# Probabilistic Gradient Estimators for Policy Gradient Methods

Emiljo Mehillaj
emehillaj@student.ethz.ch

Vasilii Kopylov
vkopylov@student.ethz.ch

Jakub Mandula
jmandula@student.ethz.ch

Christian Gasser
chgasser@student.ethz.ch

## Abstract

Policy gradient methods are among the most effective approaches 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. These techniques exhibit an advantage over traditional policy gradient estimators in sample complexity and training stability. In this work, we introduce a novel algorithm - Probabilistic Gradient Estimator for Policy Gradient (PAGE-PG) - which has fewer hyperparameters than other variance-reduced algorithms. In addition, we introduce a hybrid estimator, namely PAGESTORM-PG, whose theoretical convergence bounds promise practical improvements. Deep learning experiments on the Cart-Pole and Lunar Lander tasks confirm the practical advantages of our proposed variance reduction techniques over other state-of-the-art policy gradient algorithms.

## 1 Introduction

Reinforcement Learning (RL) is a framework where agents interact with the environment to maximize a cumulative reward based on a learnable policy [1]. There are two basic techniques to solving RL problems: model-based and model-free approaches [2]. A model-free problem is solved either by estimating a value function of a policy $\pi$ directly (e.g., Q-learning [3]), or by policy-based methods (e.g., Q-Prop [4]). Policy gradient (PG) methods learn the policy parameters 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:

$$J(\theta) = \mathbb{E}_{\tau\sim\pi_\theta} r(\tau) \tag{1}$$

where $\tau$ 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 (2) as:

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

The 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 that include several hyperparameters have been proposed to address these problems.

In this work, we propose two novel policy-gradient algorithms (PAGE-PG and PAGESTORM-PG) which adapt the PAGE gradient estimator for the PG setting. 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 [7]. Another reason why PAGE is attractive for the RL setting is that we can fix its hyperparameter (the switching probability) to an empirically suitable value as done in the original PAGE paper. In addition, drawing inspiration from the further exhibited variance-reduction in STORM, we propose a hybrid estimator, PAGESTORM-PG. With PAGESTORM-PG, we adapt PAGE-PG to include an exponential moving average SARAH instead of classical SARAH. We derive theoretical results on gradient complexity bounds for PAGESTORM-PG. Experiments on Cart-Pole and Lunar-Lander tasks show the experimental edge of PAGE-PG and PAGESTORM-PG compared to other algorithms.

This report is structured as follows. In Section 2, we give more detail about related work and our two proposed algorithms. Section 3 presents the results of this work. The implications of the obtained results and theoretical convergence bounds will be elaborated on in Section 4. We draw our conclusions in Section 5.

## 2 Models and Methods

In this section, we will introduce related groundwork and describe our line of work in detail. In Section 2.1, we explain some common variance-reduced pol-

icy gradient algorithms and classical PAGE. Then, in Section 2.2 and Section 2.3, we shift the scope of the discussion to our two proposed methods: PAGE-PG and PAGESTORM-PG.

## 2.1 Related Work

Variance-reduced policy gradient estimators aim at maximizing Equation 1. They are necessary substitutions to other gradient estimators because plain gradient-based algorithms do not account for the distribution shift in RL. Some variance-reduced algorithms include:

1. **GPOMDP:** This method reduces the variance of the gradient estimator in Equation (2) through the subtraction of a state-dependent baseline as shown below [8]:

$$\hat{\nabla} J(\theta) = \frac{1}{N} \sum_{i=1}^{N} d_i(\theta) \tag{4}$$

with:

$$d_i(\theta) = \sum_{h=0}^{H-1} \left( \sum_{t=0}^{h} \nabla \log \pi_\theta \left( x_t \mid a_t \right) \right) \left( \gamma^h r \left( x_h, a_h \right) - b_h \right) \tag{5}$$

where $b_h$ is a constant depending on the state $x_h$ and $d_i(\theta)$ is the GPOMDP unbiased estimator per trajectory. For our further discussion, we refer to $d_i(\theta)$ as the GPOMDP gradient estimator.

