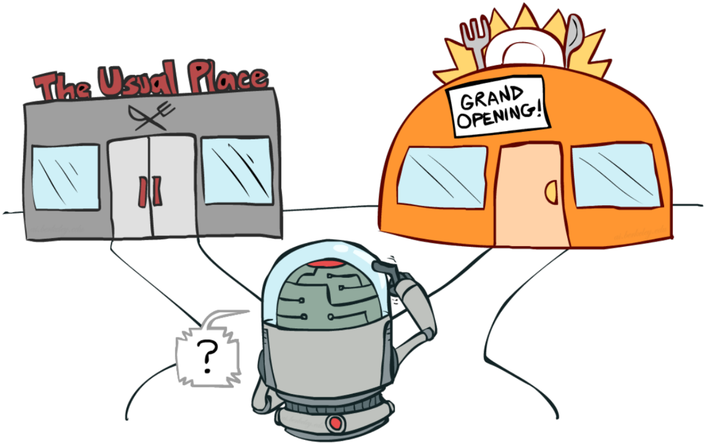
POLITECNICO
MILANO 1863

Exploration in Policy Search by Multiple Importance Sampling

Lorenzo Lupo
lorenzo.lupo@mail.polimi.it

April 16th, 2019

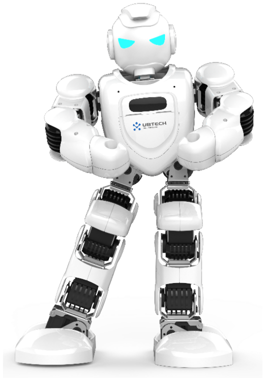
Exploration VS Exploitation



Reinforcement Learning Applications

Reinforcement Learning Applications

Reinforcement Learning Applications



Plan

1. Basics of Reinforcement Learning
2. Exploration in Policy Search
3. Problem Formalization
4. OPTIMIST
5. Experiments
6. Conclusions

