SPiDR: A Simple Approach for Zero-Shot Safety in Sim-to-Real Transfer

Yarden As[1], Chengrui Qu[2], Benjamin Unger[1], Dongho Kang[1], Max van der Hart[1], Laixi Shi[3], Stelian Coros[1], Adam Wierman[2], Andreas Krause[1]

[1]ETH Zürich, [2]Caltech, [3] John Hopkins University

TL;DR: We propose a practical algorithm for safe sim-to-real transfer

Problem Setting

Real: $\max \underbrace{\mathbb{E}_{\pi,p^\star}\left[\sum_{t=0}^{\infty}\gamma^t r(s_t,a_t)\right]}_{J_{p^\star}(\pi)}$ s.t. $\underbrace{\mathbb{E}_{\pi,p^\star}\left[\sum_{t=0}^{\infty}\gamma^t c(s_t,a_t)\right]}_{C_{p^\star}(\pi)} \leq d$

with $p^\star$ being the true dynamics.

Simulator: $s_{t+1} \sim \hat{p}_\xi(s_{t+1} \mid s_t, a_t), \xi \in \Xi \subset \mathbb{R}^{d_\xi}, \xi \overset{\text{i.i.d}}{\sim} \mu.$

Sim-to-real gap: the worst-case L_1 Wasserstein distance $\max_{\xi\in\Xi} D_W(\hat{p}_\xi, p^\star)$ is bounded.

Task: find a policy π that satisfies $C_{p^\star}(\pi) \leq d$ only by interacting with simulated environments $\hat{p}_\xi, \xi \overset{\text{i.i.d}}{\sim} \mu.$

How to find a policy that satisfies the constraints without ever interacting with the real world?

Domain Randomization is Not Safe

Apply the simulation lemma:

$$C_{p^\star}(\pi) \leq \underbrace{\mathbb{E}_{\xi\sim\mu} C_{\hat{p}_\xi}(\pi)}_{\text{Constraint in simulation}} + \mathbb{E}_{\xi\sim\mu}\left[\mathbb{E}_{(s,a)\sim d_{\hat{p}_\xi,\pi}}\left[\tfrac{\gamma L_C}{1-\gamma} D_{W(\hat{p}_\xi,p^\star)}(s,a)\right]\right]$$

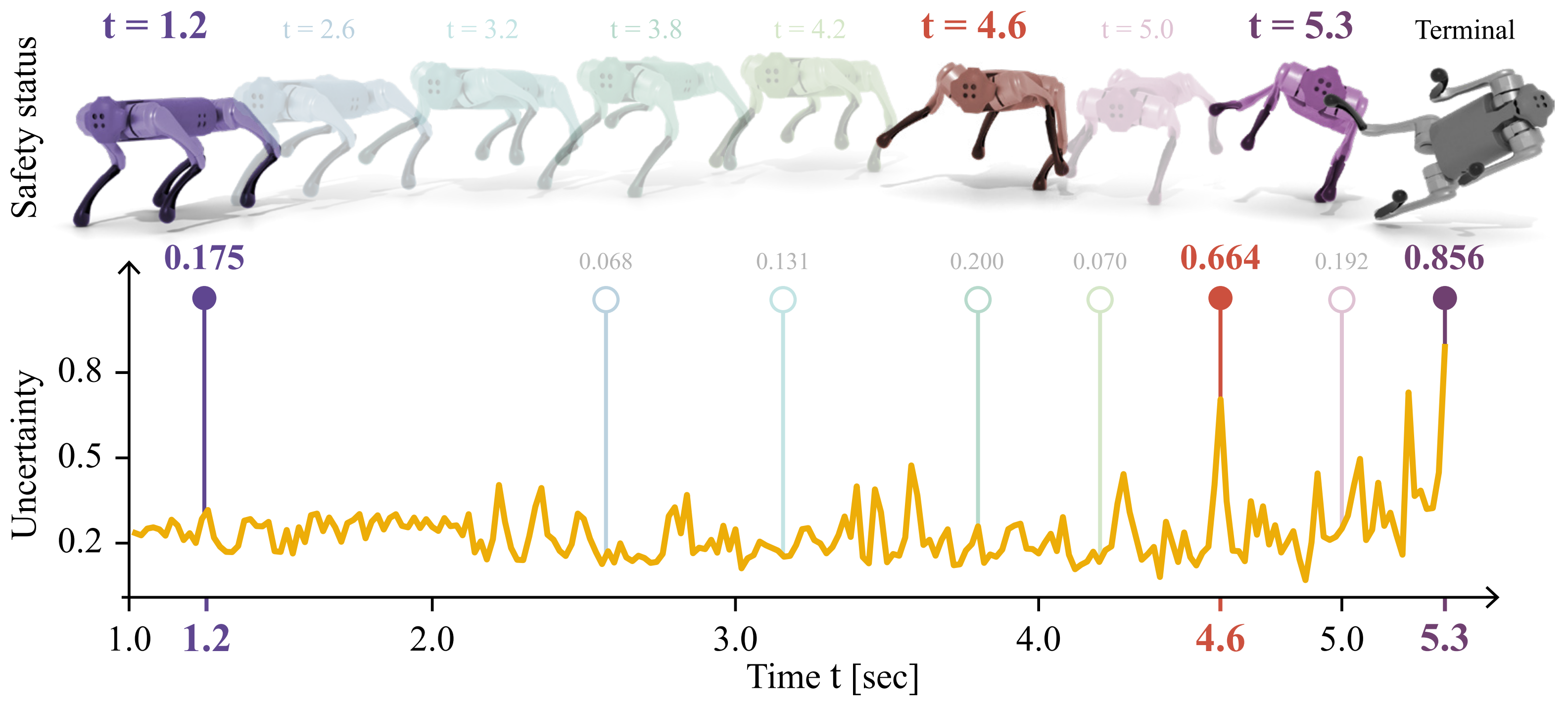
Reduction to penalized CMDPs:

