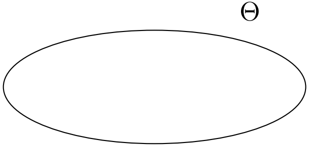
Optimistic Policy Optimization via Multiple Importance Sampling

Matteo Papini Alberto Maria Metelli
Lorenzo Lupo Marcello Restelli

11th June 2019
Thirty-sixth International Conference on Machine Learning, Long Beach, CA, USA

Policy Optimization

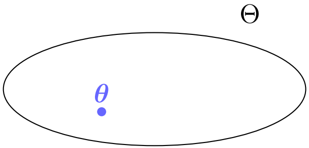
- **Parameter space** $\Theta \subseteq \mathbb{R}^d$



- A parametric **policy** for each $\boldsymbol{\theta} \in \Theta$
- Each inducing a distribution $p_{\boldsymbol{\theta}}$ over **trajectories**
- A **return** $R(\tau)$ for every trajectory τ
- **Goal:** $\max_{\boldsymbol{\theta} \in \Theta} J(\boldsymbol{\theta}) = \mathbb{E}_{\tau \sim p_{\boldsymbol{\theta}}} [R(\tau)]$
- Iterative optimization (e.g., gradient ascent)

Policy Optimization

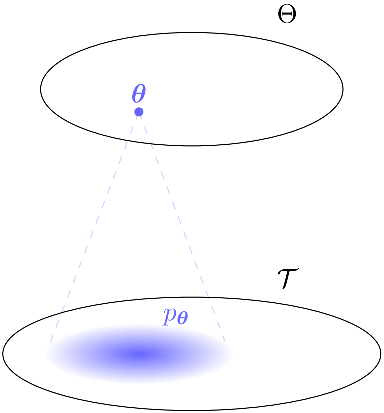
- **Parameter space** $\Theta \subseteq \mathbb{R}^d$



- A parametric **policy** for each $\boldsymbol{\theta} \in \Theta$
- Each inducing a distribution $p_{\boldsymbol{\theta}}$ over **trajectories**
- A **return** $R(\tau)$ for every trajectory τ
- **Goal:** $\max_{\boldsymbol{\theta} \in \Theta} J(\boldsymbol{\theta}) = \mathbb{E}_{\tau \sim p_{\boldsymbol{\theta}}}[R(\tau)]$
- Iterative optimization (e.g., gradient ascent)

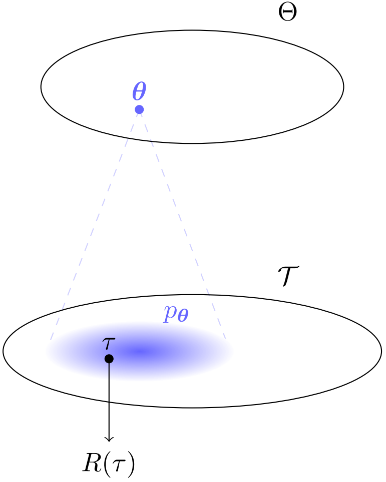
Policy Optimization

- **Parameter space** $\Theta \subseteq \mathbb{R}^d$
- A parametric **policy** for each $\boldsymbol{\theta} \in \Theta$
- Each inducing a distribution $p_{\boldsymbol{\theta}}$ over **trajectories**
- A **return** $R(\tau)$ for every trajectory τ
- **Goal:** $\max_{\boldsymbol{\theta} \in \Theta} J(\boldsymbol{\theta}) = \mathbb{E}_{\tau \sim p_{\boldsymbol{\theta}}} [R(\tau)]$
- Iterative optimization (e.g., gradient ascent)



