Policy Learning and Evaluation With Randomized Quasi-Monte Carlo

Séb Arnold, Pierre L'Ecuyer, Liyu Chen, Yi-fan Chen, Fei Sha

Contact: seb.arnold@usc.edu **More information**: sebarnold.net/projects/qrl

Summary

- We propose to combine Policy Gradients with Randomized QMC.
- This yields several advantages:
  1. Retains the flexibility of policy gradient (continuous actions, non-differentiable objectives, non-linear policies, etc.)
  2. Improves policy learning and evaluation via variance reduction.
  3. Compatible with both policy gradient and actor-critic methods.
- Empirically, we show:
  - A. Better (~ 10x) in policy evaluation.
  - B. Faster convergence in policy learning.
  - C. Improves and combines other variance reduction techniques.

Policy Gradients

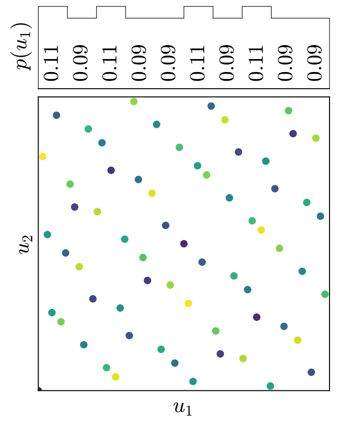
Iterate:

$$\pi \leftarrow \pi - \nabla_\pi \mathbb{E}_{s,a}[Q^\pi(s,a)]$$

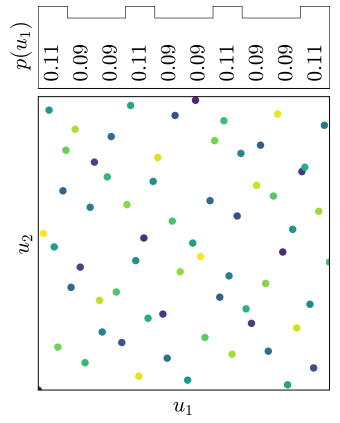
where $\pi(a \mid s) = \mu(s) + \sigma(s) \cdot F^{-1}(u), \quad u \sim U(0;1)$

Examples:

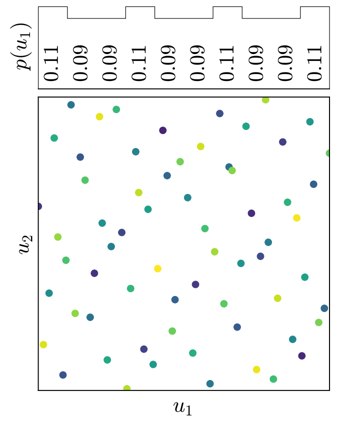
- Vanilla Policy Gradient (VPG):
  $$\nabla_\pi \mathbb{E}_{s,a}[Q^\pi(s,a)] \approx \mathbb{E}_{s,a}[Q^\pi(s,a)\nabla_\pi \log \pi(a \mid s)]$$
- Soft Actor-Critic (SAC):
  $$\nabla_\pi \mathbb{E}_{s,a}[Q^\pi(s,a)] \approx \mathbb{E}_{s,a}[\nabla_a Q^\pi(s,a)\nabla \pi(s \mid a)]$$



