Likelihood-free Inference in Reinforcement Learning: Relation to REPS and Sim-to-Real Applications

Kai Cui, Maximilian Hensel

September 26, 2019

What we will cover

- SimOpt (Chebotar et al. [2019]) for Sim-to-Real transfer
- REPS and its relation to
  - Approximate Bayesian Computation (ABC)
  - Variational Inference (VI)
- Improvement to SimOpt

SimOpt

Sim-to-Real transfer problem:

- Learn policy on simulation
- Use and tune policy on real system.

Vuong et al. [2019] formulate the problem as

$$\begin{aligned} \arg\max_{\phi} \quad & J_{z_{real}}(\pi_{\theta^*(\phi)}) \\ \text{s.t.} \quad & \theta^*(\phi) = \arg\max_{\theta} \mathop{\mathbb{E}}_{z \sim p_\phi} \left[J_z(\pi_\theta) \right]. \end{aligned} \tag{1}$$

Idea: If real system lies in class of simulated systems, the problem decomposes into:

- Find the correct simulator parameters
- Classic policy search

SimOpt

SimOpt (Chebotar et al. [2019]) alternates between:

- REPS for simulator parameter search
- TRPO for policy search

The likelihood-function is not available, hence trajectory distance between observations

$$\begin{aligned} d(x_z, x_{z_{real}}) = & w_{\ell_1} \sum_{i=0}^{T} |W(x_{i,z} - x_{i,z_{real}})| \\ & + w_{\ell_2} \sum_{i=0}^{T} \|W(x_{i,z} - x_{i,z_{real}})\|_2^2 \end{aligned}$$

is minimized.

Approximate Bayesian Computation (ABC)

- ABC can approximately sample from the posterior without likelihood function $\log p(x \mid z)$ (e.g. simulator)

ABC Rejection Sampling:

- Use discrepancy function $d(\cdot, \cdot)$ between samples and real observations to decide if samples from prior are accepted as posterior samples
- Accept if $d(x, x_{real}) < \varepsilon$
- As we let $\varepsilon \to 0$, we obtain true posterior samples

Approximate Likelihood-Free Inference with REPS

REPS is a two-part procedure:

- Generate weighted posterior samples
- Fit a distribution to samples

Difference in sampling:

- ABC Rejection Sampling rejects samples with high discrepancy
- REPS weights samples according to reward (discrepancy)

Variational Inference (VI) and REPS

VI usually solved by maximizing

$$\text{ELBO}(q) = \mathop{\mathbb{E}}_{z \sim q(z)} [\log p(x \mid z)] - D_{KL}(q(z) \parallel p(z)).$$

