Reinforcement Learning Algorithms

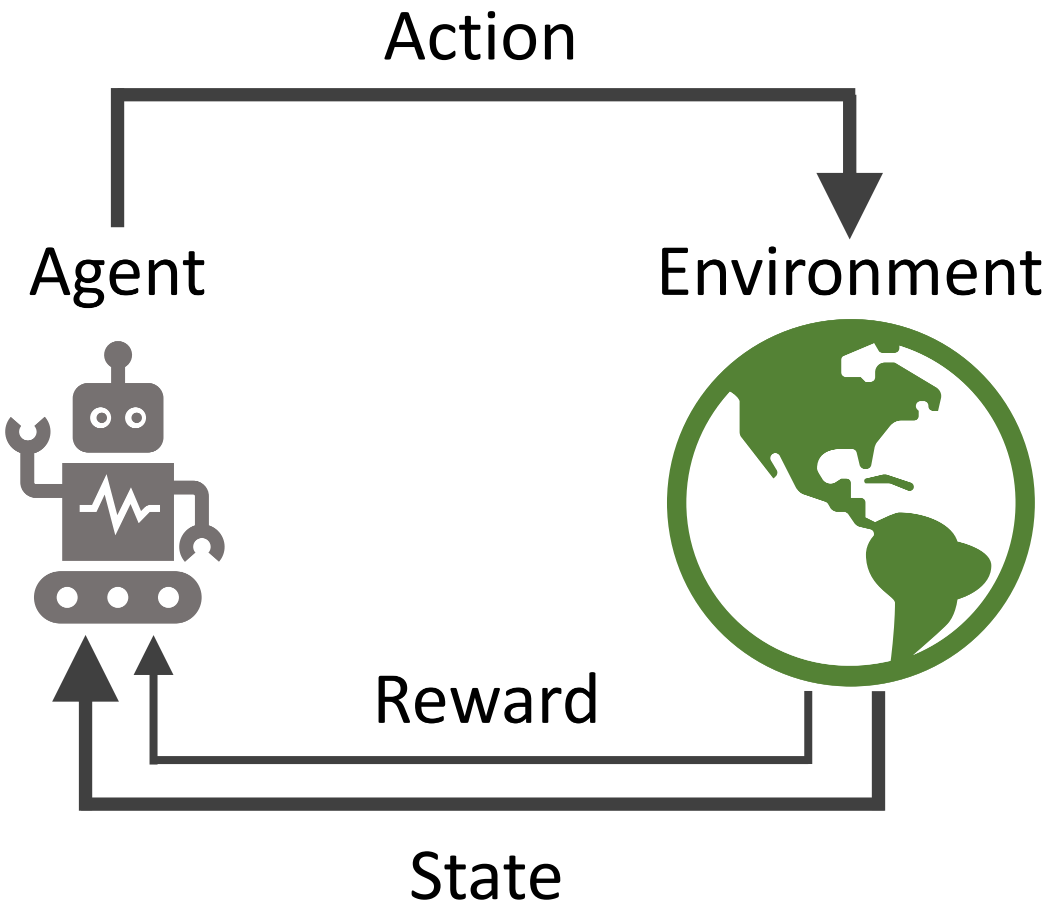
CMPT 729 G100

Jason Peng

Overview

- Anatomy of an RL algorithm
- Algorithm Characteristics
- Applications

Reinforcement Learning

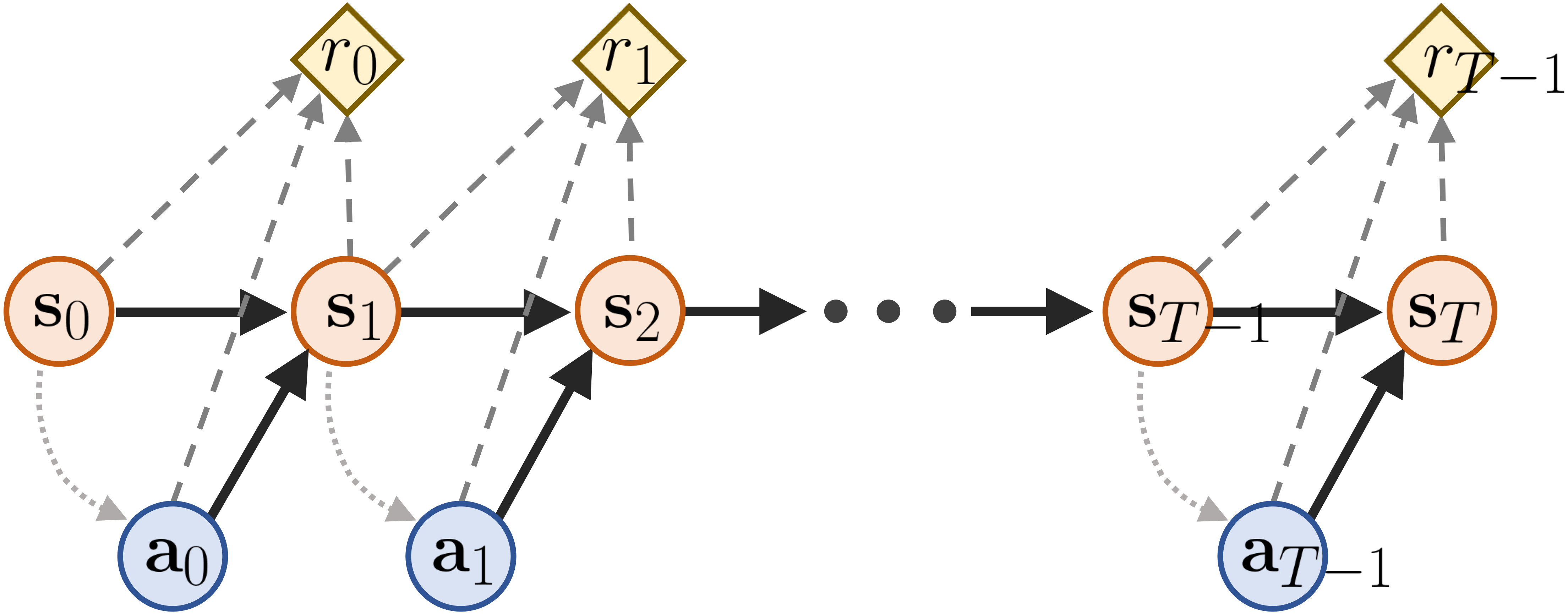


Black Box Optimization

$$\theta^* = \arg\max_{\theta} \; \underbrace{J(\pi_\theta)}_{\text{black box}}$$

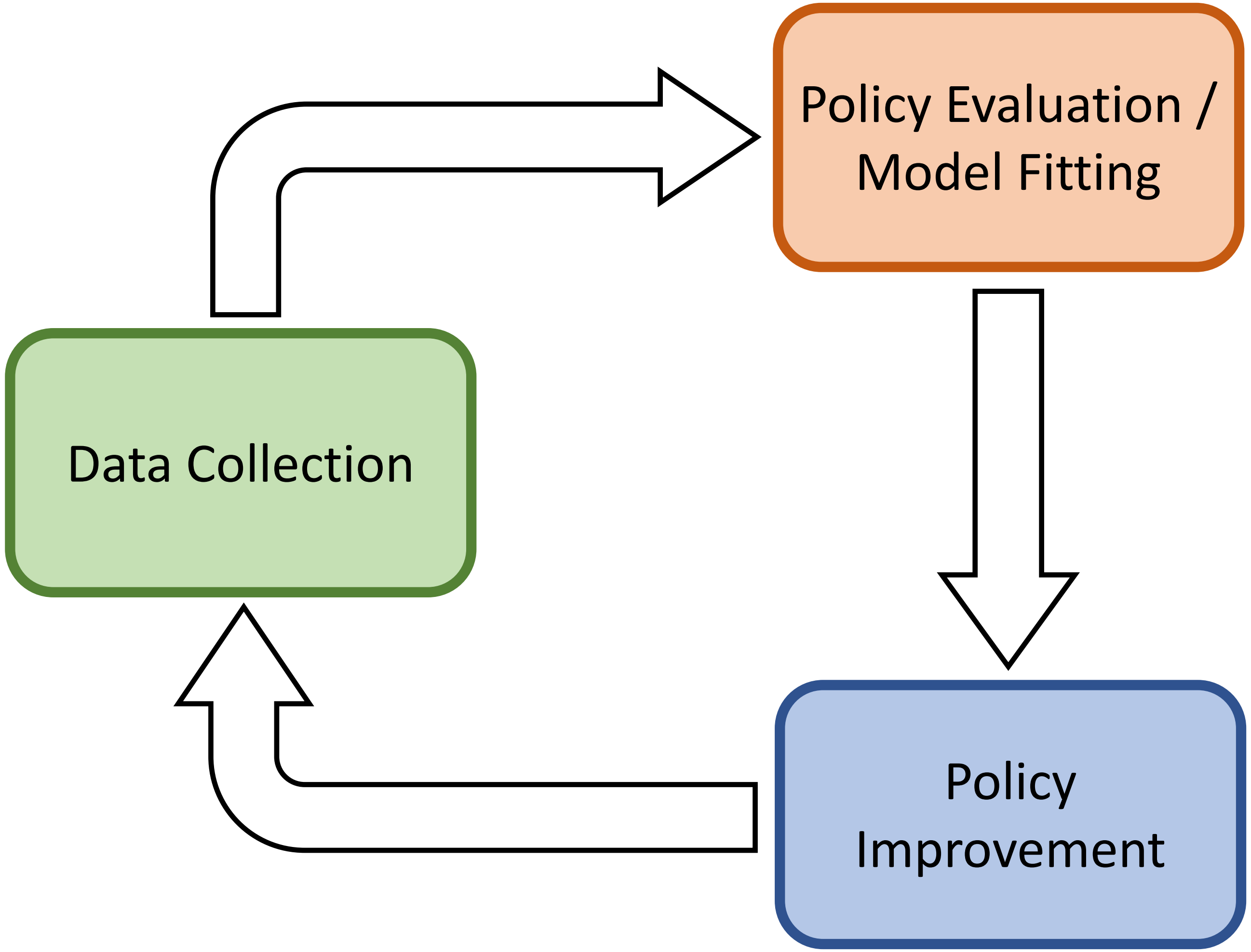
$$\theta \Rightarrow \boxed{J} \Rightarrow J(\pi_\theta)$$

MDP

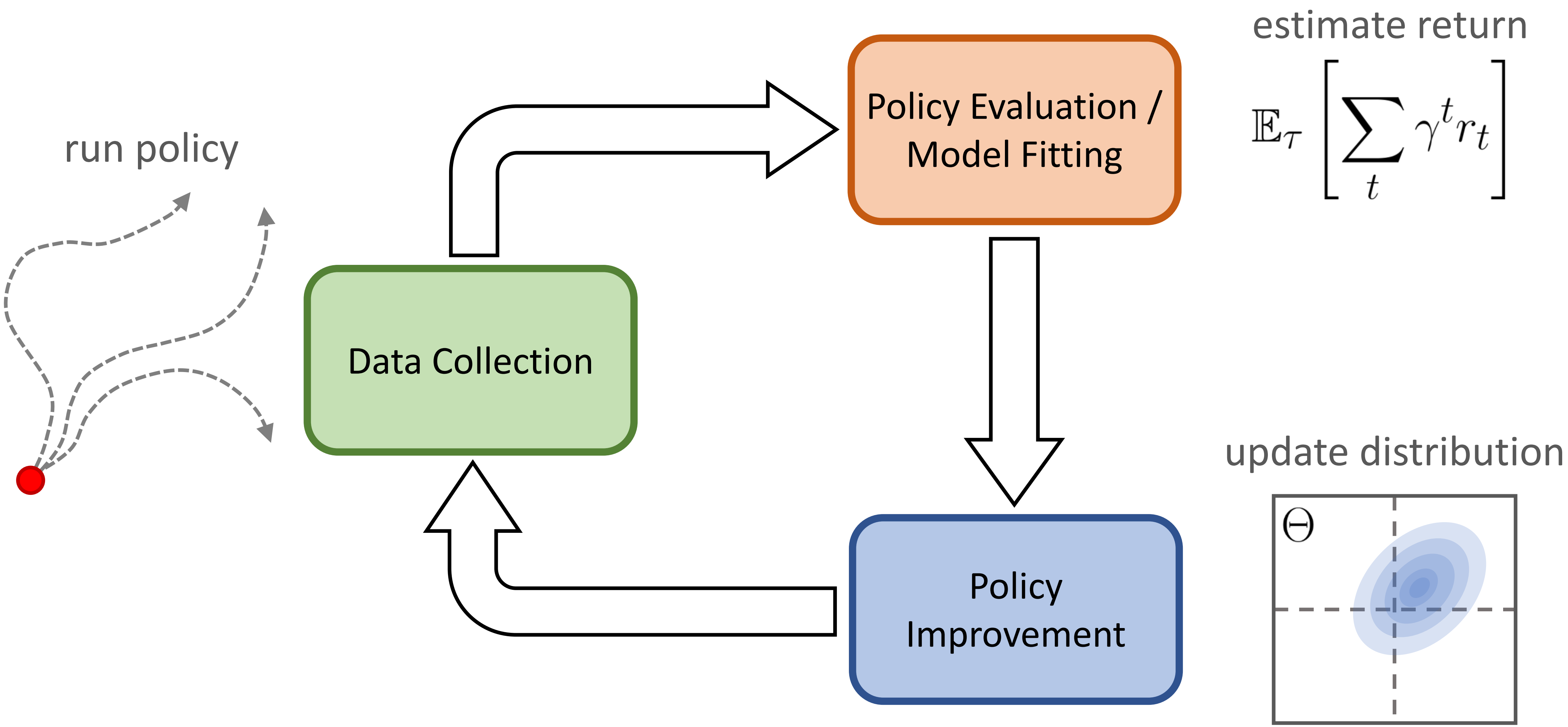


r_0
r_1
r_{T-1}
$\mathbf{s}_0$
$\mathbf{s}_1$
$\mathbf{s}_2$
$\mathbf{s}_{T-1}$
$\mathbf{s}_T$
$\mathbf{a}_0$
$\mathbf{a}_1$
$\mathbf{a}_{T-1}$