Sobol Pointset



Left Matrix Scramble



LMS + Digital Shift

Randomized Quasi-Monte Carlo

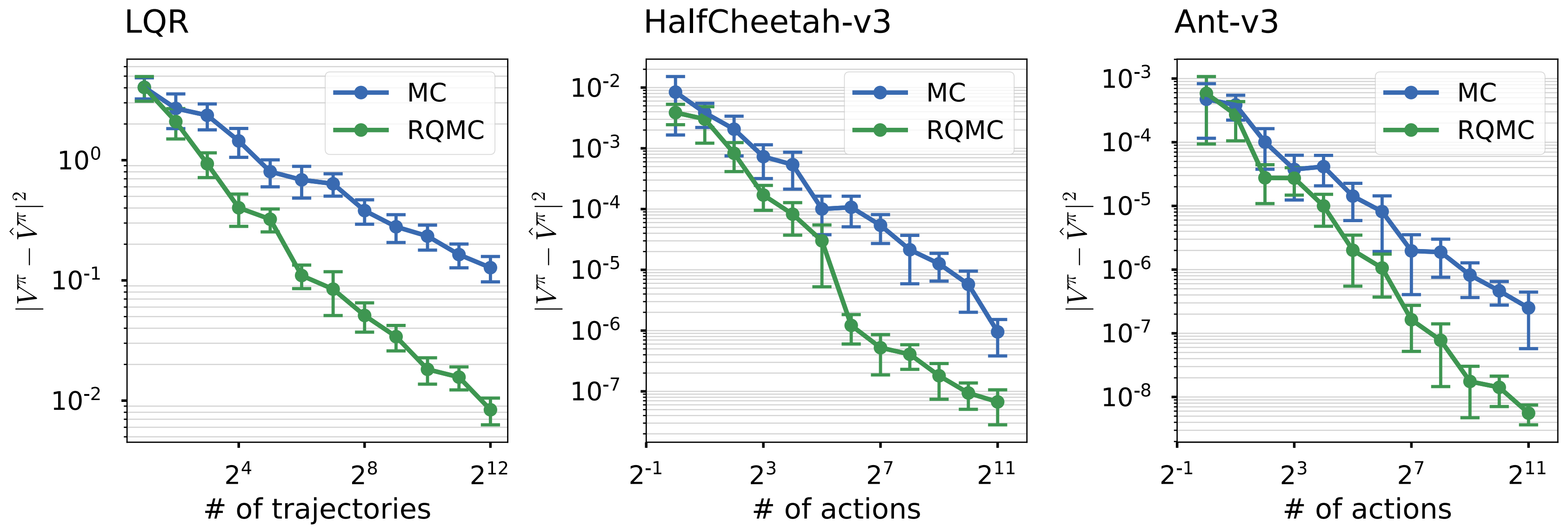
Monte-Carlo (MC; ~ $\mathcal{O}(N^{-1/2})$): $\mathbb{E}_{u \sim U(0;1)}[f(u)] \approx \frac{1}{N}\sum_{i=1}^{N} f(u^{(i)})$

Quasi-Monte Carlo (QMC; ~ $\mathcal{O}(N^{-1})$): replaces sampling with a deterministic, low discrepancy point set (e.g., Sobol).

Randomized QMC (RQMC; ~ $\mathcal{O}(N^{-3/2})$): randomizes deterministic point set with Left Matrix Scramble and a Digital Shift.