2. **SVRPG:** This estimator addresses the distribution shift issue in RL through the introduction of importance sampling between trajectories generated by different policy parametrizations [9] and can be written as:

$$\mathbf{g}_{t+1} = d_i \left( \theta_{t+1} \right) - d_i^{\theta_{t+1}}(\tilde{\theta}) + \tilde{u} \tag{6}$$

where $\tilde{u}$ is a fixed point estimation of the gradient calculated from snapshot policy parameters (which are updated less often than the current policy parameters), $d_i \left( \theta_{t+1} \right)$ is the GPOMDP gradient estimator with respect to the current policy parameters, and $d_i^{\theta_{t+1}}(\tilde{\theta})$ is the importance weight corrected GPOMDP gradient estimator with respect to the snapshot policy parameters at the current iteration. The importance weight correction is done according to the following formula: $d_i^{\theta'}(\theta) = \sum_{h=0}^{H-1} \frac{p(\tau_{i,h}|\theta)}{p(\tau_{i,h}|\theta')} d_{i,h}(\theta)$.

3. **SARAH-PG:** This method is a recursive version of SVRPG which uses previous time step policy parameters instead of a fixed snapshot policy. SARAH-PG can be written in the following way [10]:

$$\mathbf{g}_{t+1} = d_i \left( \theta_{t+1} \right) - d_i^{\theta_{t+1}} \left( \theta_t \right) + \mathbf{g}_t \tag{7}$$

4. **STORM-PG:** This gradient estimator combines the GPOMDP unbiased estimator and SARAH [11]. A tunable hyperparameter $\alpha$ controls the relative weighting between these two estimators as illustrated below:

$$g_{t+1} = (1-\alpha) \left[ d_i \left( \theta_{t+1} \right) - d_i^{\theta_{t+1}} \left( \theta_t \right) + \mathbf{g}_t \right] + \alpha d_i \left( \theta_{t+1} \right) \tag{8}$$

Recently, a novel stochastic gradient estimator was introduced for solving optimization problems in the non-convex regime: 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. [7].

**Algorithm 1** Page-PG

**Input** number of epochs $T$, large batch size $N$, min batch size $B$, initial parameter $\theta_0$
GPOMDP gradient estimator d
**Output**

Sample $N$ trajectories from $p(\cdot|\theta_0)$
$g_0 \leftarrow \frac{1}{N} \sum_{i \in N} d_i(\theta_0)$
**for** $i = 0$ to $T$ **do**
    $\theta_{t+1} \leftarrow \theta_t + \eta g_t$

$$g_{t+1} = \begin{cases} \textit{with probability } p\textit{:} \\ \text{sample } N \text{ trajectories from } p(\cdot|\theta_{t+1}) \\ \frac{1}{N} \sum_{i \in N} d_i(\theta_{t+1}) \\ \\ \textit{with probability } 1-p\textit{:} \\ \text{sample } B \text{ trajectories from } p(\cdot|\theta_{t+1}) \\ \mathrm{g}_t + \frac{1}{B} \sum_{i \in B} (d_i(\theta_{t+1}) - d_i^{\theta_{t+1}}(\theta+1)) \end{cases}$$

**end for**

## 2.2 PAGE-PG

Our goal is again to maximize Equation 1. An application of PAGE to the RL setting is not straightforward primarily due to the non-stationarity problem. Due to non-stationarity, straight application of the PAGE gradient estimator to the policy setting would make this estimator biased as the trajectories are forward sampled according to different distributions. A remedy to this problem is the introduction of importance sampling to account for the distribution shift [12].

Accounting for the distribution shift, we can propose the PAGE-PG algorithm provided in Algorithm 1. The structure of PAGE-PG is similar to standard PAGE proposed in [7] except for the importance weighting. PAGE-PG updates the gradient estimates according to a probabilistic switching between vanilla SGD and the SARAH estimator as demonstrated in

**Algorithm 2** Page-Storm-PG

**Input** number of epochs $T$, large batch size $S_0$, min batch size $B$, initial parameter $\theta_0$ GPOMDP gradient estimator d

