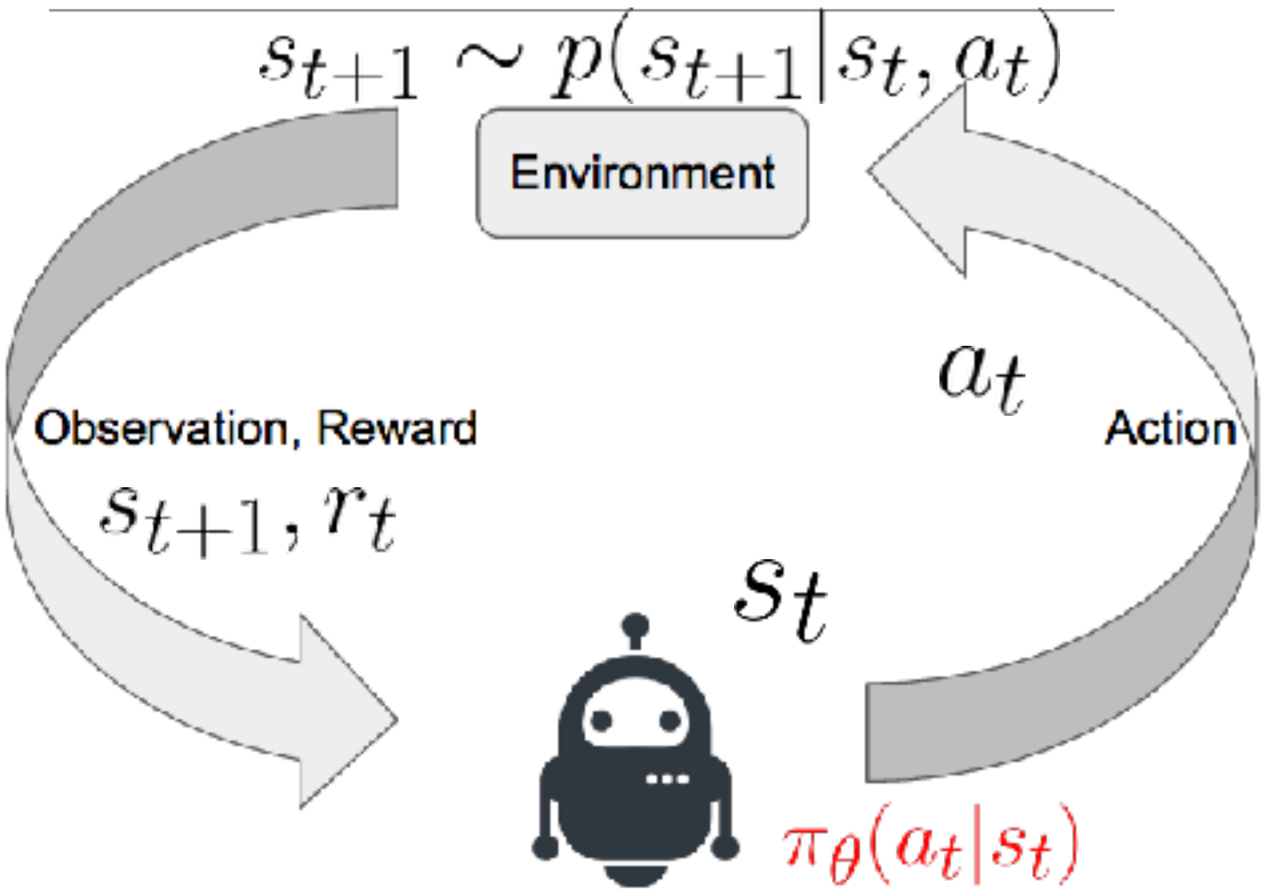
Q-Prop:
Toward Sample-Efficient & Stable Deep RL

Shixiang (Shane) Gu,
Timothy Lillicrap, Zoubin Ghahramani, Richard E. Turner, Sergey Levine

Reinforcement Learning



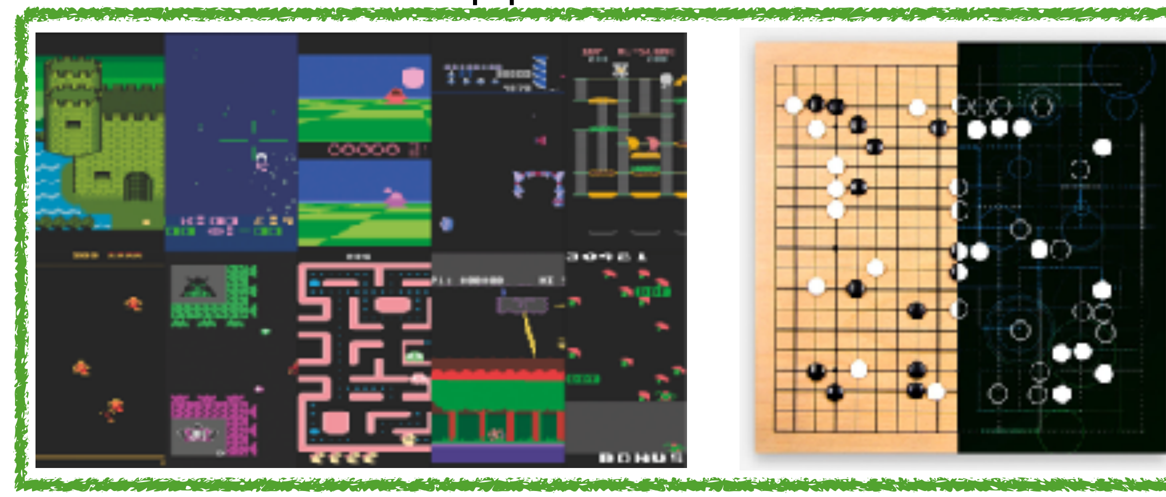
Goal: learn optimal policy that maximizes cumulative rewards