The Reinforcement Learning Framework

The Reinforcement Learning Framework

Environment

$\mathcal{P}, \mathcal{R}$

The Reinforcement Learning Framework

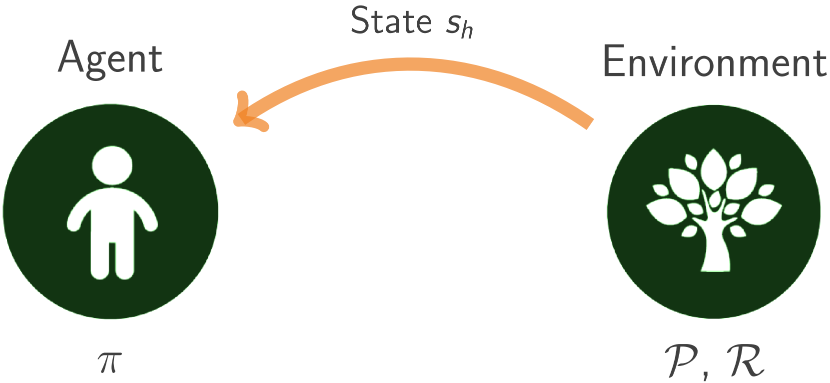
Agent

π

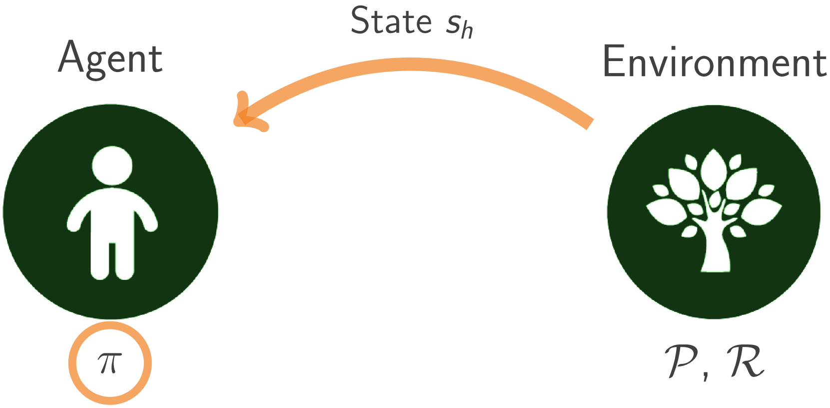
Environment

$\mathcal{P}$, $\mathcal{R}$

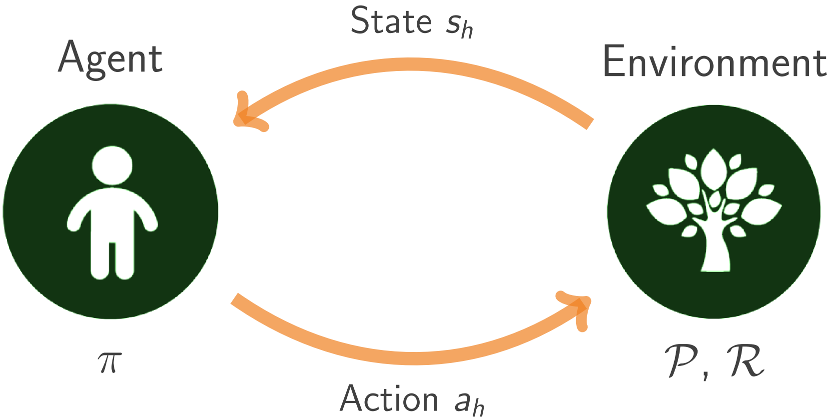
The Reinforcement Learning Framework



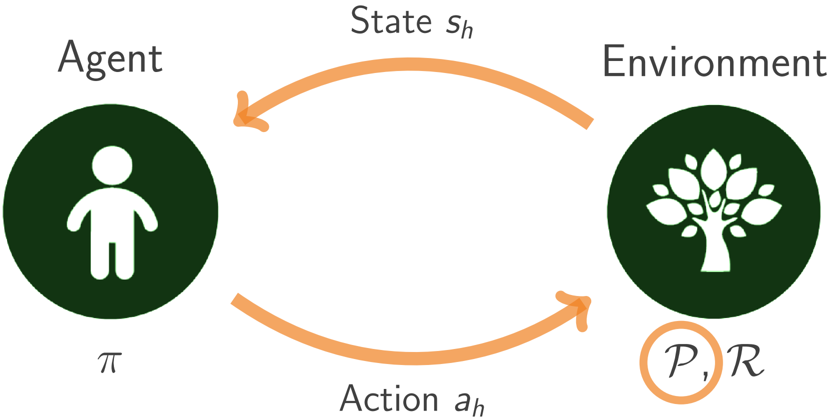
The Reinforcement Learning Framework



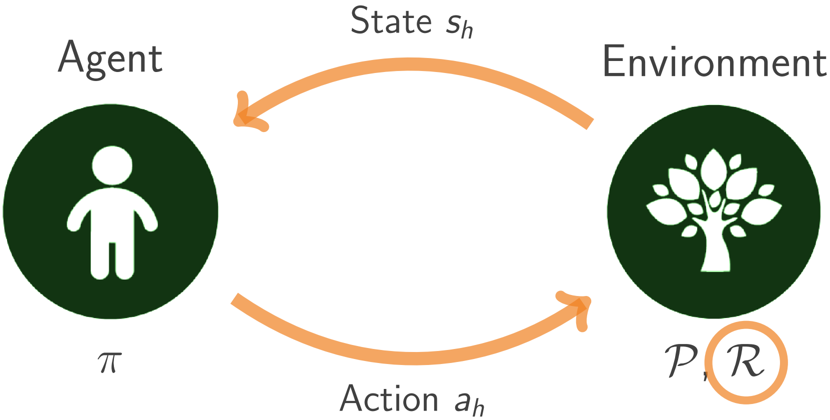
The Reinforcement Learning Framework



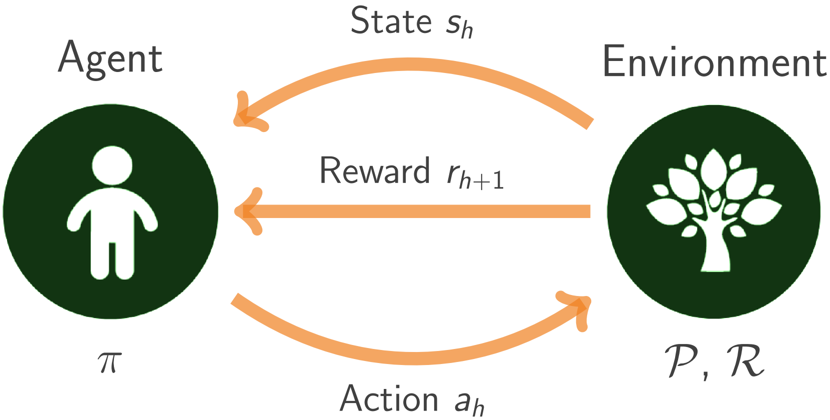
The Reinforcement Learning Framework



The Reinforcement Learning Framework



The Reinforcement Learning Framework



Policy Search Formulation

Cumulative return of a trajectory τ:

$$\mathcal{R}(\tau) = \sum_{h=0}^{H-1} \gamma^h r_{h+1}, \text{ with } \tau = [s_0, a_0, s_1, a_1, \ldots, s_{H-1}, a_{H-1}]$$

Policy Search Formulation

Cumulative return of a trajectory τ:

$$\mathcal{R}(\tau) = \sum_{h=0}^{H-1} \gamma^h r_{h+1}, \text{ with } \tau = [s_0, a_0, s_1, a_1, \ldots, s_{H-1}, a_{H-1}]$$

Parametric policy:

$$\pi_{\boldsymbol{\theta}} : \mathcal{S} \to \Delta(\mathcal{A}), \text{ i.e., } \pi_{\boldsymbol{\theta}}(a|s) = \frac{1}{\sqrt{2\pi\sigma}} \exp\left(-\frac{1}{2} \left(\frac{a - \boldsymbol{\theta}^T \phi(s)}{\sigma} \right)^2 \right)$$

Policy Search Formulation

Cumulative return of a trajectory τ:

$$\mathcal{R}(\tau) = \sum_{h=0}^{H-1} \gamma^h r_{h+1}, \text{ with } \tau = [s_0, a_0, s_1, a_1, \ldots, s_{H-1}, a_{H-1}]$$

