Transfer of Samples in Policy Search via Multiple Importance Sampling

Andrea Tirinzoni, Mattia Salvini, and Marcello Restelli

{andrea.tirinzoni, marcello.restelli}@polimi.it, mattia.salvini@mail.polimi.it

POLITECNICO MILANO 1863

GitHub

Motivation

Policy Search (PS):

- Effective RL method for **large continuous MDPs**
- Optimize a **parametric policy** $\pi_{\boldsymbol{\theta}}$ to maximize the expected return $J(\boldsymbol{\theta})$, typically via **gradient ascent**
- Needs **large batches** for accurate policy evaluation/improvement $\rightarrow$ **High sample complexity**

In practice, lots of available samples are **not used** by standard PS algorithms

- Different **policies** (e.g., from past iterations)
- Different **dynamics** (e.g., a simulator)

Goal: use these data to reduce sample complexity
$\longrightarrow$ **Transfer of samples**

Related Works

Transfer of samples: focus on **batch value-based** RL

- Learning similarity measures [4]
- Model-based settings [6]
- Theoretical properties [3]
- Shared dynamics [2]
- Transfer via **importance weighting** [7]

Contributions

1. **Algorithmic**. We propose several **MIS gradient estimators** that effectively reuse trajectories from different policies/dynamics
   - **Variance reduction** via per-decision weights [5] and control variates [1]
   - **Adaptive batch size** via ESS
   - Efficient **MSE-aware method** to estimate the weights for unknown transition models
2. **Theoretical**. We formally establish
   - **robustness to negative transfer** for the ideal case of known transition models
   - a general upper bound on the MSE of importance-weighted gradient estimators
3. **Empirical**. We empirically evaluate our algorithms on three different domains with increasing level of difficulty

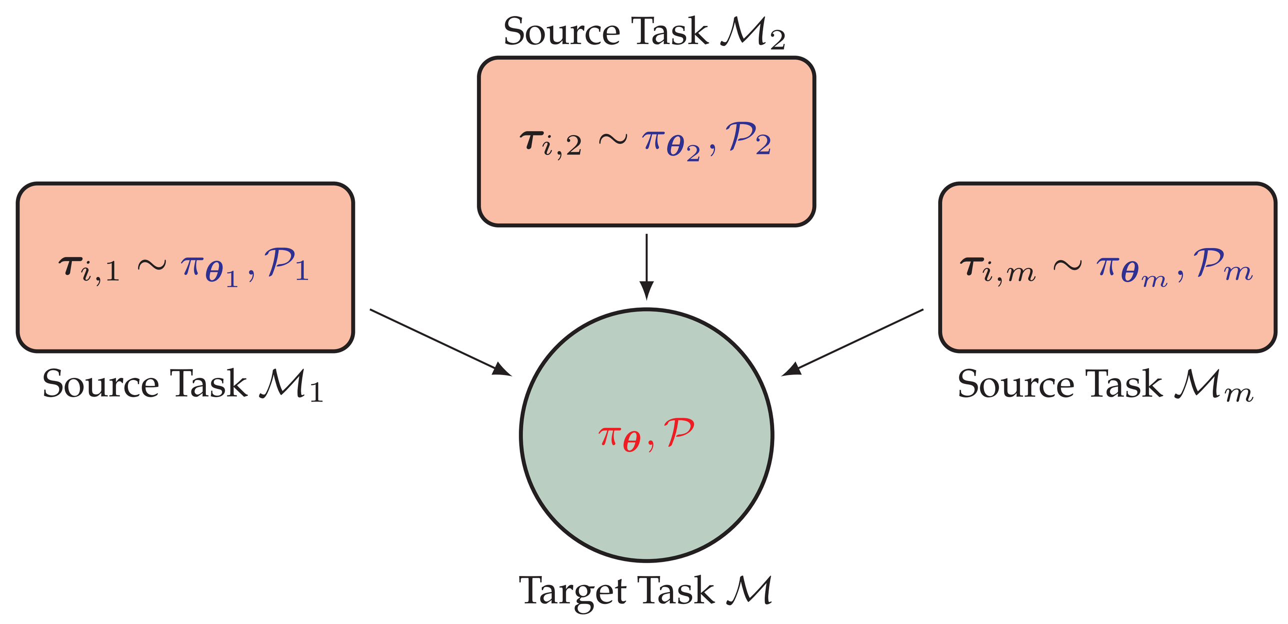
Transfer via Multiple Importance Sampling (MIS)