$$\max_{\pi} \mathbb{E}_{\pi}[\sum_{t=0}^{T} \gamma^t r(s_t, a_t)]$$

Potential applications

Computation bound

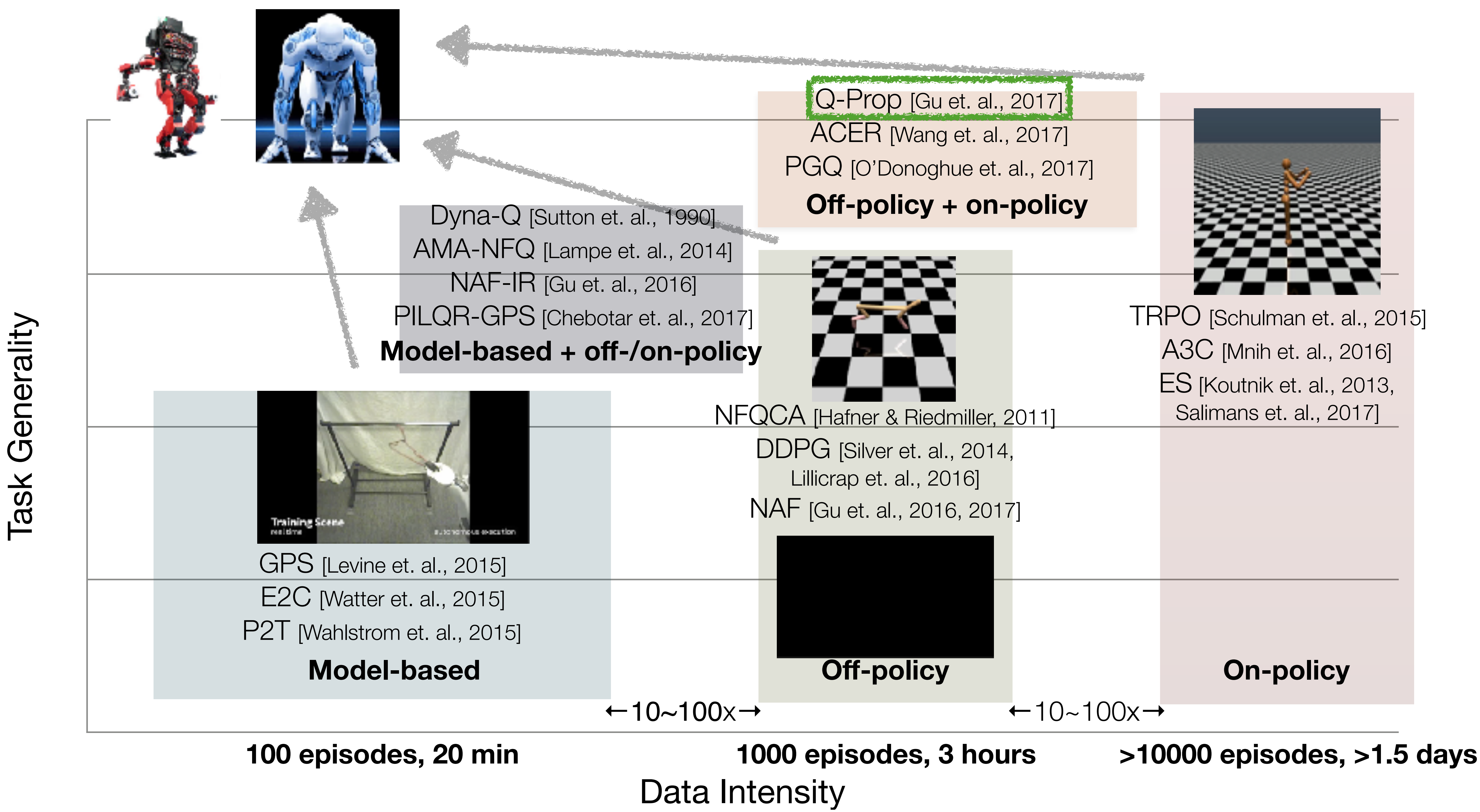
Simulation



Data bound

Real-world

Deep RL in Robotics



On-policy RL

Stability

On-policy MC policy gradient

$$\nabla_\theta J(\theta) = \mathbb{E}_\pi[\nabla_\theta \log \pi_\theta(a_t|s_t)\hat{Q}(s_t, a_t)]$$