Anatomy of an RL Algorithm



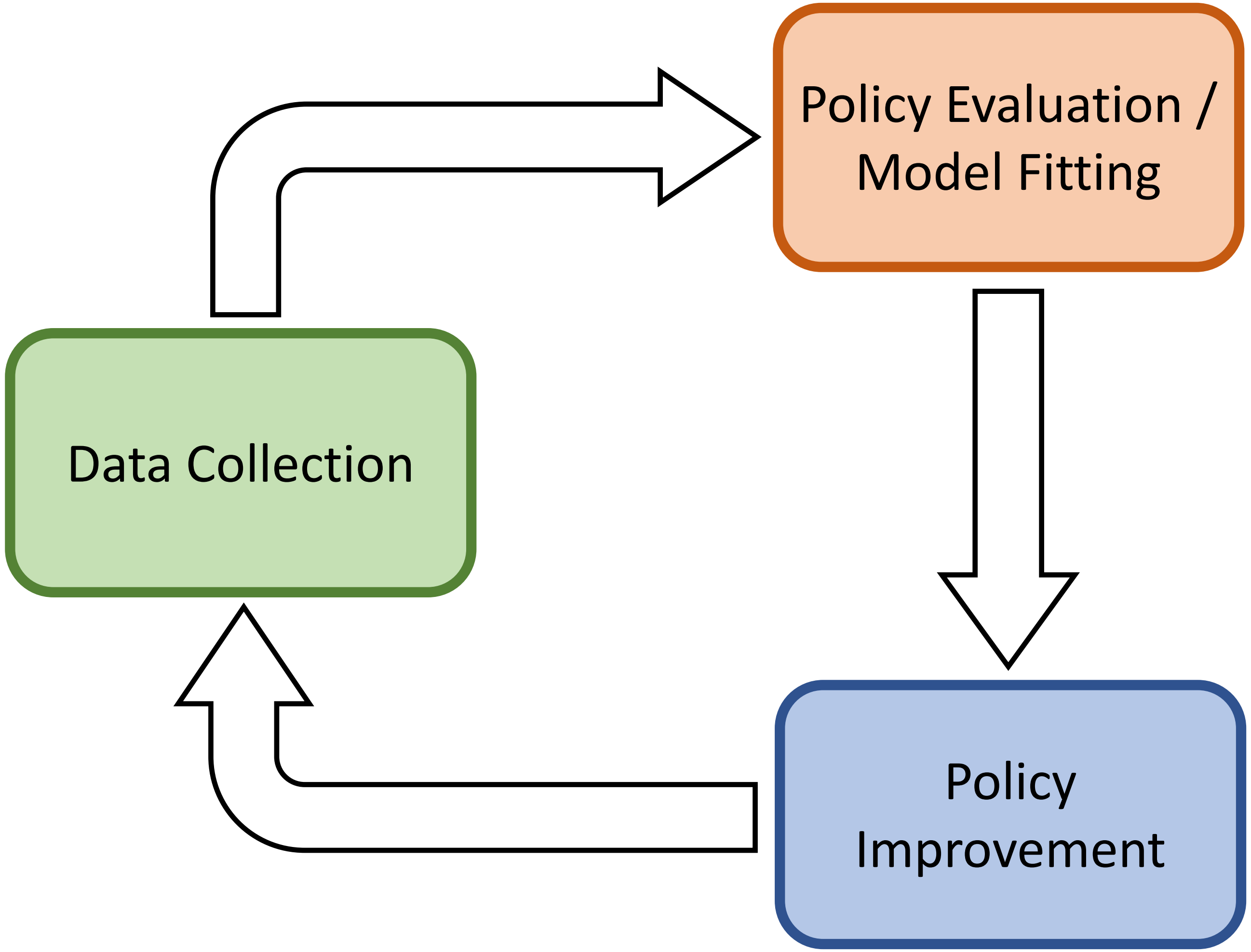
Evolutionary Methods



Taxonomy of RL Algorithms

- Policy-Based
- Value-Based
- Actor-Critic
- Model-Based

Anatomy of an RL Algorithm

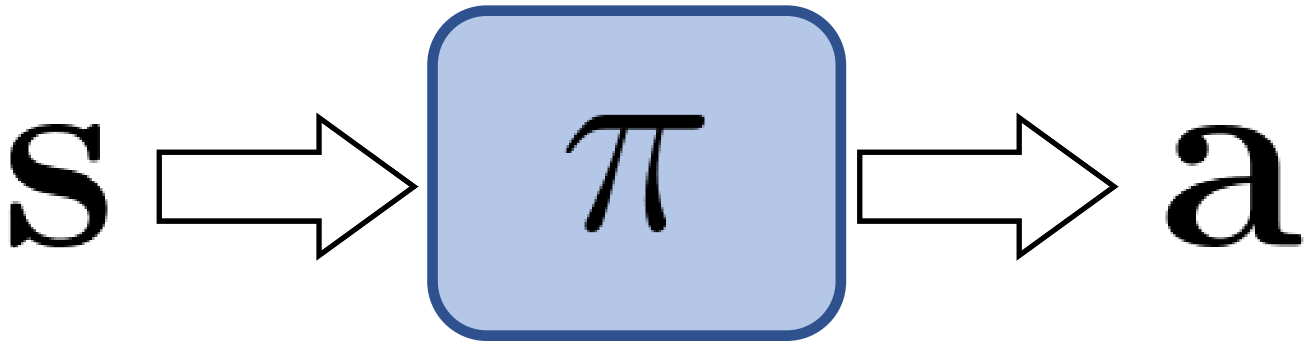


Taxonomy of RL Algorithms

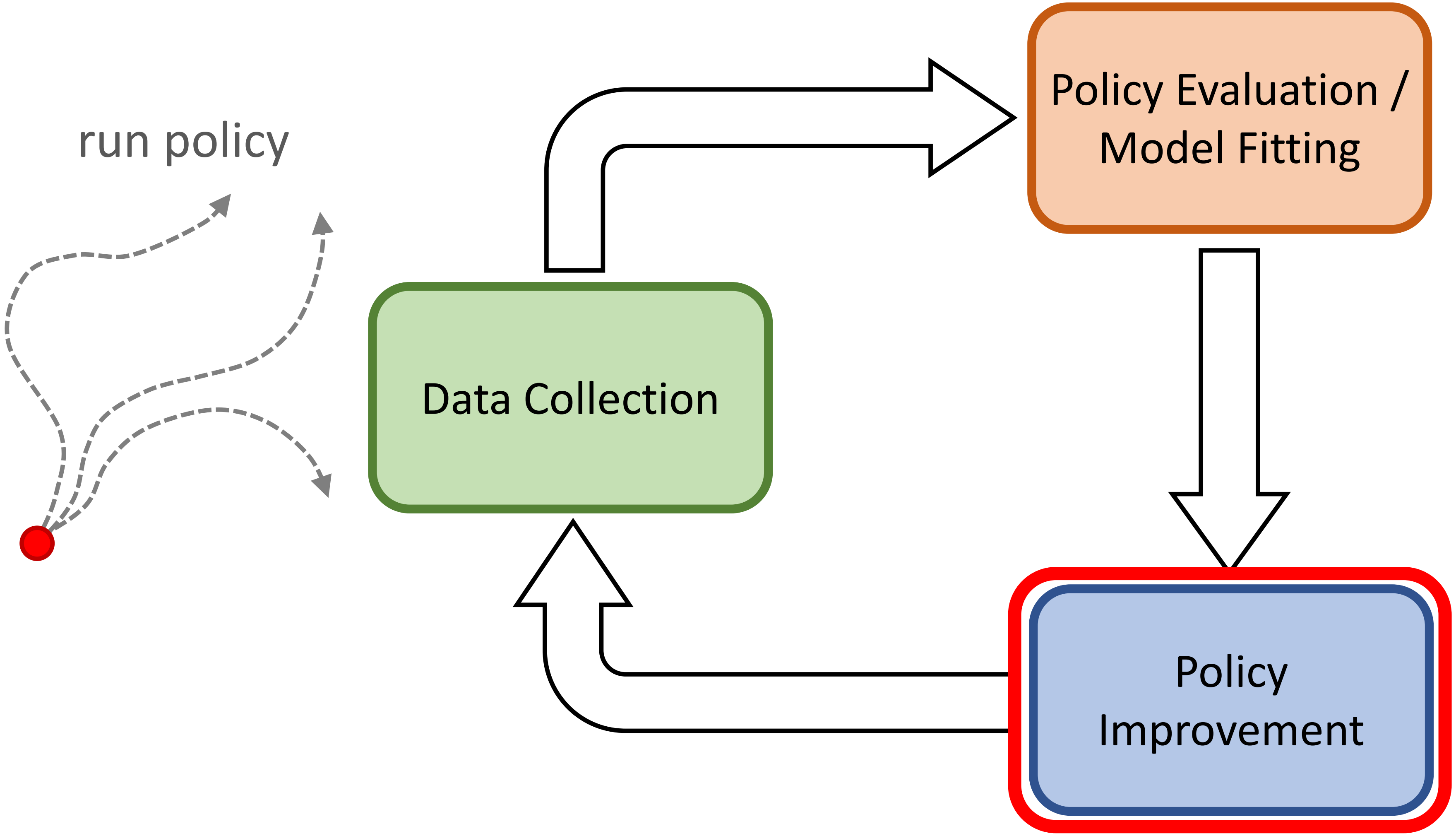
- Policy-Based
- Value-Based
- Actor-Critic
- Model-Based

