Multi-Armed Bandits

Rick Valenzano and Sheila McIlraith

Outline

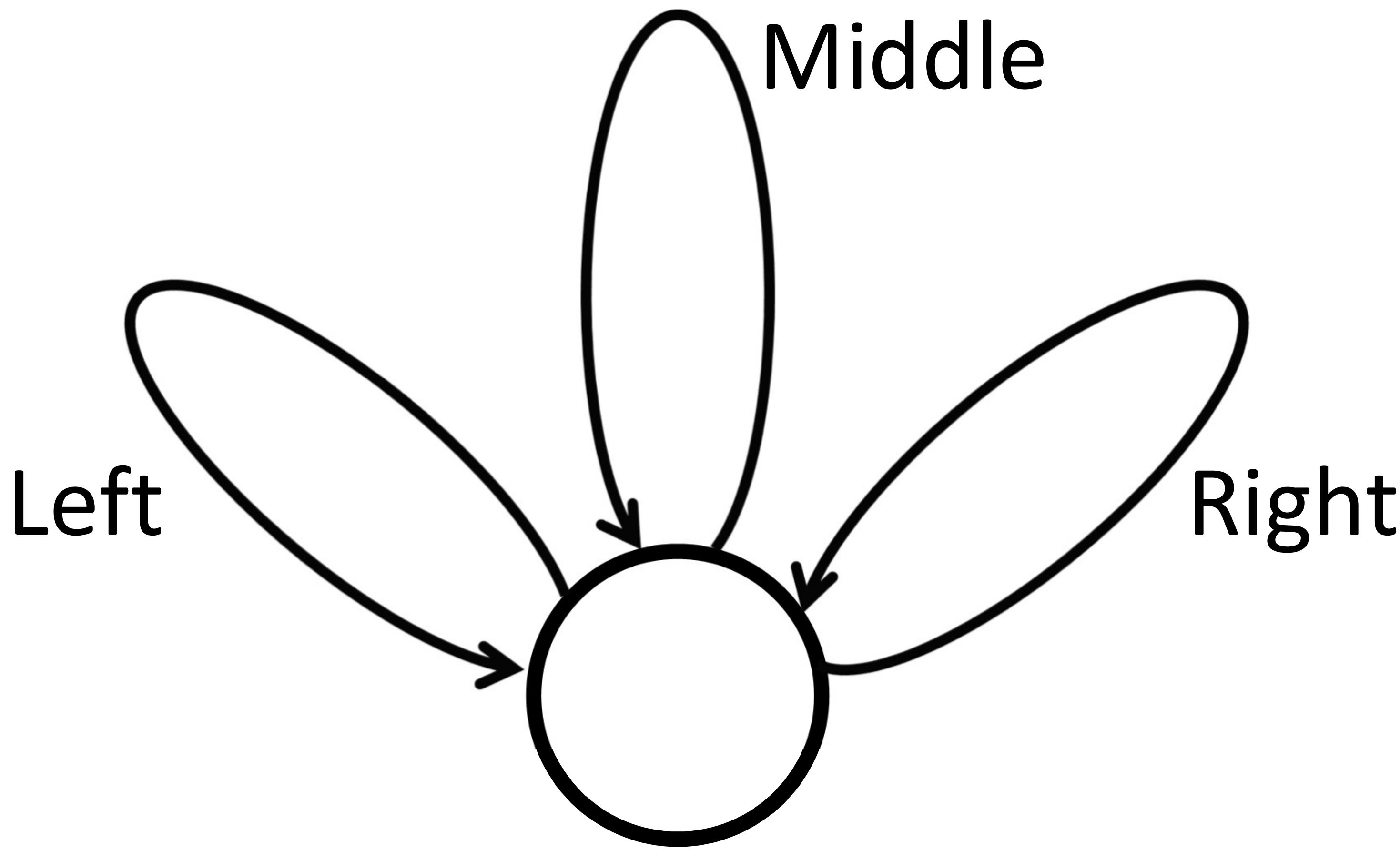
- Learning from experience
  - Exploration vs. exploitation
- Multi-armed bandits as a simple model
- Algorithms for bandit problems
- Stationary vs. non-stationary problems
  - Using incremental update rules