**Output**

Sample $S_0$ trajectories from $p(\cdot|\theta_0)$

$g_0 \leftarrow \frac{1}{S_0}\sum_{i\in S_0} d_i(\theta_0)$

**for** $i = 0$ to $T$ **do**

$\theta_{t+1} \leftarrow \theta_t + \eta g_t$

$$g_{t+1} = \begin{cases} \textit{with probability } p\textit{:} \\ \text{sample } S_0 \text{ trajectories from } p(\cdot|\theta_{t+1}) \\ \frac{1}{S_0}\sum\limits_{i\in S_0} d_i(\theta_{t+1}) \\ \\ \textit{with probability } 1-p\textit{:} \\ \text{sample } B \text{ trajectories from } p(\cdot|\theta_{t+1}) \\ (1-\alpha)g_t + \frac{1}{B}\sum\limits_{i\in B}(d_i(\theta_{t+1}) - \\ (1-\alpha)d_i^{\theta_{t+1}}(\theta+1)) \end{cases}$$

**end for**

Line 5 of Algorithm 1. Compared to other variance-reduced policy gradient methods such as SVRPG, PAGE has a single-loop structure which allows for a flexible gradient update. In addition, the switching probability can be fixed to $p_t \equiv \frac{B}{B+N}$ since the authors in [7] obtained optimal convergence rates with such a value in online settings (such as reinforcement learning).

## 2.3 PAGESTORM-PG

In contrast to PAGE, the PAGESTORM-PG estimator arises from a probabilistic switching between vanilla gradient descent and STORM-PG. Our motivation for this estimator is in part due to the theoretical advantage of STORM-PG over other variance-reduced algorithms. Additionally, STORM rectifies the disparity between the empirical performance and theoretical bounds of SARAH [11]. Thus, by substituting the SARAH inner update present in PAGE-PG with STORM, we expect a gain in performance.

The blend of PAGE-PG and STORM-PG comes about organically due to the single loop structure that these two algorithms have as shown in the pseudocode of Algorithm 2. We also note that the addition of STORM into the structure of PAGE-PG comes at the expense of an additional hyper-parameter $\alpha$ as in Equation 8.

# 3 Results

### Experiments

We conducted a set of experiments to validate our expectations regarding PAGE-PG and PAGESTORM-PG performance. The algorithms were evaluated on reinforcement learning environments from OpenAI Gym including the Cartpole and Lunar Lander problems [13]. A Neural Softmax Policy was initialized with a fixed seed for a given run. For the baselines, we compared the proposed algorithms with the related algorithms: GPOMDP, SVRPG and STORM-PG.

The learning process was repeated 10 times with consistent but different initialization seeds. The runs were averaged, and plotted together with the run's standard deviation.

The optimal estimator hyper-parameters for learning rate $\eta$, probability $p$, and $\alpha$ were found by performing a grid search on the problems with a reduced number of repetitions. The summary of the used parameters can be found in Table 1.

| | Algorithm | Cartpole | Lunar L. |
|---|---|---|---|
| NN Size of Hidden Layers | - | 32 | $(64, 64)$ |
| NN activation | - | ReLu | Tanh |
| Task horizon | - | 200 | 200 |
| Total trajectories | - | 750 | 750 |
| Discount factor $\gamma$ | - | 0.98 | 0.99 |
| Learning rate $\eta$ | GPOMDP | $3 \times 10^{-3}$ | $3 \times 10^{-3}$ |
| | PAGE-PG | $2.5 \times 10^{-2}$ | $2 \times 10^{-3}$ |
| | PAGESTORM | $3 \times 10^{-3}$ | $2 \times 10^{-3}$ |
| | STORM-PG | $3 \times 10^{-3}$ | $2 \times 10^{-3}$ |
| | SVRPG | $3 \times 10^{-3}$ | $2 \times 10^{-3}$ |
| Estimator weight $\alpha$ | - | 0.9 | 0.9 |
| Probability $p$ | - | 0.9 | 0.9 |
| Batch size $N$ | - | 5 | 5 |
| Mini-Batch size $B$ | GPOMDP | 5 | 5 |
| | PAGE-PG | 3 | 3 |
| | PAGESTORM | 3 | 3 |
| | STORM-PG | 3 | 3 |
| | SVRPG | 3 | 3 |

