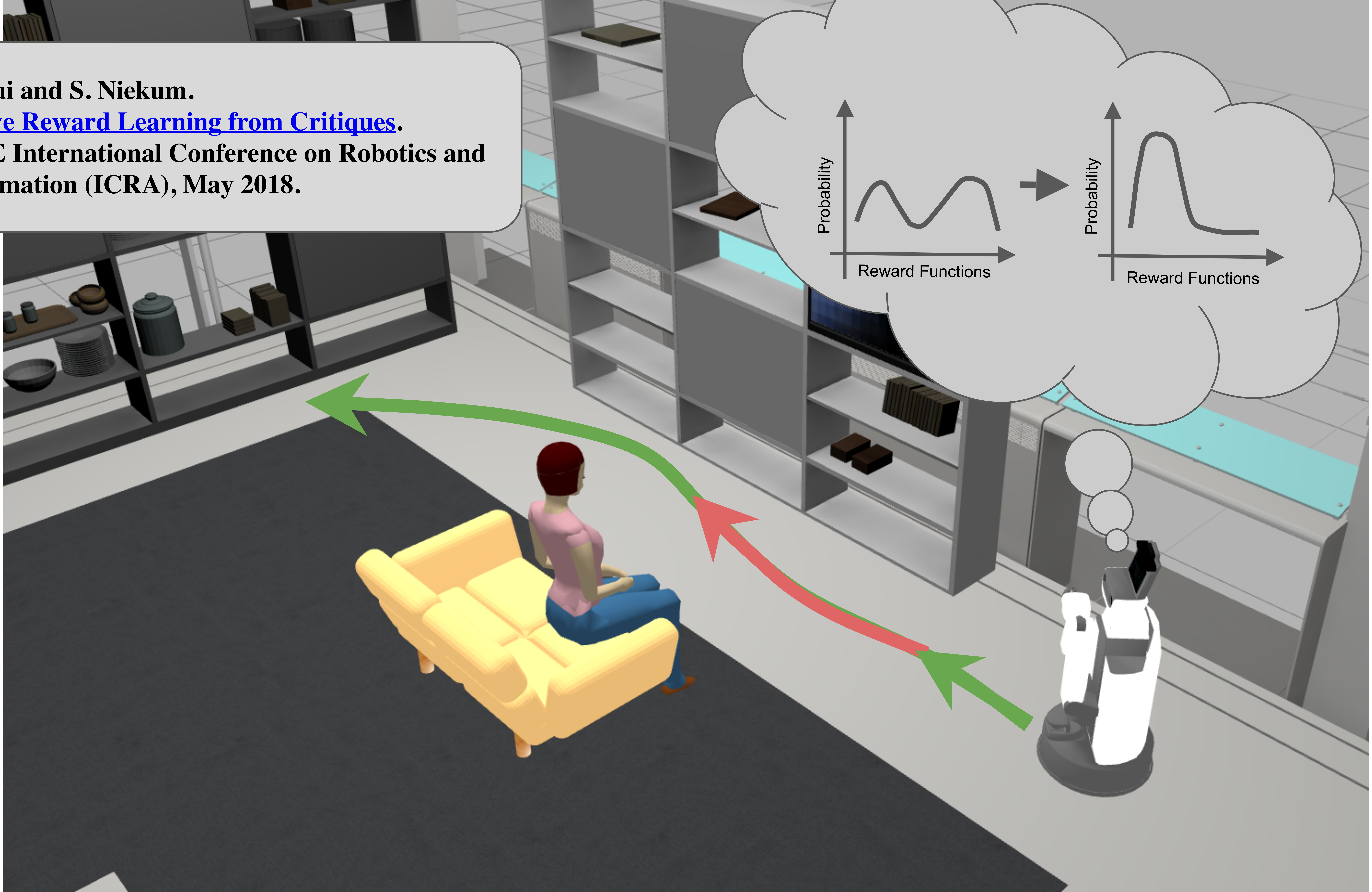
Y. Cui and S. Niekum.
[Active Reward Learning from Critiques](#).
IEEE International Conference on Robotics and Automation (ICRA), May 2018.



Y. Cui and S. Niekum.
Active Reward Learning from Critiques.
IEEE International Conference on Robotics and Automation (ICRA), May 2018.



Y. Cui and S. Niekum.
Active Reward Learning from Critiques.
IEEE International Conference on Robotics and Automation (ICRA), May 2018.



Information Gain Estimation from Reward Function Distribution

$$Pr(a_i \notin O(s_i) \mid R) = 1 - \frac{1}{Z_i} e^{\alpha Q(s_i, a_i, R)}$$

$$Pr(a_i \in O(s_i) \mid R) = \frac{1}{Z_i} e^{\alpha Q(s_i, a_i, R)}$$

- Set of optimal actions at a state:

$$O(s) = \arg\max_{a \in A} Q^{\pi}(s, a)$$

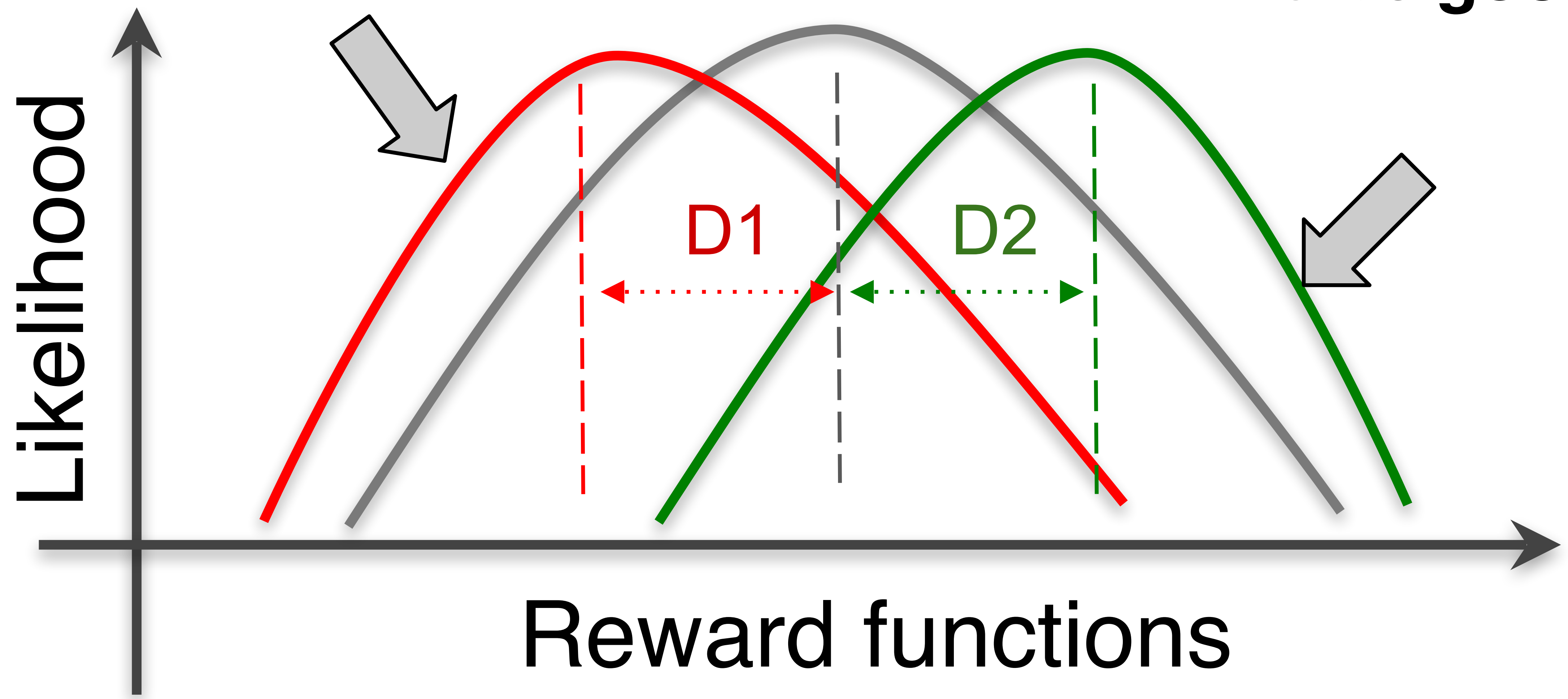
- Distance Measure:

$$D_{KL}(P||Q) = \sum_i P(i) \log \frac{P(i)}{Q(i)}$$

- Expected Information Gain:

Update an action to be bad

Update an action to be good

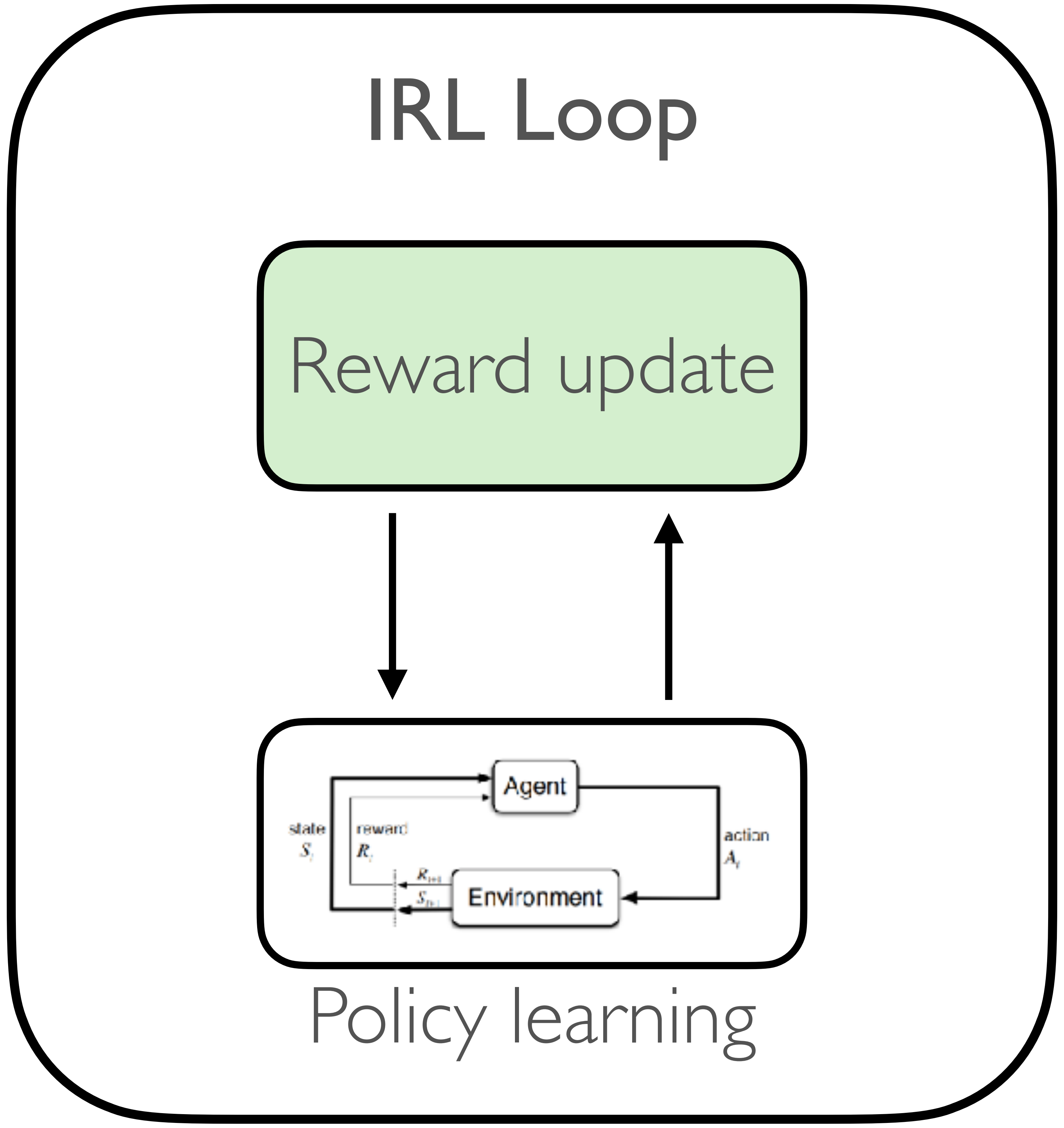


$$G^{+}(s_i, a_i) = G(D^{+} \cup (s_i, a_i) \mid Be(R)) = \boxed{Pr(a_i \in O(s_i) \mid Be(R))} D(Be'(R)||Be(R))$$

$$G^{-}(s_i, a_i) = G(D^{-} \cup (s_i, a_i) \mid Be(R)) = \boxed{Pr(a_i \notin O(s_i) \mid Be(R))} D(Be'(R)||Be(R))$$