Parametric policy:

$$\pi_{\boldsymbol{\theta}} : \mathcal{S} \to \Delta(\mathcal{A}), \text{ i.e., } \pi_{\boldsymbol{\theta}}(a|s) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2}\left(\frac{a - \boldsymbol{\theta}^T \phi(s)}{\sigma}\right)^2\right)$$

Performance:

$\mu(\boldsymbol{\theta}) = \underset{\tau \sim p_{\boldsymbol{\theta}}}{\mathbb{E}} [\mathcal{R}(\tau)]$, where $p_{\boldsymbol{\theta}}$ is the **distribution over trajectories** $\tau \in \mathcal{T}$ induced by $\pi_{\boldsymbol{\theta}}$

Policy Search Formulation

Cumulative return of a trajectory τ:

$$\mathcal{R}(\tau) = \sum_{h=0}^{H-1} \gamma^h r_{h+1}, \text{ with } \tau = [s_0, a_0, s_1, a_1, \dots, s_{H-1}, a_{H-1}]$$

Parametric policy:

$$\pi_{\boldsymbol{\theta}} : \mathcal{S} \to \Delta(\mathcal{A}), \text{ i.e., } \pi_{\boldsymbol{\theta}}(a|s) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2}\left(\frac{a - \boldsymbol{\theta}^T \phi(s)}{\sigma}\right)^2\right)$$

Performance:

$\mu(\boldsymbol{\theta}) = \mathop{\mathbb{E}}_{\tau \sim p_{\boldsymbol{\theta}}}[\mathcal{R}(\tau)]$, where $p_{\boldsymbol{\theta}}$ is the **distribution over trajectories** $\tau \in \mathcal{T}$ induced by $\pi_{\boldsymbol{\theta}}$

Objective:

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta} \in \Theta} \mu(\boldsymbol{\theta}).$$

Plan

Exploration in Policy Search

Undirected exploration

Generate actions based on randomness, without any knowledge of the learning process.

Exploration in Policy Search

Undirected exploration

Generate actions based on randomness, without any knowledge of the learning process.

- Ex1: by adopting stochastic policies.

Exploration in Policy Search

Undirected exploration

Generate actions based on randomness, without any knowledge of the learning process.

- Ex1: by adopting stochastic policies.
- Ex2: by augmenting rewards with the entropy of the policy:

$$\mathcal{R}(\tau) = \sum_{h=0}^{H-1} \gamma^h r_{h+1} + \mathcal{H}(\pi_{\boldsymbol{\theta}}(\cdot|s_h)).$$

Exploration in Policy Search

Undirected exploration

Generate actions based on randomness, without any knowledge of the learning process.

- Ex1: by adopting stochastic policies.
- Ex2: by augmenting rewards with the entropy of the policy:

$$\mathcal{R}(\tau) = \sum_{h=0}^{H-1} \gamma^h r_{h+1} + \mathcal{H}(\pi_{\boldsymbol{\theta}}(\cdot|s_h)).$$

Directed exploration

Leverage on the knowledge acquired during learning.

Exploration in Policy Search

Undirected exploration

Generate actions based on randomness, without any knowledge of the learning process.

- Ex1: by adopting stochastic policies.
- Ex2: by augmenting rewards with the entropy of the policy:

$$\mathcal{R}(\tau) = \sum_{h=0}^{H-1} \gamma^h r_{h+1} + \mathcal{H}(\pi_{\boldsymbol{\theta}}(\cdot|s_h)).$$

Directed exploration

Leverage on the knowledge acquired during learning.

- Count-based techniques.

Plan

Problem Formalization

Decision Set or Arms Set

The parameter space $\Theta \subseteq \mathbb{R}^d$.

Problem Formalization

Decision Set or Arms Set

The parameter space $\Theta \subseteq \mathbb{R}^d$.

Procedure

At every decision step $t \in [0, 1, 2, \ldots, T]$:

Problem Formalization

Decision Set or Arms Set

The parameter space $\Theta \subseteq \mathbb{R}^d$.

Procedure

At every decision step $t \in [0, 1, 2, \ldots, T]$:

1. **Select** an arm $\boldsymbol{\theta}_t \in \Theta$;

Problem Formalization

Decision Set or Arms Set

The parameter space $\Theta \subseteq \mathbb{R}^d$.

Procedure

At every decision step $t \in [0, 1, 2, \ldots, T]$:

1. **Select** an arm $\boldsymbol{\theta}_t \in \Theta$;
2. **Sample** a trajectory $\tau_t \in \mathcal{T}$ by following $\pi_{\boldsymbol{\theta}_t}$;

Problem Formalization

Decision Set or Arms Set

The parameter space $\Theta \subseteq \mathbb{R}^d$.

Procedure

At every decision step $t \in [0, 1, 2, \ldots, T]$:

1. **Select** an arm $\boldsymbol{\theta}_t \in \Theta$;
2. **Sample** a trajectory $\tau_t \in \mathcal{T}$ by following $\pi_{\boldsymbol{\theta}_t}$;
3. **Observe** the cumulative return $\mathcal{R}(\tau_t)$.

Problem Formalization

Decision Set or Arms Set

The parameter space $\Theta \subseteq \mathbb{R}^d$.