Acknowledgements

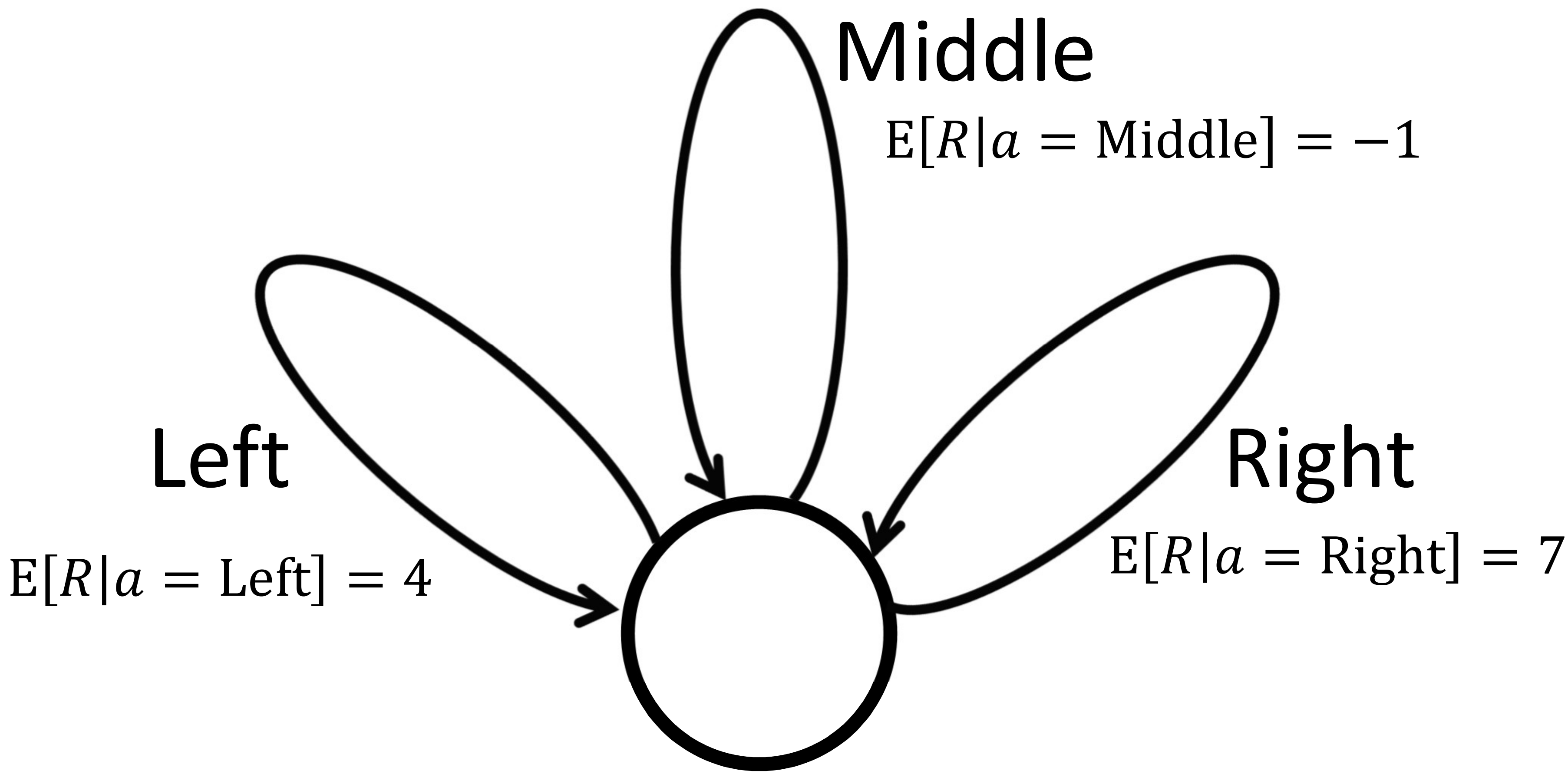
- Images from the RL book

- Based on slides by David Silver and Adam White

Simple MDP