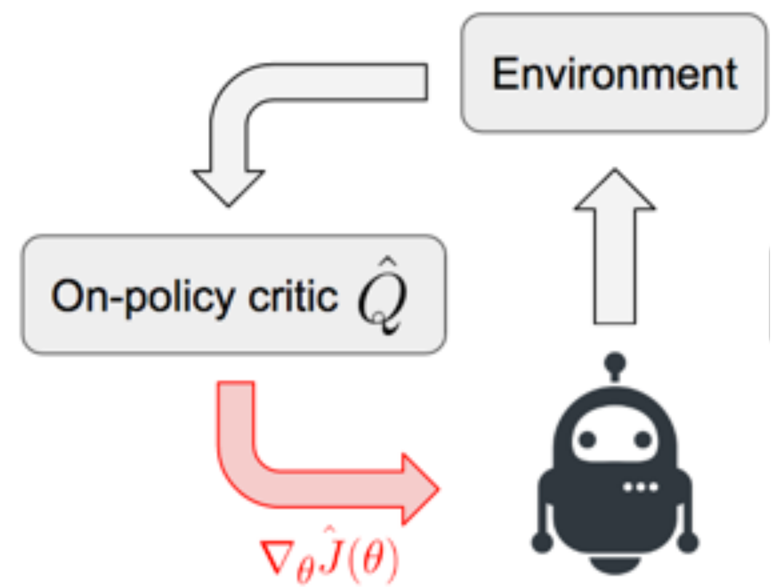
$$\hat{Q}(s_t, a_t) = \sum_{\tau \geq t} r(s_\tau, a_\tau)$$

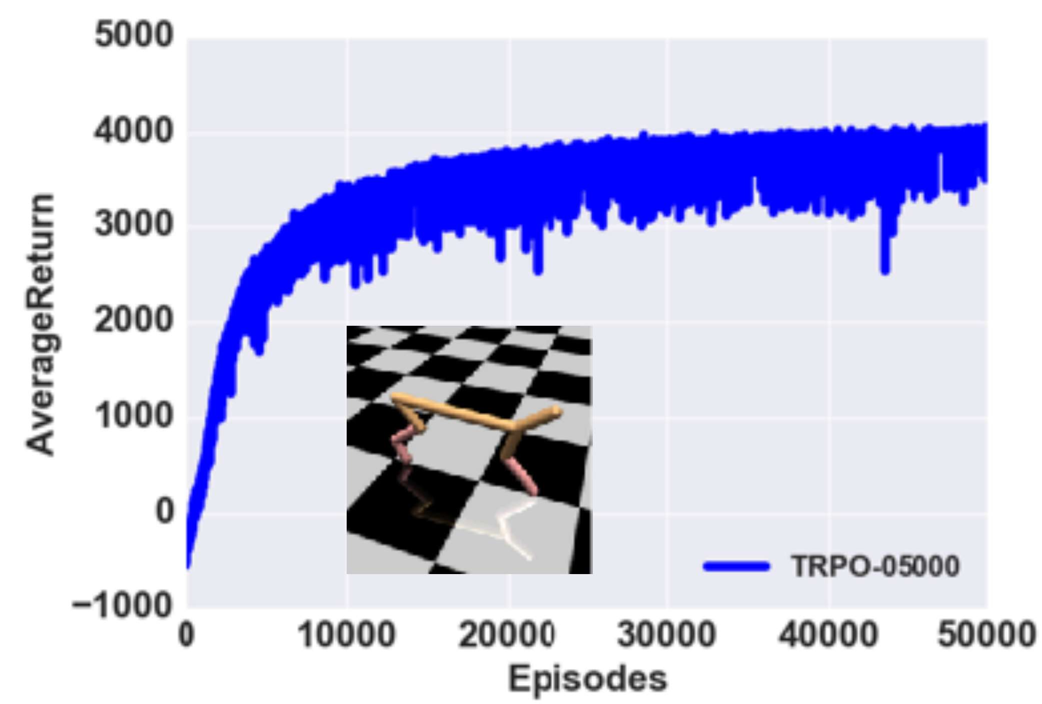
$$\pi_\theta(a|s) = \mathcal{N}(\mu_\theta(s_t), \Sigma_\theta(s_t))$$

$$\pi : \text{on-policy}$$

+ Unbiased gradient
+ Stable

- High-variance gradient
- Forgets experience
- Sample-intensive





Off-policy RL

Efficiency

Off-policy Actor-Critic

$$\min_w \mathbb{E}_\beta[(r(s_t, a_t) + \gamma Q(s_{t+1}, \mu_\theta(s_{t+1})) - Q_w(s_t, a_t))^2]$$