Procedure

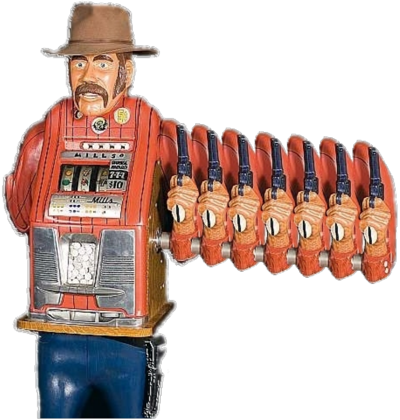
At every decision step $t \in [0, 1, 2, \ldots, T]$:

1. **Select** an arm $\boldsymbol{\theta}_t \in \Theta$;
2. **Sample** a trajectory $\tau_t \in \mathcal{T}$ by following $\pi_{\boldsymbol{\theta}_t}$;
3. **Observe** the cumulative return $\mathcal{R}(\tau_t)$.

Goal

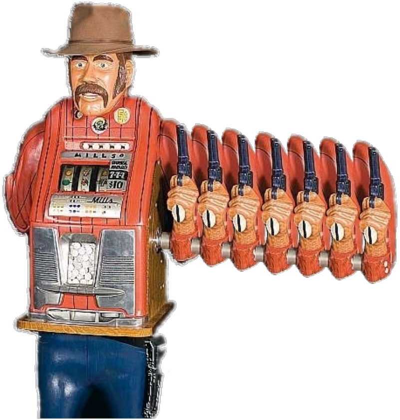
Minimize $Regret(T) = \sum_{t=0}^{T} \mu(\boldsymbol{\theta}^*) - \mu(\boldsymbol{\theta}_t)$, where $\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta} \in \Theta} \mu(\boldsymbol{\theta})$

Problem Formulation



Multi Armed Bandits

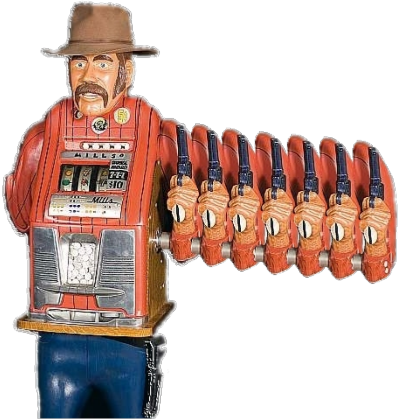
Problem Formulation



Multi Armed Bandits

- Simpler framework;

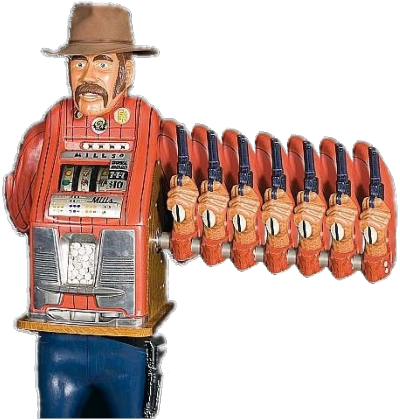
Problem Formulation



Multi Armed Bandits

- Simpler framework;
- Share the exploration-exploitation tradeoff;

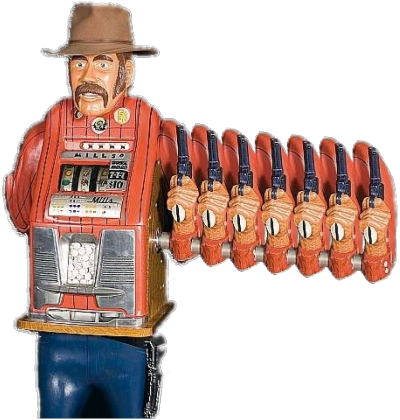
Problem Formulation



Multi Armed Bandits

- Simpler framework;
- Share the exploration-exploitation tradeoff;
- Ample literature available;

Problem Formulation



Multi Armed Bandits

- Simpler framework;
- Share the exploration-exploitation tradeoff;
- Ample literature available;

Desideratum

sub-linear $Regret(T) \Leftrightarrow \lim_{T \to \infty} Regret(T)/T = 0$

E.g. $Regret(T) = \mathcal{O}(\log T)$

Plan

1. Basics of Reinforcement Learning
2. Exploration in Policy Search
3. Problem Formalization
4. OPTIMIST
5. Experiments
6. Conclusions

Discrete Decision Set

Algorithm 1 OPTIMIST

1: **Input:** initial parameters $\boldsymbol{\theta}_0$, confidence schedule $(\delta_t)_{t=1}^T$, order $\epsilon \in (0, 1]$

Discrete Decision Set

Algorithm 2 OPTIMIST

1: **Input:** initial parameters $\boldsymbol{\theta}_0$, confidence schedule $(\delta_t)_{t=1}^T$, order $\epsilon \in (0, 1]$
2: Draw trajectory $\tau_0 \sim p_{\boldsymbol{\theta}_0}$ and observe return $\mathcal{R}(\tau_0)$

Discrete Decision Set

Algorithm 3 OPTIMIST

1: **Input:** initial parameters $\boldsymbol{\theta}_0$, confidence schedule $(\delta_t)_{t=1}^{T}$, order $\epsilon \in (0, 1]$
2: Draw trajectory $\tau_0 \sim p_{\boldsymbol{\theta}_0}$ and observe return $\mathcal{R}(\tau_0)$
3: **for** $t = 1, \ldots, T$ **do**