$$\tilde{c}(s,a) = c(s,a) + \underbrace{\frac{\gamma L_C}{1-\gamma}\max_{\xi\in\Xi} D_W(\hat{p}_\xi, p^\star)(s,a)}_{\text{penalty}}$$

Solve in simulation:

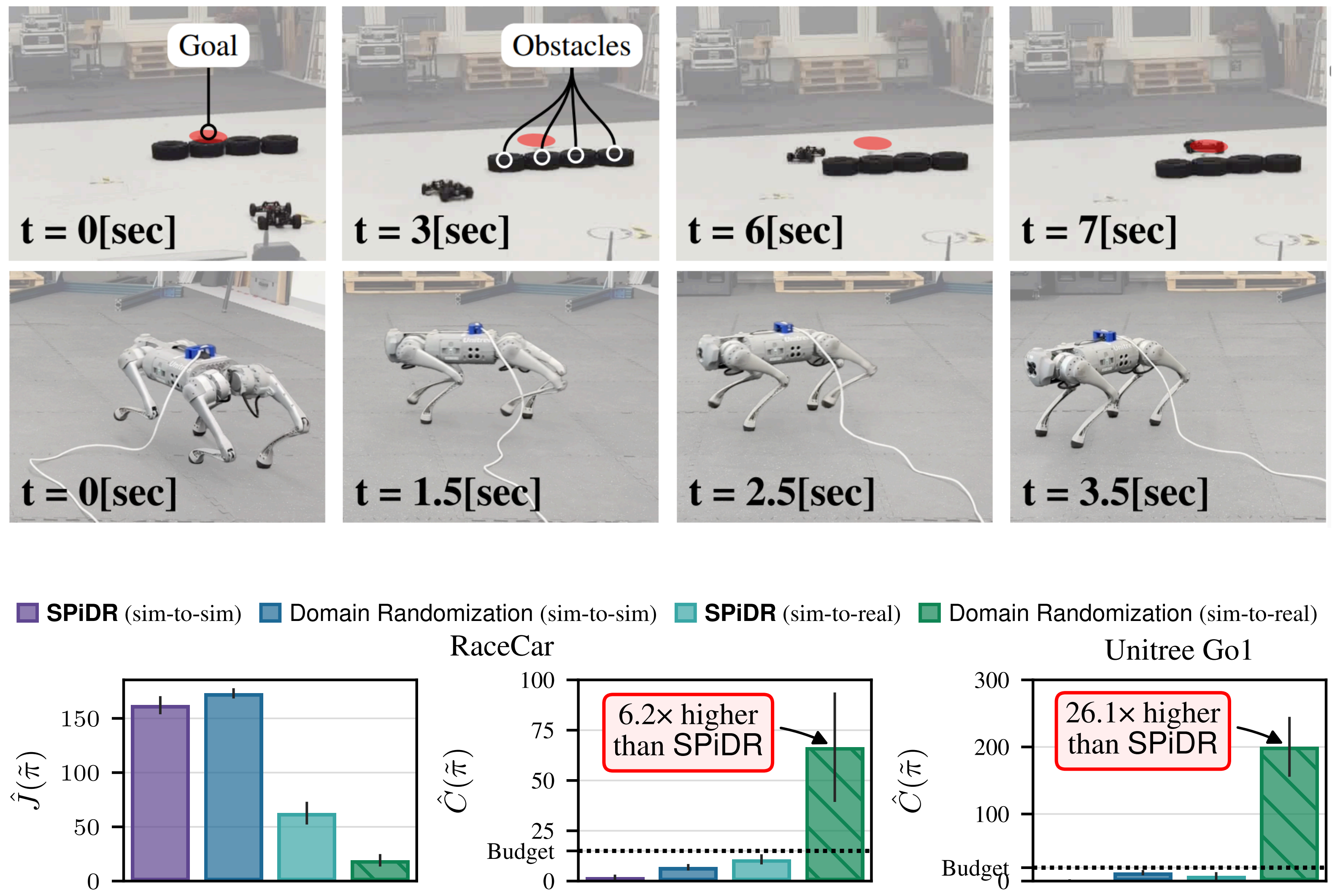
$$\max_{\pi\in\Pi} \mathbb{E}_{\xi\sim\mu} J_{\hat{p}_\xi}(\pi) \quad \text{s.t.} \quad \mathbb{E}_{\xi\sim\mu} \tilde{C}_{\hat{p}_\xi}(\pi) \leq d$$