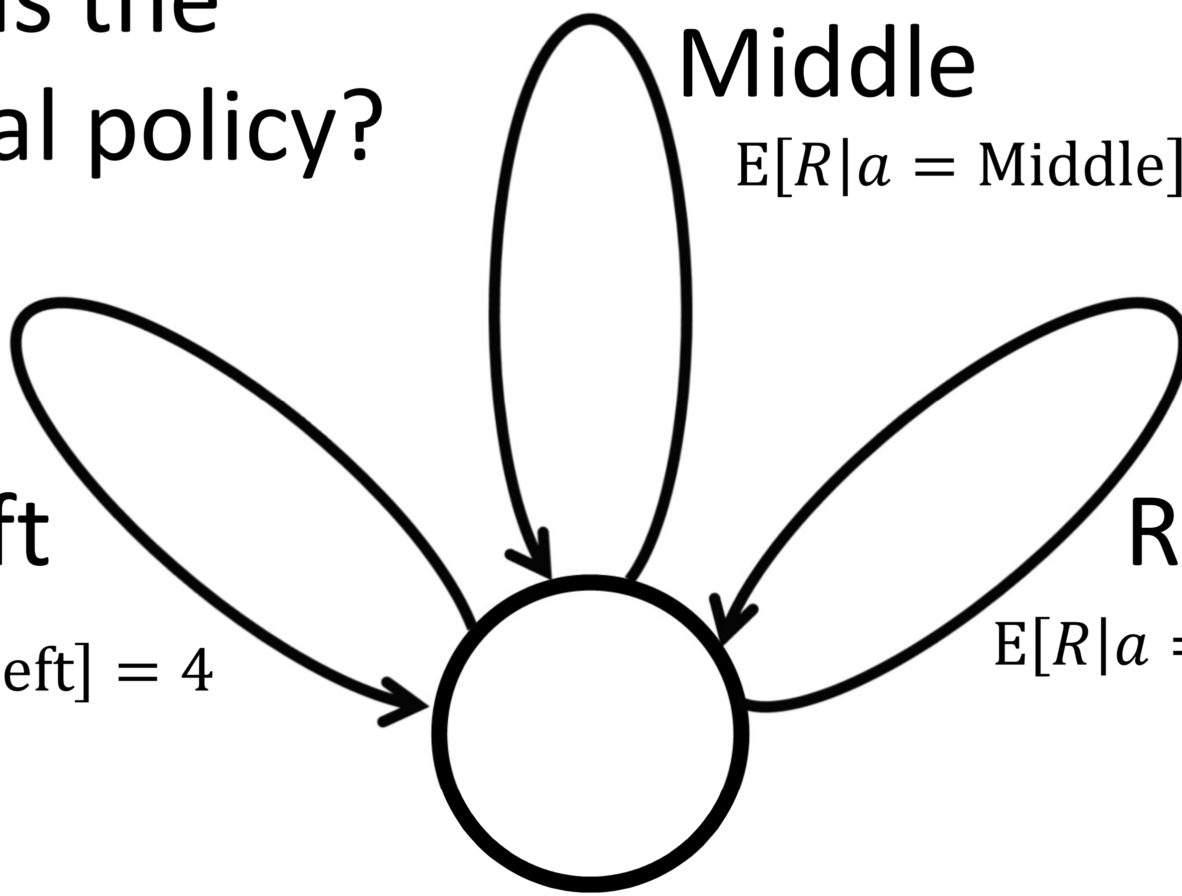


Simple MDP


Middle
$\mathrm{E}[R|a = \text{Middle}] = -1$
Left
$\mathrm{E}[R|a = \text{Left}] = 4$
Right
$\mathrm{E}[R|a = \text{Right}] = 7$

