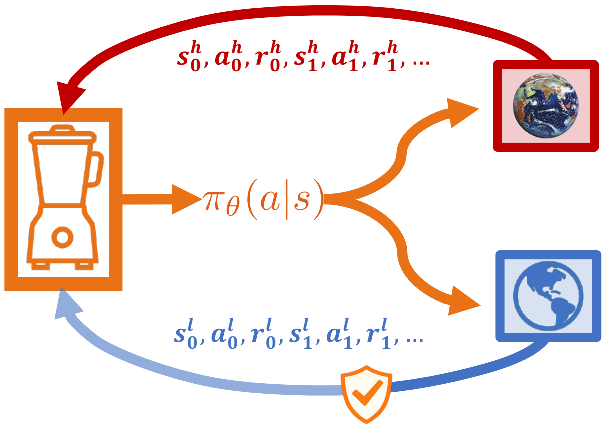
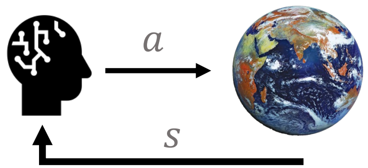
A Multi-Fidelity Control Variate Approach for Policy Gradient Estimation

Xinjie Liu*, Cyrus Neary*, Kushagra Gupta, Wesley A. Suttle, Christian Ellis, Ufuk Topcu, and David Fridovich-Keil



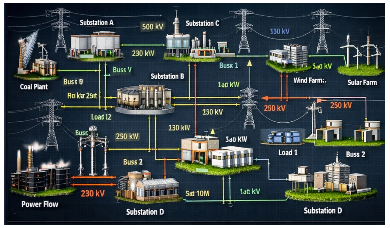
Motivation: Data Scarcity in Reinforcement Learning (RL)

- Online RL algorithms require excessive interaction with the **real environment/high-fidelity simulation**



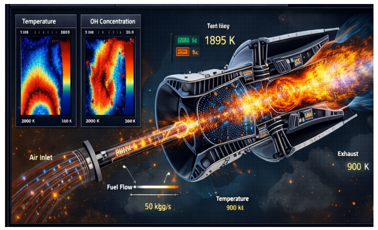
👎 $$$$, slow, even unsafe … but ✅ accurate

Autonomous driving

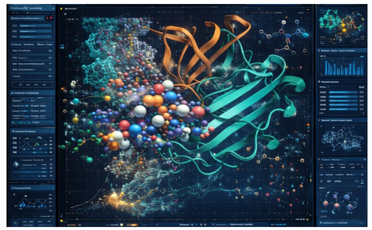


Power systems

Robotics



Combustion simulation

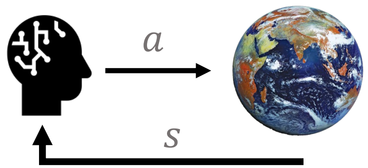


Molecular simulation

Images generated with ChatGPT and Gemini

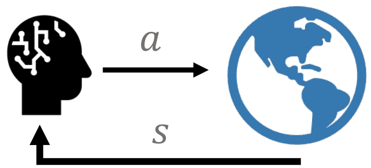
Motivation: Data Scarcity in Reinforcement Learning (RL)

- Online RL algorithms require excessive interaction with the **real environment/high-fidelity simulation**



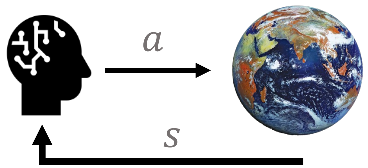
👎 **$$$$, slow, even unsafe … but ✅ accurate**

- **Low-fidelity simulation** provides low-cost ways to gather large datasets: reduced-order models, generative world models, heuristic reward functions, digital twins …



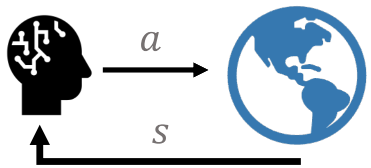
👍 **$, fast, safe … but ❌ biased**

Motivation: Data Scarcity in Reinforcement Learning (RL)