Estimating the Penalty



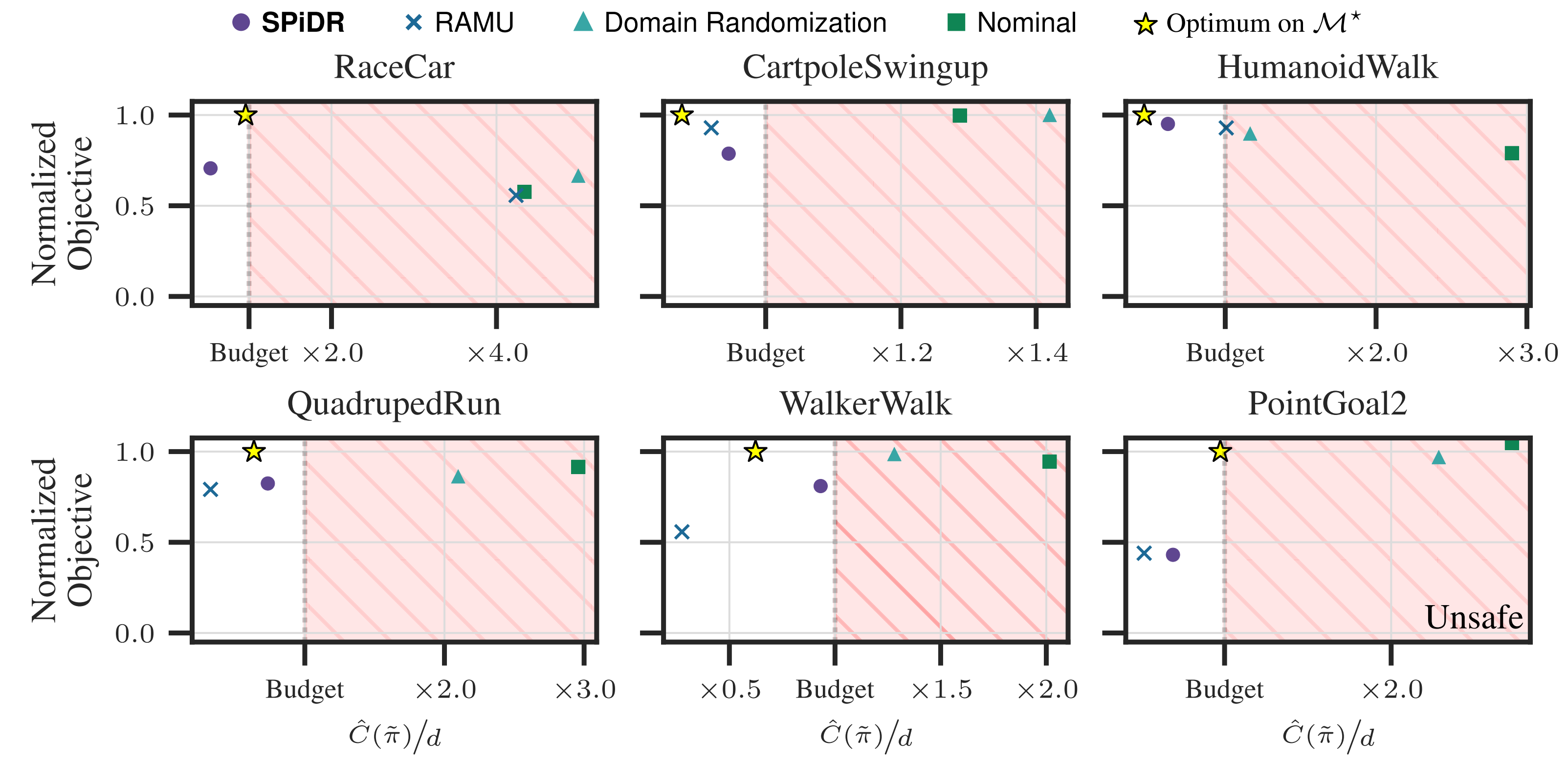
Key Idea: ensemble disagreement is a good heuristic.