Table 1: Hyper parameters used for Cartpole and LunarLander games

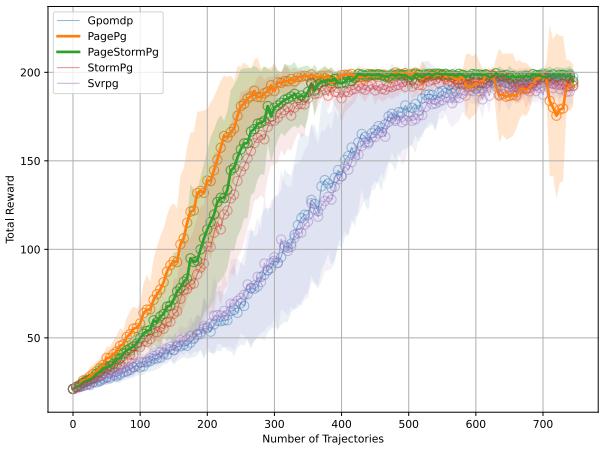


Figure 1: Best average learning rewards for cart_pole problem

### Comparison of different algorithms

In Figure 1, we notice that PAGE-PG shows the best performance under the Cart-Pole environment

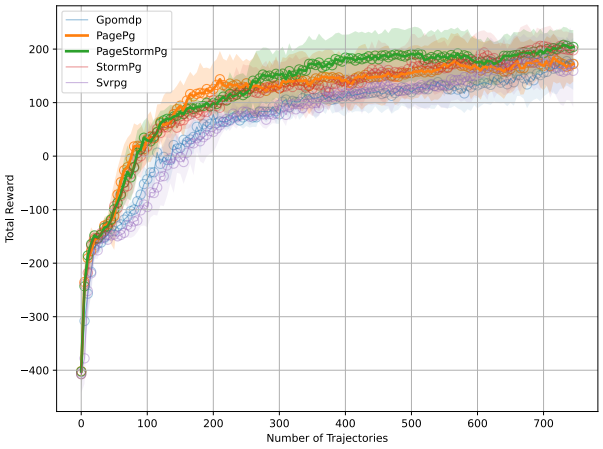


Figure 2: Best average learning rewards for lunar_lander problem.

setting followed by PAGESTORM-PG, STORM-PG, SVRPG, and GPOMDP. We see that PAGE-PG reaches the maximum value after approximately 400 trajectories while PAGESTORM-PG arrives at the maximum after 450 trajectories. Baseline algorithms lag behind: STORM-PG reaches the maximum at around 550 trajectories, SVRPG at 650, and GPOMDP at 700.

Under the Lunar-Lander environment setting, in Figure 2, we spot that PAGESTORM-PG outperforms other variance-reduced algorithms followed by STORM-PG, PAGE-PG, SVRPG, and GPOMDP. We notice that for the two games, both estimators show similar convergence speeds and variance.

## 4 Discussion

The results in Figure 1 and 2 reveal that, in general, our suggested PAGE-PG and PAGESTORM-PG algorithms have favorable performance compared to other variance-reduced algorithms. Comparing PAGE-PG and PAGESTORM-PG, we notice similar convergence properties. The extra hyperparameter involved in PAGESTORM-PG makes tuning harder and can lead to varying results. The satisfactory performance of PAGE-PG is expected given the smaller amount of tunable hyperparameters on the one hand and the possible time-to-time switch to a linear convergence rate on the other hand (as in classical PG). Even though we performed a grid search on the switching probability hyper-parameter $p$, we found out that empirically this hyper-parameter can still be fixed to $\frac{B}{B+N}$ and generate good results. To substantiate the empirical performance of PAGESTORM-PG, we will conduct a theoretical bound analysis in the next section.

### 4.1 Convergence Analysis

In this part, we present two convergence lemmas and a convergence theorem for PAGESTORM-PG. We refer the reader to Appendix A for the proofs. To state the following convergence bounds, we make assumptions about the boundedness (the rewards and the gradients with respect to the log policy are bounded by constants), the smoothness (the log policy gradient's Hessian norm, i.e, $\left\|\nabla^2_{\boldsymbol{\xi}} \log \pi_{\boldsymbol{\xi}}(a \mid s)\right\|$ is also bounded), finite-variance (the variance of the unbiased GPOMDP estimate in Equation 5 is bounded by $\sigma^2$), and finite IS variance (the variance of the importance sampling weight is also bounded). These are all valid assumptions in the RL setting.

Lemma 4.1 involves a recursive calculation of the gradient estimation error. The error between the PAGESTORM gradient estimator at a certain time point $g_{t+1}$ and the true gradient $\nabla_\theta J(\theta_{t+1})$ is bounded by $(1-p)\left[(1-\alpha)^2\right]$ times the estimation error of the previous iteration at time $t$ and some additional terms depending on the norm $||g_t||^2$ and discount-related constants $C_\gamma^2$.