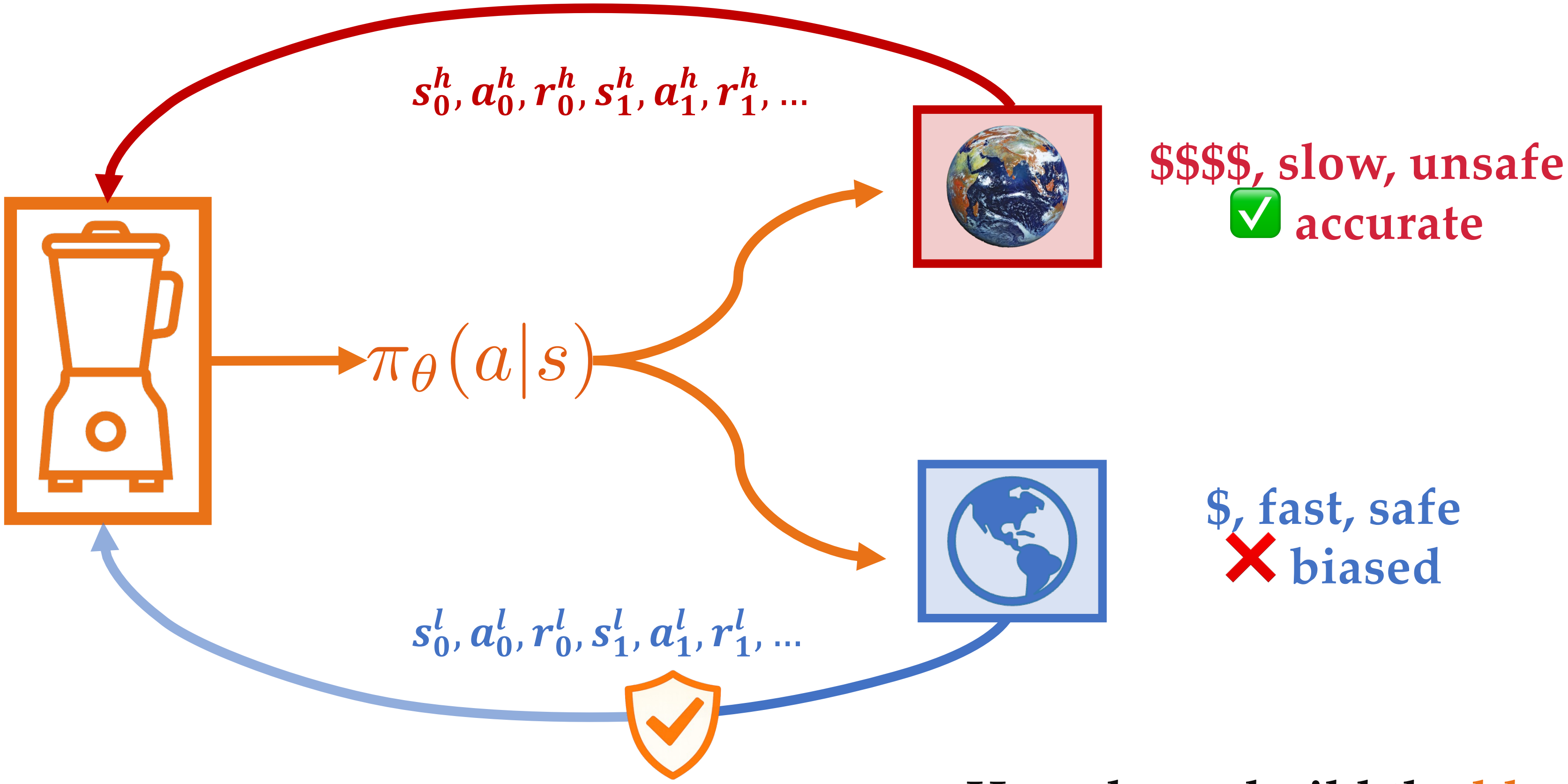
👎 $$$$, slow, even unsafe … but ✅ accurate

$\neq$



👍 $, fast, safe … but ❌ biased

How can we enable sample-efficient RL in the real world by mixing multi-fidelity data, while being robust to low-fidelity data biases?



… How do we build the blender?

Modeling Multi-Fidelity RL Problems

$$\mathcal{M}^h = \left(S,\, A,\, \Delta_{sI},\, \gamma,\, T,\, p^h,\, R^h\right)$$

$$$$
✅ accurate

$$\mathcal{M}^l = \left(S,\, A,\, \Delta_{sI},\, \gamma,\, T,\, p^l,\, R^l\right)$$

$
❌ biased

Objective: Learn a performant policy for the **high-fidelity environment**

$$\max_{\theta} \mathbb{E}\left[\sum_{t=0}^{T} \gamma^t r_t^h \,\middle|\, \tau^h \sim \mathcal{M}^h(\pi_\theta)\right], \qquad \tau^h = s_0^h, a_0^h, r_0^h, \ldots, s_T^h$$

Reminder: On-Policy Policy Gradient Algorithms

Objective: Maximize $J_\theta = \mathbb{E}[\sum_t \gamma^t r_t \mid \tau \sim \mathcal{M}(\pi_\theta)]$