Safety is Maintained Zero-Shot!

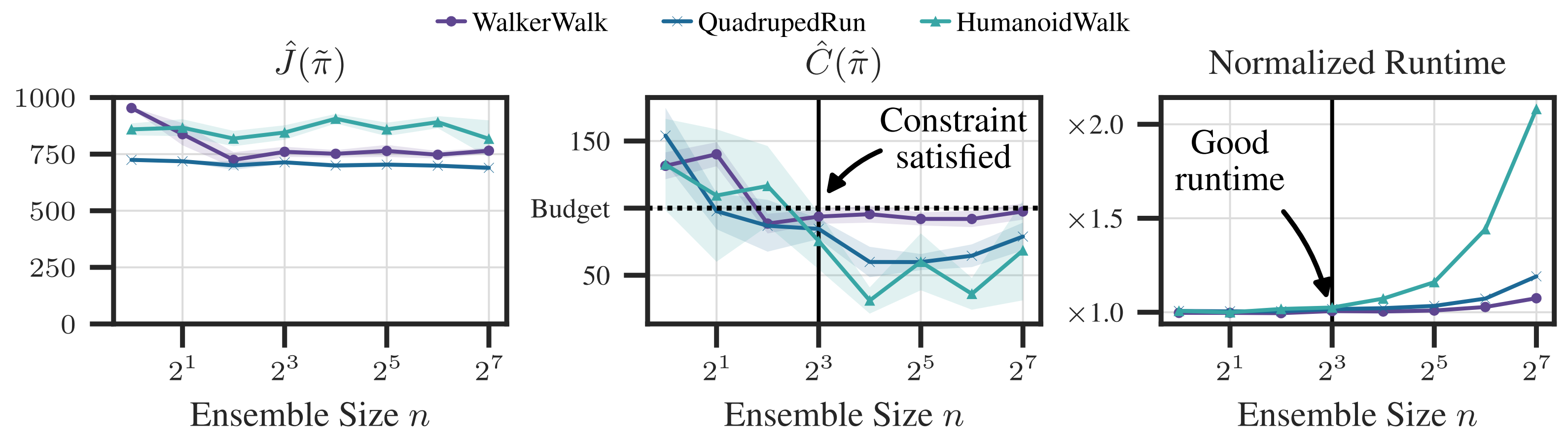


Does it work? Yes.
Check out videos @ yardenas.github.io/spidr

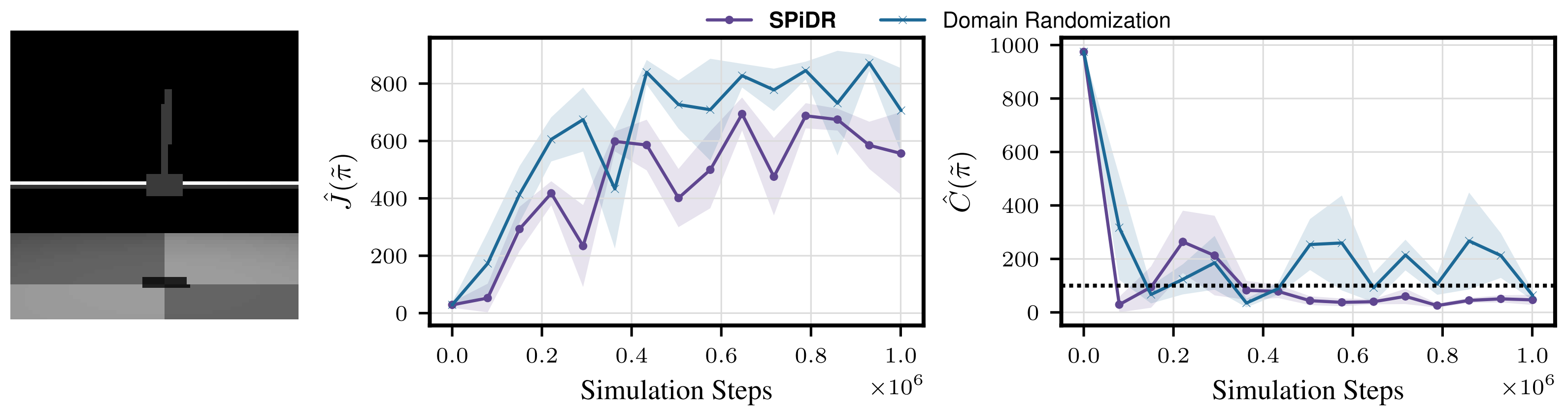
Extensive Evaluation



Fast Runtime



SPiDR Scales to Vision Control



Trick: compute disagreement on privileged state information, not directly on images.

Read the paper for a deeper dive into why it works and for more experiments
Open-source implementation @ https://github.com/yardenas/safe-learning