Transfer samples from a set of **source tasks** to **speed-up** the learning process of a **target task**

- Tasks are MDPs $\mathcal{M}_j = \langle \mathcal{S}, \mathcal{A}, \mathcal{R}, \boldsymbol{\mathcal{P}_j}, \mu \rangle$ with **different dynamics**
- **Goal**: improve the **gradient estimation** in the target task

$\longrightarrow$ Larger and less noisy steps

$\longrightarrow$ Faster convergence



MIS Gradient Estimators

Transfer *all* samples in a weighted Monte Carlo estimator

$$\widehat{\nabla}^{\mathrm{MIS}}_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) = \frac{1}{n} \sum_{j=0}^{m} \sum_{i=1}^{n_j} \underbrace{w(\boldsymbol{\tau}_{i,j})}_{\text{weights}} \underbrace{\nabla_{\boldsymbol{\theta}} \log p(\boldsymbol{\tau}_{i,j}|\boldsymbol{\theta}) \mathcal{R}(\boldsymbol{\tau}_{i,j})}_{\text{classic gradient (REINFORCE)}}$$

Can be combined with other **variance reduction** techniques

- MIS + **per-decision (PD)** weights
- MIS + **control variates (CV)**

	IS	MIS	PD	CV	PDCV
Unbiased	✓	✓	✓	✓	✓
Bounded weights	✗	✓	✓	✓	✓
Horizon-independent variance	✗	✓	✓	✓	✓
Handles long trajectories	✗	✗	✓	✗	✓
Handles high magnitudes	✗	✗	✗	✓	✓
Robustness to negative transfer	✗	✗	✗	✓	✓

MIS with Balance Heuristics

Ratio w.r.t. **mixture** of source distributions

$$w(\boldsymbol{\tau}) := \frac{p(\boldsymbol{\tau}|\boldsymbol{\theta}, \mathcal{P})}{\sum_{j=0}^{m} \alpha_j p(\boldsymbol{\tau}|\boldsymbol{\theta}_j, \mathcal{P}_j)}$$

- **Unbiased** with **bounded weigths**
- Near-optimal [8]

Algorithm

Initialize $\boldsymbol{\theta}_0 \leftarrow$ Init-Policy$(\mathcal{M}, \mathcal{D})$
while not converged **do**
  Compute $\min_{n_0 \geq n_{\min}} \{n_0 \mid \widehat{\mathrm{ESS}}(n_0; \mathcal{D}) \geq \mathrm{ESS}_{\min}\}$
  Sample n_0 trajectories from $\mathcal{M}$ under $\pi_{\boldsymbol{\theta}_k}$
  Store $\mathcal{D} \leftarrow \mathcal{D} \cup \{\langle(\boldsymbol{\tau}_1, \dots, \boldsymbol{\tau}_{n_0}), \boldsymbol{\theta}_k, \mathcal{P}\rangle\}$
  Update $\boldsymbol{\theta}_{k+1} \leftarrow \boldsymbol{\theta}_k + \eta_k \widehat{\nabla}_{\boldsymbol{\theta}} J(\boldsymbol{\theta}_k)$
end while

MSE-Aware Model Estimation

Problem: **Transition models unknown** $\rightarrow$ Importance weights cannot be computed
Solution: Online minimization of an upper-bound to the **expected MSE** of $\widehat{\nabla}^{\mathrm{MIS}}_{\boldsymbol{\theta}} J(\boldsymbol{\theta})$

Assumption 1. Source models fixed and **known**
Assumption 2. **Gaussian** transitions $\boldsymbol{s}_{t+1} = f(\boldsymbol{s}_t, \boldsymbol{a}_t) + \mathcal{N}(0, \sigma^2 \boldsymbol{I})$ with **uncertain** target model $f \in \mathcal{F}$

For $f \sim \varphi$ and $\boldsymbol{\tau}_{i,j} \sim p(\cdot|\boldsymbol{\theta}_j, f_j)$, with $\bar{f}(\boldsymbol{s}, \boldsymbol{a}) := \mathbb{E}_{f\sim\varphi}[f(\boldsymbol{s}, \boldsymbol{a})]$

$$\mathrm{MSE}\left(\widehat{\nabla}^{\mathrm{MIS}}_{\boldsymbol{\theta}} J(\boldsymbol{\theta}; \hat{f})\right) \lesssim \underbrace{\frac{1}{n} \chi^2 \left(p(\cdot|\boldsymbol{\theta}, \hat{f}) \middle\| \sum_{j=1}^{m} \alpha_j p(\cdot|\boldsymbol{\theta}_j, f_j) \right)}_{\text{Minimize the variance of the weights}} + \underbrace{\mathrm{KL}\left(p(\cdot|\boldsymbol{\theta}, \hat{f}) \middle\| p(\cdot|\boldsymbol{\theta}, \bar{f}) \right)}_{\text{Accurately predict the target model}}$$