Strategy: Gradient ascent

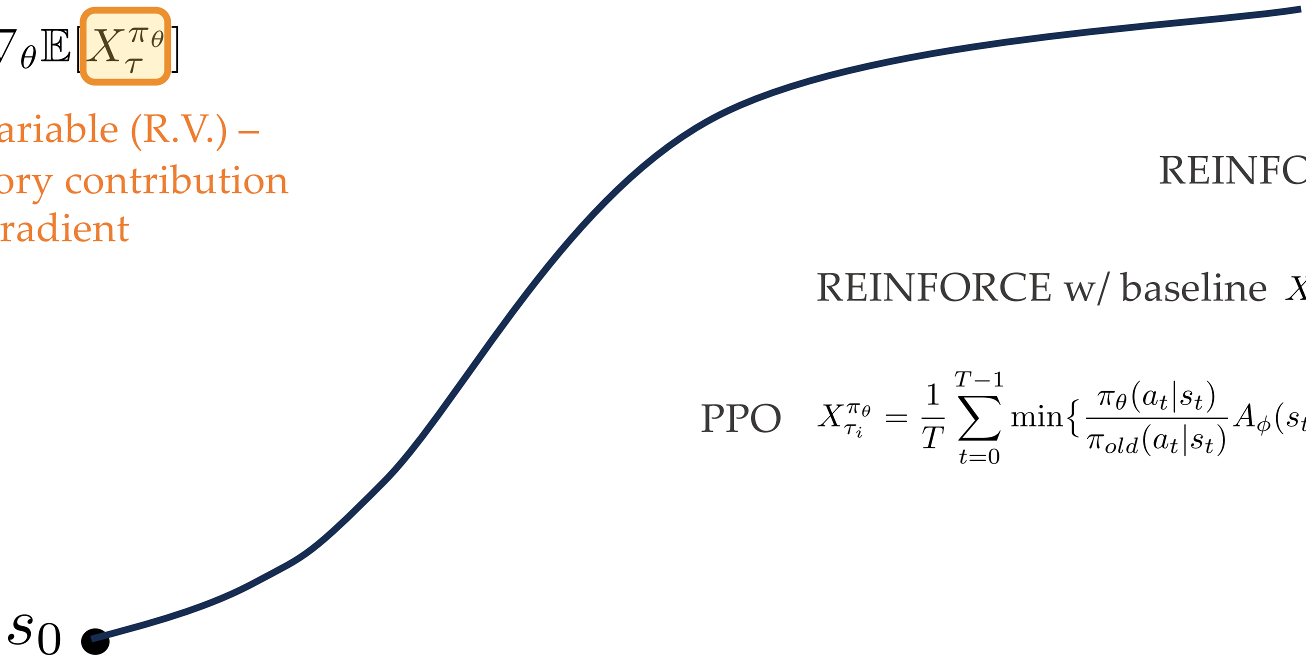
$$\theta_{k+1} = \theta_k + \alpha \nabla_{\theta_k} J_{\theta_k}$$

$$\nabla_\theta J_\theta \approx \nabla_\theta \mathbb{E}[X_\tau^{\pi_\theta}]$$

Random variable (R.V.) – per-trajectory contribution to policy gradient

Sample $\tau \sim \mathcal{M}(\pi_\theta)$

Compute R.V. $X_\tau^{\pi_\theta}$



REINFORCE $X_{\tau_i}^{\pi_\theta} = \frac{1}{T}\sum_{t=0}^{T-1} G_t \log(\pi_\theta(a_t|s_t))$

REINFORCE w/ baseline $X_{\tau_i}^{\pi_\theta} = \frac{1}{T}\sum_{t=0}^{T-1} \big(G_t - V_\phi(s_t)\big) \log(\pi_\theta(a_t|s_t))$

PPO $X_{\tau_i}^{\pi_\theta} = \frac{1}{T}\sum_{t=0}^{T-1} \min\{\frac{\pi_\theta(a_t|s_t)}{\pi_{old}(a_t|s_t)} A_\phi(s_t, a_t), clip[\frac{\pi_\theta(a_t|s_t)}{\pi_{old}(a_t|s_t)}, 1-\epsilon, 1+\epsilon] A_\phi(s_t, a_t)\}$

...

Reminder: On-Policy Policy Gradient Algorithms

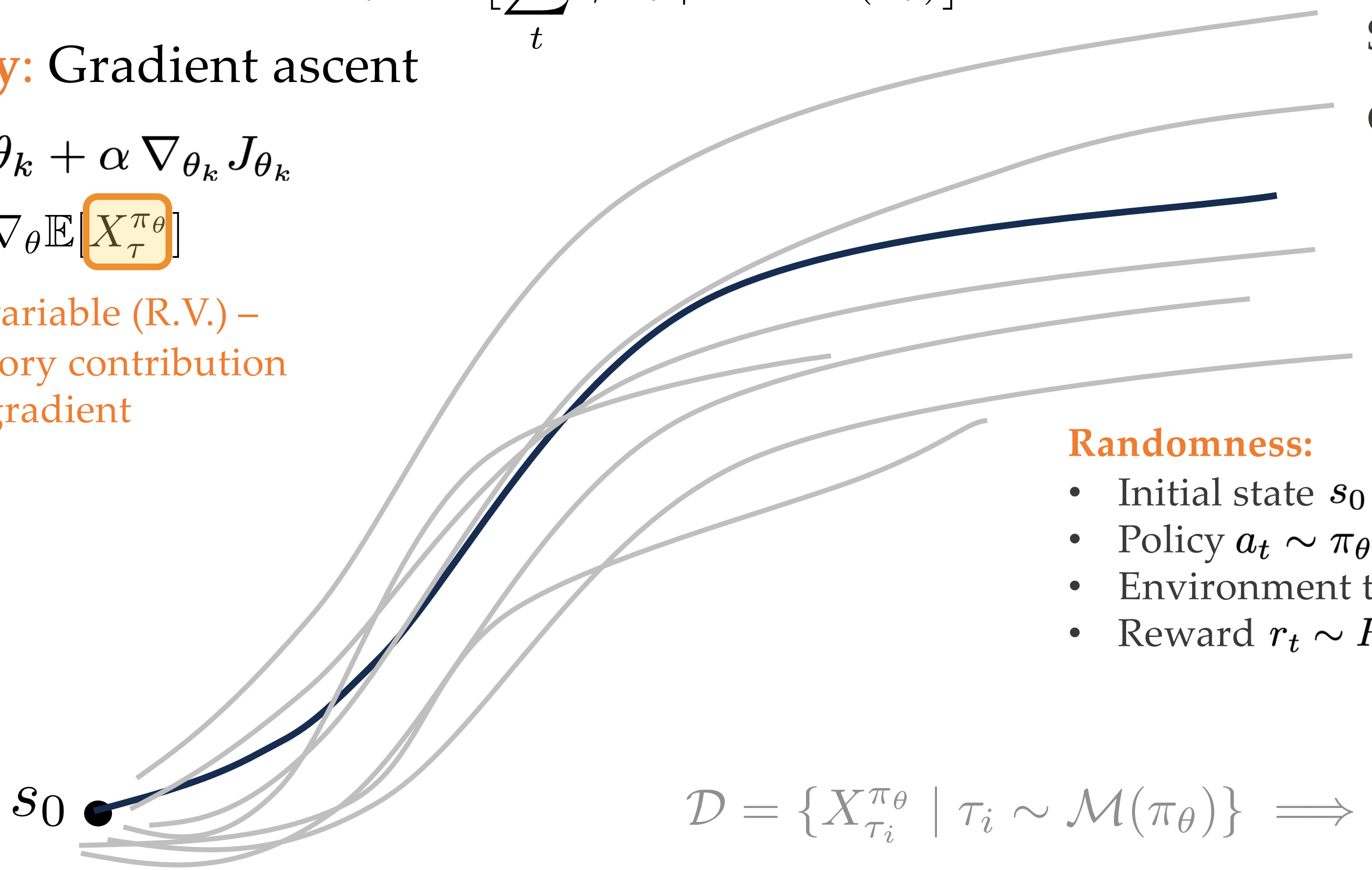
Objective: Maximize $J_\theta = \mathbb{E}[\sum_t \gamma^t r_t \mid \tau \sim \mathcal{M}(\pi_\theta)]$