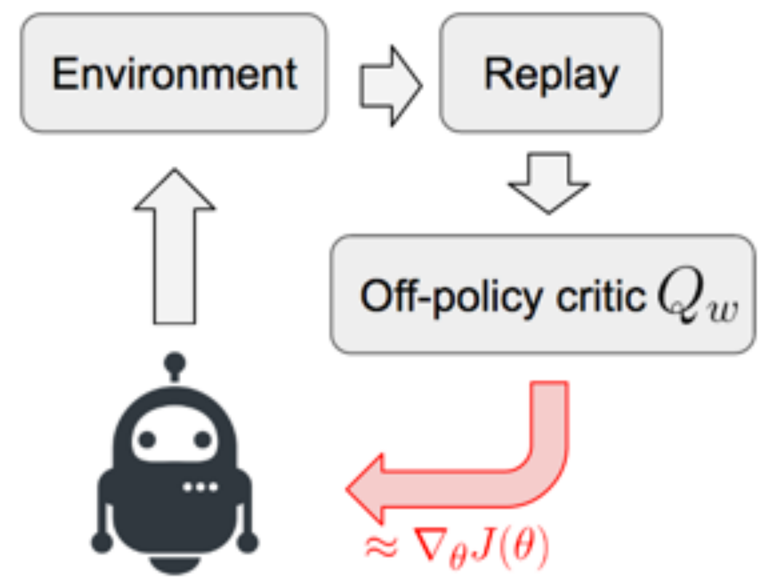
$$\max_\theta \mathbb{E}_\beta[Q_w(s_t, \mu_\theta(s_t))]$$

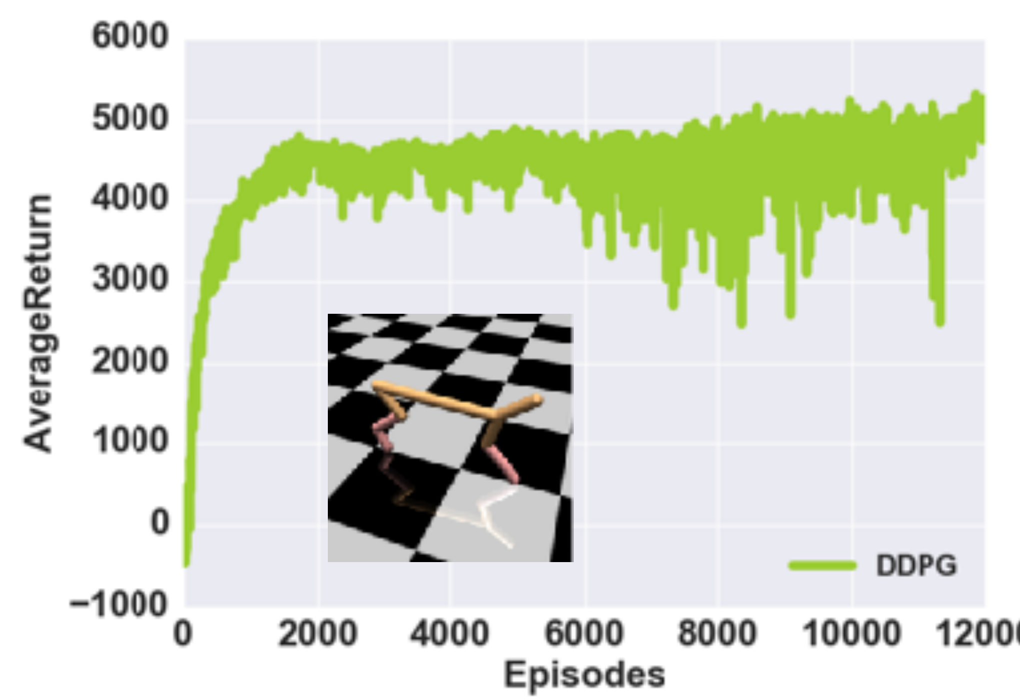
$$\nabla_\theta J(\theta) \approx \mathbb{E}_\beta[\nabla_a Q_w(s_t, a)|_{a=\mu_\theta(s_t)} \nabla_\theta \mu_\theta(s_t)]$$

β : off-policy

+ Low-variance gradient
+ Reuse experience
+ Sample-efficient (relatively)