Policy-Based Methods

$$\pi(\mathbf{a}|\mathbf{s})$$



Policy-Based Methods



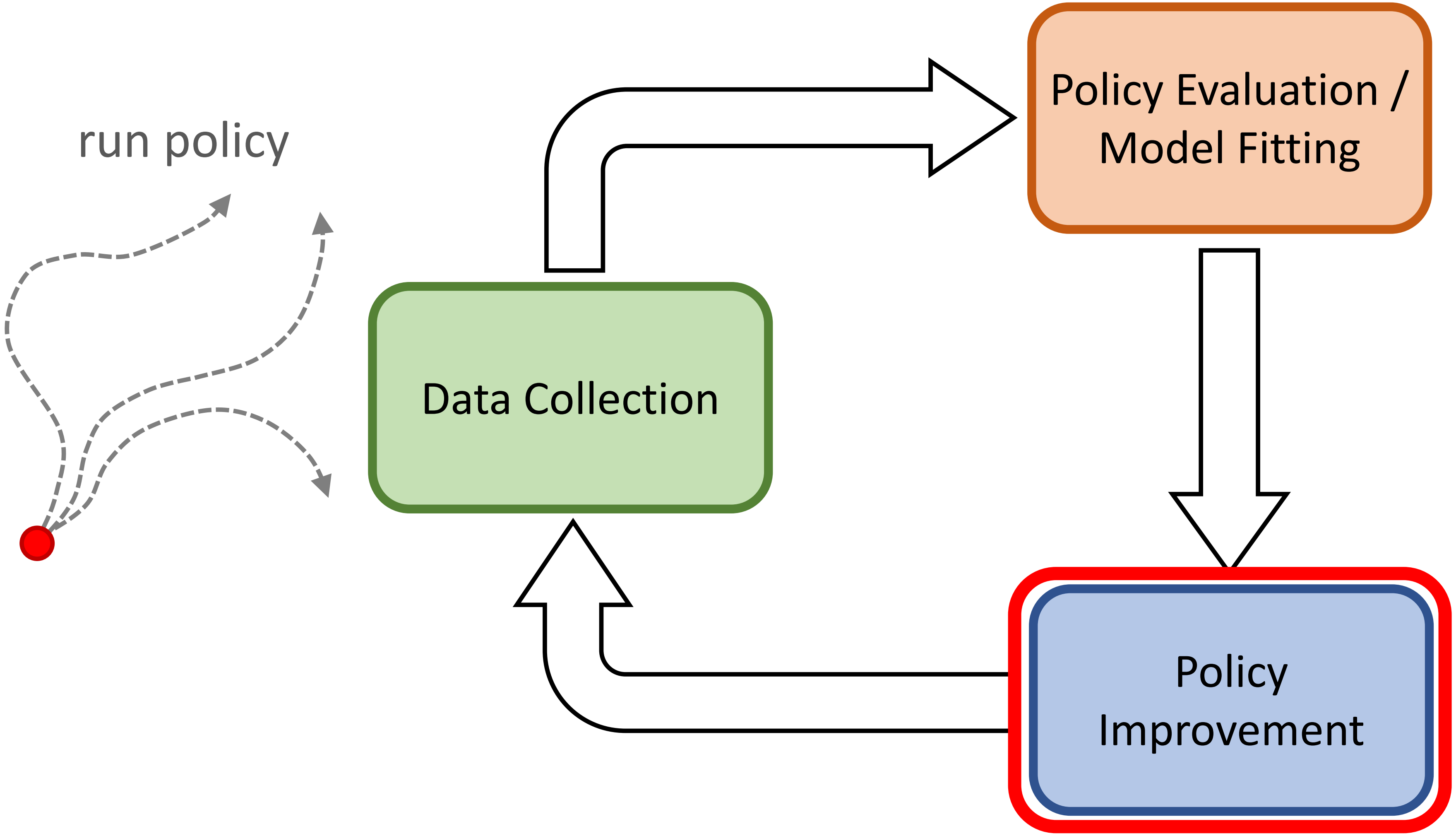
estimate return

$$J(\pi) = \mathbb{E}_{\tau \sim \pi}\left[\sum_t \gamma^t r_t\right]$$

calculate gradient

$$\nabla_\pi J(\pi)$$

Policy-Based Methods



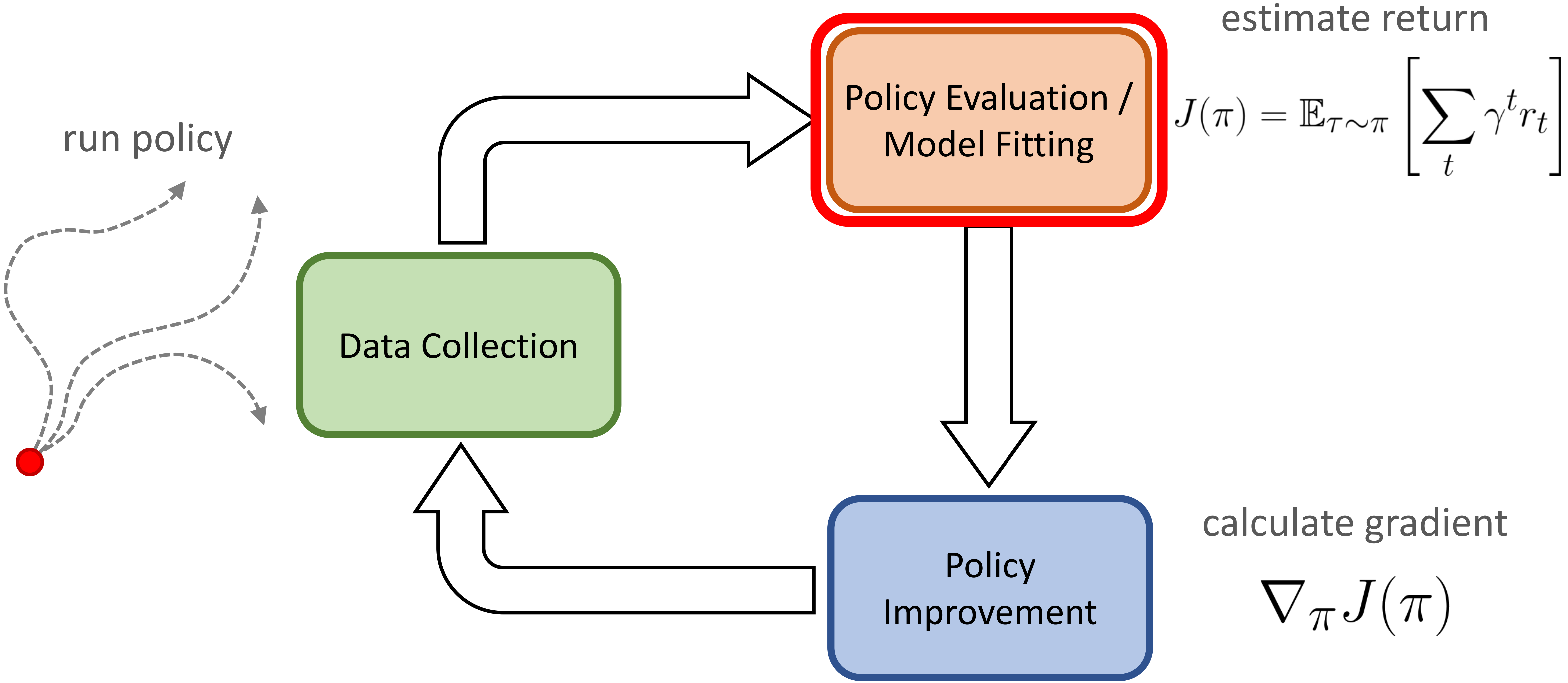
estimate return

$$J(\pi) = \mathbb{E}_{\tau\sim\pi}\left[\sum_t \gamma^t r_t\right]$$

calculate gradient

$$\nabla_\pi J(\pi)$$

Policy-Based Methods

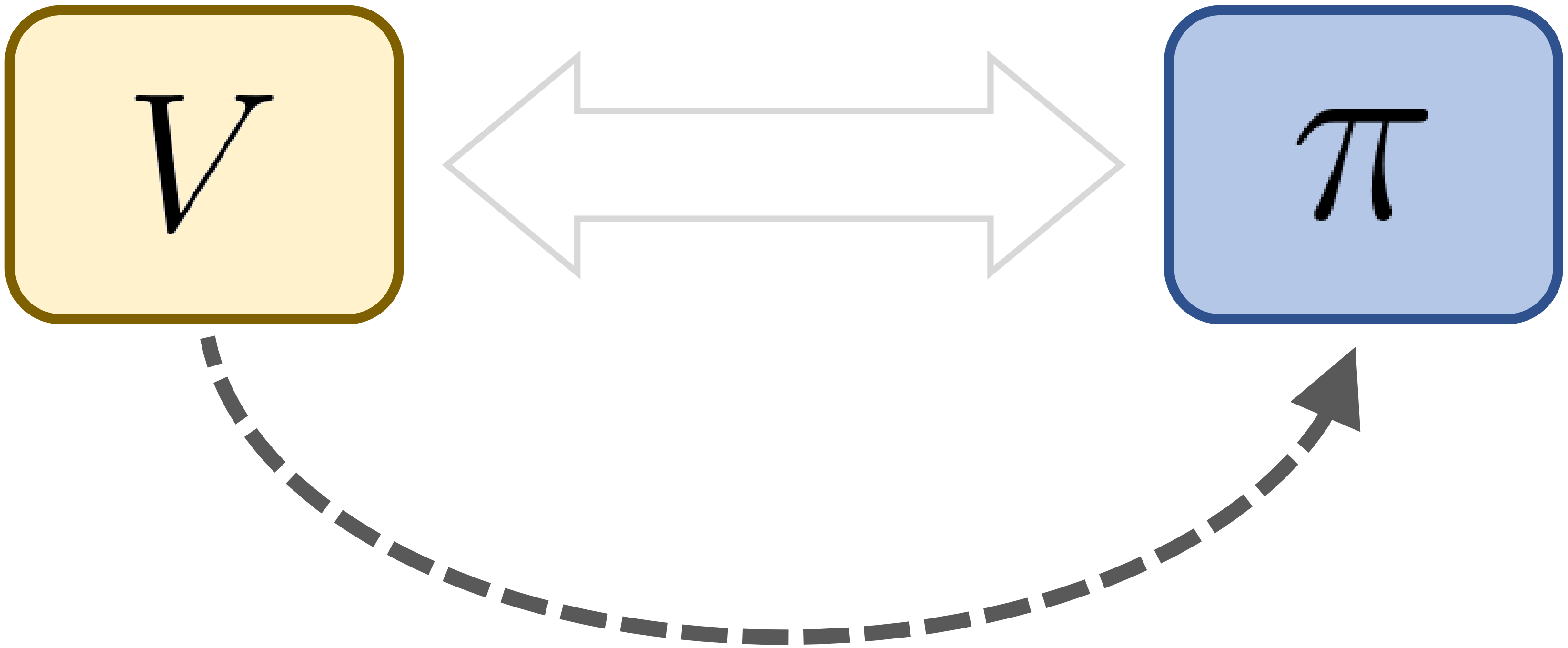


Taxonomy of RL Algorithms

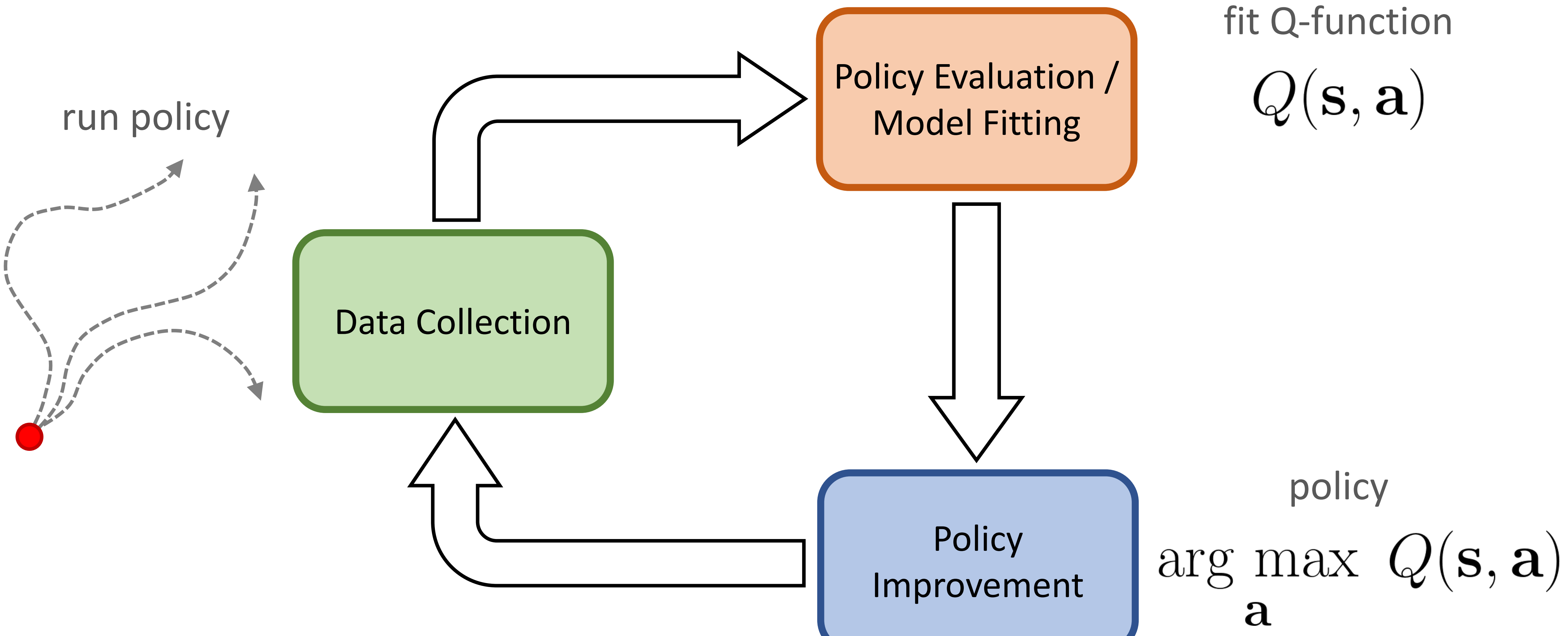
- Policy-Based
- Value-Based
- Actor-Critic
- Model-Based