Strategy: Gradient ascent

$$\theta_{k+1} = \theta_k + \alpha \nabla_{\theta_k} J_{\theta_k}$$

$$\nabla_\theta J_\theta \approx \nabla_\theta \mathbb{E}[X_\tau^{\pi_\theta}]$$

Random variable (R.V.) – per-trajectory contribution to policy gradient



Sample $\tau \sim \mathcal{M}(\pi_\theta)$

Compute R.V. $X_\tau^{\pi_\theta}$

Randomness:

- Initial state $s_0 \sim \Delta_{s_I}$
- Policy $a_t \sim \pi_\theta(\cdot \mid s_t)$
- Environment transition $s_{t+1} \sim p(\cdot \mid s_t, a_t)$
- Reward $r_t \sim R(s_t, a_t, s_{t+1})$

$$\mathcal{D} = \{X_{\tau_i}^{\pi_\theta} \mid \tau_i \sim \mathcal{M}(\pi_\theta)\} \implies \nabla_\theta J_\theta \approx \nabla_\theta \frac{1}{N} \sum_{i=1}^{N} X_{\tau_i}^{\pi_\theta}$$

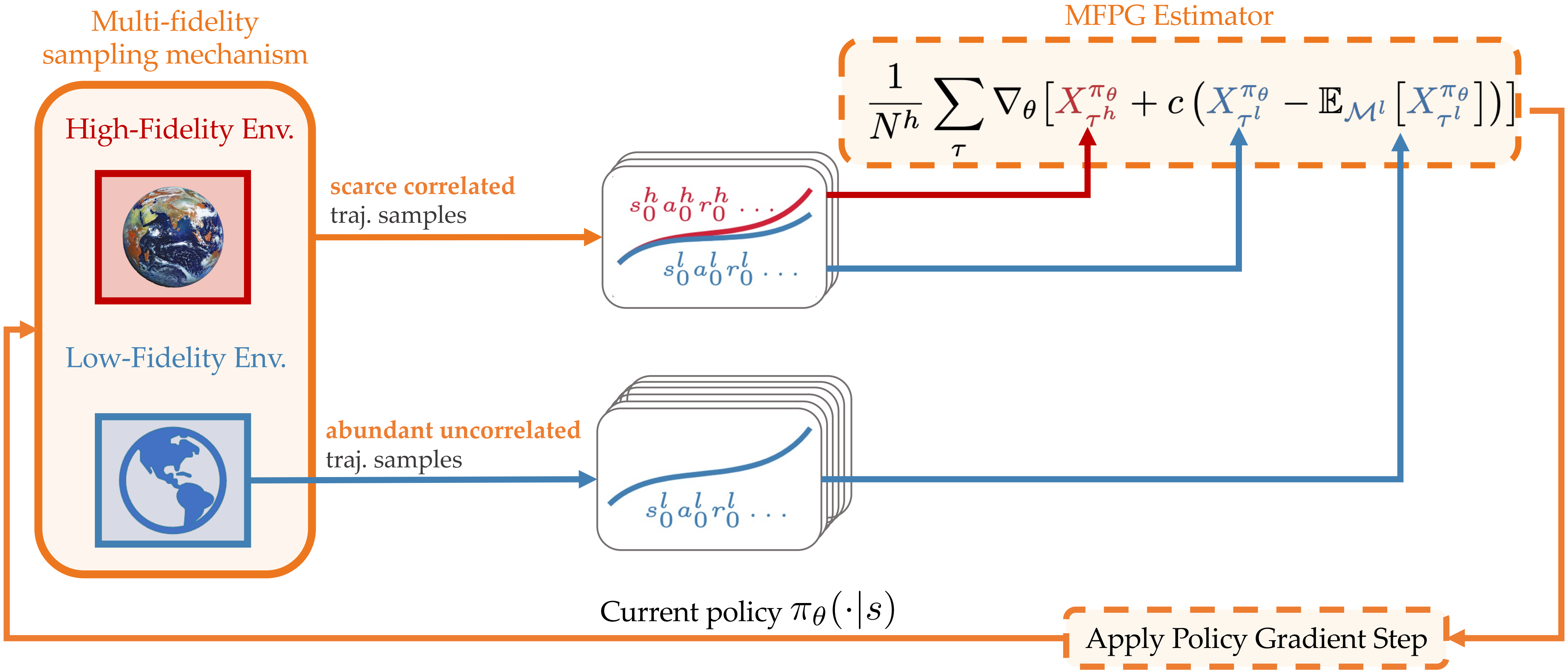
Challenge & Strategy

$$\nabla_\theta J_\theta \approx \nabla_\theta \frac{1}{N^h} \sum_{i=1}^{N^h} X_{\tau_i^h}^{\pi_\theta}$$

Challenge: high-fidelity data **scarcity** (small N^h) causing high **estimation variance** for $\mathbb{E}\left[X_{\tau^h}^{\pi_\theta}\right]$ and slow convergence

Strategy: ground learning in high-fidelity samples (**unbiased**); use abundant low-fidelity samples solely as a **variance-reduction** tool

The Multi-Fidelity Policy Gradient (MFPG) Framework



Instantiate MPFG with established policy gradient loss:

REINFORCE: $X_\tau^{\pi_\theta} = \frac{1}{T} \sum_{t=0}^{T-1} G_t \log \pi_\theta(a_t \mid s_t)$