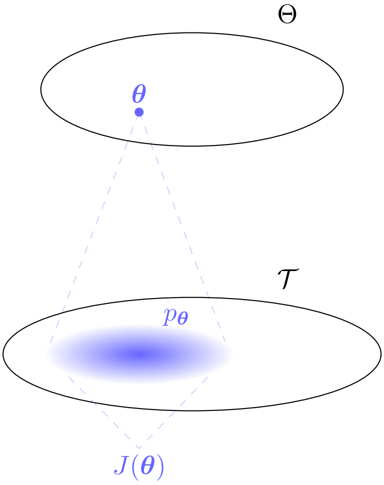
Policy Optimization

- **Parameter space** $\Theta \subseteq \mathbb{R}^d$
- A parametric **policy** for each $\boldsymbol{\theta} \in \Theta$
- Each inducing a distribution $p_{\boldsymbol{\theta}}$ over **trajectories**
- A **return** $R(\tau)$ for every trajectory τ
- **Goal:** $\max_{\boldsymbol{\theta} \in \Theta} J(\boldsymbol{\theta}) = \mathbb{E}_{\tau \sim p_{\boldsymbol{\theta}}}[R(\tau)]$
- Iterative optimization (e.g., gradient ascent)



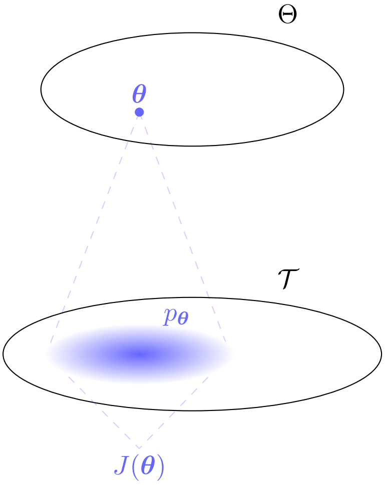
Policy Optimization

- **Parameter space** $\Theta \subseteq \mathbb{R}^d$
- A parametric **policy** for each $\boldsymbol{\theta} \in \Theta$
- Each inducing a distribution $p_{\boldsymbol{\theta}}$ over **trajectories**
- A **return** $R(\tau)$ for every trajectory τ
- **Goal:** $\max_{\boldsymbol{\theta} \in \Theta} J(\boldsymbol{\theta}) = \mathbb{E}_{\tau \sim p_{\boldsymbol{\theta}}} [R(\tau)]$
- Iterative optimization (e.g., gradient ascent)



Policy Optimization

- **Parameter space** $\Theta \subseteq \mathbb{R}^d$
- A parametric **policy** for each $\boldsymbol{\theta} \in \Theta$
- Each inducing a distribution $p_{\boldsymbol{\theta}}$ over **trajectories**
- A **return** $R(\tau)$ for every trajectory τ
- **Goal:** $\max_{\boldsymbol{\theta} \in \Theta} J(\boldsymbol{\theta}) = \mathbb{E}_{\tau \sim p_{\boldsymbol{\theta}}}[R(\tau)]$
- Iterative optimization (e.g., gradient ascent)



Exploration in Policy Optimization

- **Continuous** decision process $\implies$ difficult
- Policy gradient methods tend to be **greedy** (e.g., TRPO [6], PGPE [7])
- Mainly **undirected** (e.g., entropy bonus [2])
- **Lack of theoretical guarantees**

Exploration in Policy Optimization

- **Continuous** decision process $\implies$ difficult
- Policy gradient methods tend to be **greedy** (e.g., TRPO [6], PGPE [7])
- Mainly **undirected** (e.g., entropy bonus [2])
- **Lack of theoretical guarantees**

Exploration in Policy Optimization

- **Continuous** decision process $\implies$ difficult
- Policy gradient methods tend to be **greedy** (e.g., TRPO [6], PGPE [7])
- Mainly **undirected** (e.g., entropy bonus [2])
- **Lack of theoretical guarantees**

Exploration in Policy Optimization

- **Continuous** decision process $\implies$ difficult
- Policy gradient methods tend to be **greedy** (e.g., TRPO [6], PGPE [7])
- Mainly **undirected** (e.g., entropy bonus [2])
- **Lack of theoretical guarantees**

Exploration in Policy Optimization

- **Continuous** decision process $\implies$ difficult
- Policy gradient methods tend to be **greedy** (e.g., TRPO [6], PGPE [7])
- Mainly **undirected** (e.g., entropy bonus [2])
- **Lack of theoretical guarantees**

If only this was a Multi-Armed Bandit...