Value-Based Methods

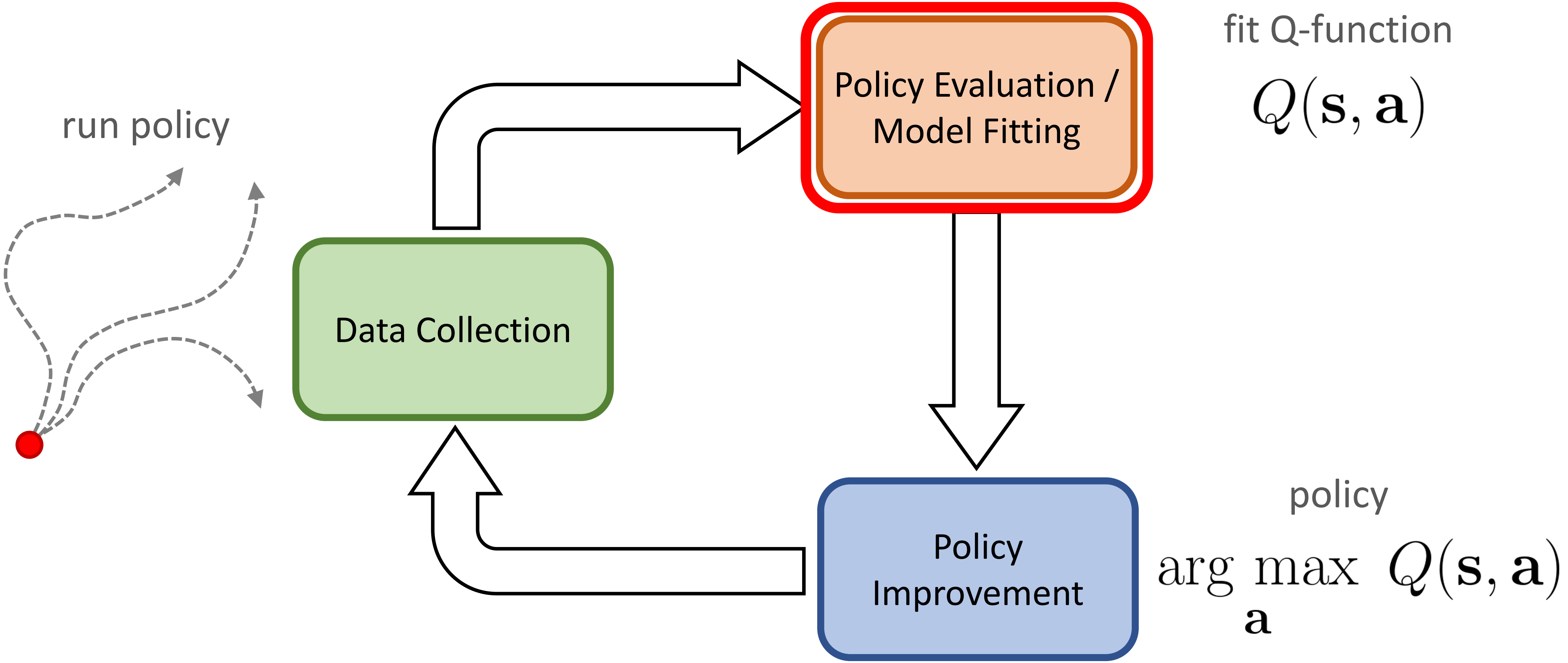
Value-Based Methods



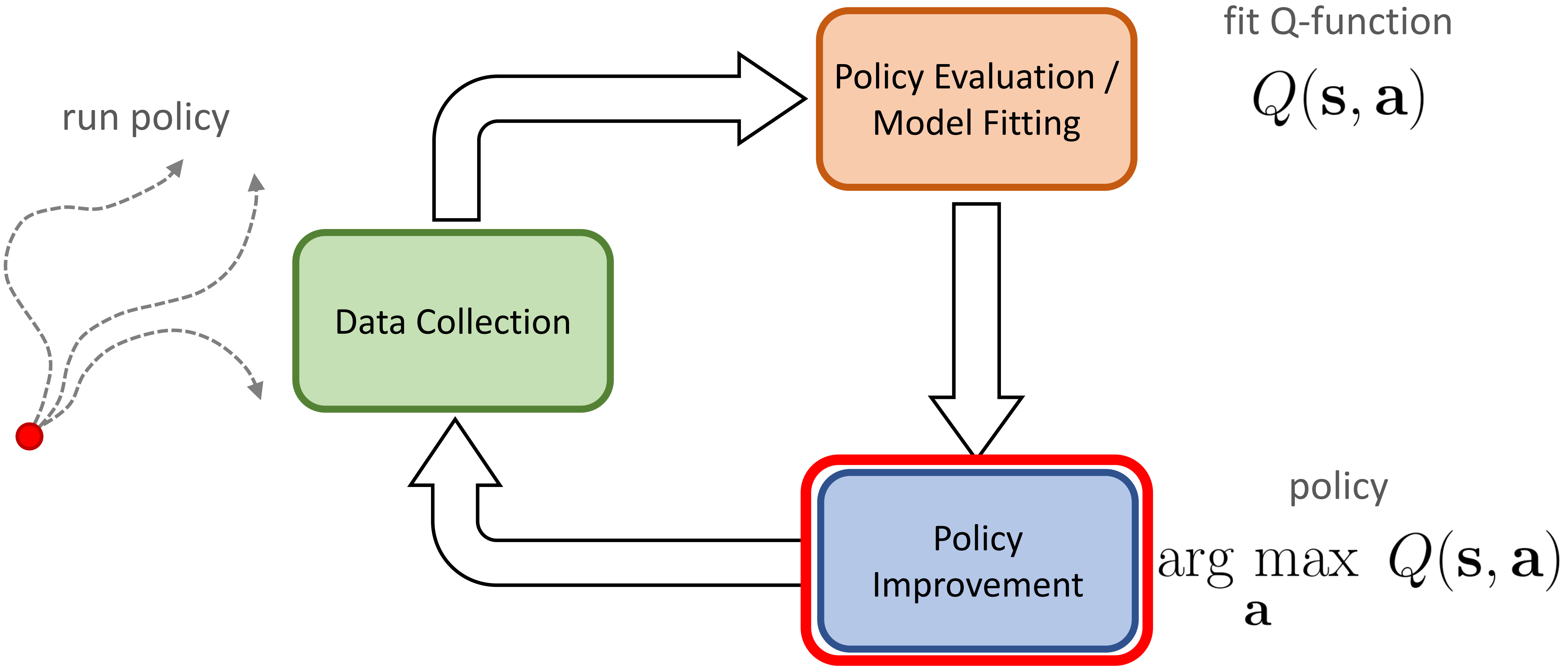
Value-Based Methods



Value-Based Methods



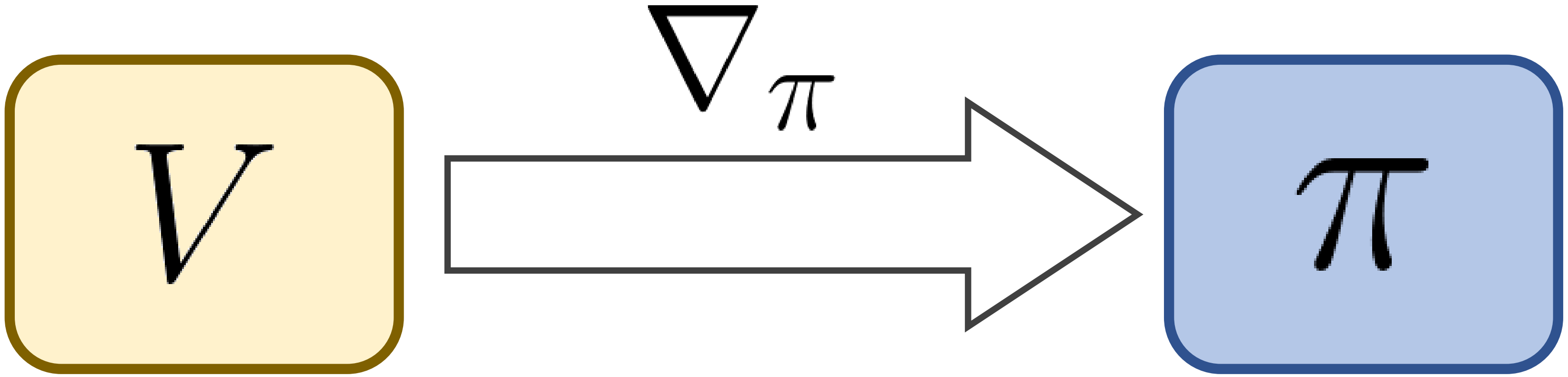
Value-Based Methods



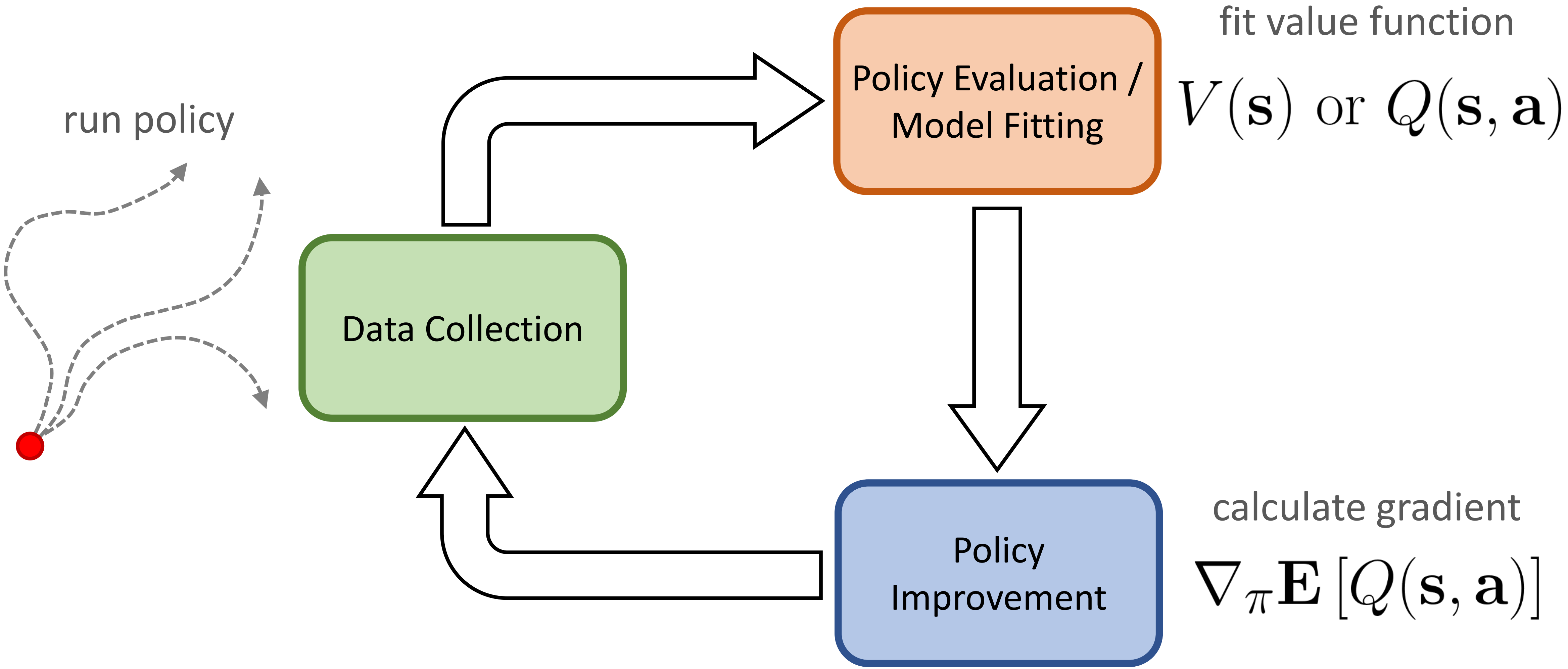
Taxonomy of RL Algorithms

- Policy-Based
- Value-Based
- Actor-Critic
- Model-Based

Actor-Critic Methods



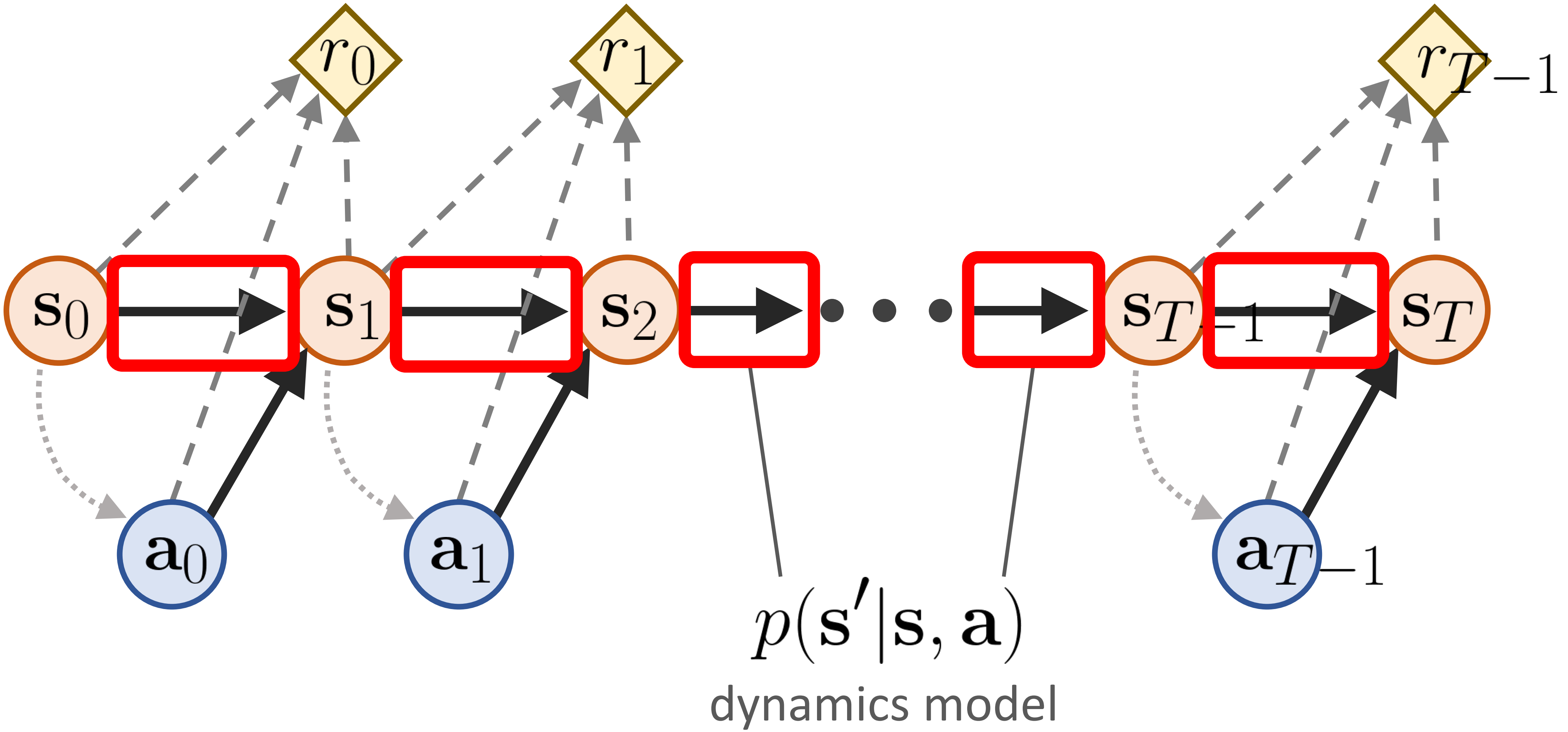
Actor-Critic Methods



Taxonomy of RL Algorithms

- Policy-Based
- Value-Based
- Actor-Critic
- Model-Based

Model-Based Method



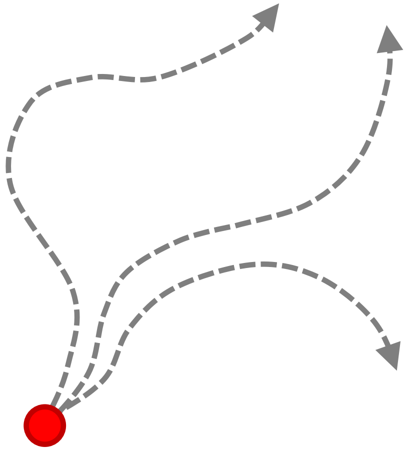