- Biased gradient
- Less Stable

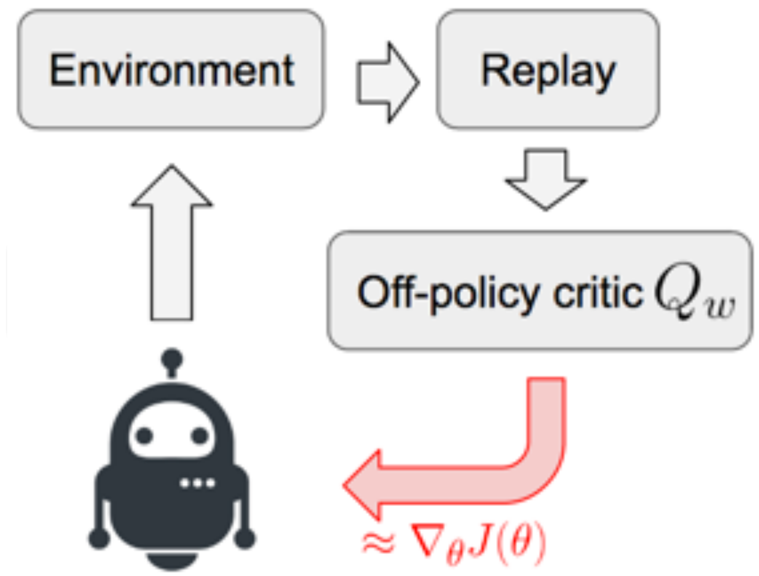




On-policy + Off-policy RL

Stability + Efficiency

Q-Prop

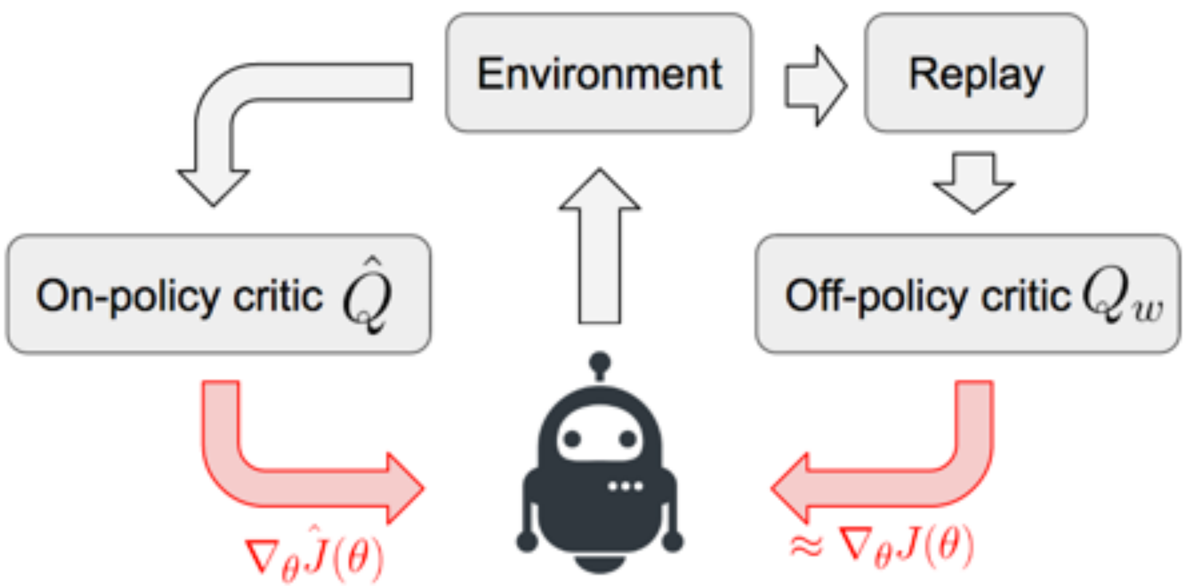


$$\nabla_\theta J(\theta) \approx \mathbb{E}_\pi[\nabla_a Q_w(s_t, a)|_{a=\mu_\theta(s_t)} \nabla_\theta \mu_\theta(s_t)]$$

On-policy + Off-policy RL

Stability + Efficiency

Q-Prop



$$\nabla_\theta J(\theta) = \mathbb{E}_\pi[\nabla_a Q_w(s_t, a)|_{a=\mu_\theta(s_t)} \nabla_\theta \mu_\theta(s_t)]$$
$$+\mathbb{E}_\pi[\nabla_\theta \log \pi_\theta(a_t|s_t)(\hat{Q}(s_t, a_t) - \bar{Q}_w(s_t, a_t))]$$

$\bar{Q}_w(s_t, a_t)$: first-order Taylor exp. of Q_w at $a_t = \mu_\theta(s_t)$

$$\nabla_\theta J(\theta) = \mathbb{E}_\pi[\nabla_a Q(s_t, a)|_{a=\mu_\theta(s_t)} \nabla_\theta \mu_\theta(s_t)]$$
$$+\mathbb{E}_\pi[\nabla_\theta \log \pi_\theta(a_t|s_t)(\hat{Q}(s_t, a_t) - Q_w(s_t, \mu_\theta(s_t)) - \nabla_{a_t} Q_w(s_t, a_t)|_{a_t=\mu_\theta(s_t)}(a_t - \mu_\theta(s_t)))]$$

Critic can be fitted **off-policy**.

Policy is fitted **on-policy**.

↓

+ unbiased policy gradient

+ low-variance grad from critic

+ sample-efficiency & stability

+modular

-more computation

-higher variance if critic is bad

Analysis

When does Q-Prop help? - When variance is reduced.