**Lemma 4.1.** *Suppose that $g_t$ and $\theta_t$ are the iteration sequence as defined in Algorithm 2. $J(\theta)$ is the objective function to be optimized. Then the estimation error can be bounded by:*

$$\begin{aligned}&\mathbb{E}||g_{t+1} - \nabla_\theta J(\theta_{t+1})||^2 \leq \\ &(1-p)[(1-\alpha)^2\mathbb{E}||g_t - \nabla_\theta J(\theta_t)||^2 + \\ &\frac{2\alpha^2\sigma^2}{B} + \frac{2\eta^2}{B}(1-\alpha)^2 C_\gamma^2 \mathbb{E}||g_t||^2] + \\ &p\left[\frac{\sigma^2}{S_0}2\right]\end{aligned} \tag{9}$$

Given Lemma 4.1, we can follow by Lemma 2 which gives a calculation of the sum of the expected gradient estimator errors over time.

**Lemma 4.2.** *The accumulated sum of the expected estimation error can be bounded by:*

$$\begin{aligned}&\sum_{t=0}^{T-1} \mathbb{E}||g_t - \nabla_\theta J(\theta_t)||^2 \leq \frac{2}{2+p(1-\alpha)^2} \\ &[\frac{(1-p)\alpha^2\sigma^2}{B}T + \frac{Tp\sigma^2}{S_0} + \mathbb{E}||g_0 - \nabla_\theta J(\theta_0)||^2 + \\ &(1-p)(1-\alpha)^2\frac{C_\gamma^2\eta^2}{B}\sum_{t=0}^{T-1}\mathbb{E}||g_t||^2]\end{aligned} \tag{10}$$

Following the two lemmas, we can set forth our main convergence theorem as shown below:

**Theorem 4.3.** *When :*

$$1 - \frac{4\eta^2(1-p)(1-\alpha)^2 C_\gamma^2}{(\alpha + p(1-\alpha)^2)B} \geq 0 \tag{11}$$

*then, after $T$ iterations :*

$$\begin{aligned}\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}||\nabla_\theta J(\theta_t)||^2 &\leq \frac{2\Delta}{\eta T} + \frac{2\alpha\sigma^2}{B}\frac{1-p}{\alpha+p(1-\alpha)^2} \\ &+ \frac{2\sigma^2}{TS_0(\alpha+p(1-\alpha)^2)} + \frac{2p\sigma^2}{S_0}\frac{1}{\alpha+p(1-\alpha)^2}\end{aligned}$$

*(for p=0, it reduces to STORM-PG)*

Our main theorem concludes that after $T$ iterations, the expected gradient norm satisfies the provided bound. We also note that if we convert to STORM by setting the switching probability $p$ to 0, our theorem reduces to the analysis presented in the original STORM-PG [11]. This serves as a valuable validation check.

From our theorem, we can deduce two things about PAGESTORM-PG. First, we can see that after T iterates the algorithm reaches a point with expected gradient norm of order $\mathcal{O}\left(\frac{1}{T}+\frac{\sigma^2}{S_0}+\frac{\sigma^2}{T(B+(S_0-B)p)}\right)$ compared to STORM's $\mathcal{O}\left(\frac{1}{T}+\frac{\sigma^2}{S_0}+\frac{\sigma^2}{TB}\right)$. These results show that PAGESTORM-PG converges faster than STORM-PG given the same amount of iterations. Second, PAGESTORM-PG exhibits higher sample complexity compared to STORM-PG which might lead to empirical performance discrepancy. In PAGESTORM-PG, we have a sample complexity of $S_0+T(S_0p+B(1-p))$ compared to $S_0+TB$ of STORM (the complexity is higher in PAGESTORM-PG given $S_0 \geqslant B$).