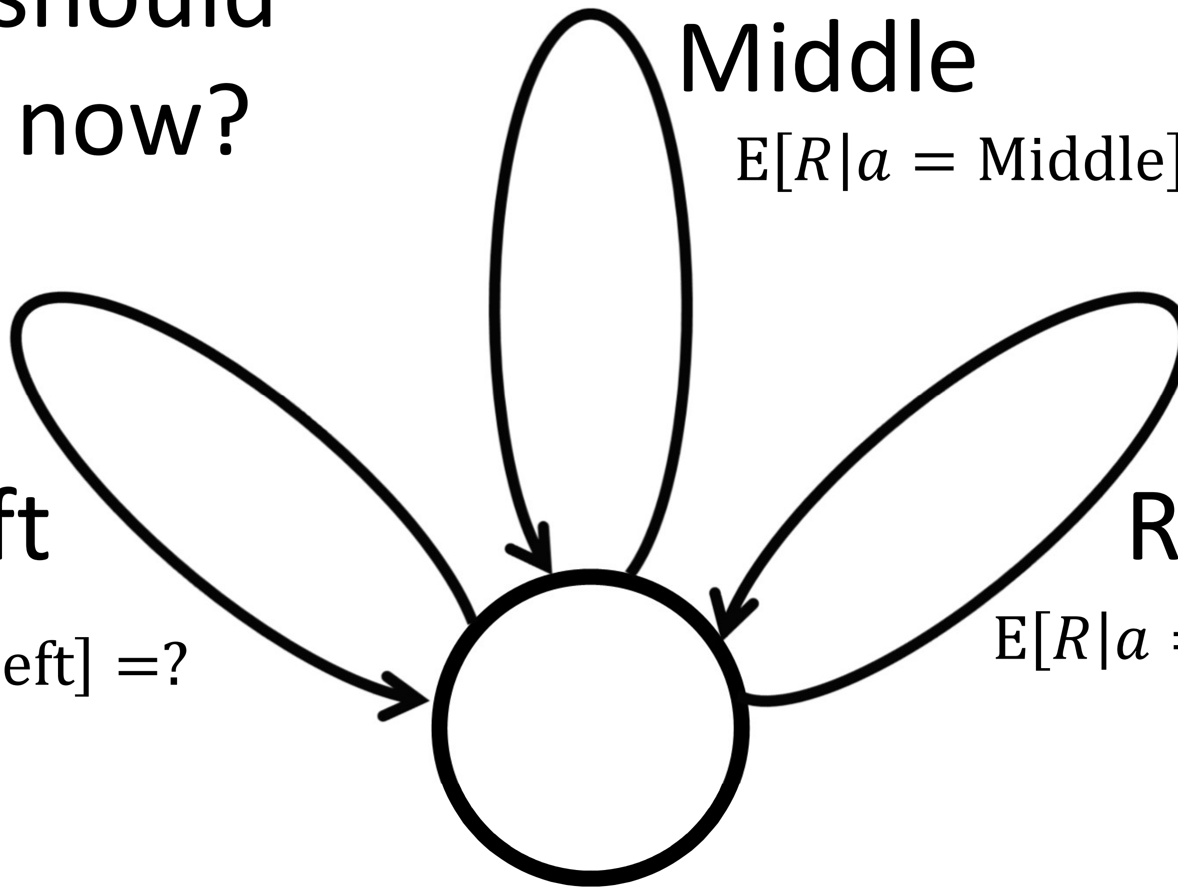
Simple MDP

What is the optimal policy?