Learning a Simulator

Simulation

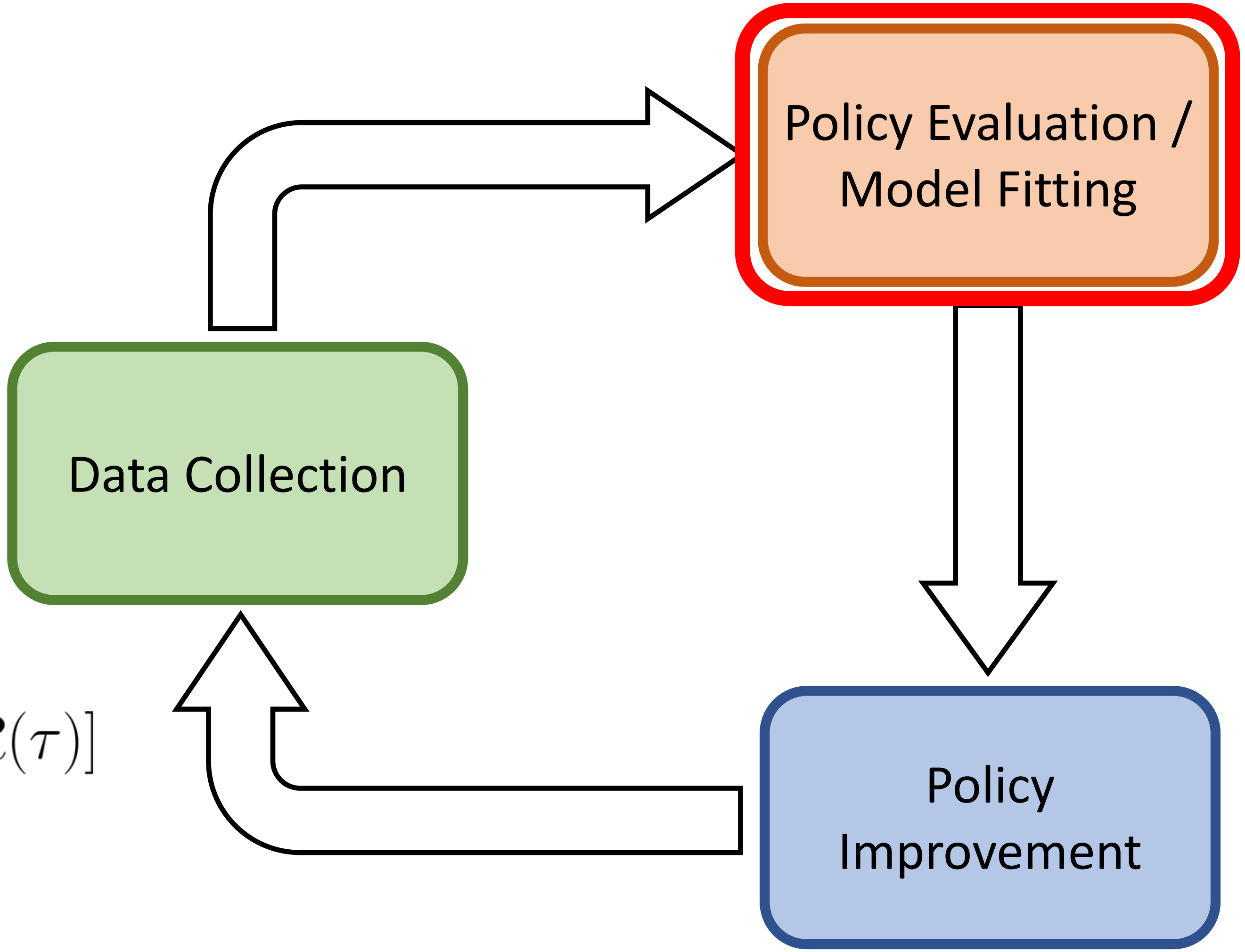
Real World

Model-Based Method

online planning



$$\arg\max_{\mathbf{a}} \; \mathbb{E}_{\tau \sim f(\tau)} [R(\tau)]$$



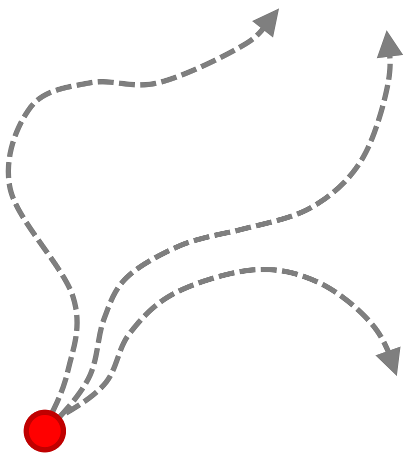
fit dynamics model

$$f(\mathbf{s}'|\mathbf{s}, \mathbf{a})$$

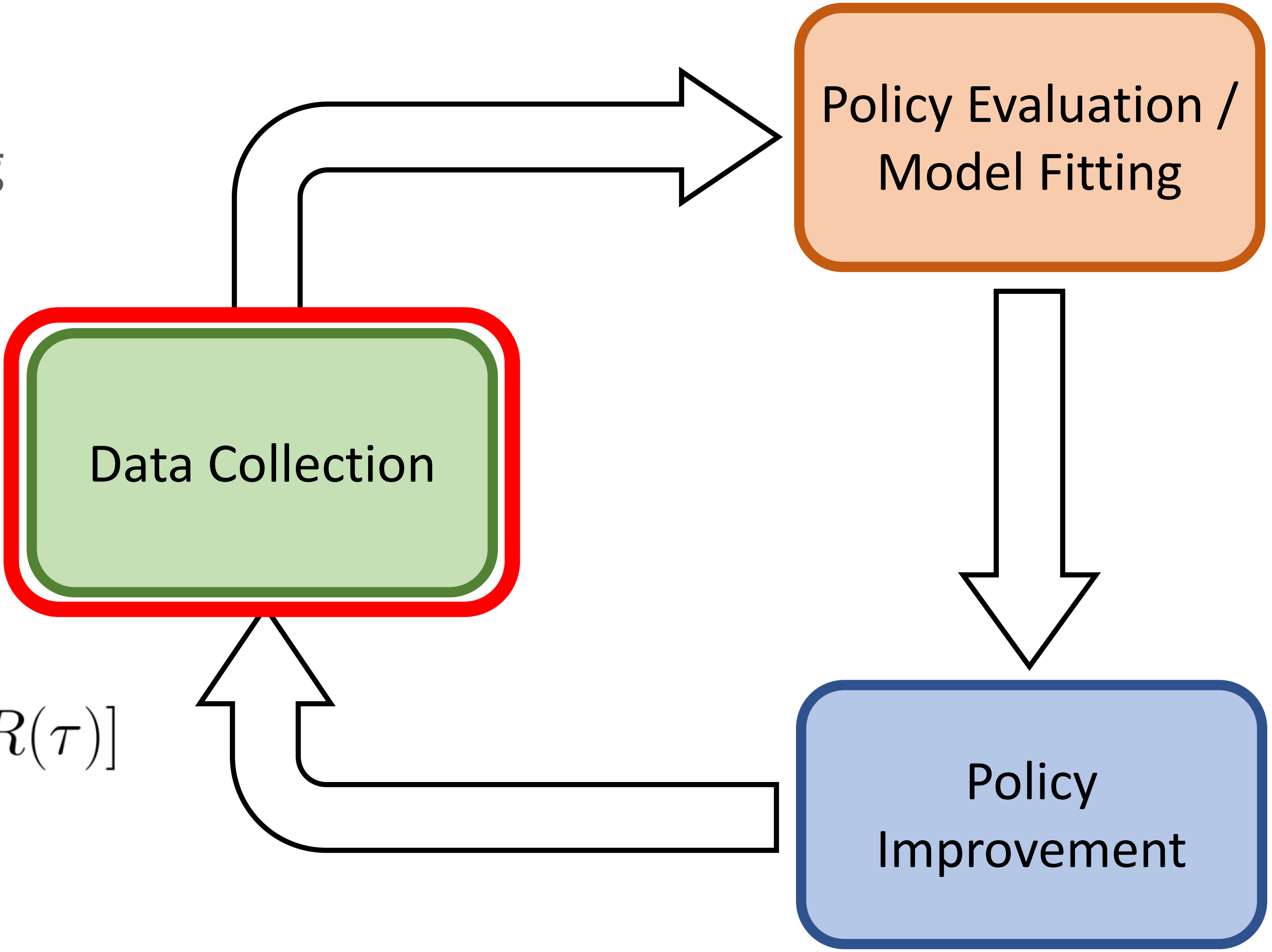
do nothing

Model-Based Method

online planning



$$\arg\max_{\mathbf{a}} \; \mathbb{E}_{\tau \sim f(\tau)} [R(\tau)]$$

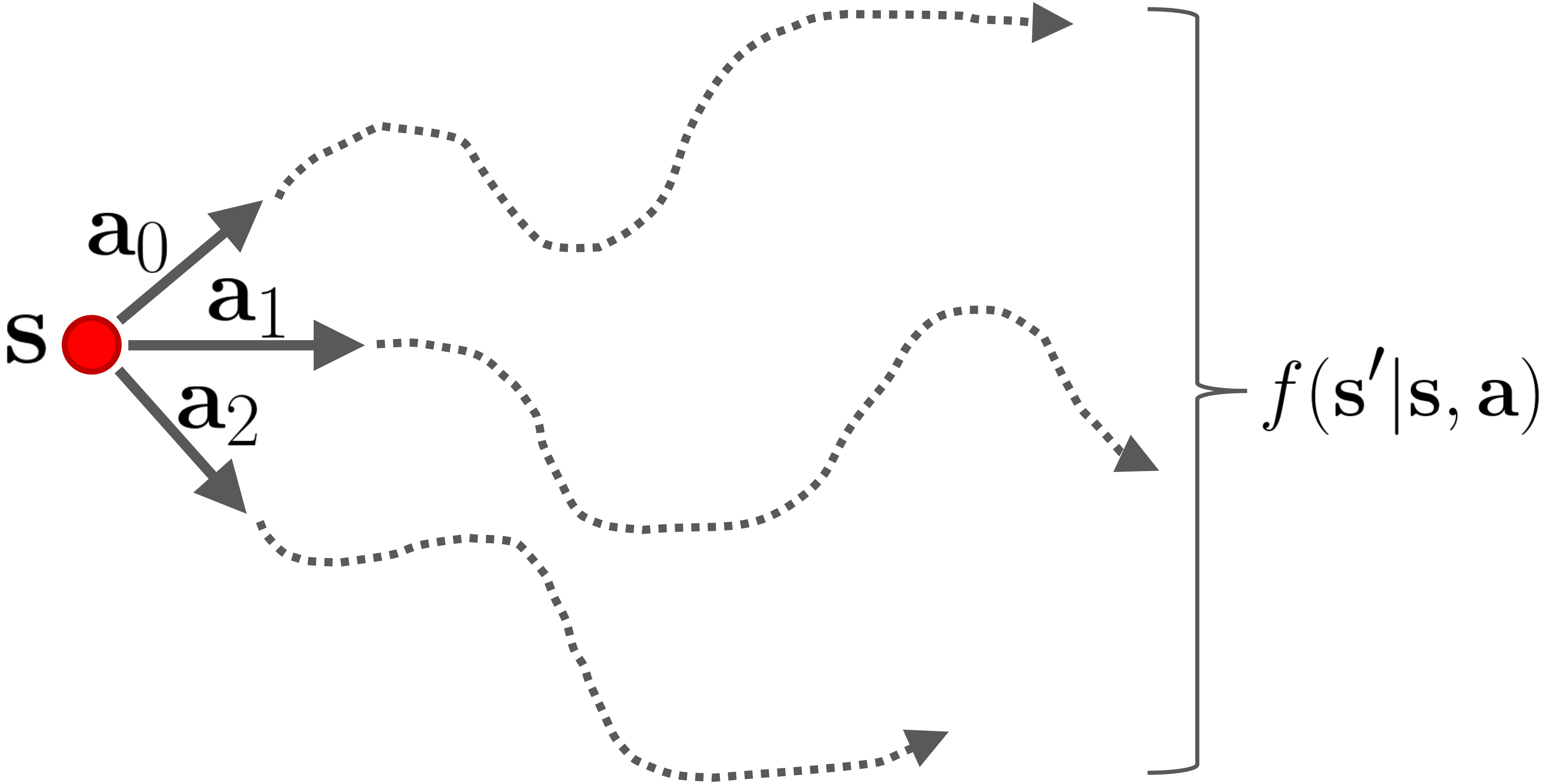


fit dynamics model

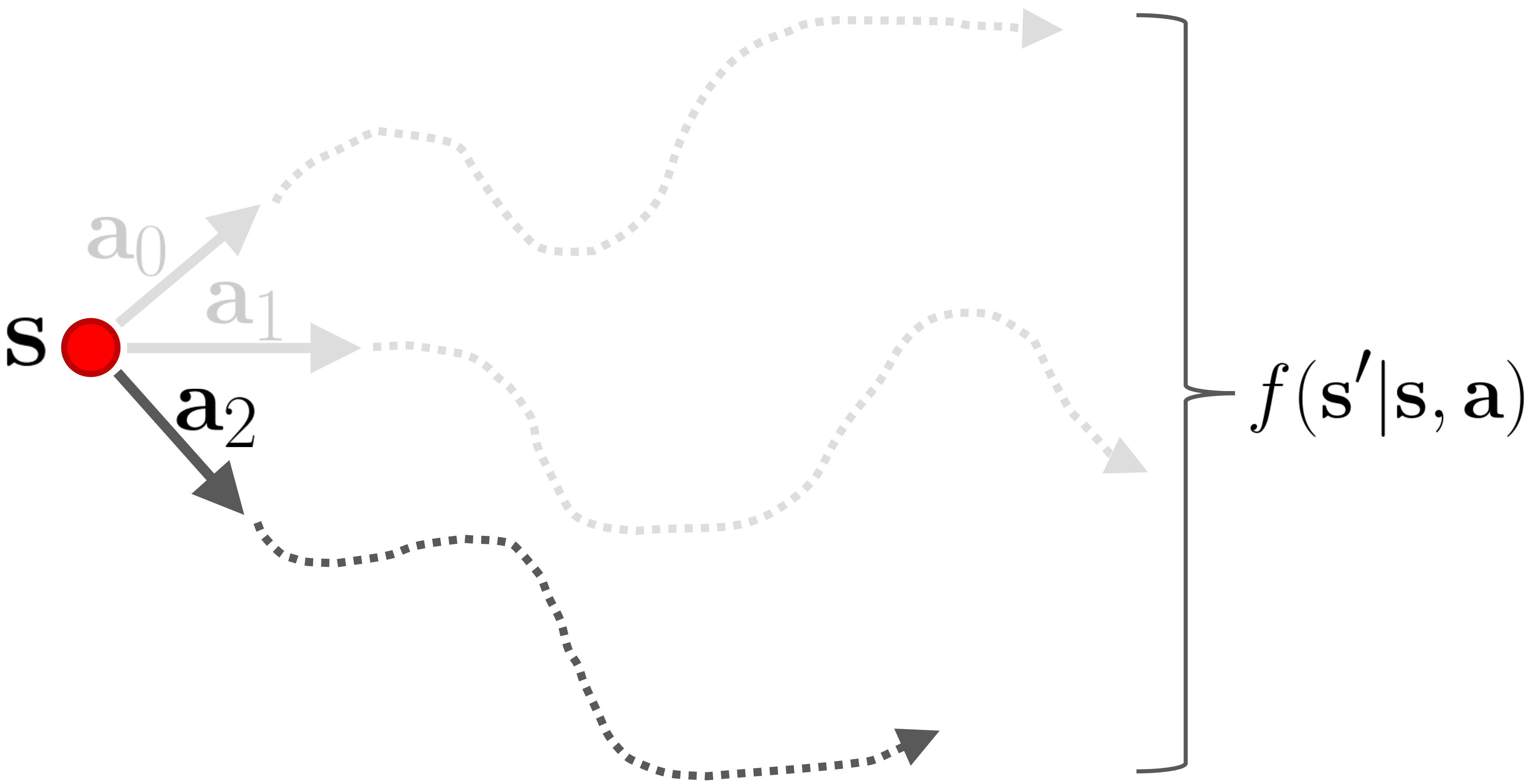
$$f(\mathbf{s}'|\mathbf{s}, \mathbf{a})$$