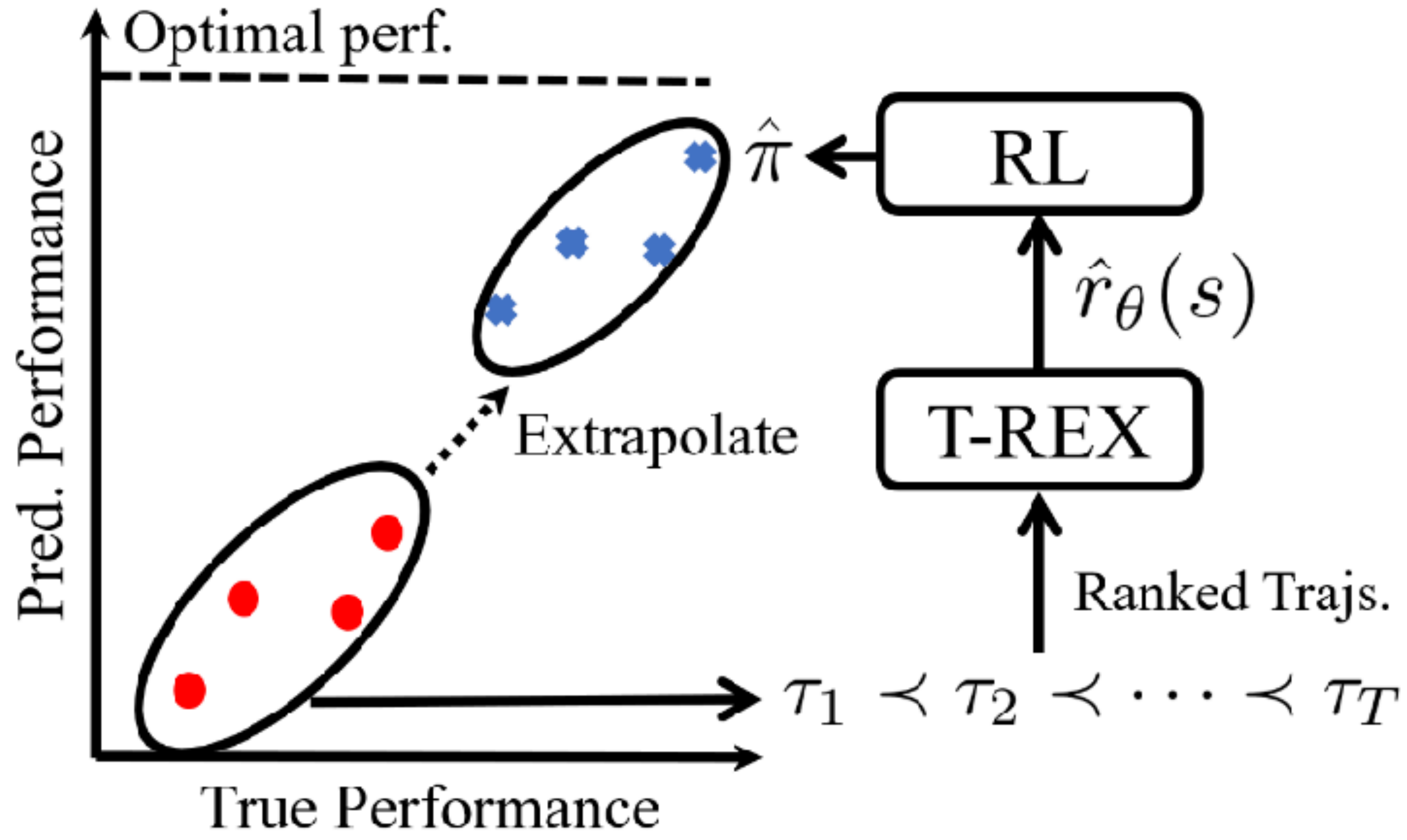
Problems with standard inverse reinforcement learning

Policy learning in inner loop



Cannot outperform demonstrator

T-REX: Trajectory-ranked Reward Extrapolation



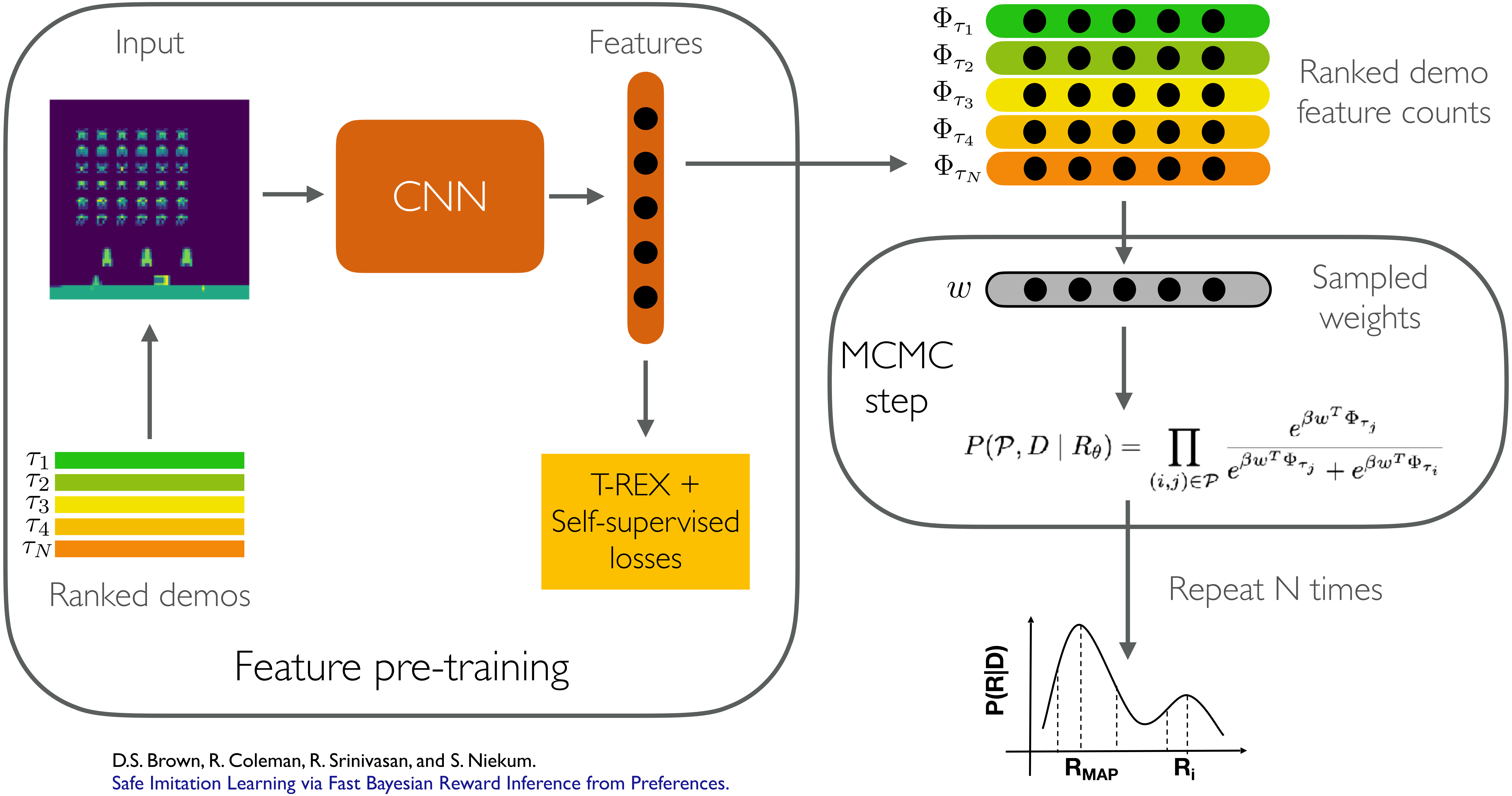
$$P(\hat{J}_\theta(\tau_i) < \hat{J}_\theta(\tau_j)) \approx \frac{\exp \sum_{s \in \tau_j} \hat{r}_\theta(s)}{\exp \sum_{s \in \tau_i} \hat{r}_\theta(s) + \exp \sum_{s \in \tau_j} \hat{r}_\theta(s)}$$

$$\mathcal{L}(\theta) = \mathbf{E}_{\tau_i, \tau_j \sim \Pi}\left[\xi\left(P(\hat{J}_\theta(\tau_i) < \hat{J}_\theta(\tau_j)), \tau_i \prec \tau_j\right)\right]$$

- Fully supervised — no policy learning
- Works on high-dim (e.g. Atari) with ~10 demos
- Auto-generated rankings: D. Brown, W. Goo, and S. Niekum. Ranking-Based Reward Extrapolation without Rankings. Conference on Robot Learning (CoRL), October 2019.