Discrete Decision Set

Algorithm 4 OPTIMIST

1: **Input:** initial parameters $\boldsymbol{\theta}_0$, confidence schedule $(\delta_t)_{t=1}^{T}$, order $\epsilon \in (0, 1]$
2: Draw trajectory $\tau_0 \sim p_{\boldsymbol{\theta}_0}$ and observe return $\mathcal{R}(\tau_0)$
3: **for** $t = 1, \ldots, T$ **do**
4: Select arm $\boldsymbol{\theta}_t = \arg\max_{\boldsymbol{\theta} \in \Theta} B_t^{\epsilon}(\boldsymbol{\theta}, \delta_t)$

Discrete Decision Set

Algorithm 5 OPTIMIST

1: **Input:** initial parameters $\boldsymbol{\theta}_0$, confidence schedule $(\delta_t)_{t=1}^{T}$, order $\epsilon \in (0, 1]$
2: Draw trajectory $\tau_0 \sim p_{\boldsymbol{\theta}_0}$ and observe return $\mathcal{R}(\tau_0)$
3: **for** $t = 1, \dots, T$ **do**
4: Select arm $\boldsymbol{\theta}_t = \arg\max_{\boldsymbol{\theta} \in \Theta} B_t^{\epsilon}(\boldsymbol{\theta}, \delta_t)$
5: Draw trajectory $\tau_t \sim p_{\boldsymbol{\theta}_t}$ and observe return $\mathcal{R}(\tau_t)$
6: **end for**

Discrete Decision Set

Algorithm 6 OPTIMIST

Input: initial parameters $\boldsymbol{\theta}_0$, confidence schedule $(\delta_t)_{t=1}^T$, order $\epsilon \in (0, 1]$
Draw trajectory $\tau_0 \sim p_{\boldsymbol{\theta}_0}$ and observe return $\mathcal{R}(\tau_0)$
for $t = 1, \dots, T$ **do**
    Select arm $\boldsymbol{\theta}_t = \arg\max_{\boldsymbol{\theta} \in \Theta} B_t^{\epsilon}(\boldsymbol{\theta}, \delta_t)$
    Draw trajectory $\tau_t \sim p_{\boldsymbol{\theta}_t}$ and observe return $\mathcal{R}(\tau_t)$
end for

$$\mu(\boldsymbol{\theta}) \leqslant B_t^{\epsilon}(\boldsymbol{\theta}, \delta_t) \;=\; \check{\mu}_t^{MIS}(\boldsymbol{\theta}) \;+\; \|f\|_{\infty} \left(\sqrt{2} + \frac{4}{3}\right) \left(\frac{d_{1+\epsilon}(p_{\boldsymbol{\theta}_t} \| \Phi_t) \;\; \log \frac{1}{\delta_t}}{t} \right)^{\frac{\epsilon}{1+\epsilon}}$$

Discrete Decision Set

Algorithm 7 OPTIMIST

Input: initial parameters $\boldsymbol{\theta}_0$, confidence schedule $(\delta_t)_{t=1}^T$, order $\epsilon \in (0,1]$
Draw trajectory $\tau_0 \sim p_{\boldsymbol{\theta}_0}$ and observe return $\mathcal{R}(\tau_0)$
for $t = 1, \dots, T$ **do**
    Select arm $\boldsymbol{\theta}_t = \arg\max_{\boldsymbol{\theta} \in \Theta} B_t^{\epsilon}(\boldsymbol{\theta}, \delta_t)$
    Draw trajectory $\tau_t \sim p_{\boldsymbol{\theta}_t}$ and observe return $\mathcal{R}(\tau_t)$
end for

Upper Confidence Bound

$$\boxed{\mu(\boldsymbol{\theta}) \leqslant B_t^{\epsilon}(\boldsymbol{\theta}, \delta_t)} = \check{\mu}_t^{MIS}(\boldsymbol{\theta}) + \|f\|_\infty \left(\sqrt{2} + \frac{4}{3}\right) \left(\frac{d_{1+\epsilon}(p_{\boldsymbol{\theta}_t} \| \Phi_t) \ \log \frac{1}{\delta_t}}{t}\right)^{\frac{\epsilon}{1+\epsilon}}$$

Discrete Decision Set

Algorithm 8 OPTIMIST

1: **Input:** initial parameters $\boldsymbol{\theta}_0$, confidence schedule $(\delta_t)_{t=1}^{T}$, order $\epsilon \in (0, 1]$
2: Draw trajectory $\tau_0 \sim p_{\boldsymbol{\theta}_0}$ and observe return $\mathcal{R}(\tau_0)$
3: **for** $t = 1, \ldots, T$ **do**
4: Select arm $\boldsymbol{\theta}_t = \arg\max_{\boldsymbol{\theta} \in \Theta} B_t^{\epsilon}(\boldsymbol{\theta}, \delta_t)$
5: Draw trajectory $\tau_t \sim p_{\boldsymbol{\theta}_t}$ and observe return $\mathcal{R}(\tau_t)$
6: **end for**

Truncated Multiple Importance Sampling Estimator

$$\mu(\boldsymbol{\theta}) \leqslant B_t^{\epsilon}(\boldsymbol{\theta}, \delta_t) = \boxed{\check{\mu}_t^{MIS}(\boldsymbol{\theta})} + \|f\|_{\infty} \left(\sqrt{2} + \frac{4}{3}\right) \left(\frac{d_{1+\epsilon}(p_{\boldsymbol{\theta}_t} \| \Phi_t) \ \log \frac{1}{\delta_t}}{t} \right)^{\frac{\epsilon}{1+\epsilon}}$$