## 5 Summary

In this work, we presented two novel algorithms: PAGE-PG and PAGESTORM-PG. PAGE-PG was inspired by a recently proposed variance-reduced algorithm called PAGE. We found that PAGE-PG and PAGE-STORM-PG surpassed the performance of other state-of-the-art counterparts. This behavior was expected given the theoretical and empirical advantages of classical PAGE. PAGESTORM-PG was motivated due to the superior convergence properties of STORM. Theoretically, we affirm that incorporating STORM into PAGE leads to superior gradient norm complexity. This superiority can also be witnessed in the empirical results for the Cart-Pole and Lunar-Lander environments. Future works may address the proof that PAGE-PG and PAGESTORM-PG enjoy faster (linear) convergence rates when the PL condition is locally satisfied for the objective function. Another direction of future efforts could relate to the establishment of theoretical lower bounds for our algorithms. Furthermore, the application of our two methods in other environments is necessary to investigate the environment translation properties of our algorithms.

## References


[1] R. S. Sutton, D. McAllester, S. Singh, and Y. Mansour, *Policy gradient methods for reinforcement learning with function approximation*, Conference Paper, 1999.

[2] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning", *Nature*, vol. 521, no. 7553, pp. 436–444, 2015, ISSN: 1476-4687. DOI: 10.1038/nature14539. [Online]. Available: https://doi.org/10.1038/nature14539.

[3] V. Mnih, K. Kavukcuoglu, D. Silver, *et al.*, "Human-level control through deep reinforcement learning", *Nature*, vol. 518, no. 7540, pp. 529–533, 2015, ISSN: 1476-4687. DOI: 10.1038/nature14236. [Online]. Available: https://doi.org/10.1038/nature14236.

[4] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning", *Machine Learning*, vol. 8, no. 3, pp. 229–256, 1992, ISSN: 1573-0565. DOI: 10.1007/BF00992696. [Online]. Available: https://doi.org/10.1007/BF00992696.

[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] 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.

[8] 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.

[9] 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.

[10] 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.

[11] 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.

[12] D. Precup, R. S. Sutton, and S. Singh, "Eligibility traces for off-policy policy evaluation", in *ICML*, 2000.

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

## A Convergence proofs

**Proof of Lemma 4.1**

*Proof.*

$$\mathbb{E}||g_{t+1} - \nabla_\theta J(\theta_{t+1})||^2 = \mathbb{E}(\mathbb{E}(||g_{t+1} - \nabla_\theta J(\theta_{t+1}||^2|F_t)) \tag{12}$$

Where $F_t$ is the information before time t

$$\begin{aligned}\mathbb{E}(||g_{t+1} - \nabla_\theta J((1-\alpha))||^2|F_t) = (1-p)\mathbb{E}(||(1-\alpha)(g_t + \frac{1}{B}\sum_{i\in B}(d_i(\theta_{t+1}) - d_i^{\theta_{t+1}}(\theta_t)))+ \\ \frac{1}{B}\sum_{i\in B}\alpha d_i(\theta_{t+1}) - \nabla_\theta J(\theta_{t+1})||^2|F_t)+ \\ p\mathbb{E}(||\frac{1}{S_0}\sum_{i\in S_0} d_i(\theta_{t+1}) - \nabla_\theta J(\theta_{t+1})||^2|F_t) \\ \leq (1-p)((1-\alpha)^2||g_t - \nabla_\theta J(\theta_t)||^2 + \frac{1}{B}2\alpha^2\sigma^2 + \frac{1}{B}2\eta^2(1-\alpha)^2C_\gamma^2\mathbb{E}||g_t||^2) \\ + p\frac{\sigma^2}{S_0}2\end{aligned} \tag{13}$$

Details for the last step can be found chapter A1 in [11]

$$\begin{aligned}\mathbb{E}||g_{t+1} - \nabla_\theta J(\theta_{t+1})||^2 = \mathbb{E}(\mathbb{E}(||g_{t+1} - \nabla_\theta J(\theta_{t+1}||^2|F_t)) \\ \leq (1-p)[(1-\alpha)^2\mathbb{E}||g_t - \nabla_\theta J(\theta_t)||^2 + \frac{2\sigma^2\alpha^2}{B} + \frac{2\eta^2}{B}(1-\alpha)^2C_\gamma^2\mathbb{E}||g_t||^2] + p\frac{\sigma^2}{S_0}2\end{aligned} \tag{14}$$