Multi-Fidelity Control Variate Estimator

$$Z^{\pi_\theta}(c) := X^{\pi_\theta}_{\tau^h} + c\left(X^{\pi_\theta}_{\tau^l} - \mathbb{E}_{\mathcal{M}^l}\left[X^{\pi_\theta}_{\tau^l}\right]\right)$$

$$\min_c \mathrm{Var}(Z^{\pi_\theta}(c)) \Longrightarrow c^\star = -\rho\left(X^{\pi_\theta}_{\tau^h}, X^{\pi_\theta}_{\tau^l}\right) \frac{\sqrt{\mathrm{Var}\left(X^{\pi_\theta}_{\tau^h}\right)}}{\sqrt{\mathrm{Var}\left(X^{\pi_\theta}_{\tau^l}\right)}}$$ (estimated from training data)

Pearson correlation

Lemma 1 Unbiasedness and variance reduction

- $\mathbb{E}_{\mathcal{M}^h}[Z^{\pi_\theta}(c)] = \mathbb{E}_{\mathcal{M}^h}\left[X^{\pi_\theta}_{\tau^h}\right]$
- $\mathrm{Var}(Z^{\pi_\theta}(c^\star)) = \left(1 - \rho^2\left(X^{\pi_\theta}_{\tau^h}, X^{\pi_\theta}_{\tau^l}\right)\right) \mathrm{Var}\left(X^{\pi_\theta}_{\tau^h}\right)$

How do we draw correlated multi-fidelity samples?

Theorem 1 Faster finite-sample convergence of MFPG-REINFORCE than plain REINFORCE

Bottom line: 👍 low-fidelity data $\Longrightarrow \rho^2\left(X^{\pi_\theta}_{\tau^h}, X^{\pi_\theta}_{\tau^l}\right) \uparrow \Longrightarrow \mathrm{Var}(Z^{\pi_\theta}(c^\star))$ ↓

faster MFPG algorithm convergence

Sampling Correlated Trajectories

Randomness:

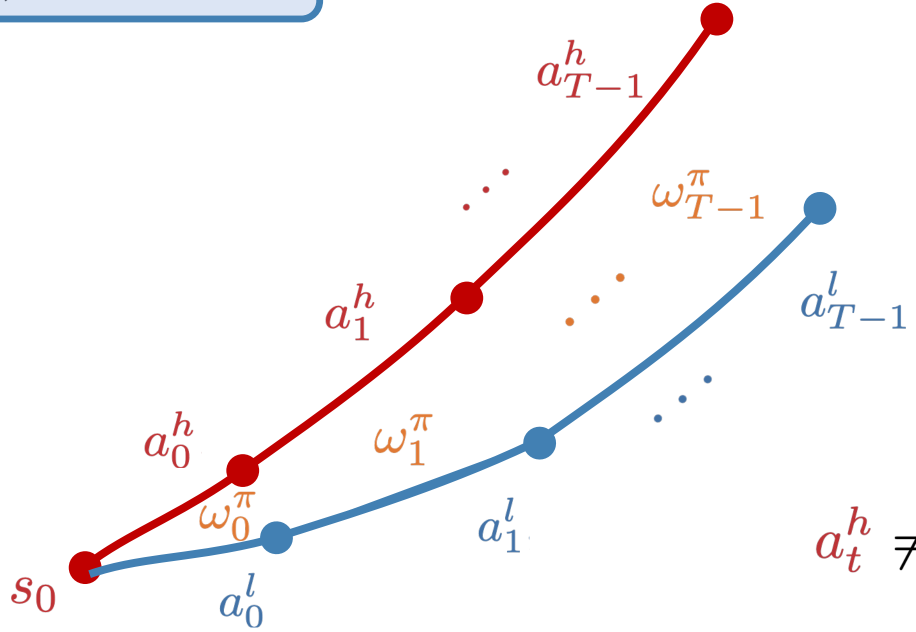
- Initial state $s_0 \sim \Delta_{s_I}$
- Policy $a_t \sim \pi_\theta(\cdot \mid s_t)$