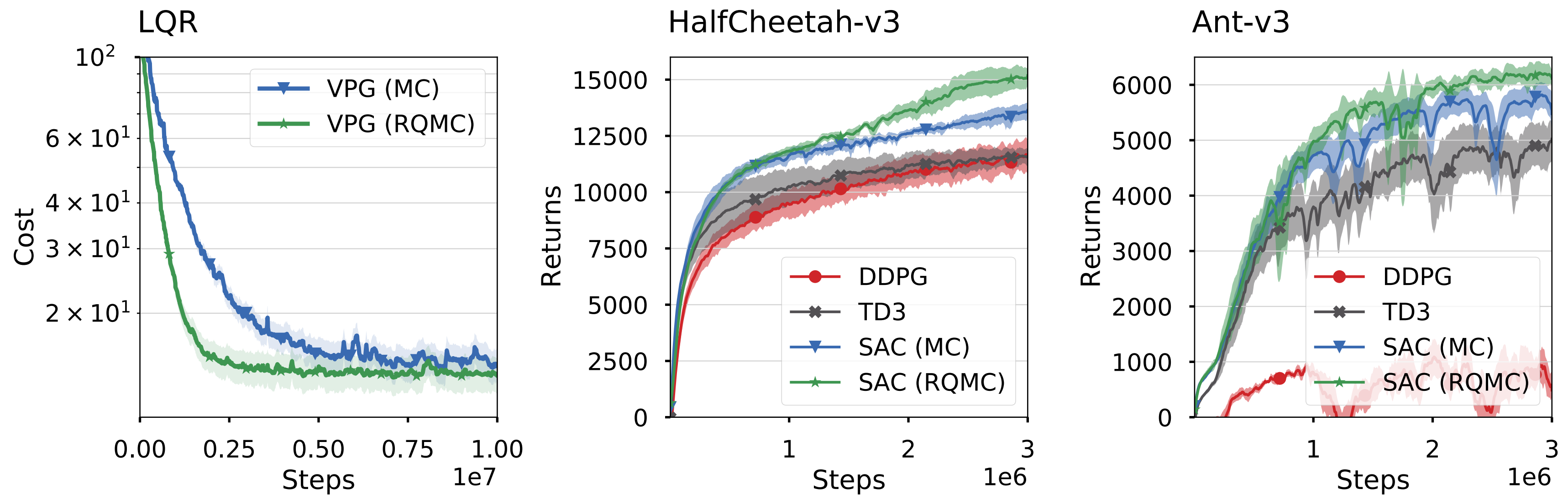
Policy Evaluation with RQMC

- RQMC: Replace $u \sim U(0;1)$ in policy with RQMC point set.
- Gather $N \leq 2^{12}$ trajectories, with MC or RQMC.
- Compare estimated value against ground-truth or with 2^{16} trajectories.
- **Results**
  - RQMC is more accurate on Brownian, LQR, and 5 MuJoCo tasks.



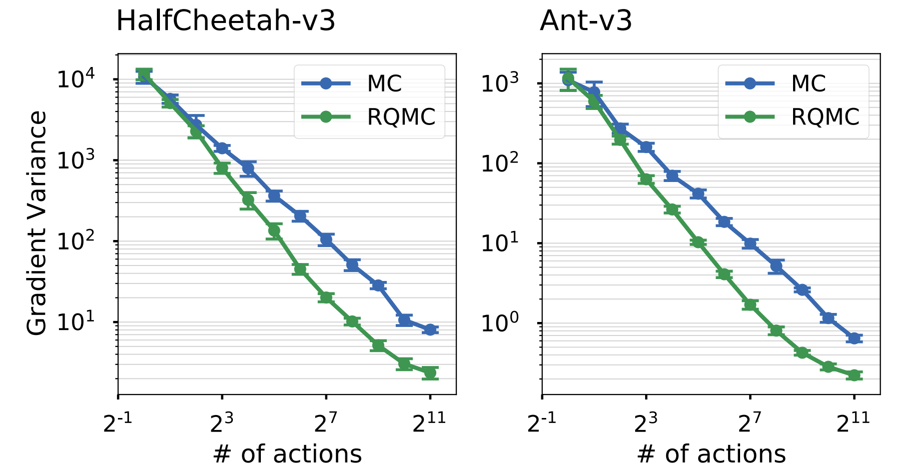
Policy Learning with RQMC

- VPG (RQMC): Roll out policy by sampling actions with RQMC, estimate $Q^\pi(s,a)$ with sum of (discounted) rewards.
- SAC (RQMC): Roll out policy as usual, estimate gradient of $Q^\pi(s,a)$ by sampling actions from policy with RQMC.
- Compare against MC policy gradient methods (e.g., DDPG, TD3, SAC).
- **Results**
  - RQMC learns faster than MC on LQR and 5 MuJoCo tasks.



Improved Gradient Estimation

- Compare (SAC) gradient variance as the number of sampled actions increases to 2^{11}.
- Use gradient estimated with 2^{16} actions as ground-truth.
- Repeat with 30 random seeds for confidence intervals.
- **Results**
  - RQMC is lower variance, and converges faster than MC.



Other Variance Reduction Techniques

- RQMC can be orthogonal to some other variance reduction techniques (VRTs), including control variates (CV) and variance-reduced optimization methods (e.g., ASGD).
- How does RQMC fare against VRTs, and can we combine them to get the best of both?
- **Results**
  - RQMC is as good as ASGD, better than CV on LQR.
  - Combination is best — RQMC complement other VRTs.