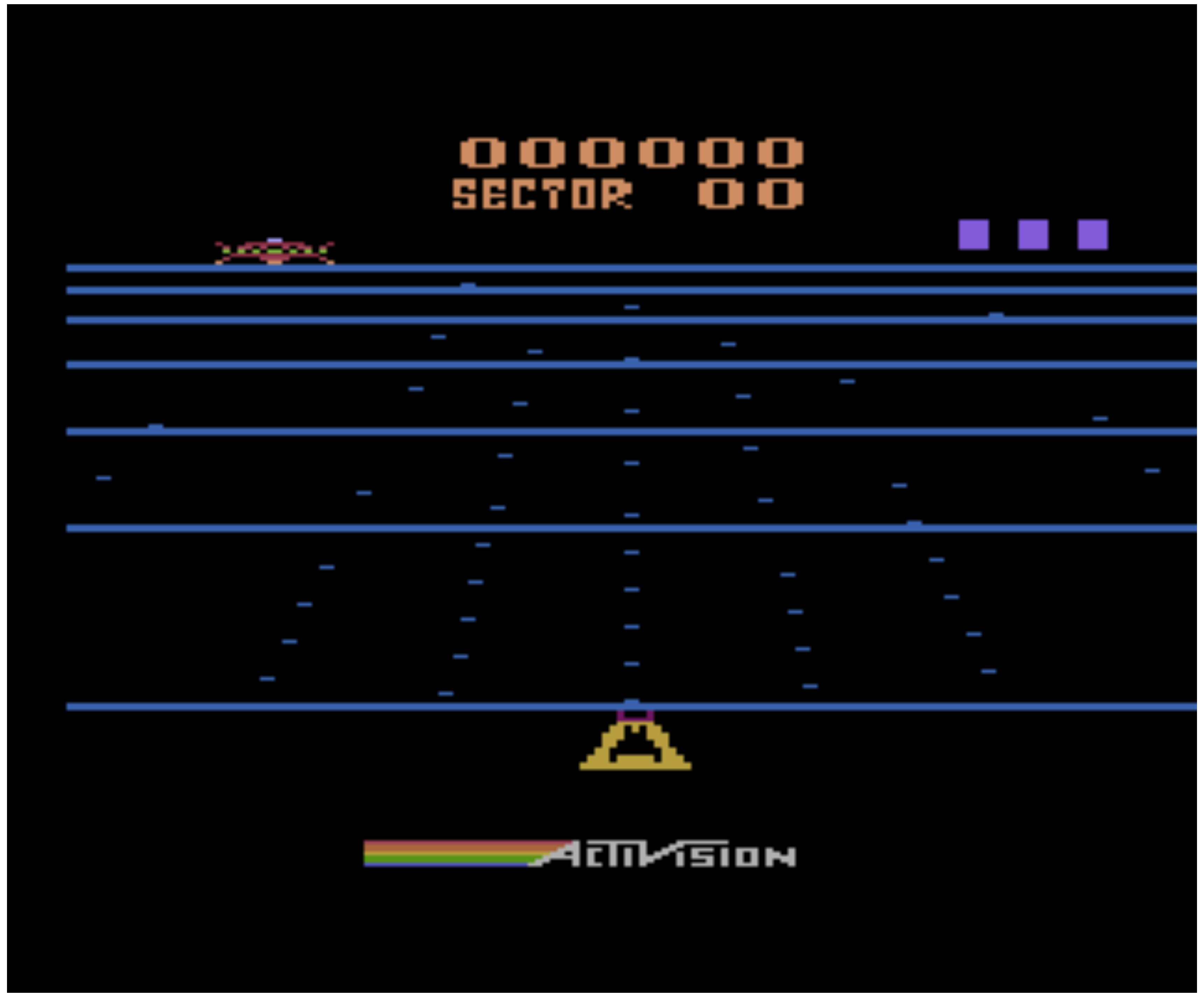
D.S. Brown, W. Goo, P. Nagarajan, and S. Niekum.
Extrapolating Beyond Suboptimal Demonstrations via Inverse Reinforcement Learning from Observations.
International Conference on Machine Learning (ICML), June 2019.

Bayesian REX



D.S. Brown, R. Coleman, R. Srinivasan, and S. Niekum.
Safe Imitation Learning via Fast Bayesian Reward Inference from Preferences.
International Conference on Machine Learning (ICML), July 2020.

Bayesian REX: Results



Beamrider

Policy	Predicted Mean	Predicted 0.05-VaR	Ground Truth Avg. Score	Ground Truth Avg. Length
A	17.1	7.9	480.6	1372.6
B	22.7	11.9	703.4	1,412.8
C	45.5	24.9	1828.5	2,389.9
D	57.6	31.5	2586.7	2,965.0
No-Op	102.5	-1557.1	0.0	99,994.0

Not restricted to policy evaluation!

Can also learn policy to balance expected return and CVaR:

D.S. Brown, S. Niekum, and M. Petrik.
Bayesian Robust Optimization for Imitation Learning.
Neural Information Processing Systems, December 2020.

Efficient value alignment verification: A driver's test for AI

- What if we want to verify a robot's value alignment with us post-learning?
- We don't want to require policy rollouts, due to both safety and efficiency concerns.
- Can we design a **driver's test** — a small set of (various types of) questions to ask an agent that verify alignment?

D.S. Brown, J. Schneider, A. Dragan, and S. Niekum.
Value Alignment Verification.
International Conference on Machine Learning, July 2021.

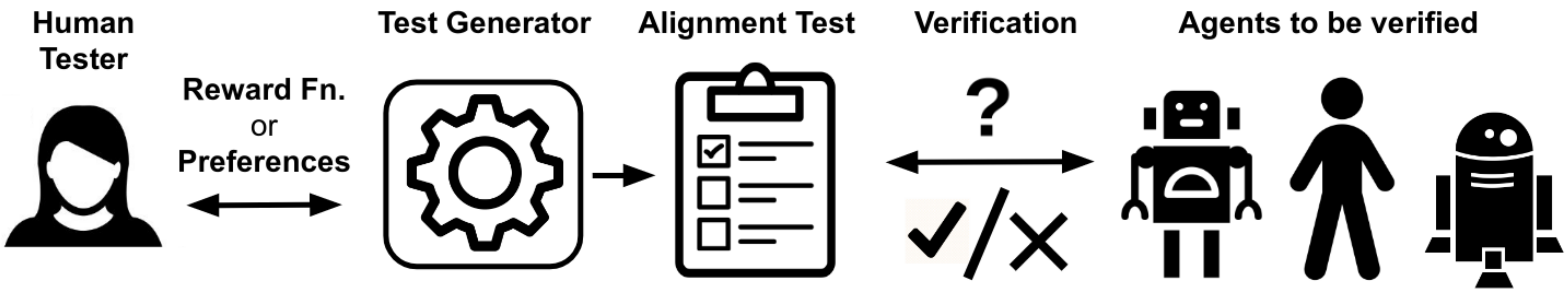
Definition: Epsilon value alignment

Definition 1. *Given reward function R, policy π' is ϵ-**value aligned** in environment E if and only if*

$$V_R^*(s) - V_R^{\pi'}(s) \leq \epsilon, \forall s \in \mathcal{S}. \qquad (1)$$

Value alignment verification

How to efficiently test whether a robot is value aligned with a human's intent?