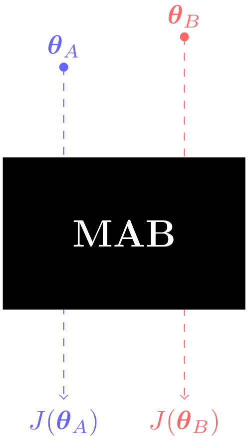
Policy Optimization as a MAB

- **Arms:** parameters $\boldsymbol{\theta}$

- **Payoff:** expected return $J(\boldsymbol{\theta})$
- **Continuous MAB** [3]: we *need* structure
- **Arm correlation** [5] through trajectory distributions

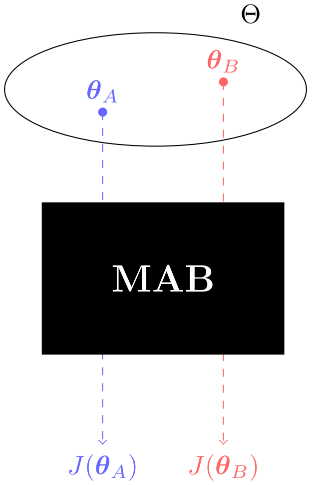
Policy Optimization as a MAB

- **Arms:** parameters $\boldsymbol{\theta}$
- **Payoff:** expected return $J(\boldsymbol{\theta})$
- **Continuous MAB** [3]
- **Arm correlation** [5] through trajectory distributions



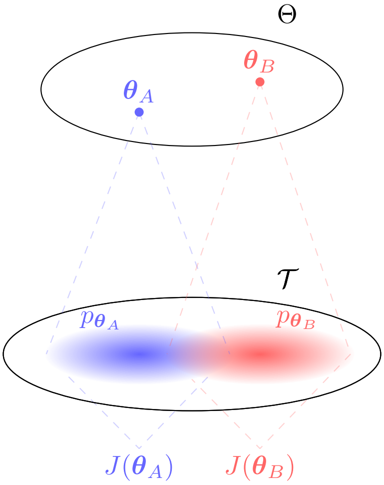
Policy Optimization as a MAB

- **Arms:** parameters $\boldsymbol{\theta}$
- **Payoff:** expected return $J(\boldsymbol{\theta})$
- **Continuous MAB** [3]
- **Arm correlation** [5] through trajectory distributions



Policy Optimization as a Correlated MAB

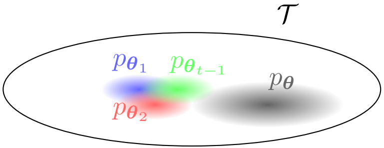
- **Arms:** parameters $\boldsymbol{\theta}$
- **Payoff:** expected return $J(\boldsymbol{\theta})$
- **Continuous MAB** [3]
- **Arm correlation** [5] through trajectory distributions



- A **UCB-like** index [4]:

$$B_t(\boldsymbol{\theta}) = \underbrace{\check{J}_t(\boldsymbol{\theta})}_{\textbf{ESTIMATE}}$$

a **truncated** **multiple** importance sampling estimator [8, 1]



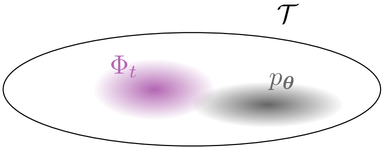
OPTIMIST

- A **UCB-like** index [4]:

$$B_t(\boldsymbol{\theta}) = \underbrace{\check{J}_t(\boldsymbol{\theta})}_{\textbf{ESTIMATE}} + \underbrace{C\sqrt{\frac{d_2(p_{\boldsymbol{\theta}}\|\Phi_t)\log\frac{1}{\delta_t}}{t}}}_{\textbf{EXPLORATION BONUS:}}$$

a **truncated** **multiple** importance sampling estimator [8, 1]

distributional distance from previous solutions



- A **UCB-like** index [4]:

$$B_t(\boldsymbol{\theta}) = \underbrace{\check{J}_t(\boldsymbol{\theta})}_{\textbf{ESTIMATE}} + \underbrace{C\sqrt{\frac{d_2(p_{\boldsymbol{\theta}} \| \Phi_t)\log\frac{1}{\delta_t}}{t}}}_{\textbf{EXPLORATION BONUS:}}$$

a **truncated** **multiple** importance sampling estimator [8, 1]

distributional distance from previous solutions

- Select $\boldsymbol{\theta}_t = \arg\max_{\boldsymbol{\theta}\in\Theta} B_t(\boldsymbol{\theta})$