Q-Prop is a **control variate** [Ross, 2002]

- use a correlated variable with known expected value to reduce variance of an estimator

$$\bar{f} = \mathbb{E}[f(x)] = \mathbb{E}[f(x) - \eta g(x) + \eta \bar{g}] = \mathbb{E}[\tilde{f}(x)]$$

$$\mathrm{Var}(\tilde{f}) = \mathrm{Var}(f) + \eta^2 \mathrm{Var}(g) - 2\eta \mathrm{Cov}(f, g)$$

+ Large variance reduction if f & g strongly correlated

$$\eta^* = \mathrm{Cov}(f, g)/\mathrm{Var}(g) \longrightarrow \mathrm{Var}(\tilde{f}) = (1 - \rho(f, g)^2)\mathrm{Var}(f)$$

+ Guaranteed variance reduction if f & g are correlated

Analysis

Adaptive Q-Prop

$$\nabla_\theta J(\theta) = \mathbb{E}_\pi[\nabla_\theta \log \pi_\theta(a_t|s_t)(\hat{Q} - \eta(s_t)\bar{Q}_w)] \\ + \mathbb{E}_\pi[\eta(s_t)\nabla_a Q_w|_{a=\mu_\theta(s_t)}\nabla_\theta \mu_\theta(s_t)]$$

"Optimal" adaptation

$$\boxed{\eta^*(s_t) = \mathrm{Cov}(\hat{Q}, \bar{Q}_w)/\mathrm{Var}(\bar{Q}_w) \longrightarrow \mathrm{Var}(\hat{Q} - \eta^*\bar{Q}_w) = (1 - \rho(\hat{Q}, \bar{Q}_w)^2)\mathrm{Var}(\hat{Q})}$$

+ guaranteed reduction on learning signal variance

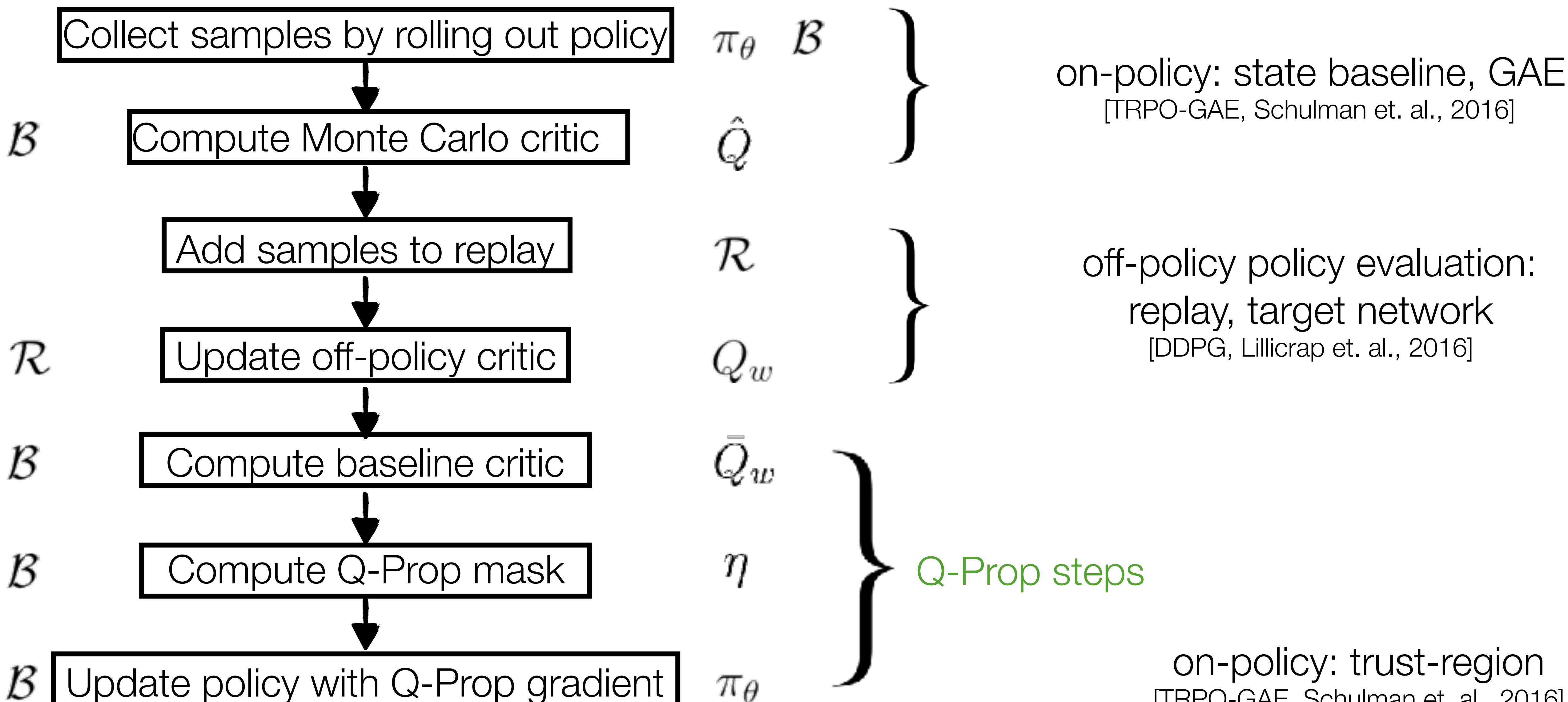
Conservative Q-Prop

$$\eta(s_t) = \begin{cases} 1, & \text{if } \hat{\mathrm{Cov}}(\hat{Q}, \bar{Q}_w) > 0 \\ 0, & \text{otherwise} \end{cases}$$