Assumptions

Non-Restrictive

- Rational Robot

$$\pi'(s) \in \arg\max_a Q^*_{R'}(s, a)$$

- Reward function is linear combination of features

$$R(s) = \mathbf{w}^\top \phi(s)$$

Restrictive

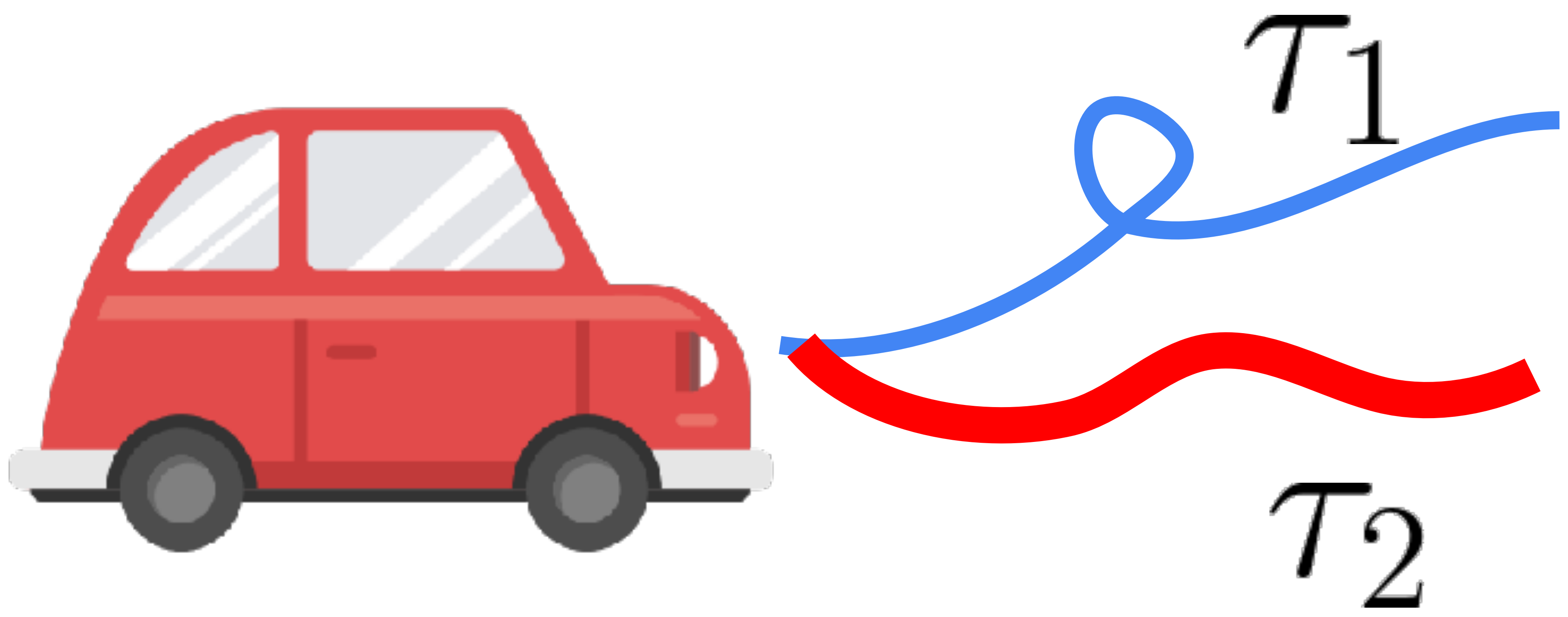
- Human and robot share same features

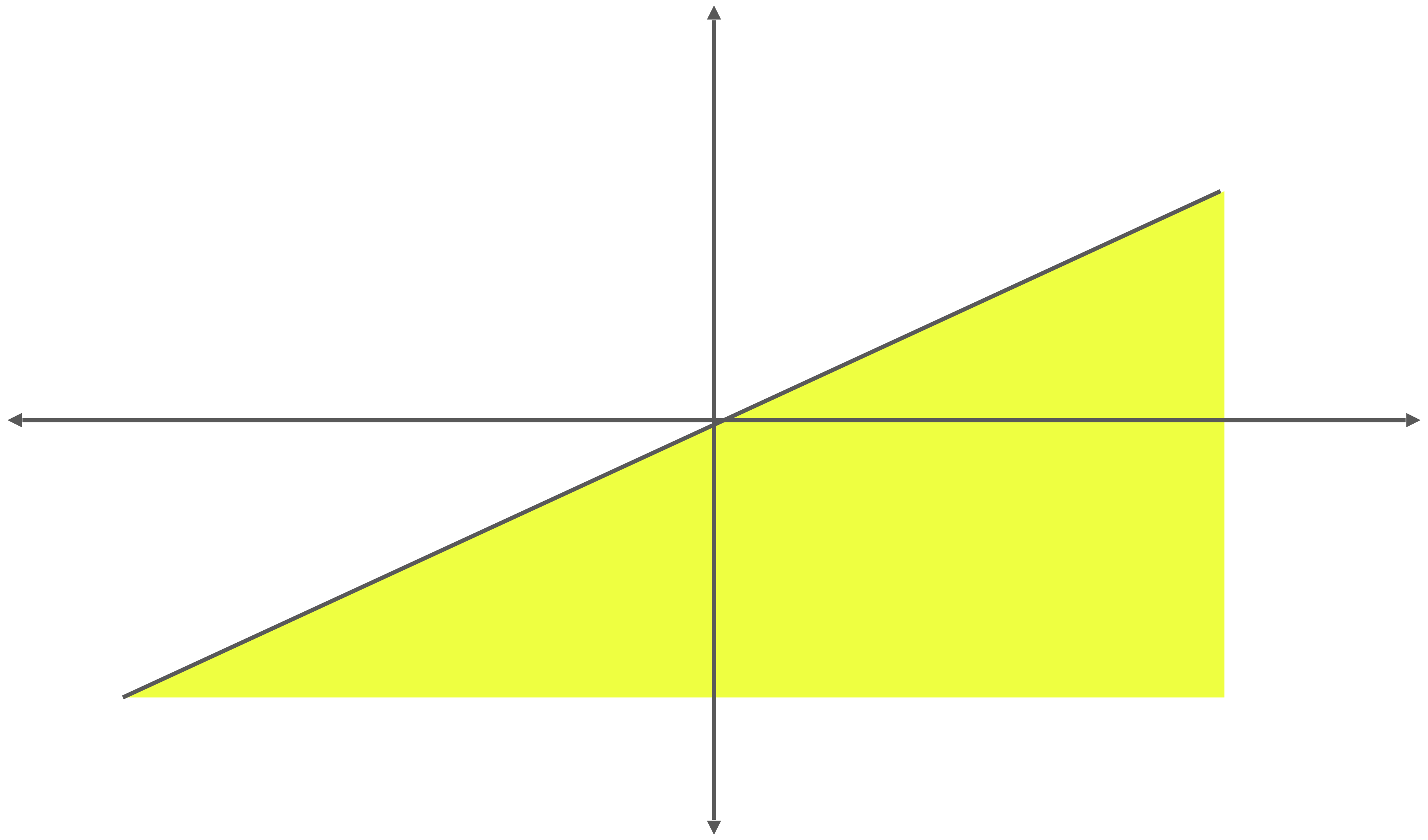
$$R(s) = \mathbf{w}^\top \boxed{\phi(s)}$$