Sublinear Regret

- $Regret(T) = \sum_{t=0}^{T} J(\boldsymbol{\theta}^*) - J(\boldsymbol{\theta}_t)$
- **Compact**, d-dimensional parameter space Θ
- Under **mild assumptions** on the policy class, with high probability:

$$Regret(T) = \widetilde{\mathcal{O}}\left(\sqrt{dT}\right)$$

Sublinear Regret

- $Regret(T) = \sum_{t=0}^{T} J(\boldsymbol{\theta}^*) - J(\boldsymbol{\theta}_t)$
- **Compact**, d-dimensional parameter space Θ
- Under **mild assumptions** on the policy class, with high probability:

$$Regret(T) = \widetilde{\mathcal{O}}\left(\sqrt{dT}\right)$$

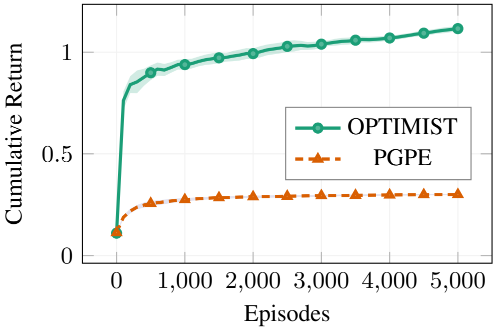
Sublinear Regret

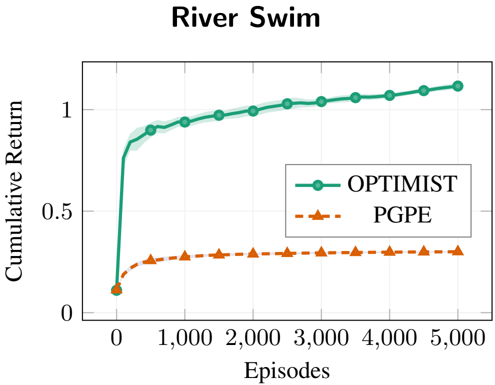
- $Regret(T) = \sum_{t=0}^{T} J(\boldsymbol{\theta}^*) - J(\boldsymbol{\theta}_t)$
- **Compact**, d-dimensional parameter space Θ
- Under **mild assumptions** on the policy class, with high probability:

$$Regret(T) = \widetilde{\mathcal{O}}\left(\sqrt{dT}\right)$$

Empirical Results

River Swim





Caveats

- Easy implementation only for parameter-based exploration [7]
- Difficult optimization
  $\implies$ discretization
- ...

Thank You for Your Attention!

Poster **#103**

Code: `github.com/WolfLo/optimist`

Contact: matteo.papini@polimi.it

Web page: `t3p.github.io/icml19`

References

[1] Bubeck, S., Cesa-Bianchi, N., and Lugosi, G. (2013). Bandits with heavy tail. *IEEE Transactions on Information Theory*, 59(11):7711–7717.

[2] Haarnoja, T., Zhou, A., Abbeel, P., and Levine, S. (2018). Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, pages 1856–1865.

[3] Kleinberg, R., Slivkins, A., and Upfal, E. (2013). Bandits and experts in metric spaces. *arXiv preprint arXiv:1312.1277*.

[4] Lai, T. L. and Robbins, H. (1985). Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics*, 6(1):4–22.

[5] Pandey, S., Chakrabarti, D., and Agarwal, D. (2007). Multi-armed bandit problems with dependent arms. In *Proceedings of the 24th international conference on Machine learning*, pages 721–728. ACM.

[6] Schulman, J., Levine, S., Abbeel, P., Jordan, M., and Moritz, P. (2015). Trust region policy optimization. In *International Conference on Machine Learning*, pages 1889–1897.

[7] Sehnke, F., Osendorfer, C., Rückstieß, T., Graves, A., Peters, J., and Schmidhuber, J. (2008). Policy gradients with parameter-based exploration for control. In *International Conference on Artificial Neural Networks*, pages 387–396. Springer.

[8] Veach, E. and Guibas, L. J. (1995). Optimally combining sampling techniques for Monte Carlo rendering. In *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques - SIGGRAPH '95*, pages 419–428. ACM Press.