do nothing

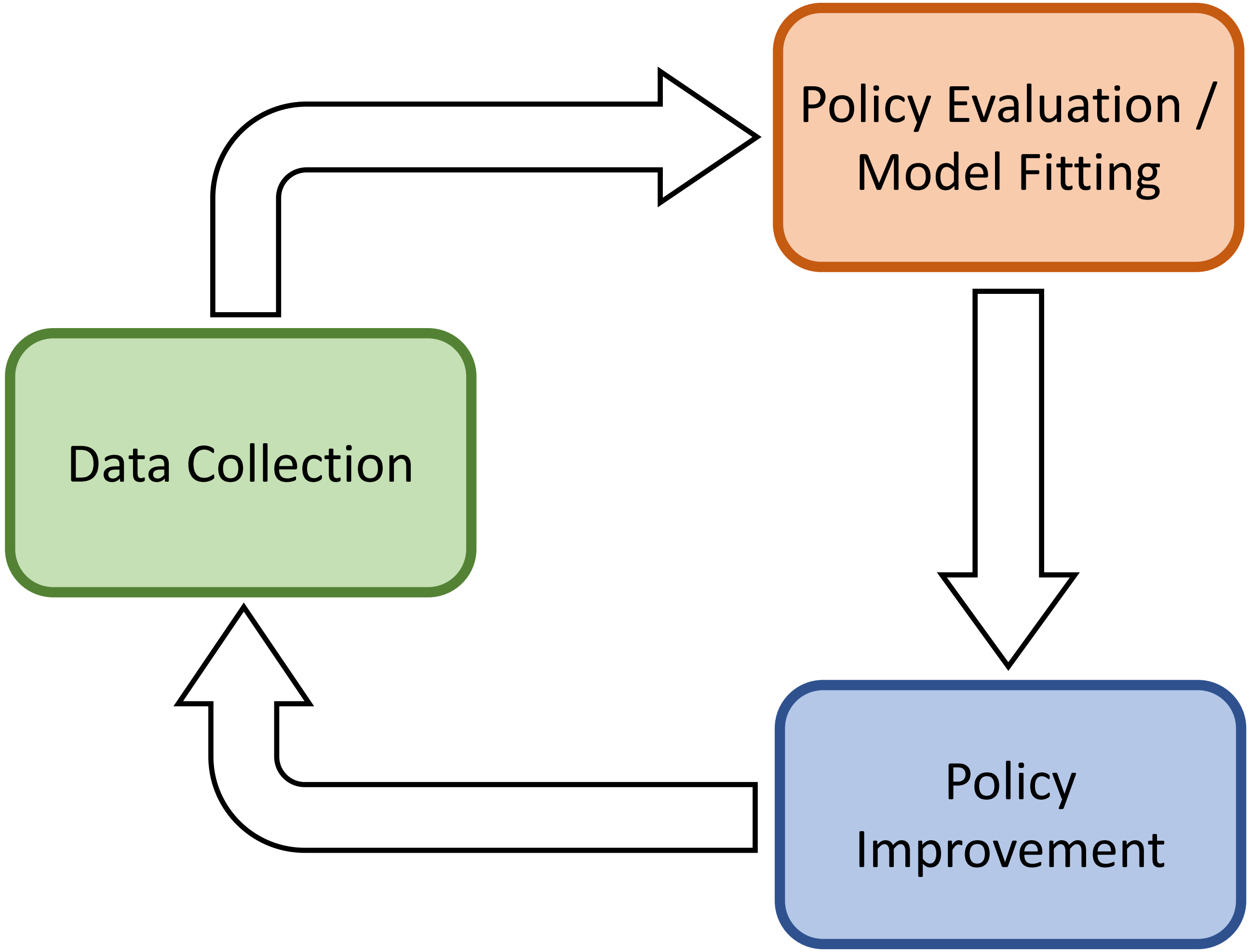
Model-Based Planning



Model-Based Planning

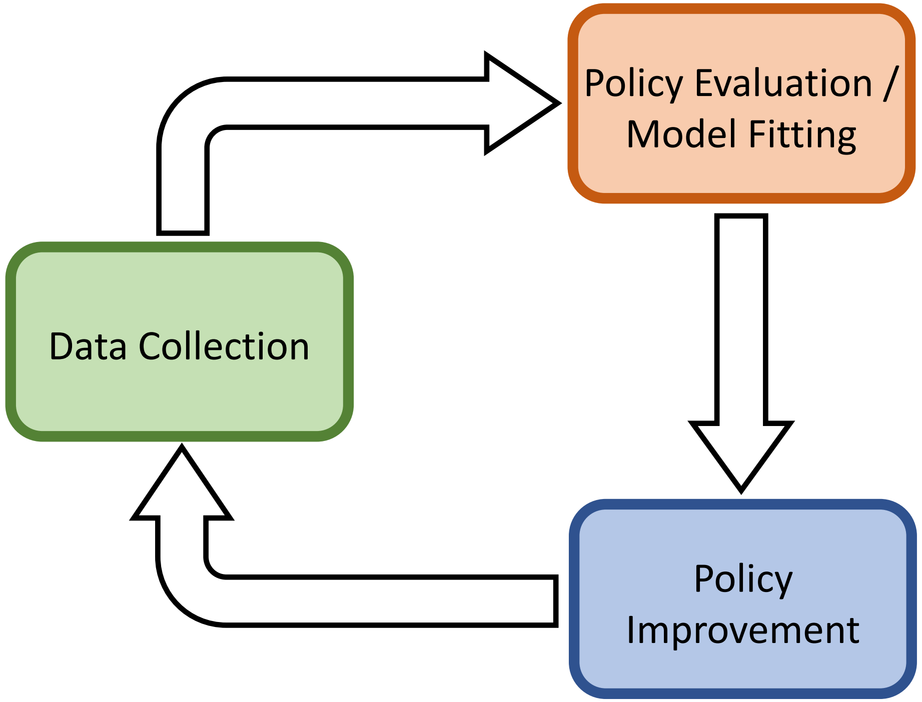


Anatomy of an RL Algorithm



Characteristics

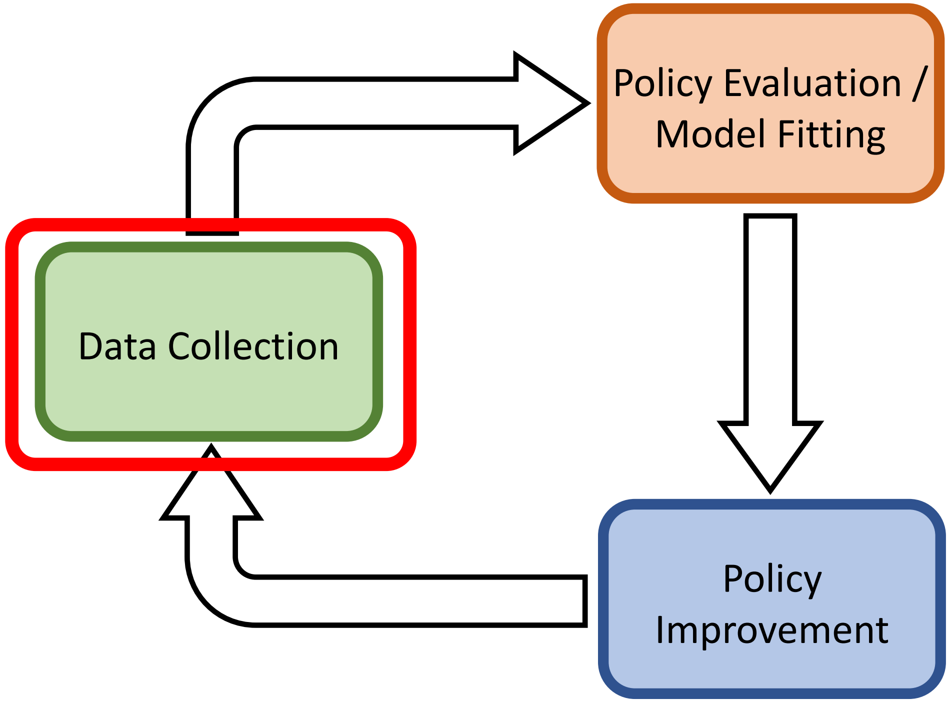
- Sample efficiency
- Wall-clock time
- Performance and stability
- Stochastic/deterministic dynamics
- Continuous/discrete actions
- Modeling challenges



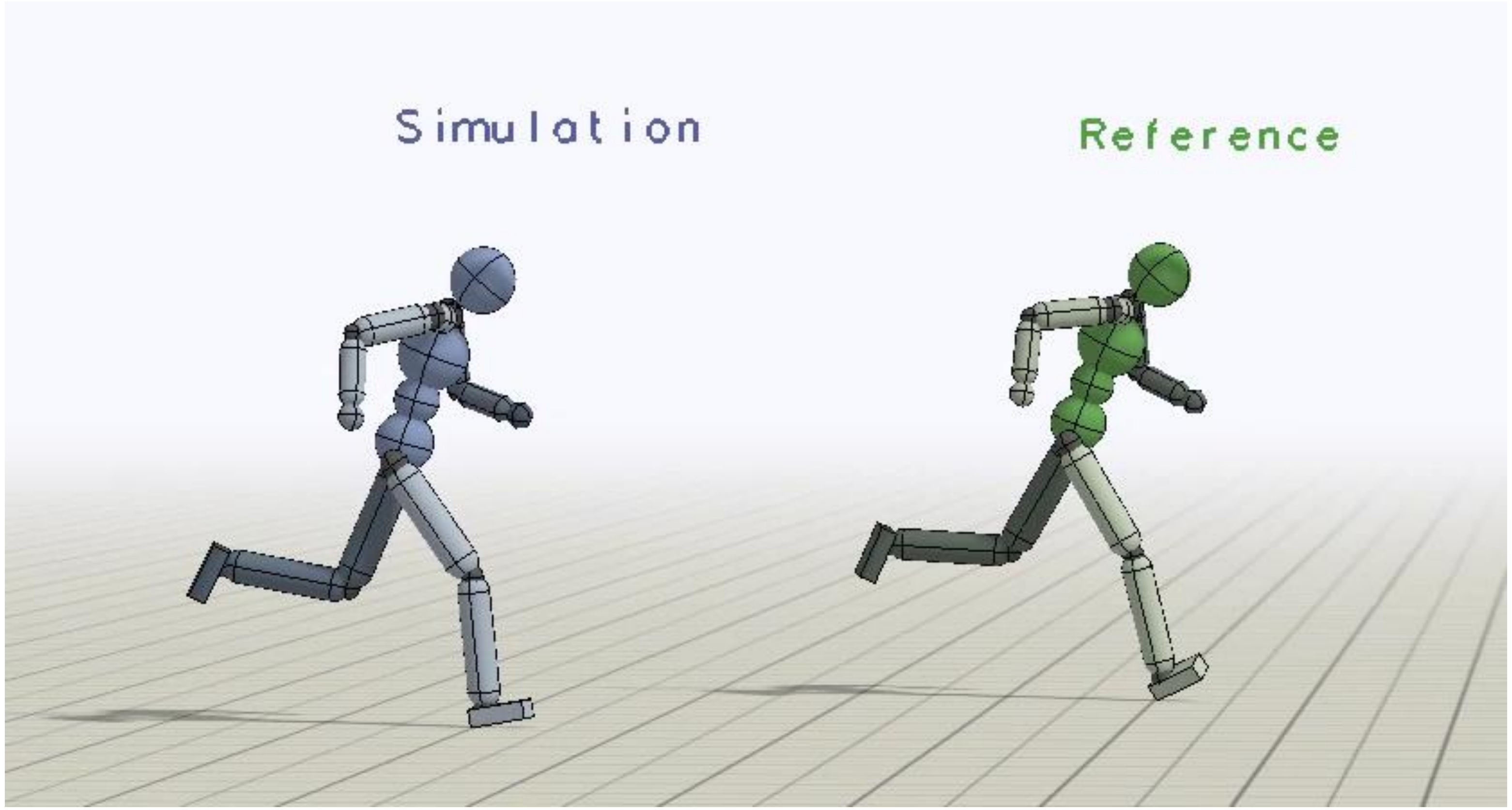
Sample Efficiency

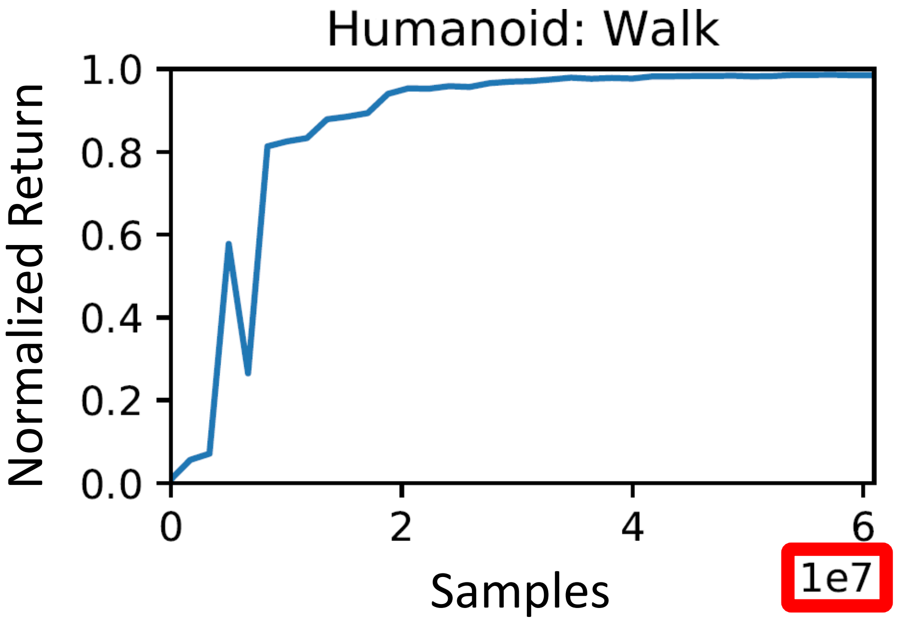
- Sample efficiency = how much data needed to get a good policy

- On-policy vs off-policy
  - Can the algorithm use data from other policies or demonstrations?
  - Can algorithm reuse data from previous iterations?