Reward function halfspaces

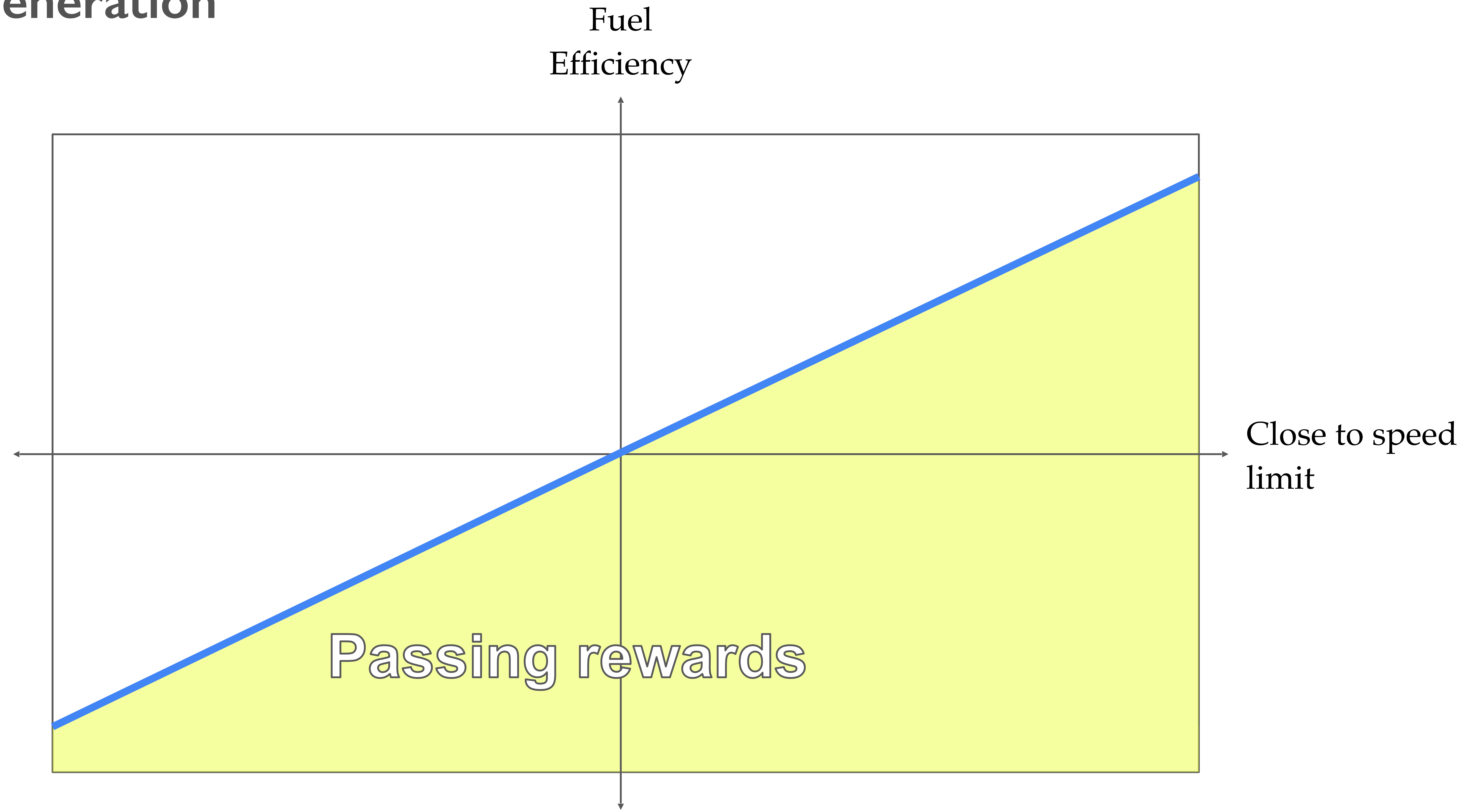
$$\tau_1 \succ \tau_2$$

$$\mathbf{w}^\top (\mathbf{\Phi}(\tau_1) - \mathbf{\Phi}(\tau_2)) > 0$$

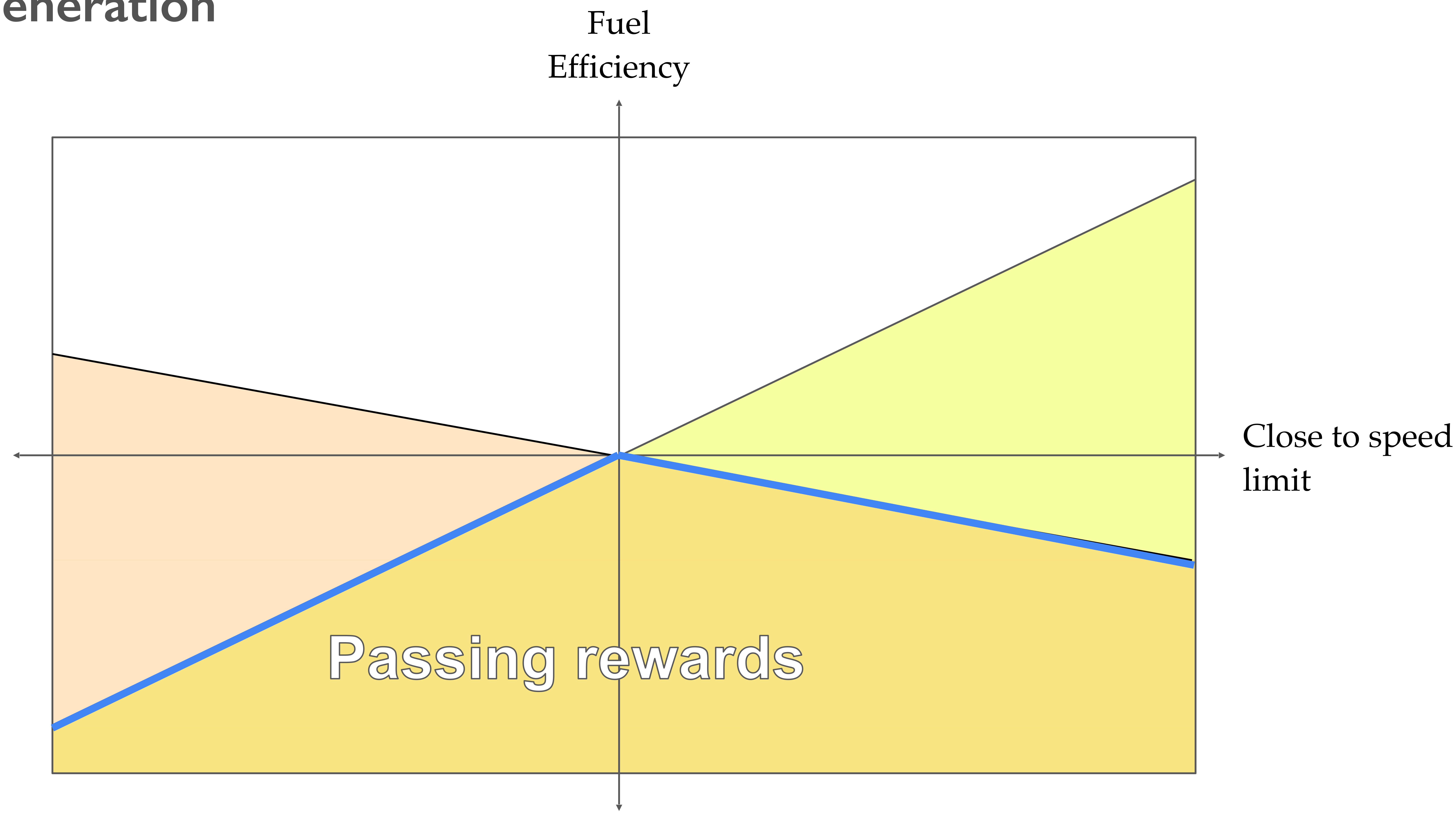




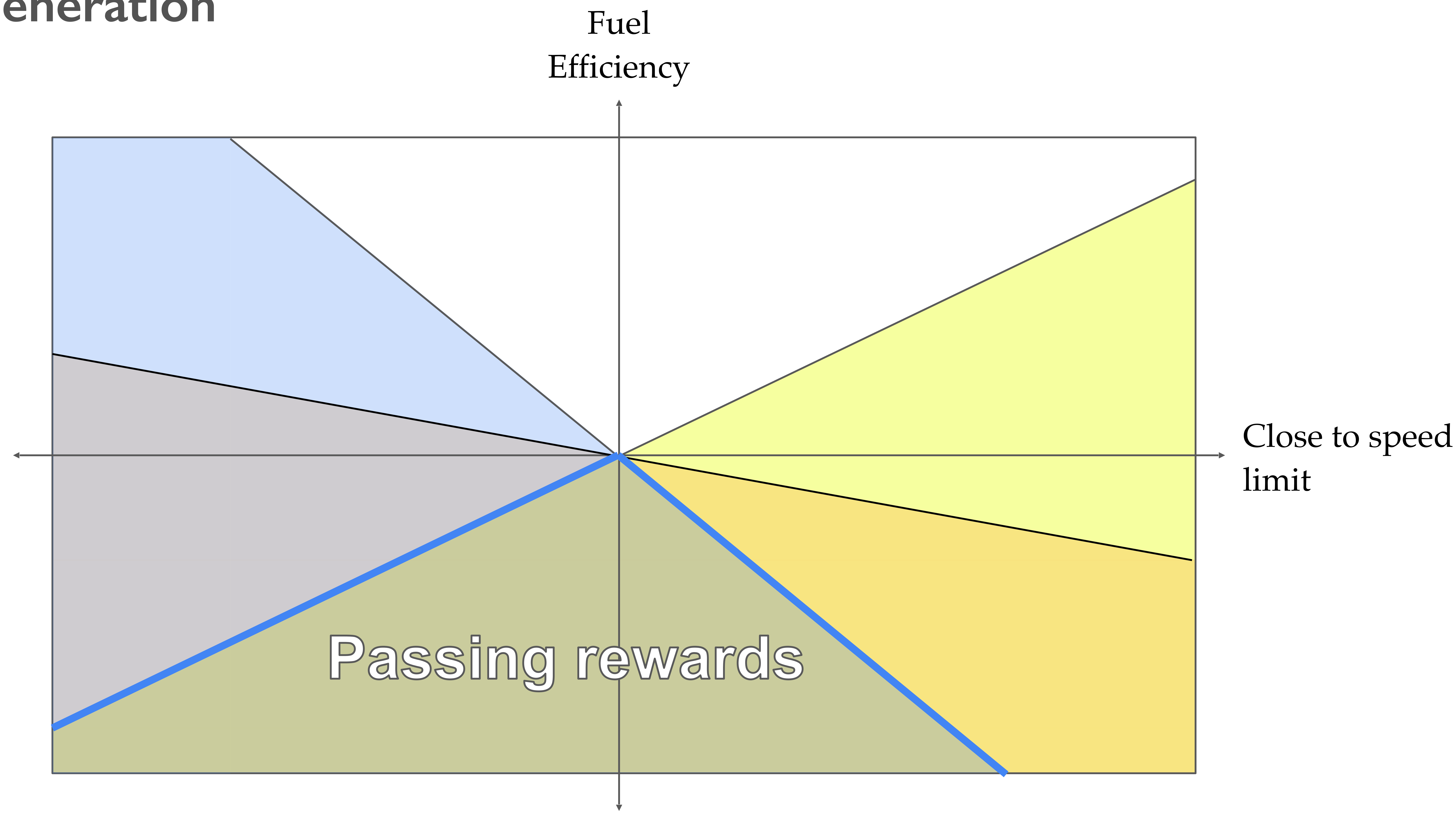
Test Generation



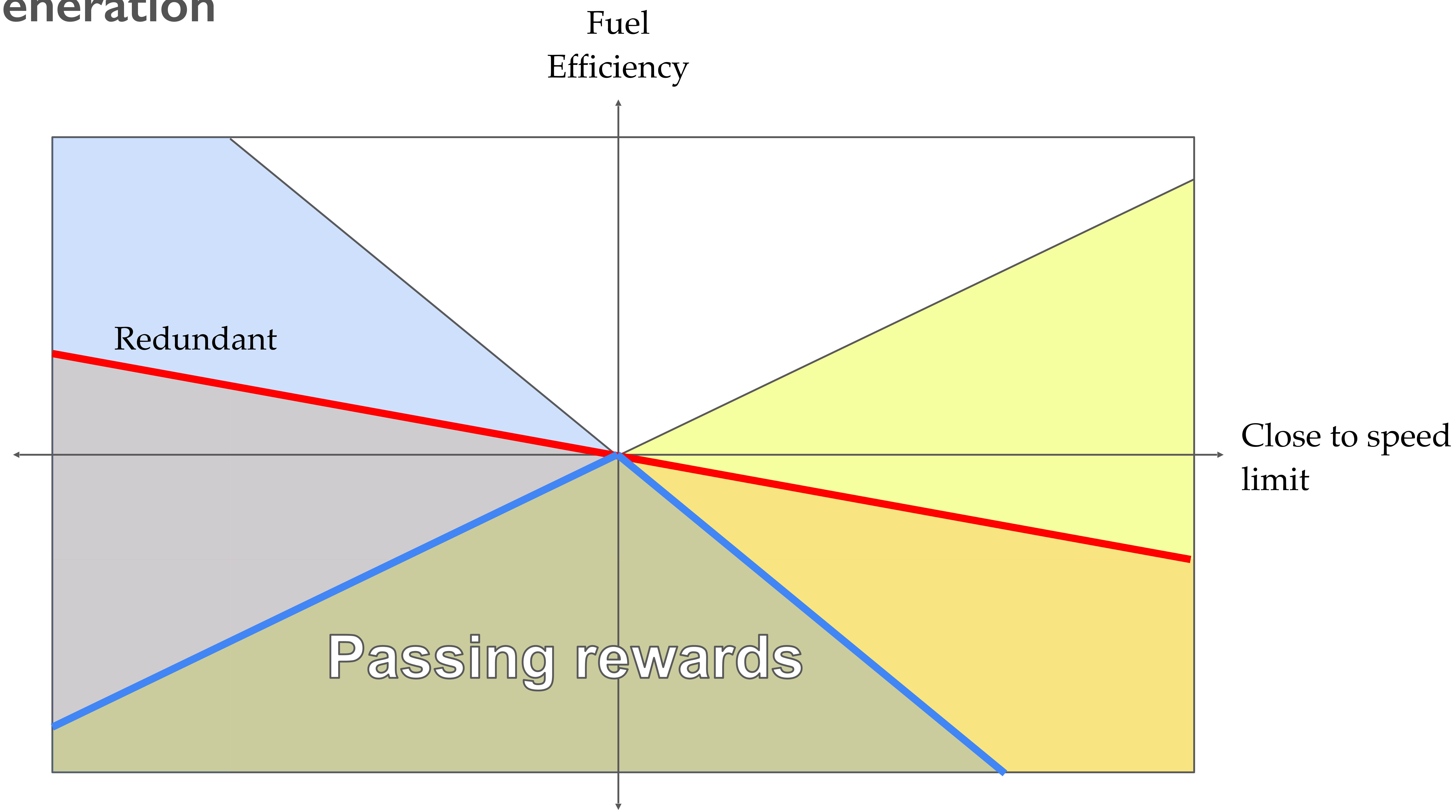