Discrete Decision Set

Algorithm 9 OPTIMIST

1: **Input:** initial parameters $\boldsymbol{\theta}_0$, confidence schedule $(\delta_t)_{t=1}^T$, order $\epsilon \in (0, 1]$
2: Draw trajectory $\tau_0 \sim p_{\boldsymbol{\theta}_0}$ and observe return $\mathcal{R}(\tau_0)$
3: **for** $t = 1, \dots, T$ **do**
4: Select arm $\boldsymbol{\theta}_t = \arg\max_{\boldsymbol{\theta} \in \Theta} B_t^{\epsilon}(\boldsymbol{\theta}, \delta_t)$
5: Draw trajectory $\tau_t \sim p_{\boldsymbol{\theta}_t}$ and observe return $\mathcal{R}(\tau_t)$
6: **end for**

Exploration Bonus

$$\mu(\boldsymbol{\theta}) \leqslant B_t^{\epsilon}(\boldsymbol{\theta}, \delta_t) = \check{\mu}_t^{MIS}(\boldsymbol{\theta}) + \boxed{\|f\|_\infty \left(\sqrt{2} + \frac{4}{3}\right) \left(\frac{d_{1+\epsilon}(p_{\boldsymbol{\theta}_t} \| \Phi_t) \ \log \frac{1}{\delta_t}}{t} \right)^{\frac{\epsilon}{1+\epsilon}}}$$

Discrete Decision Set

Algorithm 10 OPTIMIST

Input: initial parameters $\boldsymbol{\theta}_0$, confidence schedule $(\delta_t)_{t=1}^T$, order $\epsilon \in (0, 1]$
Draw trajectory $\tau_0 \sim p_{\boldsymbol{\theta}_0}$ and observe return $\mathcal{R}(\tau_0)$
for $t = 1, \dots, T$ **do**
    Select arm $\boldsymbol{\theta}_t = \arg\max_{\boldsymbol{\theta}\in\Theta} B_t^{\epsilon}(\boldsymbol{\theta}, \delta_t)$
    Draw trajectory $\tau_t \sim p_{\boldsymbol{\theta}_t}$ and observe return $\mathcal{R}(\tau_t)$
end for

Exploration Bonus

$$\mu(\boldsymbol{\theta}) \leqslant B_t^{\epsilon}(\boldsymbol{\theta}, \delta_t) = \check{\mu}_t^{MIS}(\boldsymbol{\theta}) + \boxed{\|f\|_{\infty} \left(\sqrt{2} + \frac{4}{3}\right) \left(\frac{\boxed{d_{1+\epsilon}(p_{\boldsymbol{\theta}_t} \| \Phi_t)} \log \frac{1}{\delta_t}}{t}\right)^{\frac{\epsilon}{1+\epsilon}}}$$

Continuous Decision Set

Algorithm 11 OPTIMIST2

Input: initial arm $\boldsymbol{\theta}_0$, confidence schedule $(\delta_t)_{t=1}^{T}$, discretization schedule $(\nu_t)_{t=1}^{T}$, order $\epsilon \in (0, 1]$

Draw sample $\nu_0 \sim p_{\boldsymbol{\theta}_0}$ and observe return $f(\tau_0)$

for $t = 1, \dots, T$ **do**

    Discretize Θ with a uniform grid $\widetilde{\Theta}_t$ of ν_t^d points

    Select arm $\boldsymbol{\theta}_t = \arg\max_{\boldsymbol{\theta} \in \widetilde{\Theta}_t} B_t^{\epsilon}(\boldsymbol{\theta}, \delta_t)$

    Draw sample $\tau_t \sim p_{\boldsymbol{\theta}_t}$ and observe return $\mathcal{R}(\tau_t)$

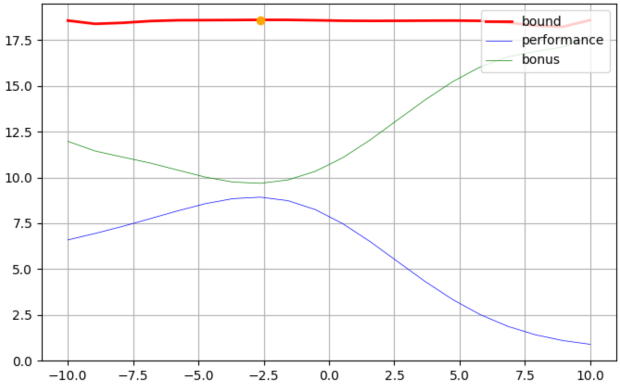
end for

Continuous Decision Set

Algorithm 12 OPTIMIST2

Input: initial arm $\boldsymbol{\theta}_0$, confidence schedule $(\delta_t)_{t=1}^T$, $\boxed{\text{discretization schedule } (\nu_t)_{t=1}^T}$, order $\epsilon \in (0, 1]$
Draw sample $\nu_0 \sim p_{\boldsymbol{\theta}_0}$ and observe return $f(\tau_0)$
for $t = 1, \ldots, T$ **do**
    Discretize Θ with a uniform grid $\widetilde{\Theta}_t$ of ν_t^d points
    Select arm $\boldsymbol{\theta}_t = \arg\max_{\boldsymbol{\theta} \in \widetilde{\Theta}_t} B_t^\epsilon(\boldsymbol{\theta}, \delta_t)$
    Draw sample $\tau_t \sim p_{\boldsymbol{\theta}_t}$ and observe return $\mathcal{R}(\tau_t)$
end for