Sample Efficiency





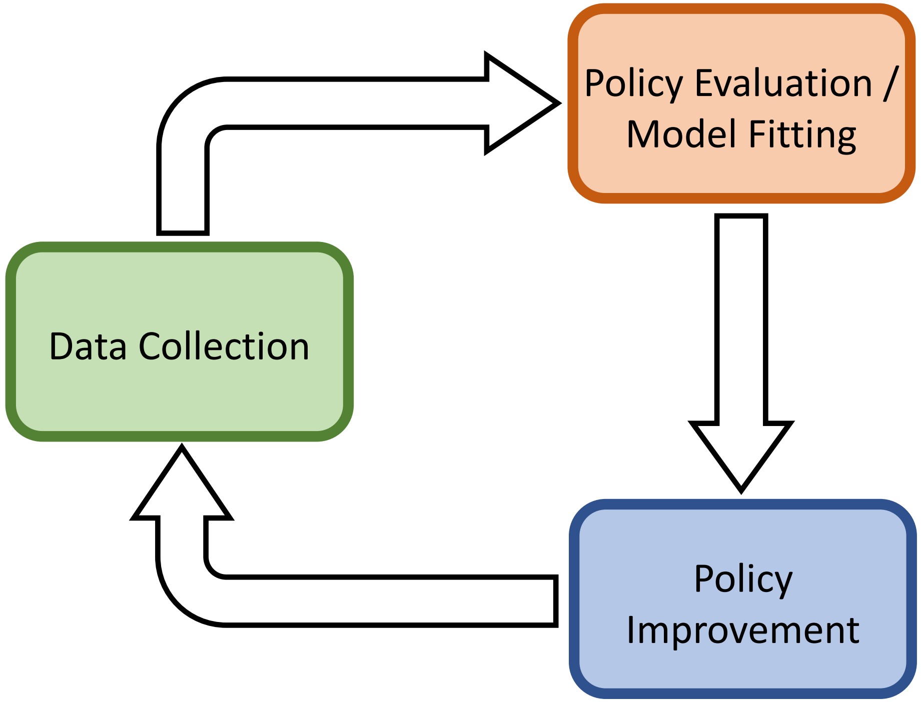
DeepMimic: Example-Guided Deep Reinforcement Learning of Physics-Based Character Skills
[Peng et al. 2018]

Sample Efficiency

DayDreamer: World Models for Physical Robot Learning
[Wu et al. 2022]

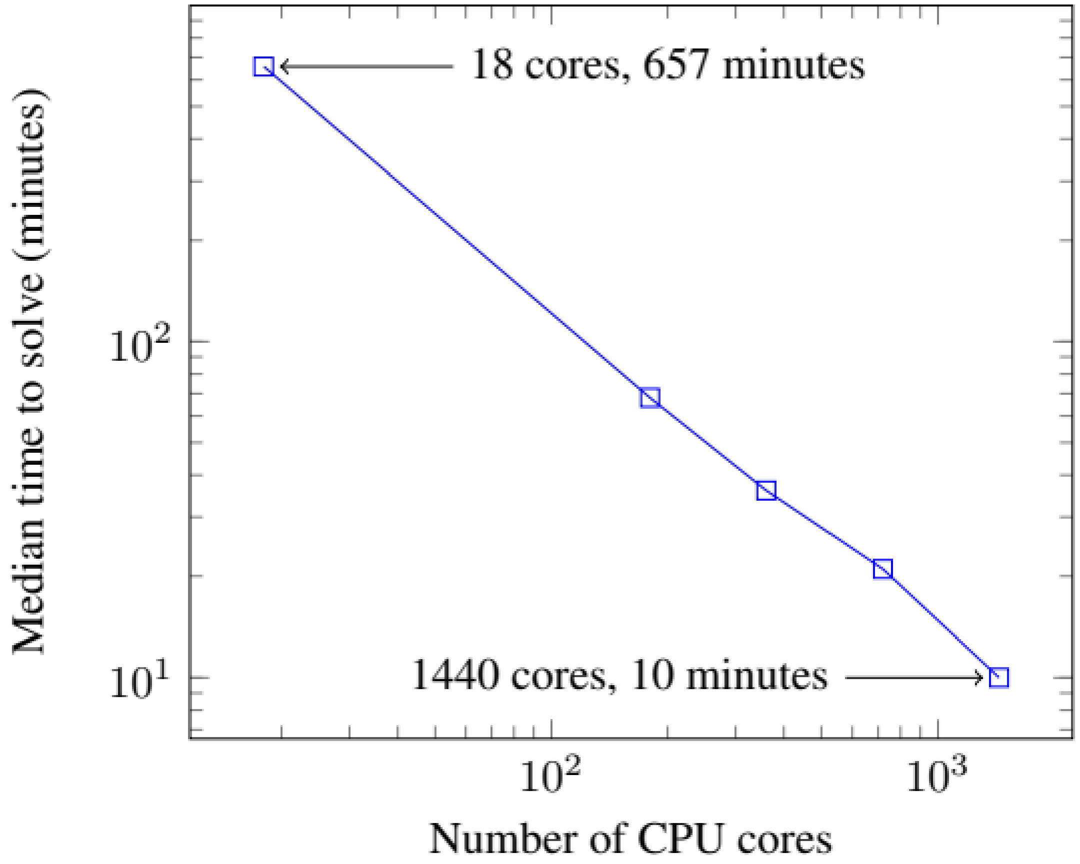
Characteristics

- Sample efficiency
- Wall-clock time
- Performance and stability
- Stochastic/deterministic dynamics
- Continuous/discrete actions
- Modeling challenges



Wall-Clock Time

- How much compute?
- How parallelizable?
- sample efficiency ≠ wall-clock time



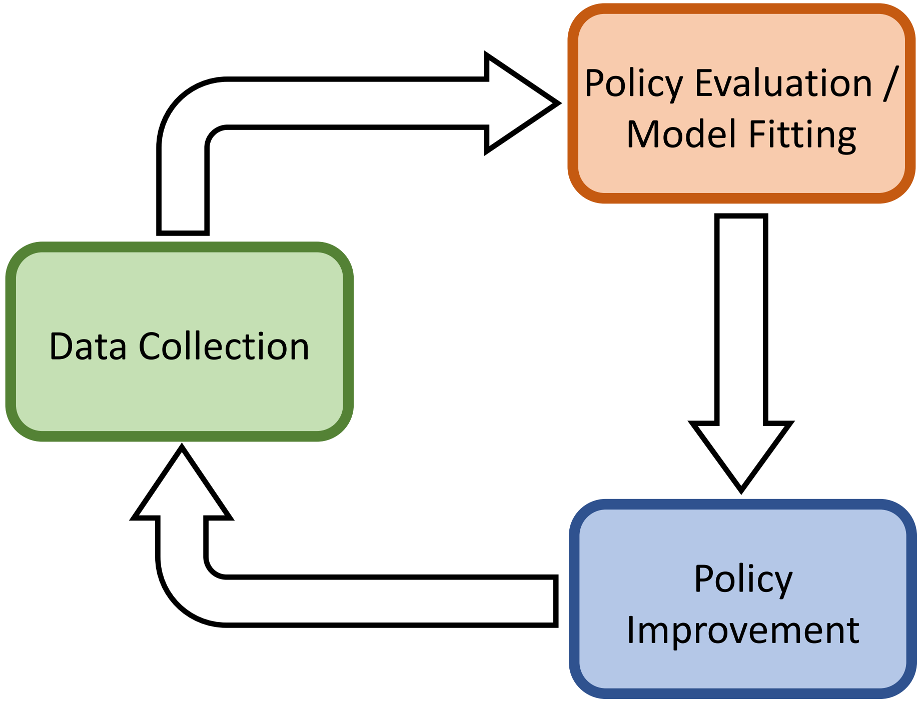
Evolution Strategies as a Scalable Alternative to Reinforcement Learning [Salimans et al. 2017]

Wall-Clock Time

Human-Level Control Through Deep Reinforcement Learning
[Mnih et al. 2015]

Characteristics

- Sample efficiency
- Wall-clock time
- Performance and stability
- Stochastic/deterministic dynamics
- Continuous/discrete actions
- Modeling challenges



Performance and Stability

- Does it converge?
- What does it converge to?
- Does it converge every time?

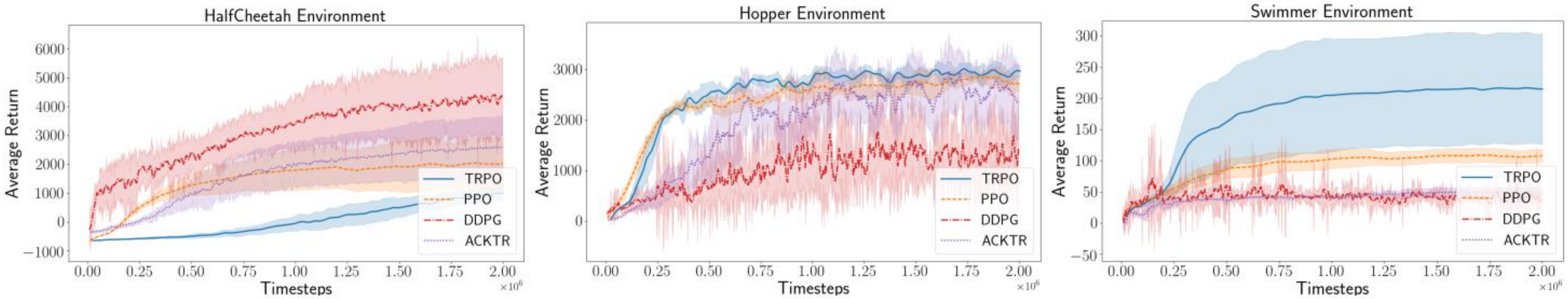


Figure 4: Performance of several policy gradient algorithms across benchmark MuJoCo environment suites

Deep Reinforcement Learning that Matters
[Henderson et al. 2018]

Performance and Stability

- Does it converge?
- What does it converge to?
- Does it converge every time?

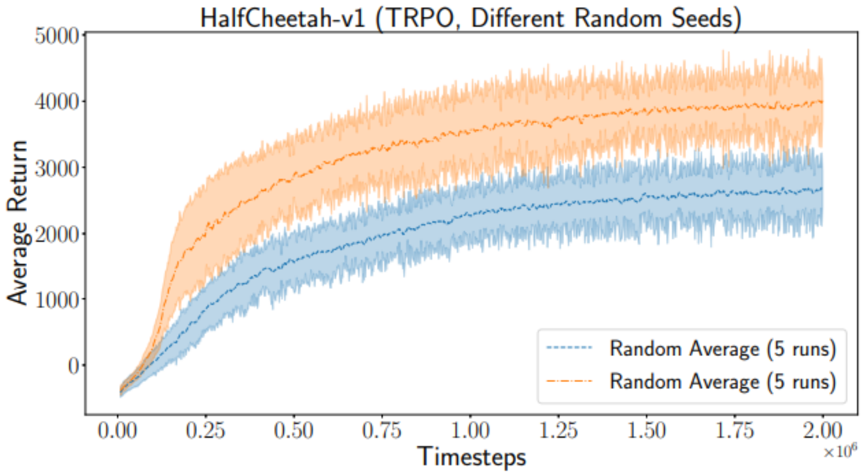


Figure 5: TRPO on HalfCheetah-v1 using the same hyperparameter configurations averaged over two sets of 5 different random seeds each. The average 2-sample t-test across entire training distribution resulted in $t = -9.0916$, $p = 0.0016$.

Deep Reinforcement Learning that Matters
[Henderson et al. 2018]

Performance and Stability

- Does it converge?
- What does it converge to?
- Does it converge every time?

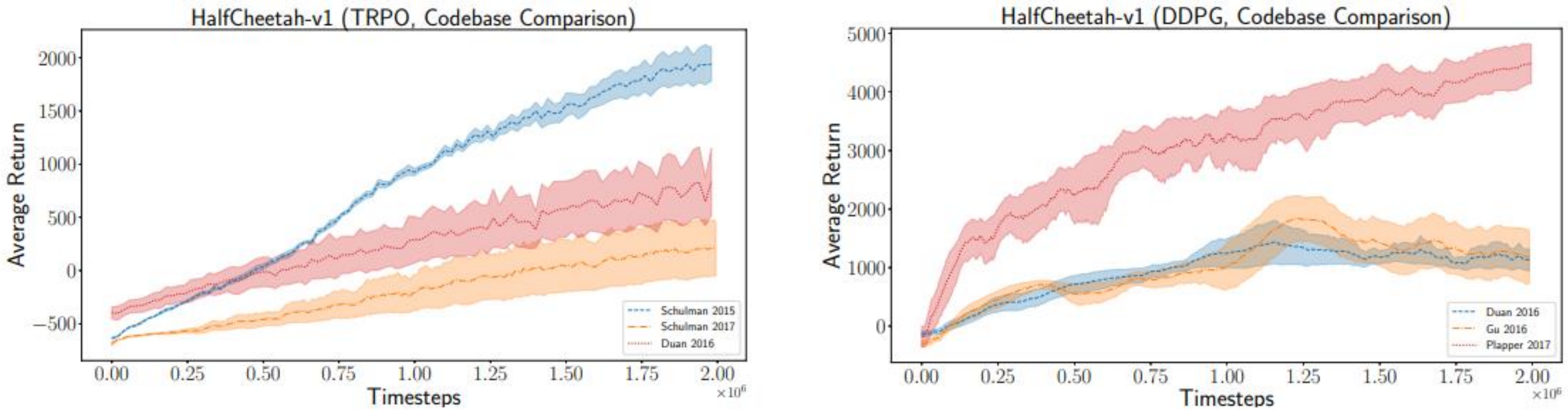


Figure 6: TRPO codebase comparison using our default set of hyperparameters (as used in other experiments).