Test Generation



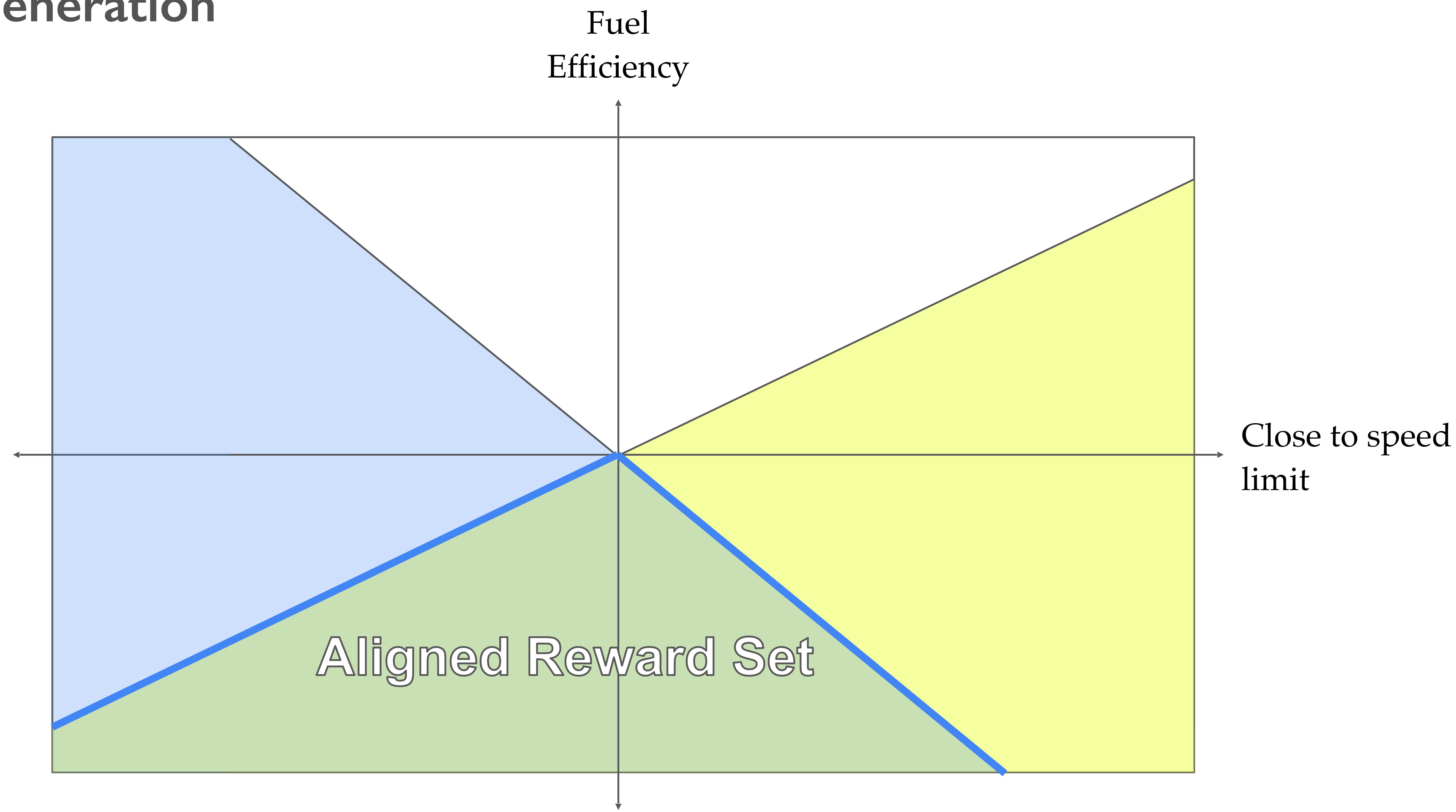
Test Generation



Test Generation



Test Generation



$$ARS(R) = \{R' \mid OPT(R') \subseteq OPT(R)\}.$$

Alignment test conditions

- If the human can write down their reward function, an exact alignment test can be performed in the following query-access settings:
  - Reward function weights $\mathbf{w}_{\text{Robot}}$
  - Reward samples $R_{\text{Robot}}(s)$
  - Value samples $V^*_{\text{Robot}}(s),\ Q^*_{\text{Robot}}(s, a)$
  - Trajectory Preferences $\xi_1 \prec \xi_2$
- Otherwise, must perform preference elicitation to construct test