Can be **efficiently optimized** for

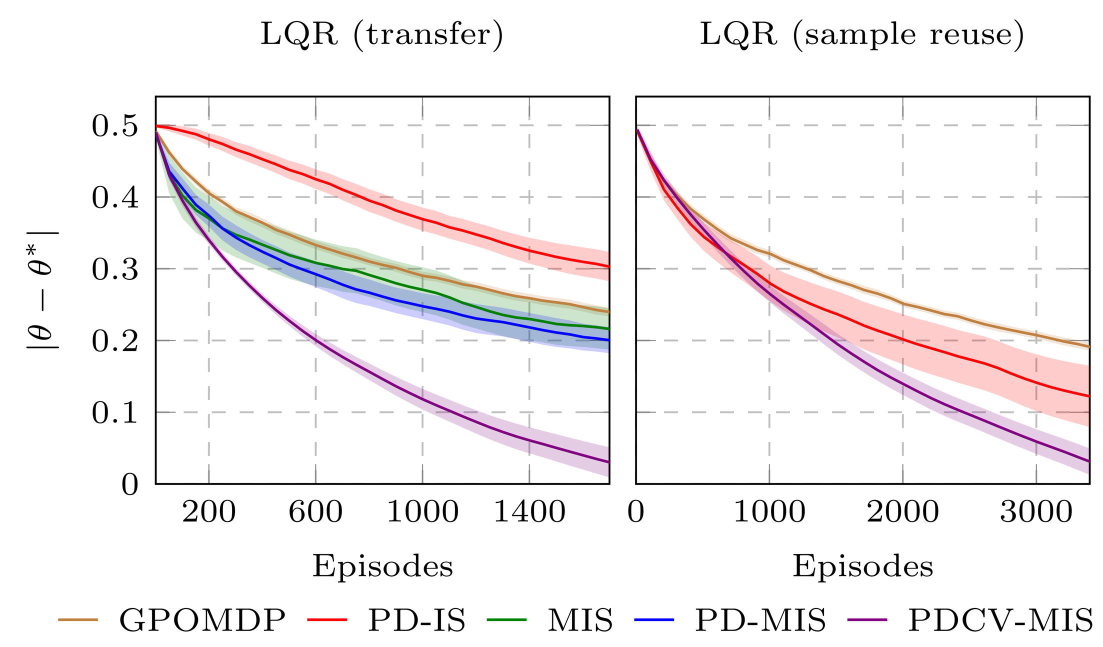
- Discrete set of models
- **Reproducing Kernel Hilbert Spaces** $\rightarrow$ **Closed-form solution**