👍 Can be controlled by the algorithm! (share initial state + action sampling noise)

- Reset low-fidelity simulator to matched s_0
- Policy reparameterization trick $a_t \leftarrow \pi_\theta(s_t, \omega_t^{\pi_\theta})$

- Environment transition $s_{t+1} \sim p(\cdot \mid s_t, a_t)$
- Reward $r_t \sim R(s_t, a_t, s_{t+1})$

Uncontrolled randomness



$a_t^h \neq a_t^l \,!\, (s_t^h \neq s_t^l)$

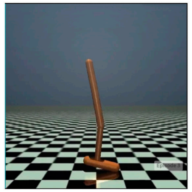
Experimental Results

- Variance reduction
- Reliability & robustness to fidelity gaps
  - Dynamics shift
  - Misspecified (negated) reward

Experimental Results

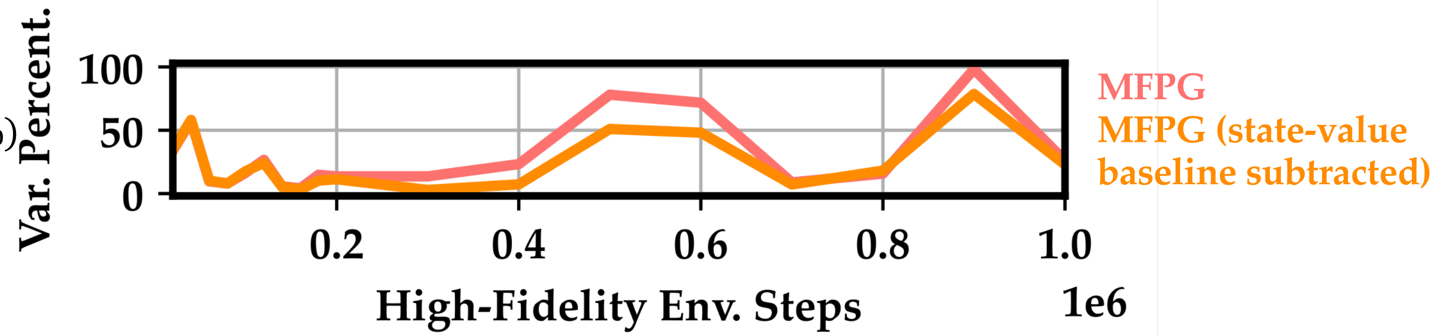
- **Variance reduction**
- Reliability & robustness to fidelity gaps
  - Dynamics shift
  - Misspecified (negated) reward

MFPG substantially reduces PG estimation variance



Robot control task: MuJoCo Hopper
High-fidelity environment: changed friction (1.2×)
Baseline: High-Fidelity Only

$$\frac{\mathrm{Var(MFPG)}}{\mathrm{Var(High-Fidelity\ Only)}}\ (\%)$$



When high-fidelity data are scarce, MFPG reduces variance significantly—(see paper) **far more substantial** than common state-value baseline subtraction

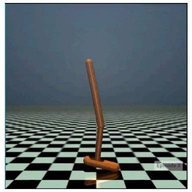
Experimental Results

- Variance reduction
- Reliability & robustness to fidelity gaps
  - Dynamics shift
  - Reward misspecification

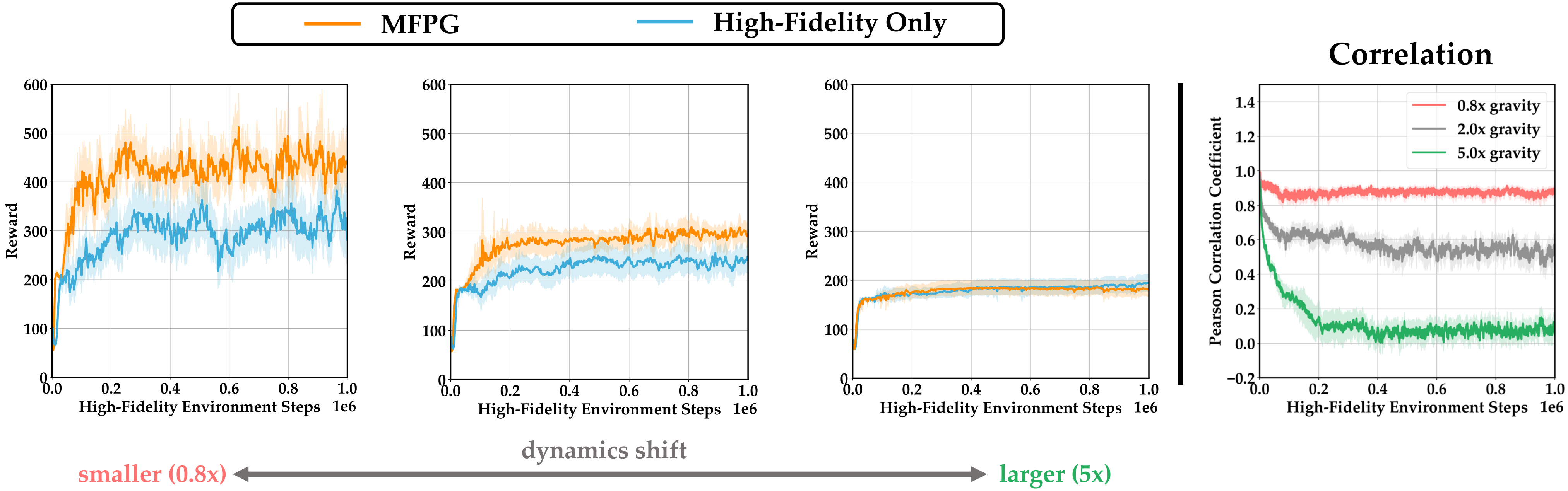
Experimental Results

- Variance reduction
- **Reliability & robustness to fidelity gaps**
  - **Dynamics shift**
  - Reward misspecification