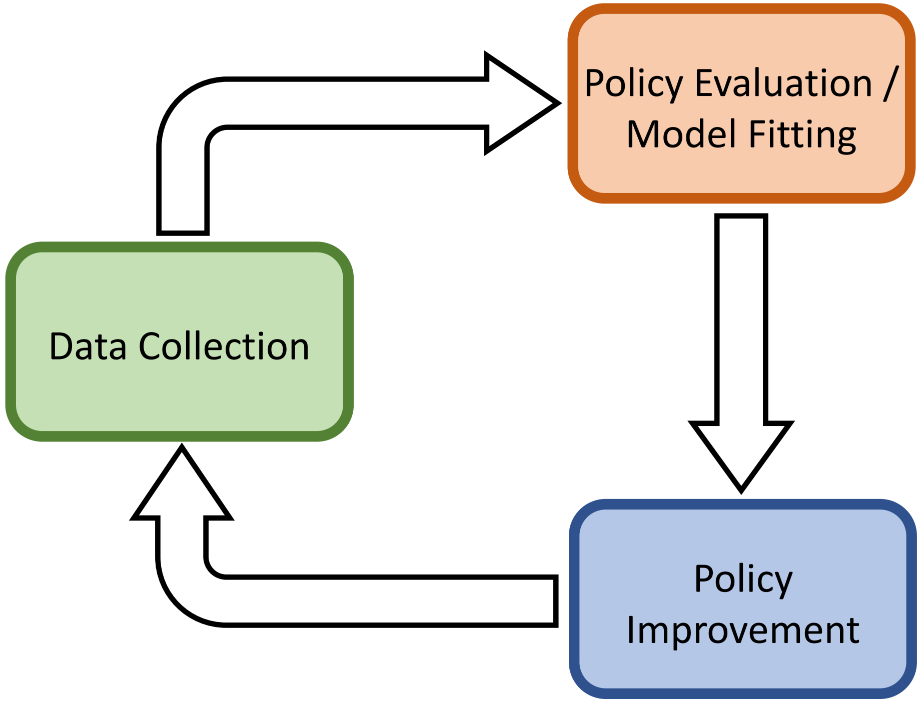
Deep Reinforcement Learning that Matters
[Henderson et al. 2018]

Performance and Stability

- Supervised learning: almost *always* gradient descent
- Reinforcement learning: often *not* gradient descent
  - Q-learning: fixed point iteration
  - Mode-based RL: model not optimized for expected reward
  - Policy gradient: *is* gradient descent, but often very inefficient

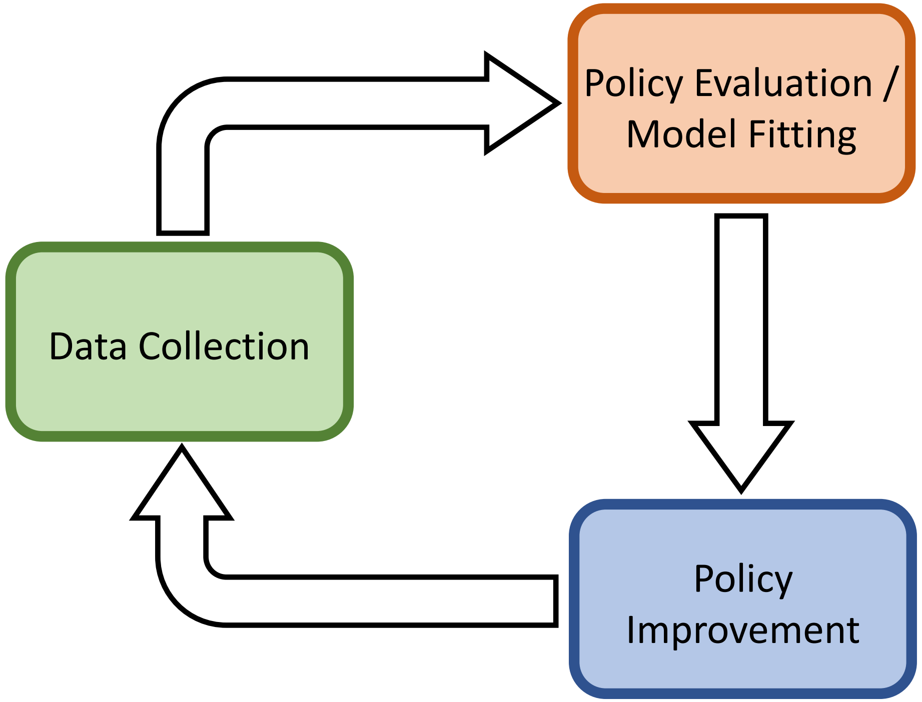
Characteristics

- Sample efficiency
- Wall-clock time
- Performance and stability
- Stochastic/deterministic dynamics
- Continuous/discrete actions
- Modeling challenges



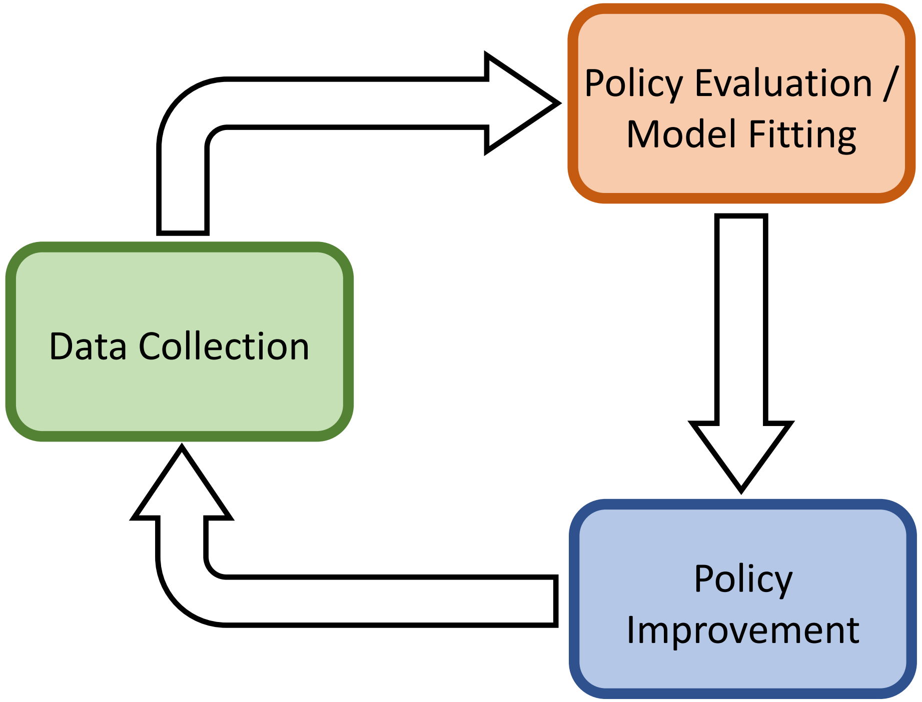
Characteristics

- Sample efficiency
- Wall-clock time
- Performance and stability
- Stochastic/deterministic dynamics
- Continuous/discrete actions
- Modeling challenges



Characteristics

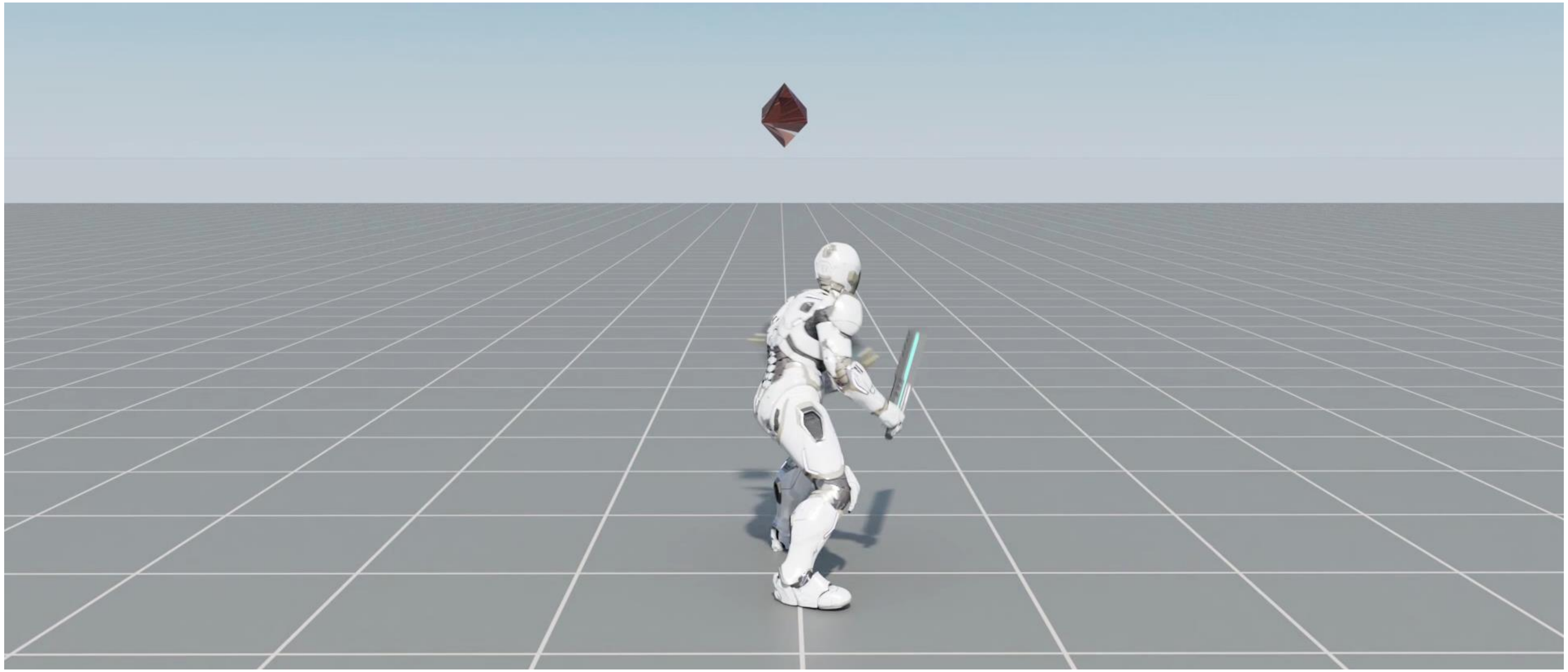
- Sample efficiency
- Wall-clock time
- Performance and stability
- Stochastic/deterministic dynamics
- Continuous/discrete actions
- Modeling challenges



Applications

Policy Gradients

- Directly optimize policy via gradient ascent
  - E.g. TRPO, PPO, A2C, DDPG, SAC, TD3, MBPO



ASE: Large-Scale Reusable Adversarial Skill Embeddings for Physically Simulated Characters
[Peng et al. 2022]

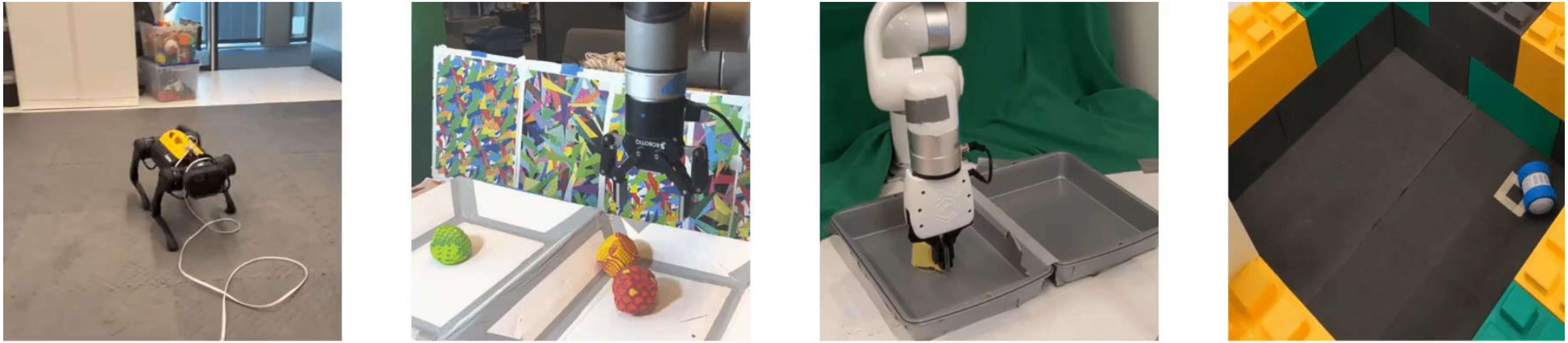
Q-Learning

- Learn Q-function that implicitly encodes policy
  - E.g. DQN, double-DQN, Dueling DQN, Rainbow

Human-Level Control Through Deep Reinforcement Learning
[Mnih et al. 2015]

Model-Based RL

- Learn a model of the dynamics for planning or simulation
  - E.g. Dyna, GPS, MBPO, PETS, Dreamer, MOReL, AlphaGo



A1 Quadruped Walking | UR5 Multi-Object Visual Pick Place | XArm Visual Pick and Place | Sphero Ollie Visual Navigation

DayDreamer: World Models for Physical Robot Learning
[Wu et al. 2022]

Summary

- Anatomy of an RL algorithm
- Algorithm Characteristics
- Applications