Safely Exploring Policy Gradient

Matteo Papini, Andrea Battistello and Marcello Restelli
{matteo.papini, andrea.battistello, marcello.restelli}@polimi.it
POLITECNICO MILANO 1863

Problem

- **Reinforcement Learning** for **continuous** control [Deisenroth et al., 2013]
- **Policy Gradient (PG)**: iteratively update **parametric** policy $\pi_{\boldsymbol\theta}$ via **gradient ascent** on performance $J(\boldsymbol\theta)$ (expected cumulative reward):

$$\boldsymbol\theta^{t+1} \leftarrow \boldsymbol\theta^t + \alpha \nabla J(\boldsymbol\theta^t)$$

- Convergence to local optimum guaranteed
- **Intermediate policies may be arbitrarily bad!**
- **Safe Exploration** [Amodei et al., 2016]: limit risks/costs of novel behavior

Motivation

- A working controller $\boldsymbol\theta^0$ is provided
- **Fine tuning**: improve it online via policy gradient
- Intermediate policies should never be worse than the initial $\boldsymbol\theta^0$ (baseline)
- *Economic safety*: losses and gains cancel out

State of the Art

Existing **safe PG** approaches [Pirotta et al., 2013, Papini et al., 2017]:

- Apply only to **Gaussian** policies with fixed variance:

$$\pi_{\boldsymbol\theta}(a|s) \sim \mathcal{N}(\mu_{\boldsymbol\theta}(s), \sigma^2)$$

⚠ The variance parameter regulates exploration and has a big impact on convergence speed

- Focus on **monotonic improvement** guarantees:

$$J(\boldsymbol\theta^{t+1}) - J(\boldsymbol\theta^t) \geq 0$$

⚠ Too strict for most practical scenarios

Contributions

- We adopt a more general definition of safety
- We extend the existing guarantees for Gaussian policies to the adaptive-variance case
- We introduce a **surrogate objective** for variance updates that explicitly encourages **exploration**
- We provide an algorithm (**SEPG**) for the **fine-tuning** scenario

Setting

- *Shallow* Gaussian policy parametrization:

$$\pi_{\boldsymbol\theta}(a|s) = \frac{1}{\sqrt{2\pi}\sigma_{\boldsymbol\theta}} \exp\left\{-\frac{1}{2}\left(\frac{a-\mu_{\boldsymbol\theta}(s)}{\sigma_{\boldsymbol\theta}}\right)^2\right\} \qquad \begin{aligned}\mu_{\boldsymbol\theta}(s) &= \boldsymbol v^T \boldsymbol\phi(s)\\ \sigma_{\boldsymbol\theta} &= e^w\end{aligned} \qquad \boldsymbol\theta = \begin{cases}\boldsymbol v \text{ mean parameter}\\ w \text{ variance parameter}\end{cases}$$

- **Safety requirement** (similar to Thomas et al. [2015]):

$$J(\boldsymbol\theta^{t+1}) - J(\boldsymbol\theta^t) \geq C^t \quad \text{with probability at least } 1-\delta$$

$C^t \geq 0$: *required improvement*
$C^t < 0$: *bounded worsening*

- Base algorithm: REINFORCE with separate mean and variance updates

$$\begin{cases}\boldsymbol v^{t+1} \leftarrow \boldsymbol v^t + \alpha\nabla_{\boldsymbol v} J(\boldsymbol v^t, w^t)\\ w^{t+1} \leftarrow w^t + \eta\nabla_w J(\boldsymbol v^{t+1}, w^t)\end{cases}$$

naive update: **too greedy!**

- **Adaptive PG**: we look for the *largest* step sizes guaranteeing safety at each iteration

Safe-Exploratory Updates

We introduce a surrogate **exploration objective** that accounts for long-term advantages of high policy variance:

$$\mathcal{L}(\boldsymbol v, w) = \underbrace{\frac{\|\nabla_{\boldsymbol v} J(\boldsymbol v, w)\|_2^2}{4mc_w}}$$

Largest **safe** performance improvement obtainable by updating the mean parameter $\boldsymbol v$ when $\sigma = e^w$

$$c_w = \frac{\overbrace{R}^{\text{max reward}}\ \overbrace{M^2}^{\text{max feature}}}{(1-\gamma)^2 e^{2w}}\left(\frac{\overbrace{|\mathcal{A}|}^{\text{volume of action space}}}{\sqrt{2\pi}e^w} + \frac{\overbrace{\gamma}^{\text{discount factor}}}{2(1-\gamma)}\right)$$

We provide a **safe** way to update the variance parameter according to this objective:

$$\begin{cases}\boldsymbol v^{t+1} \leftarrow \boldsymbol v^t + \overline\alpha\nabla_{\boldsymbol v} J(\boldsymbol v^t, w^t)\\ w^{t+1} \leftarrow w^t + \overline\eta\nabla_w \mathcal{L}(\boldsymbol v^{t+1}, w^t)\end{cases}$$

Largest safe step size $\overline\alpha$ from Papini et al. [2017]:

$$\overline\alpha = \frac{1}{2c_w}\left(1 + \sqrt{1 - \frac{4c_w C^t}{\|\nabla_{\boldsymbol v} J(\boldsymbol v^t, w^t)\|_\infty^2}}\right)$$

