Stochastic Variance-Reduced Policy Gradient

M. Papini, D. Binaghi, G. Canonaco, M. Pirotta and M. Restelli
{matteo.papini, marcello.restelli}@polimi.it, {damiano.binaghi, giuseppe.canonaco}@mail.polimi.it, matteo.pirotta@inria.fr

POLITECNICO MILANO 1863

informatics mathematics Inria

Motivation

▶ We want to solve continuous Markov Decision Processes (MDPs), e.g., robot locomotion

▶ **Policy Gradient** (PG): optimize a parametric policy $\pi_{\boldsymbol{\theta}}$ via **gradient ascent** on performance $J(\boldsymbol{\theta})$.

▶ Two main strategies for gradient computation:

- **Full Gradient (FG)**: sample infefficient $\longrightarrow O(N/\epsilon)$
- **Stochastic Gradient (SG)**: slow convergence $\longrightarrow O(1/\epsilon^2)$

Sample efficiency is crucial in Reinforcement Learning (RL), where collecting samples is extremely expensive
$\implies$ **FG** often unfeasible

Slower convergence of **SG** due to high gradient variance.

▶ Solution from finite-sum optimization in Supervised Learning (SL):

- **Stochastic Variance-Reduced Gradient (SVRG)** $\longrightarrow O(N + N^{2/3}/\epsilon)$

SVRG in RL: challenges

Goal: replace SG in RL with SVRG

RL introduces three challenges

1. **Non-concavity**: policy performance $J(\boldsymbol{\theta})$ is typically a non-concave objective
2. **Infinite dataset**: expectation over all possible trajectories cannot be expressed as a finite sum
3. **Non-stationarity**: the data-generating distribution changes as we learn (policy changes)

What is out there?

From the **SL** literature, *separate* study of:

- **Non-concavity** [Allen-Zhu and Hazan, 2016, Reddi et al., 2016]
- **Infinite dataset** [Harikandeh et al., 2015, Bietti and Mairal, 2017]

From the **RL** literature:

- **SVRPGE** [Du et al., 2017]: policy *evaluation*
- **SVRPO** [Xu et al., 2017]: direct application of SVRG to TRPO

Non-stationarity never addressed explicitly!

SVRPG: stochastic variance-reduced policy gradient

We design an **SVRG**-like algorithm for the **PG** framework.

The easy part:

- Take any **unbiased** policy gradient estimator $g(\tau|\boldsymbol{\theta})$ (e.g., REINFORCE)
- **SVRG** idea: combine frequent **SG** with rare **FG** into a corrected gradient estimator $\blacktriangledown J(\boldsymbol{\theta})$
- Two-time scales: 1 **epoch** $= m$ **iterations**

Solving the RL challenges:

- Manage **non-concavity** with **smooth** policies (e.g., Gaussian)
- Approximate **infinite dataset** with a large **batch size** N
- Correct **nonstationarity** with **importance weighting**

Introducing the **SVRPG** estimator $\blacktriangledown J(\boldsymbol{\theta})$:

$$\blacktriangledown J(\boldsymbol{\theta}) := \widehat{\nabla}_N J(\widetilde{\boldsymbol{\theta}}) + \widehat{\nabla}_B J(\boldsymbol{\theta}) - \omega(\boldsymbol{\theta}, \widetilde{\boldsymbol{\theta}}) \widehat{\nabla}_B J(\widetilde{\boldsymbol{\theta}})$$

FG term
$\frac{1}{N}\sum_{i=1}^{N} g(\tau_i|\widetilde{\boldsymbol{\theta}})$
$\tau_i \sim \widetilde{\boldsymbol{\theta}}$
N trajectories
Accurate but expensive
↓
Updated once per **epoch**

SG term
$\frac{1}{B}\sum_{i=1}^{B} g(\tau_i|\boldsymbol{\theta})$
$\tau_i \sim \boldsymbol{\theta}$
$B << N$ trajectories
Computed at each iteration
Cheap but high-variance
↓
Stabilized by other terms

Correction term
$\frac{1}{B}\sum_{i=1}^{B} \underbrace{\omega(\tau_i|\boldsymbol{\theta}, \widetilde{\boldsymbol{\theta}})}_{\text{imp. weight}} g(\tau_i|\widetilde{\boldsymbol{\theta}})$
Trajectories from SG term
Ensures SVRG properties
Distribution mismatch
↓
importance weighting
↓
new source of variance

Fundamental SVRG properties:

1. **Unbiasedness:** $\mathbb{E}\left[\blacktriangledown J(\boldsymbol{\theta})\right] = \nabla J(\boldsymbol{\theta})$
2. **Vanishing variance:** $\mathbb{V}\mathrm{ar}\left[\blacktriangledown J(\boldsymbol{\theta})\right] \to 0$ as $\boldsymbol{\theta} \to \boldsymbol{\theta}^*$

Importance weighting

Use samples from $\boldsymbol{\theta}$ as taken with $\widetilde{\boldsymbol{\theta}}$:

$$w(\tau|\boldsymbol{\theta}, \widetilde{\boldsymbol{\theta}}) = \frac{p(\tau|\widetilde{\boldsymbol{\theta}})}{p(\tau|\boldsymbol{\theta})} \quad \begin{matrix} \leftarrow \textit{target distribution} \\ \leftarrow \textit{source distribution} \end{matrix}$$