Q-Prop Diagram

$\mathcal{B}$: use on-policy batch samples

$\mathcal{R}$: use off-policy samples from replay



+allow any on-policy policy gradient & any policy evaluation

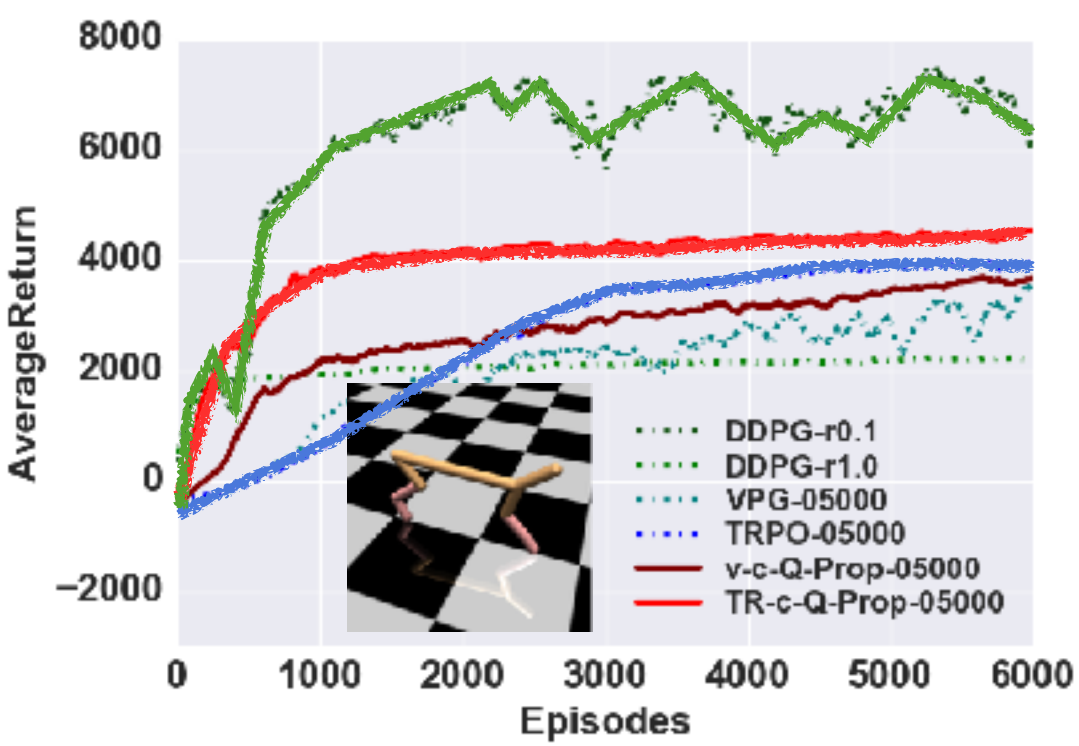
Experiments

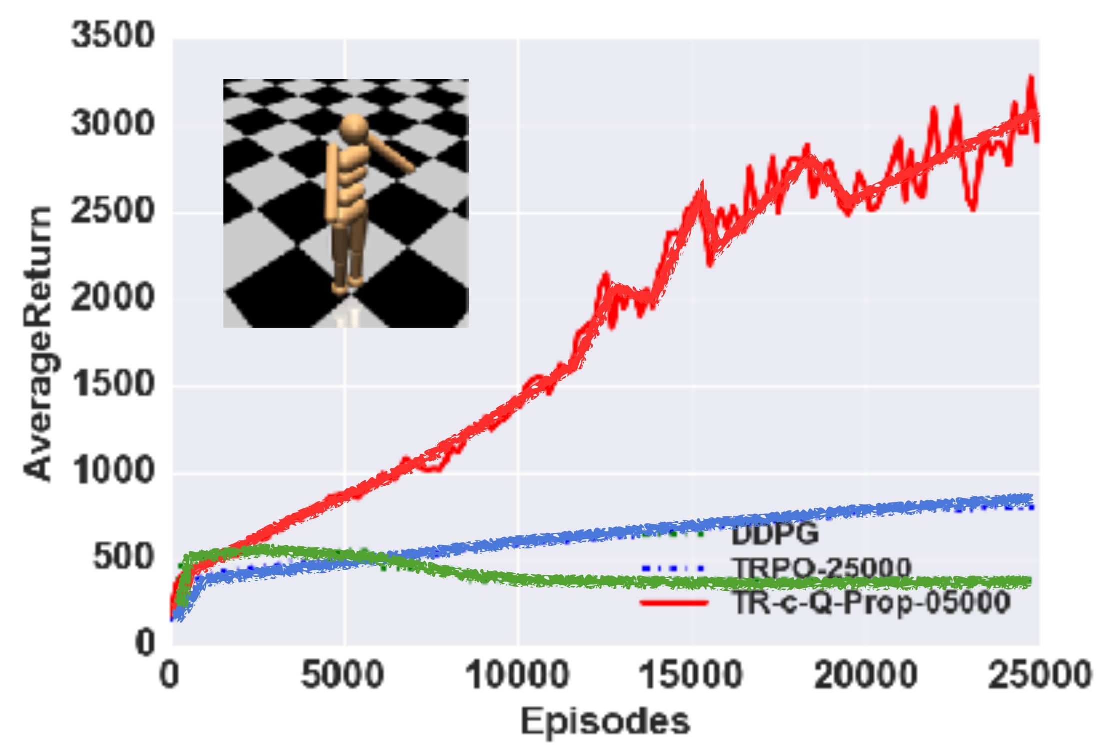
Q-Prop + TRPO-GAE vs. TRPO-GAE vs. DDPG

[Gu et. al., 2017]

