Project proposal : Probabilistic Gradient Estimator for Policy Gradient Methods

Emiljo Mehillaj[*], Vasilii Kopylov[†], Jakub Mandula[‡], Christian Gasse[§]

Policy gradient methods are among the most effective methods in challenging reinforcement learning problems with large state and/or action spaces but suffer from a high variance of gradient estimates. Variance reduction techniques for solving non-convex problems have been recently explored for policy gradient-based problems [1], [2]. They showed an advantage over traditional policy gradient estimators (e.g., REINFORCE[3] and GPOMDP[4]) both in sample complexity and training stability. However, most of these methods have a large number of tunable hyperparameters. In this work, we introduce a novel algorithm - Probabilistic Gradient Estimator for Policy Gradient (PAGE-PG) - which has fewer hyperparameters and is expected to reach a superior convergence rate.

1 Introduction & Literature Review

Policy gradient (PG) methods learn the policy parameter based on the gradient of some scalar performance measure $J(\theta)$ with respect to the policy parameters [5]. One way of expressing $J(\theta)$ is as the expected total reward given below:

$$J(\theta) = E_{\tau\sim\pi_\theta} r(\tau) \tag{1}$$

where τ is a trajectory that contains states and actions involved in an episodic task. Usually, this problem is solved by gradient descent (GD) where the gradients are calculated from Equation (1) as:

$$\nabla J(\theta) = \nabla E_{\tau\sim\pi_\theta} r(\tau) = E_{\tau\sim\pi_\theta}\left[r(\tau)\nabla \log \pi_\theta(\tau)\right] \tag{2}$$

Computation of this gradient is costly and we need to estimate it. A straightforward estimator is the Monte Carlo estimator shown below:

$$\hat{\nabla} J(\theta) = \frac{1}{N}\sum_{i=0}^{N}\left[r(\tau)\nabla \log \pi_\theta(\tau)\right] \tag{3}$$

However, two common problems that arise during training via PG are the distribution shift and the high inherent training variance [6]. Therefore, several variance-reduced policy gradient methods have been proposed to address these problems. Some examples of such methods include:

1. **GPOMDP:** This method reduces the variance of the gradient estimator in Equation (3) through the subtraction of a state-dependent baseline [4]. We will refer to $d_{i,unb}(\theta)$ as the GPOMDP unbiased estimator per trajectory i that is in turn forward sampled according to the policy π_θ. The total GPOMDP estimator will be an average of all trajectory estimators.
2. **SVRG:** This estimator addresses the distribution shift issue in RL through the introduction of importance sampling between trajectories generated by different policy parametrizations [7].
3. **SARAH:** This method is similar to SVRG but is more memory efficient by waiving the requirement of storing past gradients [8].
4. **STORM-PG:** This gradient estimator combines the unbiased estimator introduced by GPOMDP and SARAH. A tunable hyperparameter controls the relative weights of these two estimators. This estimator showed improved gradient complexity in RL setting [1].

Recently, a novel stochastic gradient estimator has been introduced for non-convex approximation: Probabilistic Gradient Estimator (PAGE). This method relies on probabilistic switching between the vanilla SGD and SARAH and shows superior convergence results in models trained on MNIST, LeNet, etc. [9].

2 Methods

In this project, we will adapt PAGE for policy gradients in RL. To the best of our knowledge, a PAGE adjustment to the RL setting has not yet been implemented. Our motivation for the PAGE-PG lies in the favorable convergence results of PAGE in online settings. PAGE exhibits convergence rates that are on par with SARAH and STORM when the PL condition is not satisfied. When such a condition is met locally, the convergence rate becomes linear [9]. Another reason why PAGE is attractive for the RL setting is that we can fix the hyperparameter p_t to an empirically suitable value as done in the original PAGE paper. This is another advantage over most other variance-reduced methods which have a lot of trainable hyperparameters.