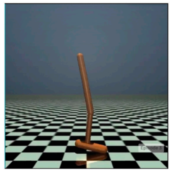
MFPG improves performance by leveraging multi-fidelity correlation

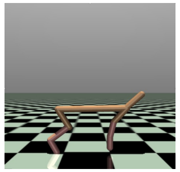


Robot control task: MuJoCo Hopper
High-fidelity environment: changed gravity
Baseline: High-Fidelity Only



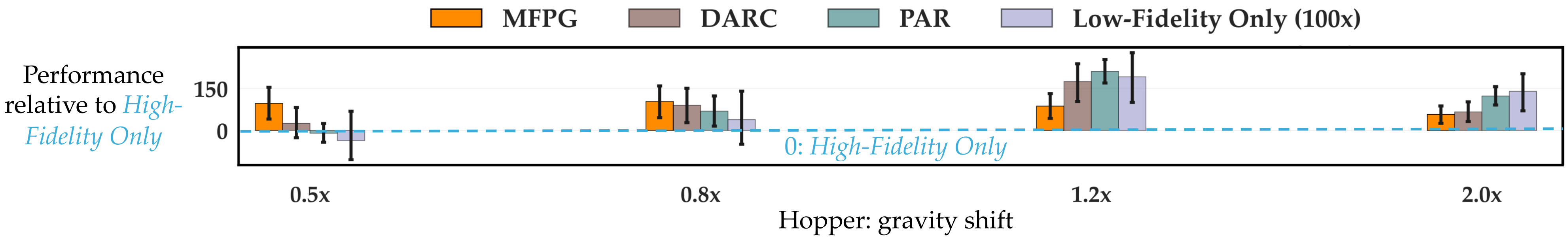
MFPG presents the strongest consistency and robustness compared to the evaluated off-dynamics RL baselines





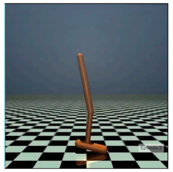
Robot control tasks: MuJoCo Hopper, HalfCheetah
High-fidelity environment: changed gravity, friction
Baselines: off-dynamics RL (DARC [1], PAR [2]), Low-Fidelity Only
Common baseline: High-Fidelity Only

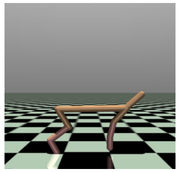
- **When low-fidelity data are neutral/beneficial and dynamics gaps are mild/moderate, MFPG is the only method that consistently outperforms High-Fidelity Only across all settings**
  - Error bars: 95% bootstrap confidence intervals; bars strictly above 0 indicate significant improvement vs. High-Fidelity Only



[1] Eysenbach et al. "Off-Dynamics Reinforcement Learning: Training for Transfer with Domain Classifiers", ICLR 2021.
[2] Lyu et al. "Cross-Domain Policy Adaptation by Capturing Representation Mismatch", ICML 2024.

MFPG presents the strongest consistency and robustness compared to the evaluated off-dynamics RL baselines

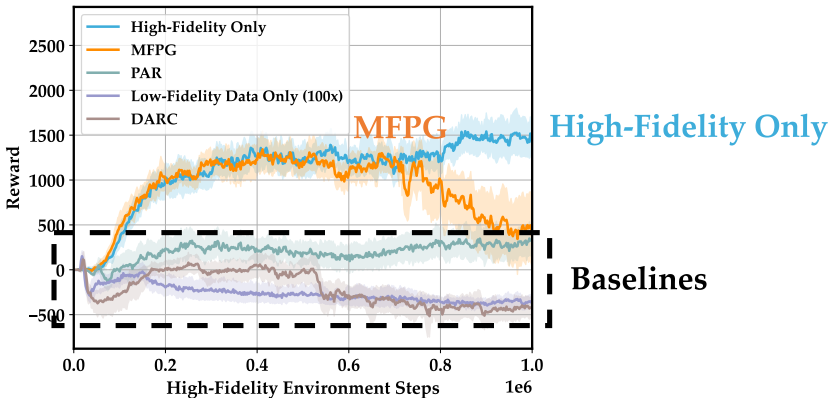




Robot control tasks: MuJoCo Hopper, HalfCheetah
High-fidelity environment: changed gravity, friction
Baselines: off-dynamics RL (DARC [1], PAR [2]), Low-Fidelity Only
Common baseline: High-Fidelity Only

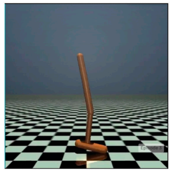
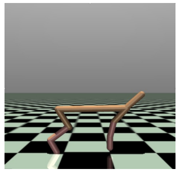
- **When low-fidelity data are neutral/beneficial and dynamics gaps are mild/moderate, MFPG is the only method that consistently outperforms High-Fidelity Only across all settings**
- **When low-fidelity data are harmful, MFPG presents the strongest robustness**
  - MFPG tracks High-Fidelity Only for most of training (cautious use of low-fidelity data only for variance reduction)
  - Baselines fail catastrophically (aggressive exploitation of low-fidelity data)

Extreme case: HalfCheetah (5× friction)



[1] Eysenbach et al. "Off-Dynamics Reinforcement Learning: Training for Transfer with Domain Classifiers", ICLR 2021.
[2] Lyu et al. "Cross-Domain Policy Adaptation by Capturing Representation Mismatch", ICML 2024.