Largest safe-exploratory step size $\overline\eta$

$$\eta^* = \frac{\nabla_w J(\boldsymbol v^{t+1}, w^t)}{2d\nabla_w \mathcal{L}(\boldsymbol v^{t+1}, w^t)} \quad \text{(corresponds to naive update)}$$

$$\overline\eta = \eta^* + |\eta^*|\sqrt{1 - \frac{4dC^t}{\|\nabla_w J(\boldsymbol v^{t+1}, w^t)\|_\infty^2}}$$

$$d = \frac{R}{(1-\gamma)^2}\left(\frac{2(\sqrt{7}-2)e^{\sqrt{7}/2-2}|\mathcal{A}|}{\sqrt{2\pi}e^w} + \frac{\gamma}{1-\gamma}\right)$$

- Policy gradient $\nabla_w J$ (greedy) and surrogate $\nabla_w \mathcal{L}$ (explorative) typically *disagree* $\implies \eta^*$ typically negative
- Largest safe step size $\overline\eta$ can be positive (*exploration is allowed*) or negative (*greediness is required*) according to safety constraint C^t

Approximate Framework

- In practice, gradients $\nabla_{\boldsymbol v} J, \nabla_w J, \nabla_w \mathcal{L}$ must be estimated from batches of N trajectories
- We characterize estimation error $\epsilon = \left|\widehat\nabla_N J - \nabla J\right|$ using known statistical inequalities
- We obtain corrected step sizes with high-probability safety guarantees

References

Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete problems in ai safety. 2016.

Marc Peter Deisenroth, Gerhard Neumann, Jan Peters, et al. A survey on policy search for robotics. *Foundations and Trends® in Robotics*, 2(1–2):1–142, 2013.

Matteo Papini, Matteo Pirotta, and Marcello Restelli. Adaptive batch size for safe policy gradients. In *NIPS*, 2017.

Matteo Pirotta, Marcello Restelli, and Luca Bascetta. Adaptive step-size for policy gradient methods. In *NIPS*, pages 1394–1402, 2013.

Philip Thomas, Georgios Theocharous, and Mohammad Ghavamzadeh. High confidence policy improvement. In *ICML*, 2015.

Fine Tuning

- We define the **exploration budget**

$$B^t := J(\boldsymbol\theta^t) - J(\boldsymbol\theta^0)$$

- "Never do worse than the initial policy" is equivalent to

$$J(\boldsymbol\theta^{t+1}) - J(\boldsymbol\theta) \geq -B^t$$

- **SEPG** algorithm

1. Keep track of the budget:

$$B^0 \leftarrow 0$$
$$B^{t+1} \leftarrow B^t + J(\boldsymbol\theta^{t+1}) - J(\boldsymbol\theta^t)$$

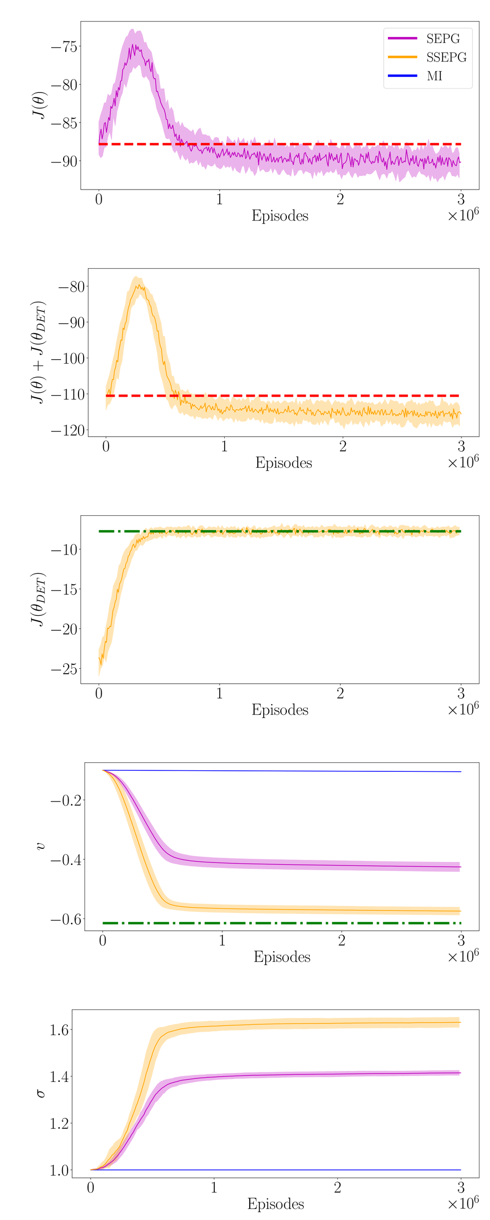
2. Update $\boldsymbol v$ and w alternately
3. Select $\overline\alpha$ and $\overline\eta$ according to $C^t = -B^t$

- **SSEPG** algorithm (*heuristic* variant):

+ Provide initial budget $B^0 > 0$ to encourage initial exploration
+ Test the corresponding **deterministic** policy ($\sigma = 0$) at each iteration to capture long-term advantages

Experiments

LQR



MOUNTAIN CAR