□

**Proof of Lemma 4.2**

*Proof.*

$$\begin{aligned}\alpha\sum_{t=0}^{T-1}||g_t - \nabla_\theta J(\theta_t)||^2 \leq &\sum_{t=1}^{T}\mathbb{E}||g_t - \nabla_\theta J(\theta_t)||^2 - (1-\alpha)^2\sum_{t=0}^{T-1}\mathbb{E}||g_t - \nabla_\theta J(\theta_t)||^2 - \\ &\mathbb{E}(||g_t - \nabla_\theta(\theta_t)||^2 - ||g_0 - \nabla_\theta J(\theta_0)||^2) \\ &\leq \text{[lemma 1]} \\ &\leq (1-p)(1-\alpha)^2\sum_{t=0}^{T-1}\mathbb{E}||g_t - \nabla_\theta J(\theta_t)||^2 + (1-p)\frac{2\alpha^2\sigma^2}{B}T + \\ &(1-p)\frac{2\eta^2c_\gamma^2}{B}(1-\alpha)^2\sum_{t=0}^{T-1}\mathbb{E}||g_t||^2 + p\frac{\sigma^2}{S_0}2T - (1-\alpha)^2 - (1-\alpha)^2\sum_{t=0}^{T-1}\mathbb{E}||g_t - \nabla_\theta J(\theta_t)||^2 - \\ &\mathbb{E}(||g_T - \nabla_\theta J(\theta_T)||^2 - ||g_0 - \nabla_\theta J(\theta_0)||^2)\end{aligned} \tag{15}$$

Details for the first step can be found chapter A2 in [11]

$$\begin{aligned}\implies (\alpha + p(1-\alpha)^2)\sum_{t=0}^{T-1}\mathbb{E}||g_t - \nabla_\theta J(\theta_t)||^2 \leq &(1-p)\frac{2\alpha^2\sigma^2}{B}T + (1-p)(1-\alpha)^2\frac{2\eta^2}{B}c_\gamma^2\sum_{t=0}^{T-1}\mathbb{E}||g_t||^2 + \\ &p\frac{2\sigma^2}{S_0}T + 2\mathbb{E}||g_0 - \nabla_\theta J(\theta_0)||^2\end{aligned} \tag{16}$$

$$\begin{aligned}\implies \sum_{t=0}^{T-1}\mathbb{E}||g_t - \nabla_\theta J(\theta_t)||^2 \leq &\frac{1-p}{\alpha + p(1-\alpha)^2}\frac{2\alpha^2\sigma^2}{B}T + \frac{(1-p)(1-\alpha)^2}{\alpha + p(1-\alpha)^2}\frac{2\eta^2}{B}c_\gamma^2\sum_{t=0}^{t-1}\mathbb{E}||g_t||^2 \\ &+ \frac{2*p*T\sigma^2}{S_0} * \frac{1}{\alpha + p(1-\alpha)^2} + \frac{2}{(\alpha + p(1-\alpha)^2)}\mathbb{E}||g_0 - \nabla_\theta J(\theta_0)||^2\end{aligned} \tag{17}$$

□

**Proof of the main theorem 4.3**

. Let Us prove that PAGE-STORM-PG can reach $\epsilon$-accurate solution after $T(\epsilon)$ iterations. In other words,

$$\forall \epsilon,\ T(\epsilon) \text{ such that}$$

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}||\nabla_\theta J(\theta_t)||^2 \leq \epsilon$$

$$\text{where } J(\theta) = \mathbb{E}_{\tau\sim p(\cdot|\theta)}R(\tau) \coloneqq \mathbb{E}_{\tau\sim p(\cdot|\theta)}[\sum_{t=0}^{T-1}\gamma^t r(s_t,a_s)]$$

Let us apply the $L_d$-smoothness of $J(\theta)$ to get a general estimation bound of $J(\theta_{t+1})$ :

$$\begin{aligned} J(\theta_{t+1}) = J(\theta_t+\eta g_t) &\geq J(\theta_t) + \eta\langle g_t, \nabla_\theta J(\theta_t)\rangle - \frac{\nabla^2 L_d}{2}||g_t||^2 = J(\theta_t) + (\frac{\eta}{2} - \frac{\eta^2 L_d}{2})||g_t||^2 + \frac{\eta}{2}||\nabla_\theta J(\theta_t)||^2 - \\ &\frac{\eta}{2}||g_t - \nabla_\theta J(\theta_t)||^2 \\ &\geq J(\theta_t) + \frac{\eta}{4}||g_t||^2 + \frac{\eta}{2}||\nabla_\theta J(\theta_t)||^2 - \frac{\eta}{2}||g_t - \nabla_\theta J(\theta_t)||^2 \end{aligned} \tag{18}$$