SOA on-policy
[Schulman et. al., 2016]

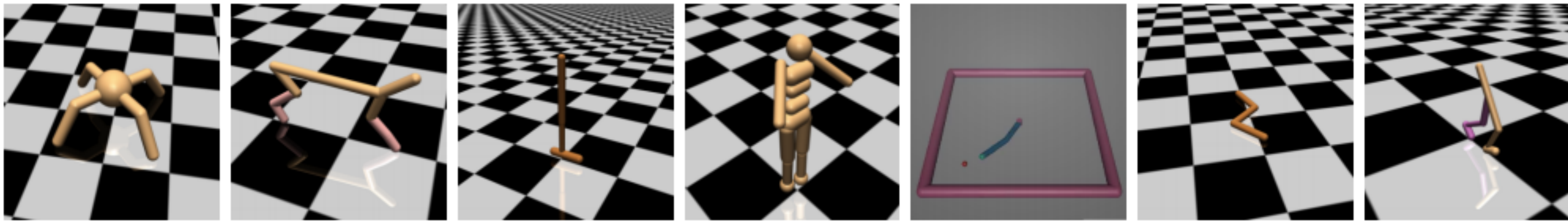
SOA off-policy
[Lillicrap et. al., 2016]





+ more sample-efficient than TRPO-GAE
+ more stable than DDPG
+ requires smaller batch size than TRPO-GAE

Experiments



MuJoCo, OpenAI Gym

		TR-c-Q-Prop		TRPO		DDPG	
Domain	Threshold	MaxReturn.	Episodes	MaxReturn	Epsisodes	MaxReturn	Episodes
Ant	3500	3534	**4975**	**4239**	13825	957	N/A
HalfCheetah	4700	4811	20785	4734	26370	**7490**	**600**
Hopper	2000	**2957**	5945	2486	5715	2604	**965**
Humanoid	2500	**>3492**	**14750**	918	>30000	552	N/A
Reacher	-7	**-6.0**	**2060**	-6.7	2840	-6.6	**1800**
Swimmer	90	103	2045	110	3025	**150**	**500**
Walker	3000	**4030**	3685	3567	18875	3626	**2125**

+ results appear consistent across multiple domains

Relation to other work

Q-Prop

$$\nabla_\theta J(\theta) = \mathbb{E}_\pi[\nabla_\theta \log \pi_\theta(a_t|s_t)(\hat{Q} - \bar{Q}_w)] + \mathbb{E}_\pi[\nabla_a Q_w|_{a=\mu_\theta(s_t)} \nabla_\theta \mu_\theta(s_t)]$$

Directly mixing on-policy and off-policy

$$\nabla_\theta J(\theta) \approx \nu \mathbb{E}_\pi[\nabla_\theta \log \pi_\theta(a_t|s_t)\hat{Q}] + (1-\nu)\mathbb{E}_\beta[\nabla_a Q_w|_{a=\mu_\theta(s_t)} \nabla_\theta \mu_\theta(s_t)]$$

Mixing on-policy and off-policy deep RL
- ACER [Wang et. al., 2017], PGQ [O'Donoghue et. al., 2017]

Take-away Messages

Q-Prop: take off-policy algorithm and correct it with on-policy algorithm on residuals

For RL:

- Toward sample-efficient & stable algorithm
- Toward off-policy policy gradient

For ML:

- An efficient, biased algorithm with a correct algorithm on the residuals
  - Stochastic discrete networks
    - MuProp [Gu et. al., 2016], REBAR [Tucker et. al., 2017]
  - Model-based RL?
    - PILQR [Chebotar et. al., 2017]
  - Synthetic gradients?
  - GANs?

Thank you!

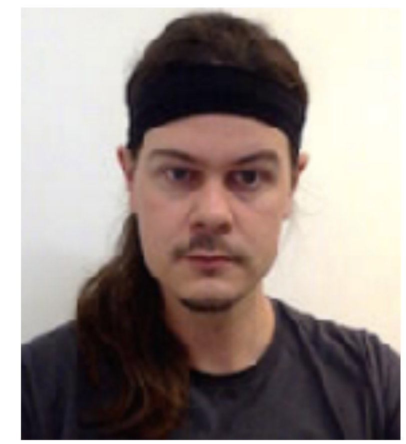
Google

UNIVERSITY OF CAMBRIDGE

UBER

Berkeley
UNIVERSITY OF CALIFORNIA

DeepMind



Acknowledgements:

openai/rllab
OpenAI Gym