REPS maximizes

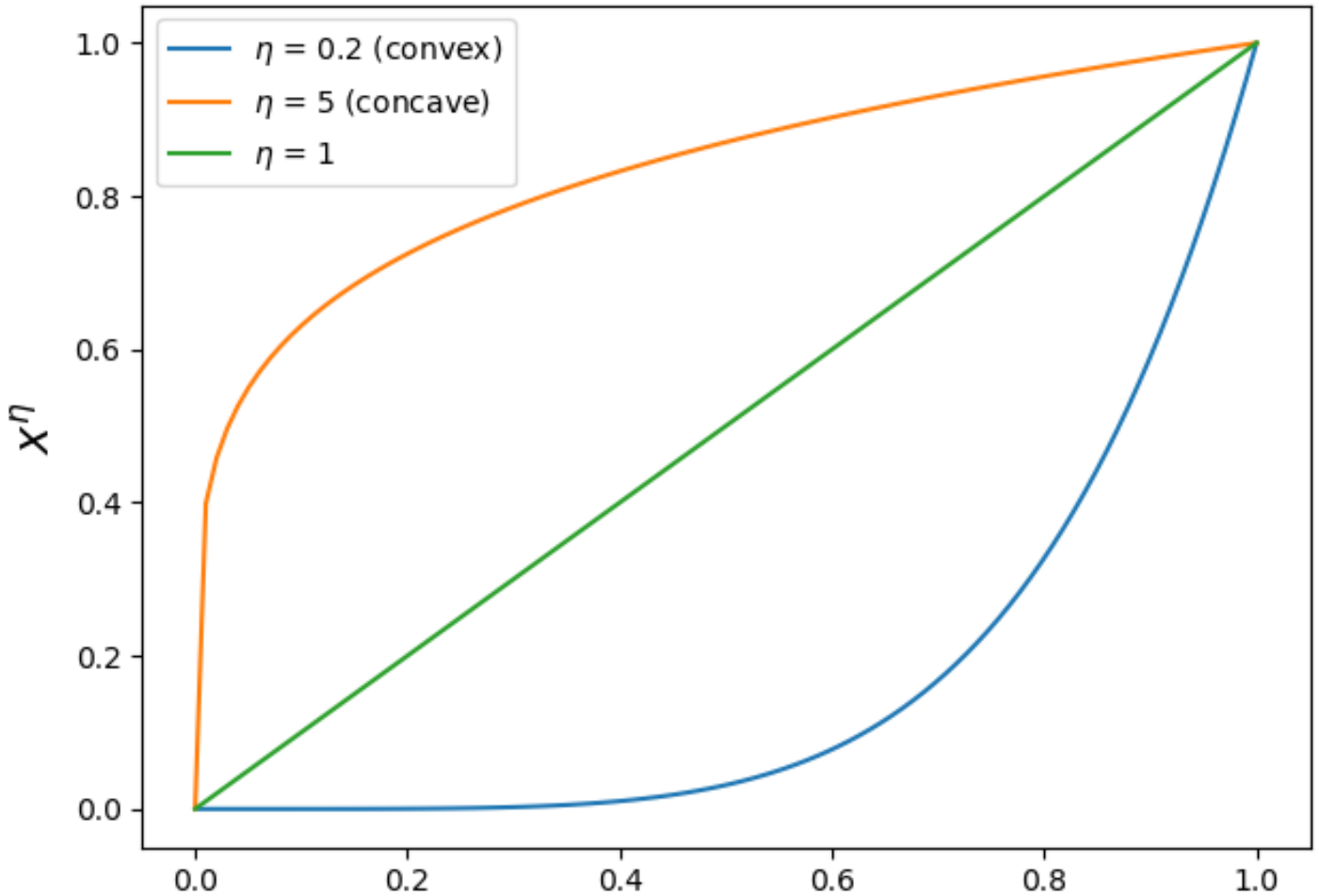
$$\mathop{\mathbb{E}}_{z \sim q(z)} [R(z)] - \eta(D_{KL}(q(z) \parallel p(z)) - \epsilon).$$

With $R(z) = \log p(x \mid z)$ we obtain

$$\underbrace{p_{n+1}(z \mid x)}_{\text{posterior}} \propto p_n(z) \exp\left(\frac{R(z)}{\eta}\right) = \underbrace{p_n(z)}_{\text{prior}} \underbrace{p_n(x \mid z)^{\frac{1}{\eta}}}_{\text{weighted likelihood}} .$$