Summing $J(\theta_{t+1}) - J(\theta_t)$ over T, we get the inequality bellow :

$$J(\theta_0) - J(\theta_T) \leq -\frac{\eta}{2}\sum_{t=0}^{T-1}||\nabla_\theta J(\theta_t)||^2 - \frac{\eta}{4}\sum_{t=0}^{T-1}||g_t||^2 + \frac{\eta}{2}\sum_{t=0}^{T-1}||g_t - \nabla_\theta J(\theta_t)||^2 \tag{19}$$

Definition : $\Delta \coloneqq f^*$-$J(\theta_0)$-constant representing the function value gap between the initialization and the optimal value $f^*$

taking the expectation :

$$-\Delta \leq J(\theta_0) - J(\theta_T) \leq -\frac{\eta}{2}\sum_{t=0}^{T-1}||\nabla_\theta J(\theta_t)||^2 - \frac{\eta}{4}\sum_{t=0}^{T-1}\mathbb{E}||g_t||^2 + \frac{\eta}{2}\sum_{t=0}^{T-1}\mathbb{E}||g_t - \nabla_\theta J(\theta_t)||^2$$

$\leq$ [apply lemma 4.2] $\leq -\frac{\eta}{2}\sum_{t=0}^{T-1}\mathbb{E}||\nabla_\theta J(\theta_t)||^2 - \frac{\eta}{4}\sum_{t=0}^{T-1}\mathbb{E}||g_t||^2 + \frac{\eta}{\alpha+p(1-\alpha)^2}\frac{(1-p)\alpha^2\sigma^2 T}{B} + \frac{Tp\sigma^2\eta}{S_0(\alpha+p(1-\alpha)^2)} + \frac{\sigma^2\eta}{s_0(\alpha+p(1-\alpha^2))} + \frac{\eta}{\alpha+p(1-\alpha)^2}(1-p)(1-\alpha)^2\frac{C_\gamma^2\eta^2}{B}\sum_{t=0}^{T-1}\mathbb{E}||g_t||^2$

$$= -\frac{\eta}{2}\sum_{t=0}^{T-1}\mathbb{E}||\nabla_\theta J(\theta_t)||^2 - \frac{\eta}{4}\left(1 - \frac{4\eta^2(1-p)(1-\alpha)^2}{\alpha+p(1-\alpha)^2}\frac{C_\gamma^2}{B}\right)\sum_{t=0}^{T-1}\mathbb{E}||g_t||^2$$

$$+\frac{\eta}{\alpha+p(1-\alpha)^2}\frac{(1-p)\alpha^2\sigma^2}{B}T + \frac{Tp\sigma^2}{S_0}\frac{\eta}{(\alpha+p(1-\alpha)^2)} + \frac{\sigma^2\eta}{S_0(\alpha+p(1-\alpha)^2)}$$

Our pick for $\eta$ that satisfies

$$1 - \frac{4\eta^2(1-p)(1-\alpha)^2C_\gamma^2}{(\alpha+p(1-\alpha)^2)B} \geq 0$$

Hence :

$$\frac{\eta}{2}\sum_{t=0}^{T-1}\mathbb{E}||\nabla_\theta J(\theta_t)||^2 \leq \Delta + \frac{\eta}{\alpha+p(1-\alpha)^2}\frac{(1-p)\alpha^2\sigma^2}{B}T + \frac{Tp\sigma^2}{S_0}\frac{\eta}{\alpha+p(1-\alpha)^2} + \frac{\sigma^2\eta}{S_0(\alpha+p(1-\alpha)^2)}$$

$$\implies \frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}||\nabla_\theta J(\theta_t)||^2 \leq \frac{2}{\eta}\frac{\Delta}{T} + \frac{2\alpha^2(1-p)}{(\alpha+p(1-\alpha)^2)}\frac{\sigma^2}{B} + \frac{2p\sigma^2}{S_0}\frac{1}{(\alpha+p(1-\alpha)^2)} + \frac{2\sigma^2}{TS_0}\frac{1}{(\alpha+p(1-\alpha)^2)}$$

□