MFPG presents the strongest consistency and robustness compared to the evaluated off-dynamics RL baselines

Robot control tasks: MuJoCo Hopper, HalfCheetah
High-fidelity environment: changed gravity, friction
Baselines: off-dynamics RL (DARC [1], PAR [2]), Low-Fidelity Only
Common baseline: High-Fidelity Only

- **When low-fidelity data are neutral/beneficial and dynamics gaps are mild/moderate, MFPG is the only method that consistently outperforms High-Fidelity Only across all settings**
- **When low-fidelity data are harmful, MFPG presents the strongest robustness**
  - MFPG tracks High-Fidelity Only for most of training (cautious use of low-fidelity data only for variance reduction)
  - Baselines fail catastrophically (aggressive exploitation of low-fidelity data)
  - Sweep of 39 scenarios (paper): MFPG is the most robust among the evaluated methods

[1] Eysenbach et al. “Off-Dynamics Reinforcement Learning: Training for Transfer with Domain Classifiers”, ICLR 2021.
[2] Lyu et al. “Cross-Domain Policy Adaptation by Capturing Representation Mismatch”, ICML 2024.

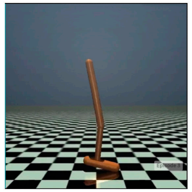
Experimental Results

- Variance reduction
- Reliability & robustness to fidelity gaps
  - Dynamics shift
  - Reward misspecification

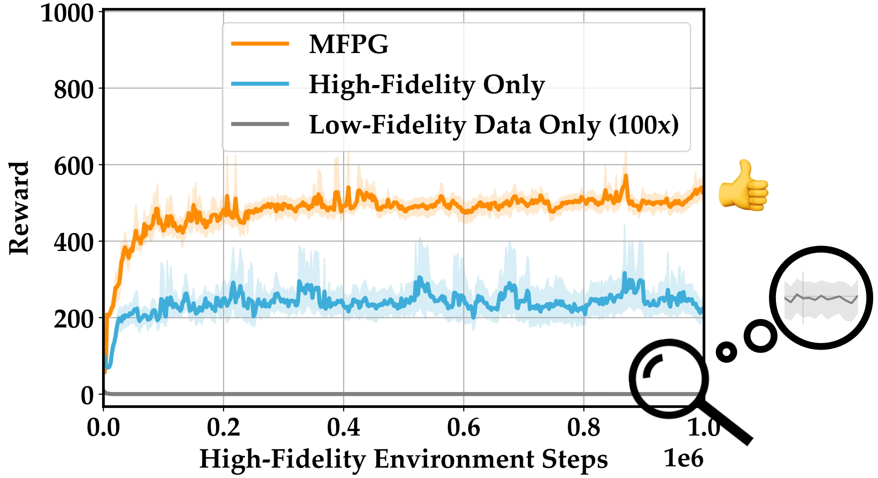
Experimental Results

- Variance reduction
- **Reliability & robustness to fidelity gaps**
  - Dynamics shift
  - **Reward misspecification**

MFPG benefits from negative correlation (negated low-fidelity reward)



Robot control task: MuJoCo Hopper
Low-fidelity environment: negated reward model
Baseline: High-Fidelity Only, Low-Fidelity Only



Even when the low-fidelity environment is **substantially different** or even **adversarial**, it might still provide useful information for **multi-fidelity training**, e.g., **negative correlation**

Summary

MFPG: **sample-efficient** RL framework by **mixing** scarce high-fidelity data with abundant low-fidelity simulation data

- **grounded** to high-fidelity data (**unbiased**)
- low-fidelity data and cross-fidelity **correlation** for **variance reduction**
- handles **dynamics** gaps and **reward** misspecification
- more **robust** to low-fidelity data biases than off-dynamics RL baselines

Future work:

- Broader algorithms (Appendix G; actor-critic, model-based, off-policy, offline RL)
- Enhancing multi-fidelity correlation
- More general settings (e.g., multiple fidelities, different state-action spaces)
- Real-world RL

Summary

MFPG: **sample-efficient** RL framework by **mixing** scarce high-fidelity data with abundant low-fidelity simulation data

- **grounded** to high-fidelity data (**unbiased**)
- low-fidelity data and cross-fidelity **correlation** for **variance reduction**
- handles **dynamics** gaps and **reward** misspecification
- more **robust** to low-fidelity data biases than off-dynamics RL baselines


Xinjie Liu*


Cyrus Neary*


Kushagra Gupta


Wesley A. Suttle


Christian Ellis


Ufuk Topcu


David Fridovich-Keil

(code available)

https://xinjie-liu.github.io/mfpg-rl/

*Indicates equal contribution