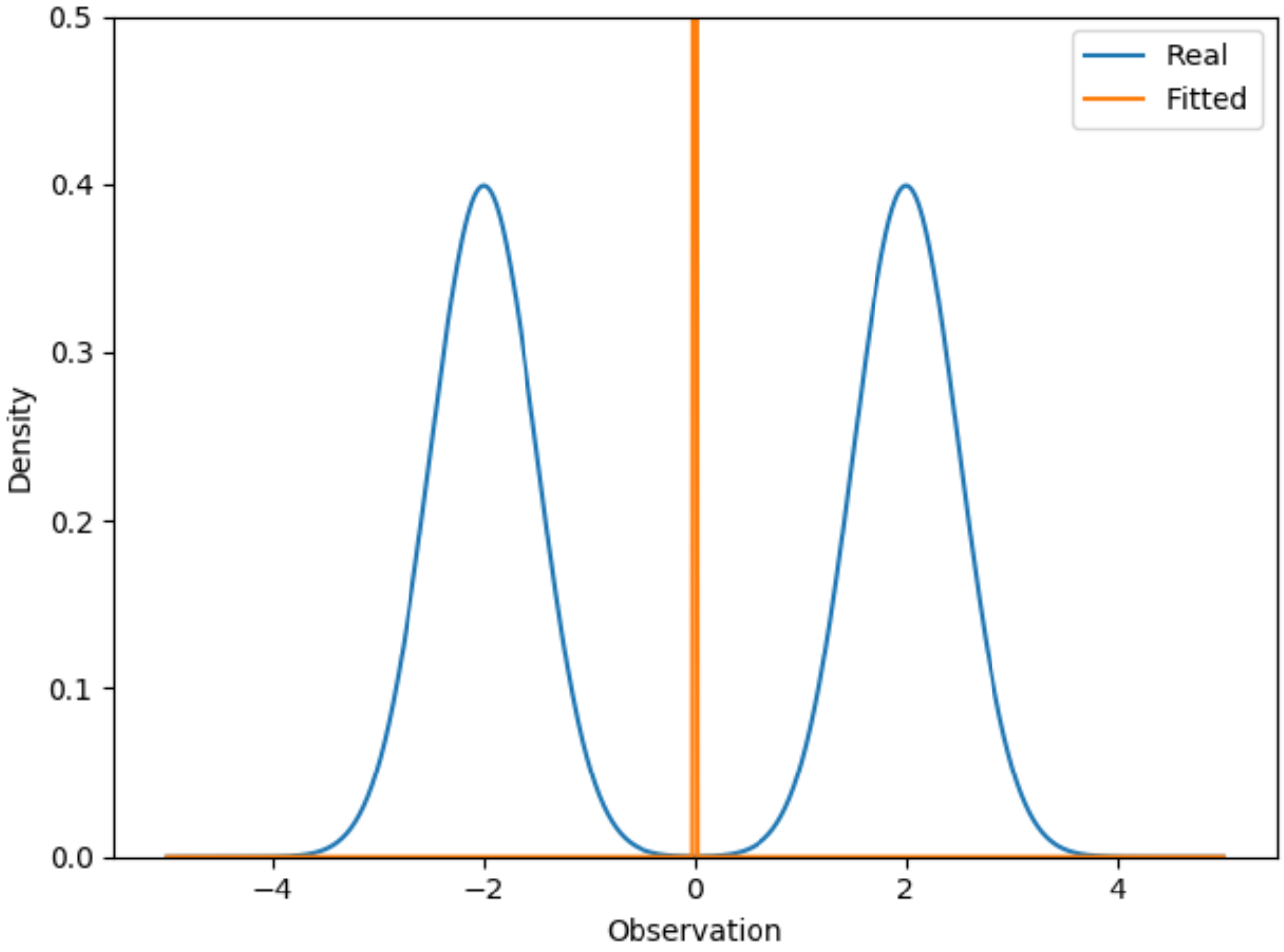
Impact Of η

$$\underbrace{p_{n+1}(z \mid x)}_{\text{posterior}} \propto p_n(z) \exp\left(\frac{R(z)}{\eta}\right) = \underbrace{p_n(z)}_{\text{prior}} \quad \underbrace{p_n(x \mid z)^{\frac{1}{\eta}}}_{\text{weighted likelihood}} .$$



Note on discrepancies

The choice of discrepancy implicitly assumes a model.



Step-based SimOpt

We use a transition discrepancy instead of a trajectory discrepancy. For simulator f_z parameterized by z, we choose

$$R(z) = - \mathop{\mathbb{E}}_{(s,a,s') \sim D} [(f_z(s,a) - s')^2]. \tag{2}$$

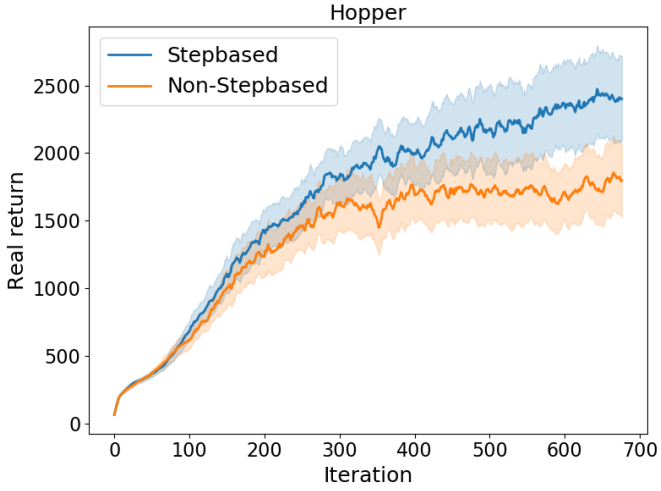
Advantage:

- Significantly improved sample efficiency

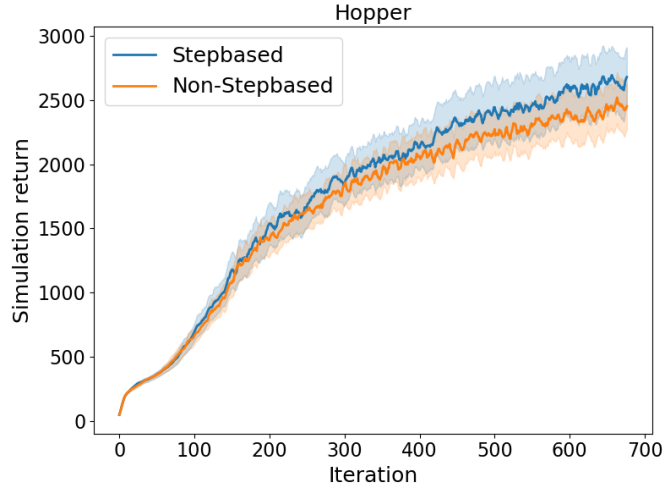
Drawback:

- Requires full state observations and state-settable simulator

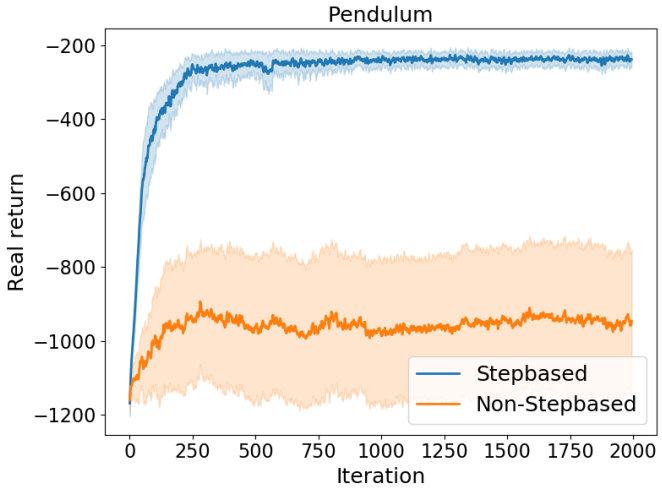
Step-based SimOpt Experiments



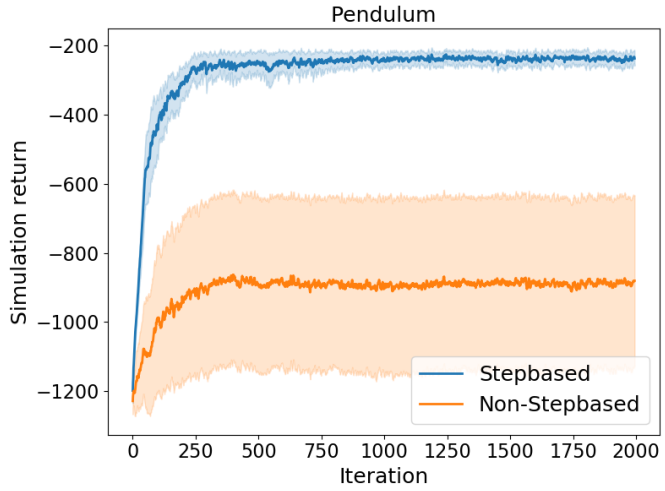
(a) Hopper real return



(b) Hopper sim return



(c) Pendulum real return



(d) Pendulum sim return

Thanks for your attention!

Yevgen Chebotar, Ankur Handa, Viktor Makoviychuk, Miles Macklin, Jan Issac, Nathan Ratliff, and Dieter Fox. Closing the sim-to-real loop: Adapting simulation randomization with real world experience. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 8973–8979. IEEE, 2019.

Quan Vuong, Sharad Vikram, Hao Su, Sicun Gao, and Henrik I Christensen. How to pick the domain randomization parameters for sim-to-real transfer of reinforcement learning policies? *arXiv preprint arXiv:1903.11774*, 2019.