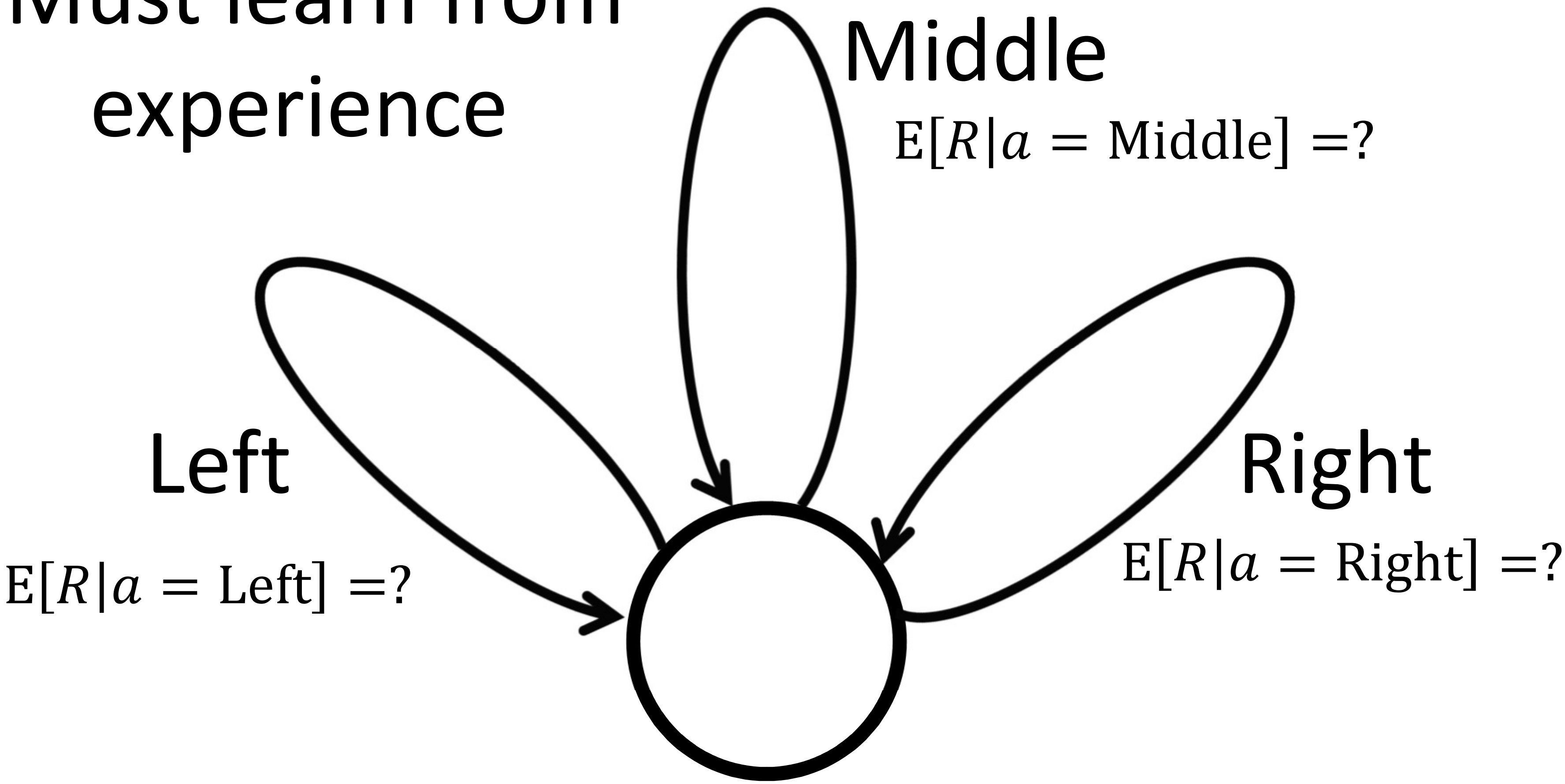
Simple MDP

What should
we do now?



Simple MDP

Must learn from experience



Simple MDP Demo

Simple MDP Demo

- Possible strategies?

- What information seems useful to keep track of?

Multi-Armed Bandits

- There are n actions $A = \{a_1, \dots, a_n\}$

- All actions applicable on all of discrete time steps
  - Infinite time steps 1, 2, 3, …
  - On each time step, pick one to execute. Denoted A_t

- $q^*(s, a_i) = q^*(a_i) = \mathrm{E}[R_t | a_i]$

- Agent is trying to maximize total reward over time

Applications

- Youtube, ad, news recommendations
  - Or extension to “associative” bandits
- Parameter selection on a batch of problems
- Clinical trials or treatment

Greedy Policy

- Let $q_t(a_i)$ be the average value of a_i after t steps
- On each step, choose the action with the best average return thus far

$$A_t = \text{argmax}_{a \in A} q_t(a)$$

- What are the issues with this approach?