Algorithm

For $s = 1, \ldots$
  Sample N trajectories using $\widetilde{\theta}$
  Compute FG $= \widehat{\nabla}_N J(\widetilde{\theta})$
  For $t = 1, \ldots, m$
    Sample B trajectories using θ
    Compute SG $= \widehat{\nabla}_B J(\theta)$
    Compute correction $= \omega(\theta, \widetilde{\theta}) \widehat{\nabla}_B J(\widetilde{\theta})$
    Update $\theta \leftarrow \theta + \alpha \blacktriangledown J(\theta)$ } iteration
  Update $\widetilde{\theta} \leftarrow \theta$ } epoch

Convergence

Non-concavity $\longrightarrow$ *local* maximum
Linear rate under increasing batch sizes:

$$\mathbb{E}\left[\|\nabla J(\boldsymbol{\theta})\|^2\right] \leq \frac{J(\boldsymbol{\theta}^*) - J(\boldsymbol{\theta}_0)}{\psi T} + \frac{\zeta}{N} + \frac{\xi}{B}$$

Infinite-dataset error from FG approximation
↓
take *large* N

Non-stationarity error from importance weighting
↓
take *large* B (still $<< N$)

- Constants (ψ, ζ, ξ) depend only on step size α and epoch size m
- Theory dictates very conservative meta-parameters (α, m, N, B) to guarantee convergence

Heuristics

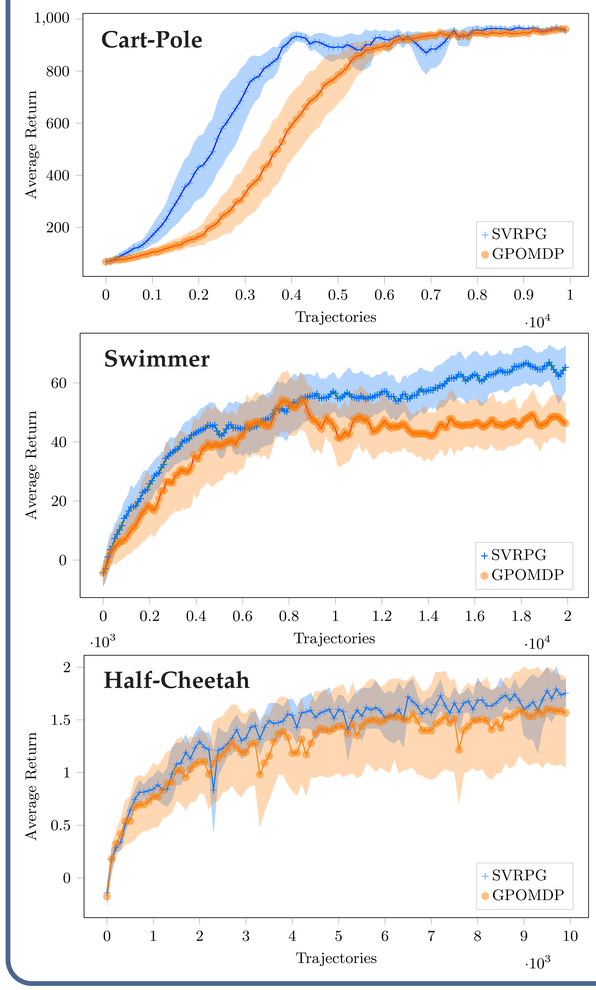
Heuristics can improve performance in practice

- Step size: **ADAM** to counteract gradient variance
  ⚠ FG and SG variance have completely different magnitude $\rightarrow$ use two separate annealing schedules α_{FG} and α_{SG}
- Epoch size: cut epoch when the effective step size becomes too small

$$\frac{\alpha_{FG}}{N} > \frac{\alpha_{SG}}{B} \implies \textbf{new epoch}$$

- Normalized importance weights: reduce variance at the price of introducing a small bias
- Critic: an orthogonal variance reduction technique from the **PG** literature

Empirical Results



References

Zeyuan Allen-Zhu and Elad Hazan. Variance reduction for faster non-convex optimization. In *International Conference on Machine Learning*, 2016.

Alberto Bietti and Julien Mairal. Stochastic optimization with variance reduction for infinite datasets with finite sum structure. In *Advances in Neural Information Processing Systems*, 2017.

Simon S. Du, Jianshu Chen, Lihong Li, Lin Xiao, and Dengyong Zhou. Stochastic variance reduction methods for policy evaluation. In *ICML*, volume 70 of *Proceedings of Machine Learning Research*, 2017.

Reza Harikandeh, Mohamed Osama Ahmed, Alim Virani, Mark Schmidt, Jakub Konečnỳ, and Scott Sallinen. Stopwasting my gradients: Practical svrg. In *Advances in Neural Information Processing Systems*, 2015.

Sashank J Reddi, Ahmed Hefny, Suvrit Sra, Barnabas Poczos, and Alex Smola. Stochastic variance reduction for nonconvex optimization. In *International conference on machine learning*, 2016.

Tianbing Xu, Qiang Liu, and Jian Peng. Stochastic variance reduction for policy gradient estimation. *CoRR*, abs/1710.06034, 2017.