In this project, we will first implement the vanilla SGD part of our PAGE-PG using the Monte Carlo estimator shown in Equation (3). Then, drawing inspiration from STORM-PG, we will implement another version of PAGE-PG using the GPOMDP unbiased estimator. In this case, the PAGE-PG gradient estimate will look as shown below:

$$\mathbf{g}_{t+1} = \begin{cases} \frac{1}{b}\sum_{i\in I} d_{i,unb} \\ \text{with probability } p_t \\ \\ \mathbf{g}_{t+1} + \frac{1}{b'}\sum_{i\in I'}\left(d_{i,unb}\left(\theta_{t+1}\right) - d_{i,unb}^{\theta_{t+1}}(\theta_t)\right) \\ \text{with probability } 1-p_t \end{cases} \tag{4}$$

Furthermore, we plan on exploring better baselines other than the one used in GPOMDP to test the performance of our suggested algorithm.

3 Experiments

We will conduct a set of experiments to validate our expectations regarding PAGE-PG and benchmark its performance. We will compare it with the baseline algorithms such as SARAH, SVRG, GPOMDP, STORM-PG in the Cart-Pole and Mountain-Car tasks to evaluate both its computational complexity and convergence. And if time allows, we would also benchmark on larger-scale problems using the OpenAI Gym [10] or the Multi-Agent Emergence Environments [11].

[*]Emiljo Mehillaj emehillaj@student.ethz.ch 20-952-420
[†]Vasilii Kopylov vkopylov@student.ethz.ch - 20-943-825
[‡]Jakub Mandula jmandula@student.ethz.ch - 20-961-861
[§]Christian Gasse chgasser@student.ethz.ch - 15-830-516

References

[1] H. Yuan, X. Lian, J. Liu, and Y. Zhou, "Stochastic recursive momentum for policy gradient methods", *arXiv pre-print server*, 2020. DOI: Nonearxiv:2003.04302. [Online]. Available: https://arxiv.org/abs/2003.04302.

[2] M. Papini, D. Binaghi, G. Canonaco, M. Pirotta, and M. Restelli, "Stochastic variance-reduced policy gradient", in *International conference on machine learning*, PMLR, 2018, pp. 4026–4035.

[3] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning", *Machine learning*, vol. 8, no. 3, pp. 229–256, 1992.

[4] P. L. Bartlett and J. Baxter, "Infinite-horizon policy-gradient estimation", *CoRR*, vol. abs/1106.0665, 2011. arXiv: 1106.0665. [Online]. Available: http://arxiv.org/abs/1106.0665.

[5] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[6] M. Papini, D. Binaghi, G. Canonaco, M. Pirotta, and M. Restelli, "Stochastic variance-reduced policy gradient", *CoRR*, vol. abs/1806.05618, 2018. arXiv: 1806.05618. [Online]. Available: http://arxiv.org/abs/1806.05618.

[7] T. Xu, Q. Liu, and J. Peng, "Stochastic variance reduction for policy gradient estimation", *CoRR*, vol. abs/1710.06034, 2017. arXiv: 1710.06034. [Online]. Available: http://arxiv.org/abs/1710.06034.

[8] Lam, J. Liu, K. Scheinberg, and M. Taká˘, "Sarah: A novel method for machine learning problems using stochastic recursive gradient", *arXiv pre-print server*, 2017, c. DOI: Nonearxiv:1703.00102. [Online]. Available: https://arxiv.org/abs/1703.00102.

[9] Z. Li, H. Bao, X. Zhang, and P. Richtárik, "Page: A simple and optimal probabilistic gradient estimator for nonconvex optimization", *arXiv pre-print server*, 2020. DOI: Nonearxiv:2008.10898. [Online]. Available: https://arxiv.org/abs/2008.10898.

[10] G. Brockman, V. Cheung, L. Pettersson, *et al.*, "Openai gym", *arXiv preprint arXiv:1606.01540*, 2016.

[11] B. Baker, I. Kanitscheider, T. Markov, *et al.*, *Emergent tool use from multi-agent autocurricula*, 2020. arXiv: 1909.07528 [cs.LG].