The Upper Confidence Bound



Regret Analysis

Theorem (1)

*Let $\mathcal{X}$ be a discrete arm set with $|\mathcal{X}| = K \in \mathbb{N}_+$. Under Assumption (**??**), Algorithm 1 with confidence schedule $\delta_t = \dfrac{3\delta}{t^2\pi^2 K}$ guarantees, with probability at least $1 - \delta$:*

$$\mathit{Regret}(T) \leqslant \Delta_0 + CT^{\frac{1}{1+\epsilon}} \left[v_\epsilon \left(2 \log T + \log \frac{\pi^2 K}{3\delta} \right) \right]^{\frac{\epsilon}{1+\epsilon}},$$

where $C = (1+\epsilon)\left(2\sqrt{2} + \dfrac{5}{3}\right) \|f\|_\infty$, and Δ_0 is the instantaneous regret of the initial arm $\boldsymbol{x}_0$.

This yields a $\mathcal{O}(\sqrt{T \log T})$ regret when $\epsilon = 1$.

Regret Analysis

Theorem (2)

*Let $\mathcal{X}$ be a d-dimensional compact arm set with $\mathcal{X} \subseteq [-D, D]^d$. For any $\kappa \geqslant 2$, under Assumptions (**??**) and (**??**), Algorithm 11 with confidence schedule*

$\delta_t = \dfrac{6\delta}{\pi^2 t^2 \left(1 + \lceil t^{1/\kappa} \rceil^d\right)}$ *and discretization schedule* $\tau_t = \lceil t^{\frac{1}{\kappa}} \rceil$ *guarantees, with probability at least* $1 - \delta$*:*

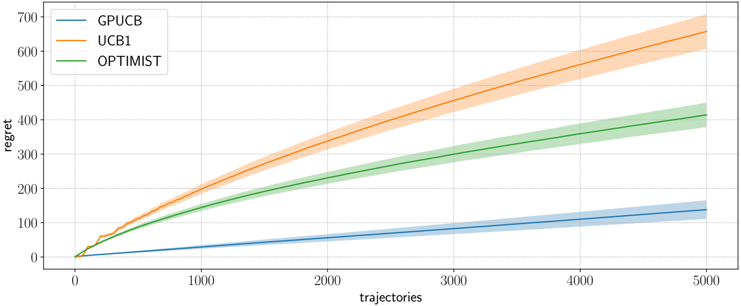
$$\text{Regret}(T) \leqslant \Delta_0 + C_1 T^{\left(1-\frac{1}{\kappa}\right)} d + C_2 T^{\frac{1}{1+\epsilon}} \cdot \left[v_\epsilon \left((2 + {d}/{\kappa}) \log T + d \log 2 + \log \frac{\pi^2}{3\delta} \right) \right]^{\frac{\epsilon}{1+\epsilon}},$$

where $C_1 = \dfrac{\kappa}{\kappa - 1} LD$, $C_2 = (1+\epsilon)\left(2\sqrt{2} + \dfrac{5}{3}\right) \|f\|_\infty$, *and* Δ_0 *is the instantaneous regret of the initial arm* $\boldsymbol{x}_0$.

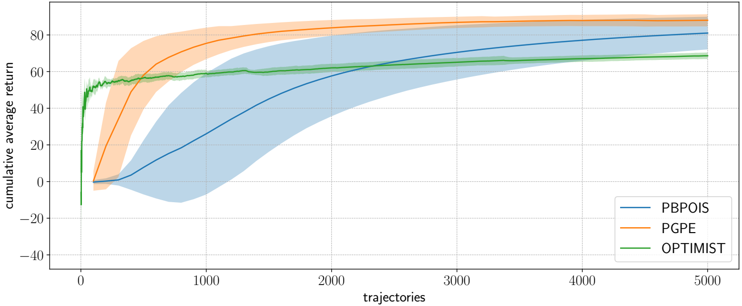
Plan

1. Basics of Reinforcement Learning
2. Exploration in Policy Search
3. Problem Formalization
4. OPTIMIST
5. Experiments
6. Conclusions