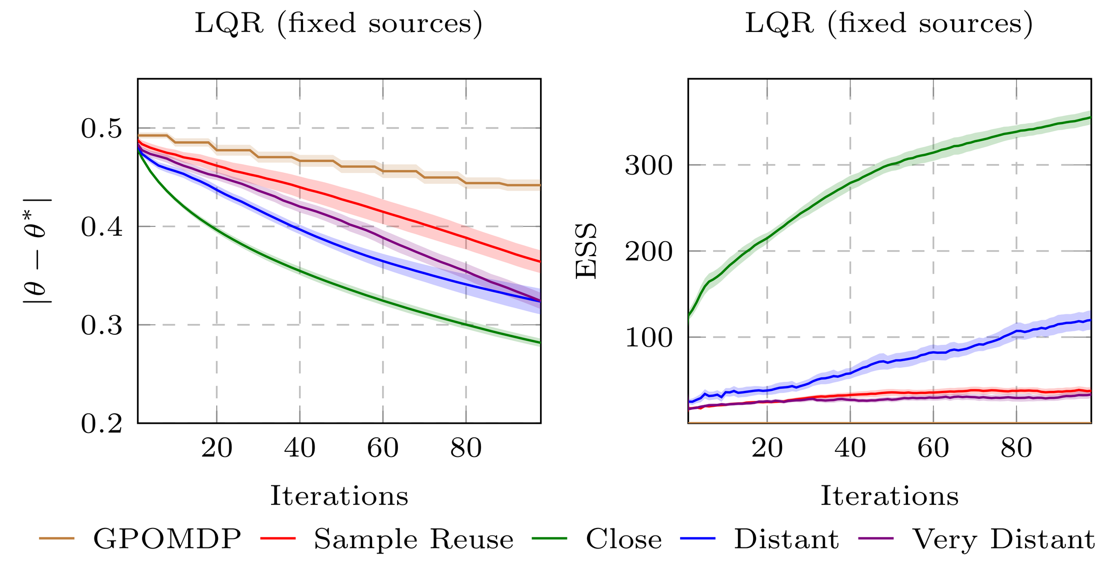
In case of **unknown source models**

- **Offline** estimation
- Fixed during learning

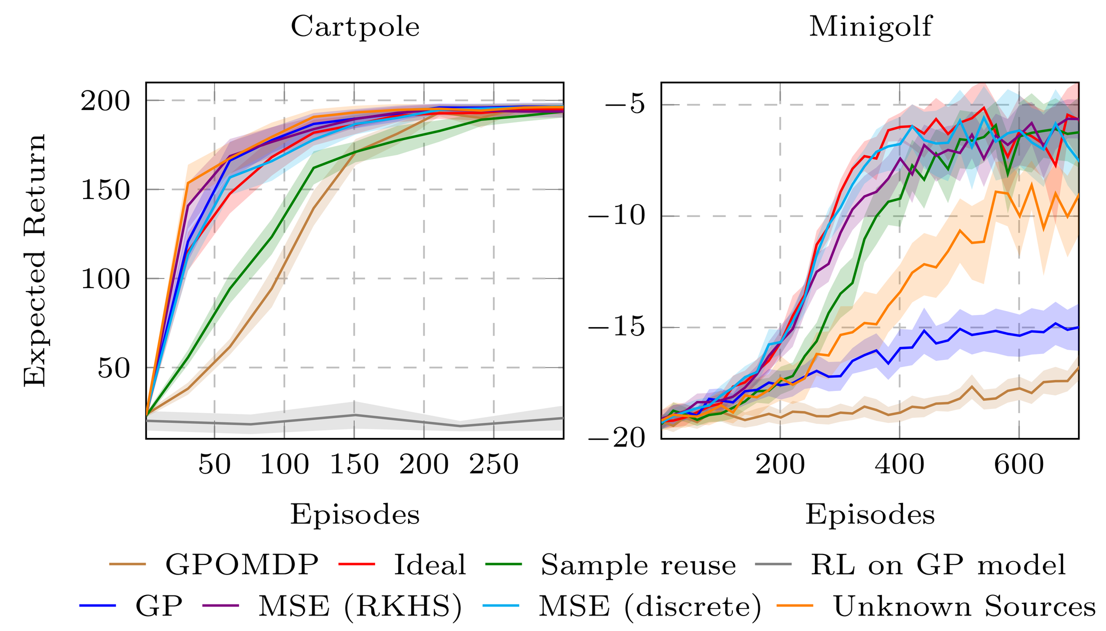
Empirical Results

Known Models





Unknown Models



References

[1] J.M. Hammersley and D.C. Handscomb. *Monte Carlo Methods*. Methuen's monographs on applied probability and statistics. Methuen, 1964.

[2] Romain Laroche and Merwan Barlier. Transfer reinforcement learning with shared dynamics. In *AAAI*, 2017.

[3] Alessandro Lazaric and Marcello Restelli. Transfer from multiple mdps. In *Advances in Neural Information Processing Systems*, 2011.

[4] Alessandro Lazaric, Marcello Restelli, and Andrea Bonarini. Transfer of samples in batch reinforcement learning. In *ICML*, 2008.

[5] Doina Precup, Richard S. Sutton, and Satinder P. Singh. Eligibility traces for off-policy policy evaluation. In *ICML*, 2000.

[6] Matthew E Taylor, Nicholas K Jong, and Peter Stone. Transferring instances for model-based reinforcement learning. In *ECML PKDD*, 2008.

[7] Andrea Tirinzoni, Andrea Sessa, Matteo Pirotta, and Marcello Restelli. Importance weighted transfer of samples in reinforcement learning. In *Proceedings of the 35th International Conference on Machine Learning*. PMLR, 2018.

[8] Eric Veach and Leonidas J Guibas. Optimally combining sampling techniques for monte carlo rendering. In *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pages 419–428. ACM, 1995.