ϵ-Greedy Policy

- Don't always pick the best looking action
  - May not actually be the best

ϵ-greedy policy:

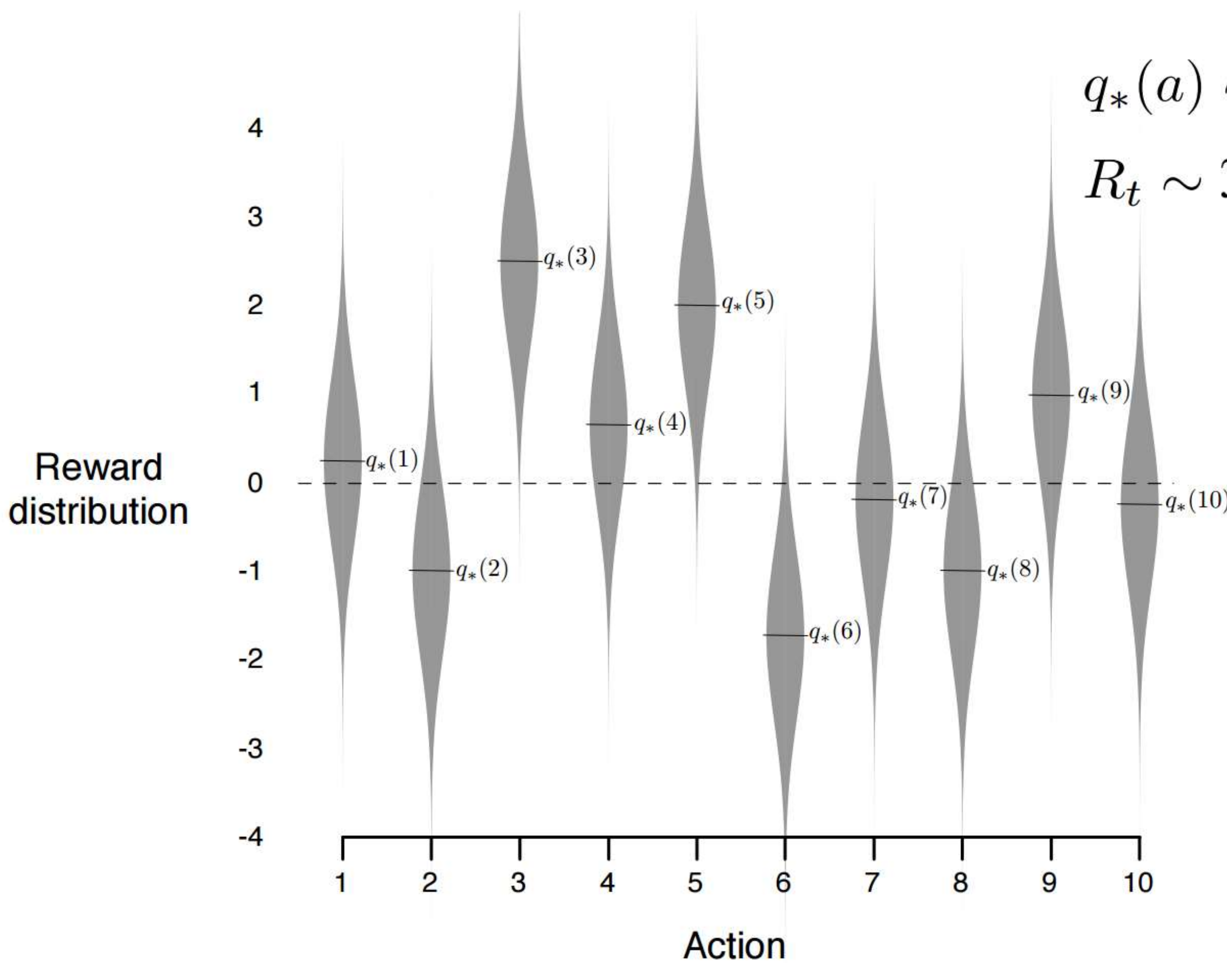
With probability $(1 - \epsilon)$:

$$A_t = \text{argmax}_{a \in A} q_t(a)$$

With probability ϵ:

A_t is selected randomly from A

10-Armed Bandit Testbed

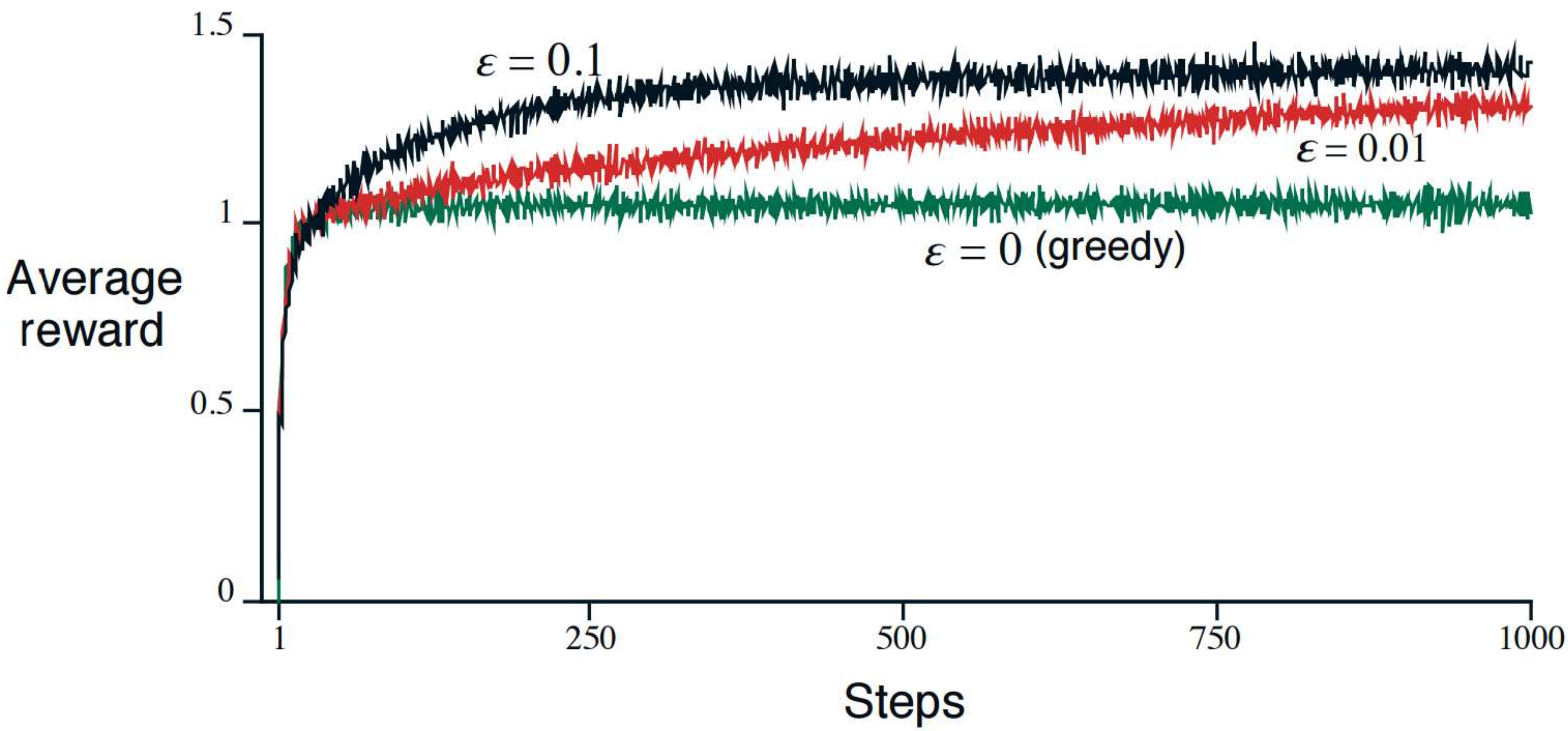


$$q_*(a) \sim \mathcal{N}(0, 1)$$

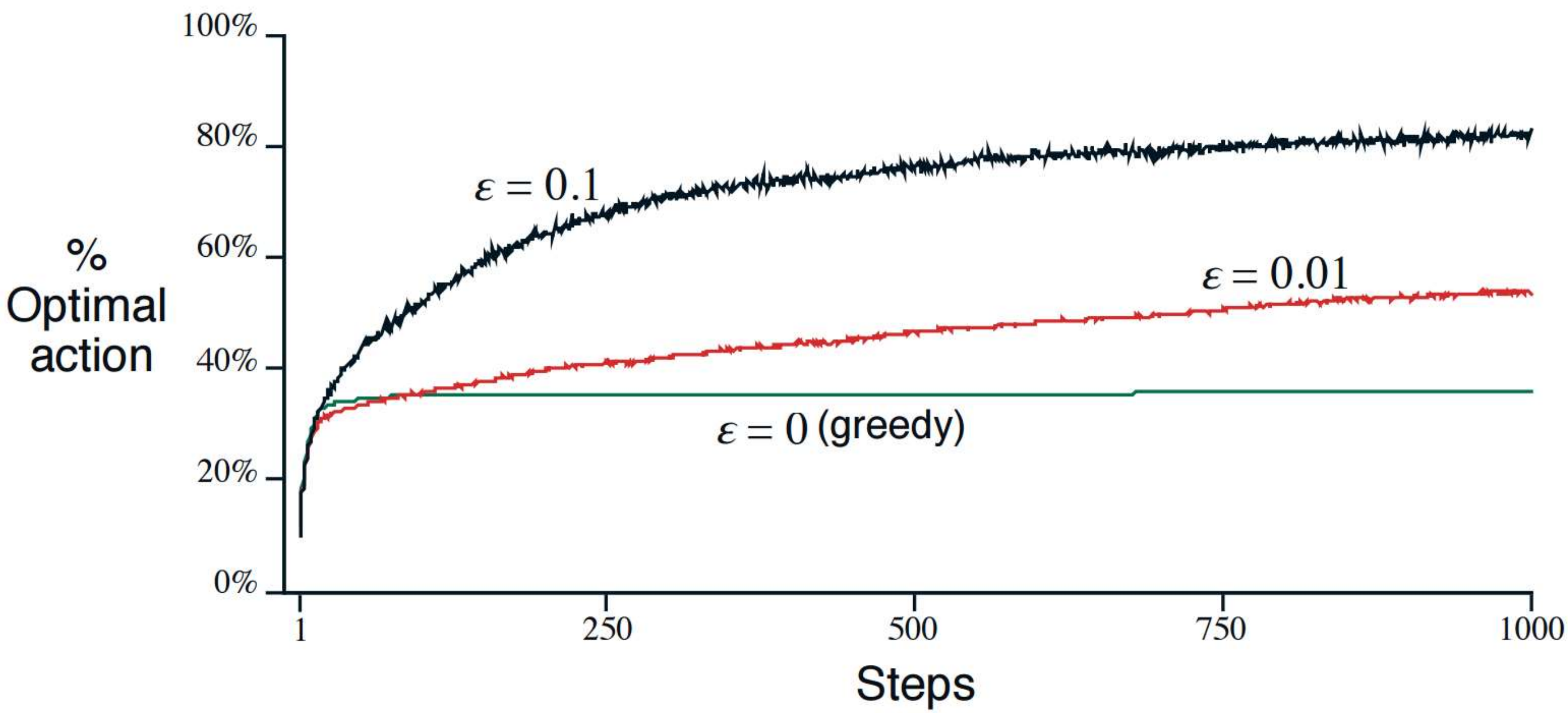
$$R_t \sim \mathcal{N}(q_*(a), 1)$$

- Made 2,000 such problems

10-Armed Bandit Results



10-Armed Bandit Results



$\boldsymbol{\epsilon}$-Greedy Policy

- $q_t(a_i)$ converges to $q^*(a_i)$ in the limit
- Needs to make exploratory actions for this to hold
- But exploratory actions may be “sacrificing” potential reward

Exploration vs. Exploitation

- When select greedily, agent is **exploiting** its information
- When selects randomly, it is **exploring**
- If we exploit to much, can get stuck with suboptimal values
- If we explore too much, we may be sacrificing a lot of reward that we could have gotten
- Need to balance between the two
  - A central dilemma in reinforcement learning