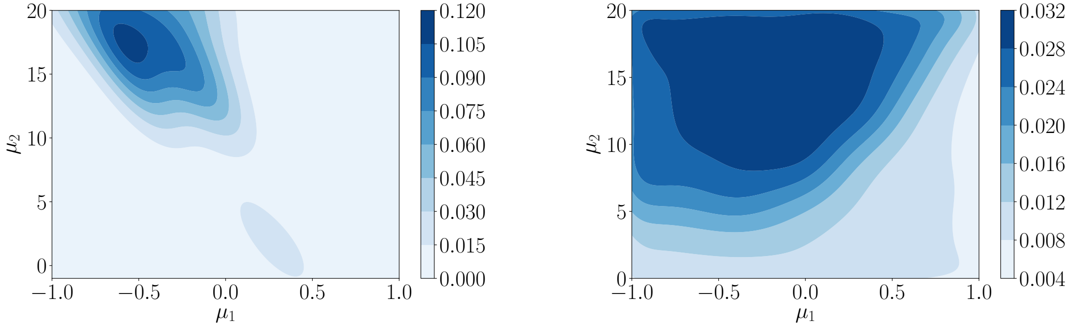
Linear Quadratic Gaussian Regulator



Mountain Car - Performance



Mountain Car - Exploration



Plan

Conclusions

Original Contributions

1. Algorithmic contributions: OPTIMIST and OPTIMIST2.

Conclusions

Original Contributions

1. Algorithmic contributions: OPTIMIST and OPTIMIST2.
2. Theoretical contributions:

Conclusions

Original Contributions

1. Algorithmic contributions: OPTIMIST and OPTIMIST2.
2. Theoretical contributions:
   2.1 novel problem formalization of Policy Search;

Conclusions

Original Contributions

1. Algorithmic contributions: OPTIMIST and OPTIMIST2.
2. Theoretical contributions:
   2.1 novel problem formalization of Policy Search;
   2.2 proved sub-linear regret for both algorithms.

Conclusions

Original Contributions

1. Algorithmic contributions: OPTIMIST and OPTIMIST2.
2. Theoretical contributions:
   2.1 novel problem formalization of Policy Search;
   2.2 proved sub-linear regret for both algorithms.
3. Experimental contributions

Conclusions

Original Contributions

1. Algorithmic contributions: OPTIMIST and OPTIMIST2.
2. Theoretical contributions:
    2.1 novel problem formalization of Policy Search;
    2.2 proved sub-linear regret for both algorithms.
3. Experimental contributions
    3.1 Linear Quadratic Gaussian regulation;

Conclusions

Original Contributions

1. Algorithmic contributions: OPTIMIST and OPTIMIST2.
2. Theoretical contributions:
   2.1 novel problem formalization of Policy Search;
   2.2 proved sub-linear regret for both algorithms.
3. Experimental contributions
   3.1 Linear Quadratic Gaussian regulation;
   3.2 Continuous Mountain Car;

Conclusions

Original Contributions

1. Algorithmic contributions: OPTIMIST and OPTIMIST2.
2. Theoretical contributions:
   2.1 novel problem formalization of Policy Search;
   2.2 proved sub-linear regret for both algorithms.
3. Experimental contributions
   3.1 Linear Quadratic Gaussian regulation;
   3.2 Continuous Mountain Car;
   3.3 Inverted Pendulum.

Conclusions

Original Contributions

1. Algorithmic contributions: OPTIMIST and OPTIMIST2.
2. Theoretical contributions:
   2.1 novel problem formalization of Policy Search;
   2.2 proved sub-linear regret for both algorithms.
3. Experimental contributions
   3.1 Linear Quadratic Gaussian regulation;
   3.2 Continuous Mountain Car;
   3.3 Inverted Pendulum.

Paper submission at **ICML2019** (International Conference on Machine Learning)

Conclusions

Original Contributions

1. Algorithmic contributions: OPTIMIST and OPTIMIST2.
2. Theoretical contributions:
   2.1 novel problem formalization of Policy Search;
   2.2 proved sub-linear regret for both algorithms.
3. Experimental contributions
   3.1 Linear Quadratic Gaussian regulation;
   3.2 Continuous Mountain Car;
   3.3 Inverted Pendulum.

Paper submission at **ICML2019** (International Conference on Machine Learning)

Future Works

Conclusions

Original Contributions

1. Algorithmic contributions: OPTIMIST and OPTIMIST2.
2. Theoretical contributions:
   2.1 novel problem formalization of Policy Search;
   2.2 proved sub-linear regret for both algorithms.
3. Experimental contributions
   3.1 Linear Quadratic Gaussian regulation;
   3.2 Continuous Mountain Car;
   3.3 Inverted Pendulum.

Paper submission at **ICML2019** (International Conference on Machine Learning)

Future Works

1. Optimization problem

Conclusions

Original Contributions

1. Algorithmic contributions: OPTIMIST and OPTIMIST2.
2. Theoretical contributions:
   2.1 novel problem formalization of Policy Search;
   2.2 proved sub-linear regret for both algorithms.
3. Experimental contributions
   3.1 Linear Quadratic Gaussian regulation;
   3.2 Continuous Mountain Car;
   3.3 Inverted Pendulum.

Paper submission at **ICML2019** (International Conference on Machine Learning)

Future Works

1. Optimization problem
2. Posterior sampling

Thank you for your attention!

References

Abbasi-Yadkori, Y., Pál, D., and Szepesvári, C. (2011).
Improved algorithms for linear stochastic bandits.
In *Advances in Neural Information Processing Systems*, pages 2312–2320.

Agrawal, R. (1995a).
The continuum-armed bandit problem.
SIAM Journal on Control and Optimization, 33(6):1926–1951.

Agrawal, R. (1995b).
Sample mean based index policies by o (log n) regret for the multi-armed bandit problem.
Advances in Applied Probability, 27(4):1054–1078.

Agrawal, S. and Goyal, N. (2013).
Further optimal regret bounds for thompson sampling.
In *Artificial intelligence and statistics*, pages 99–107.

Amari, S.-I. (1998).