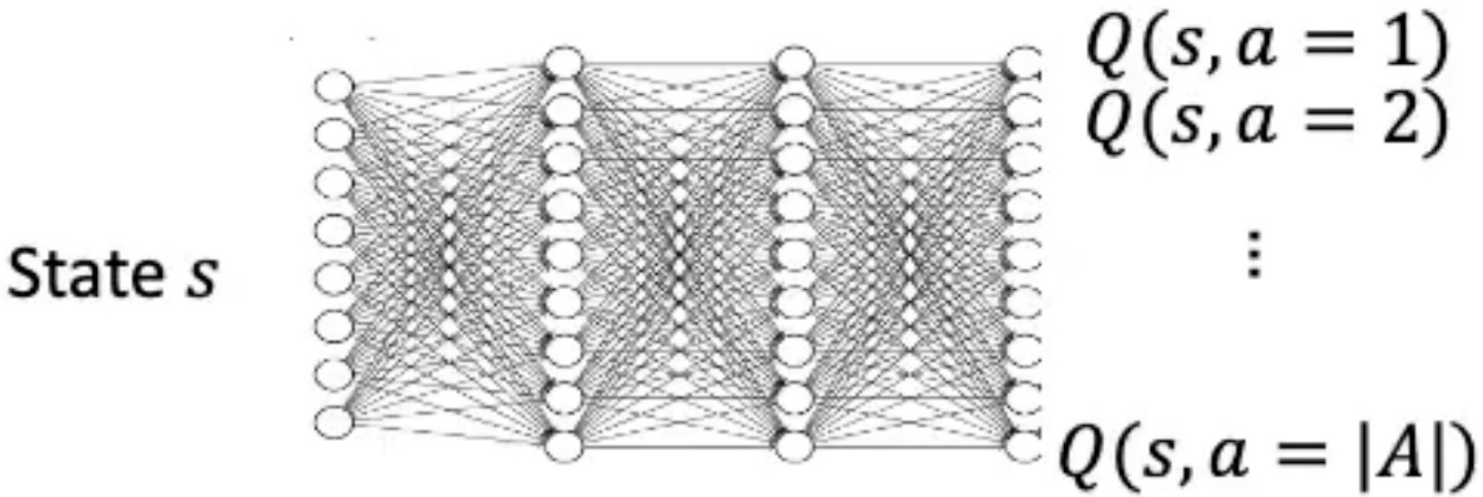
Continuous control with deep reinforcement learning

Lillicrap, T.P., Hunt, J.J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D. and Wierstra, D., Google Deepmind, 2015.

Presenter: Zhanzhan Zhao

Extending DQN to Continuous Action Spaces

DQN: Discrete and Finite Action Space



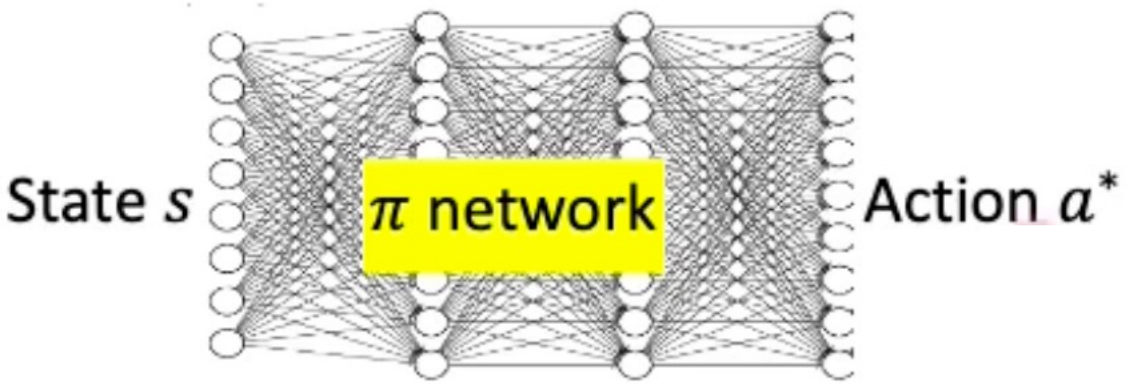
Policy:

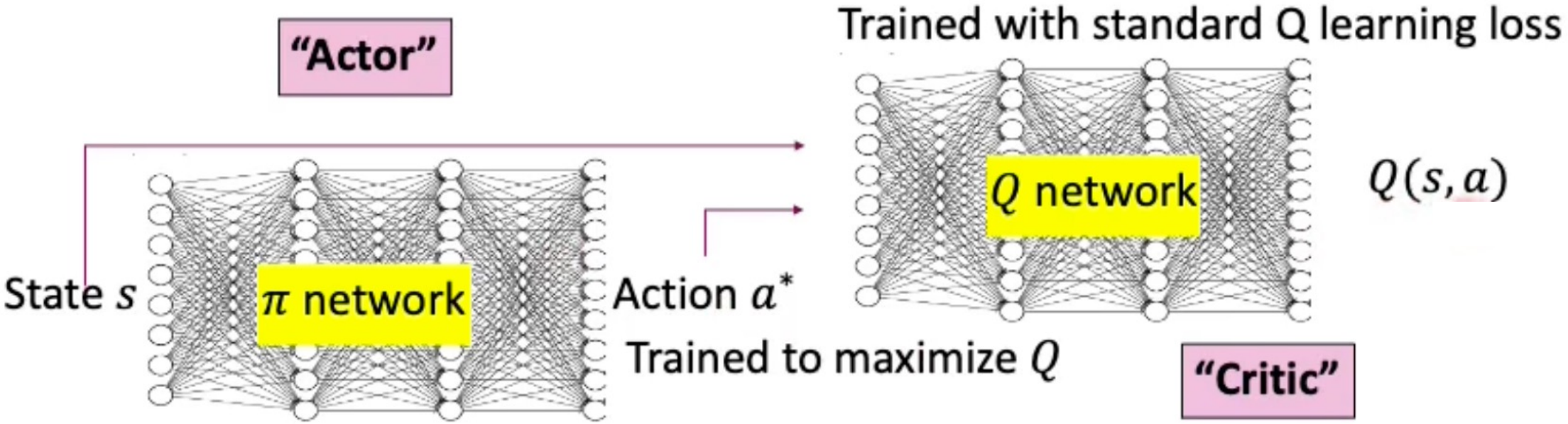
$$\pi(s) = a^* = \underset{a}{\operatorname{argmax}}\, Q(s, a)$$

DQN with Continuous Actions?

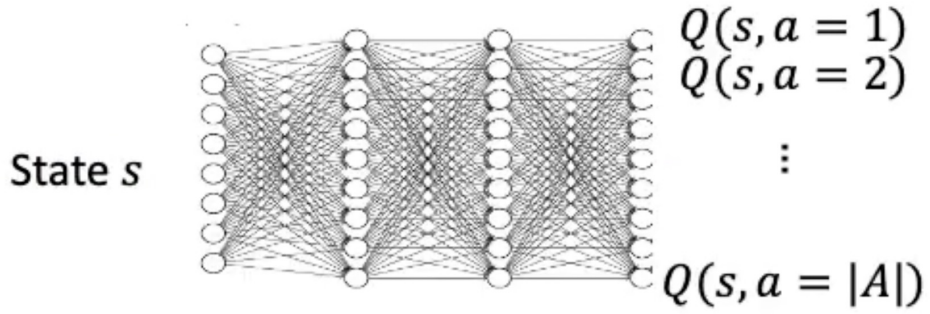
Could we train a neural network to produce the output of this optimization problem?

$$a^*(s) = \underset{a}{\operatorname{argmax}}\, Q(s, a)$$





Deep Deterministic Policy Gradient (DDPG)



DQN

Policy:

$$\pi(s) = a^* = \underset{a}{\operatorname{argmax}}\, Q(s, a)$$

DDPG

- for iter = 1, 2, …

  Roll-outs:
  Execute roll-outs under current policy (+some noise for exploration)

  Q function update:

$$g \propto \nabla_\phi \sum_t (Q_\phi(s_t, u_t) - \hat{Q}(s_t, u_t))^2$$

  Policy update:
  Backprop through Q to compute gradient estimates for all t:

$$g \propto \sum_t \nabla_\theta Q_\phi(s_t, \pi_\theta(s_t, v_t))$$

QUIZ 1:

1.1. Is DDPG a model-free or model-based algorithm?

1.2 Is DDPG an on-policy or off-policy learning algorithm?

DDPG

- for iter = 1, 2, …

  Roll-outs:
  Execute roll-outs under current policy (+some noise for exploration)

  Q function update:

$$g \propto \nabla_\phi \sum_t (Q_\phi(s_t, u_t) - \hat{Q}(s_t, u_t))^2$$

  Policy update:
  Backprop through Q to compute gradient estimates for all t:

$$g \propto \sum_t \nabla_\theta Q_\phi(s_t, \pi_\theta(s_t, v_t))$$

QUIZ 1:

1.1. Is DDPG a model-free or model-based algorithm?

model-free

DDPG

- for iter = 1, 2, …

  Roll-outs:
  Execute roll-outs under current policy (+some noise for exploration)

  Q function update:

$$g \propto \nabla_\phi \sum_t (Q_\phi(s_t, u_t) - \hat{Q}(s_t, u_t))^2$$

  Policy update:
  Backprop through Q to compute gradient estimates for all t:

$$g \propto \sum_t \nabla_\theta Q_\phi(s_t, \pi_\theta(s_t, v_t))$$

QUIZ 1:

1.2 Is DDPG an on-policy or off-policy learning algorithm?
off-policy

Target Policy: It is the policy that an agent is trying to learn, i.e agent is learning value function for this policy.

Behavior Policy: It is the policy that is being used by an agent for action select, i.e agent follows this policy to interact with the environment.

DDPG

- for iter = 1, 2, …

  Roll-outs:
  Execute roll-outs under current policy (+some noise for exploration)

  Q function update:

$$g \propto \nabla_\phi \sum_t (Q_\phi(s_t, u_t) - \hat{Q}(s_t, u_t))^2$$

  Policy update:
  Backprop through Q to compute gradient estimates for all t:

$$g \propto \sum_t \nabla_\theta Q_\phi(s_t, \pi_\theta(s_t, v_t))$$

QUIZ 2:

2.1 What is the trick here used related to experience replay and target network?

2.2 Why we want to use the trick?

DDPG

- for iter = 1, 2, …

  Roll-outs:
  Execute roll-outs under current policy (+some noise for exploration)

  Q function update:

$$g \propto \nabla_\phi \sum_t (Q_\phi(s_t, u_t) - \hat{Q}(s_t, u_t))^2$$

  Policy update:
  Backprop through Q to compute gradient estimates for all t:

$$g \propto \sum_t \nabla_\theta Q_\phi(s_t, \pi_\theta(s_t, v_t))$$

QUIZ 2:

2.1 What is the trick here used related to experience replay and target network?

Use lagged (Polyak-averaging) version of Q_ϕ and π_θ for target values $\hat{Q}_t$

$$\hat{Q}_t = r_t + \gamma Q_{\phi'}(s_{t+1}, \pi_{\theta'}(s_{t+1}))$$

DDPG

- for iter = 1, 2, …

  Roll-outs:
  Execute roll-outs under current policy (+some noise for exploration)

  Q function update:

$$g \propto \nabla_\phi \sum_t (Q_\phi(s_t, u_t) - \hat{Q}(s_t, u_t))^2$$

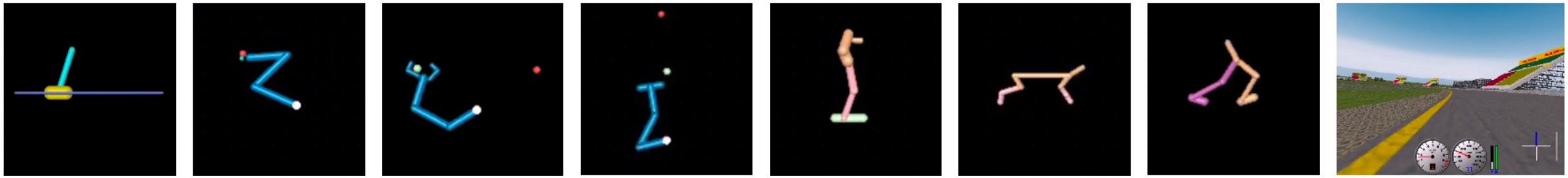
  Policy update:
  Backprop through Q to compute gradient estimates for all t:

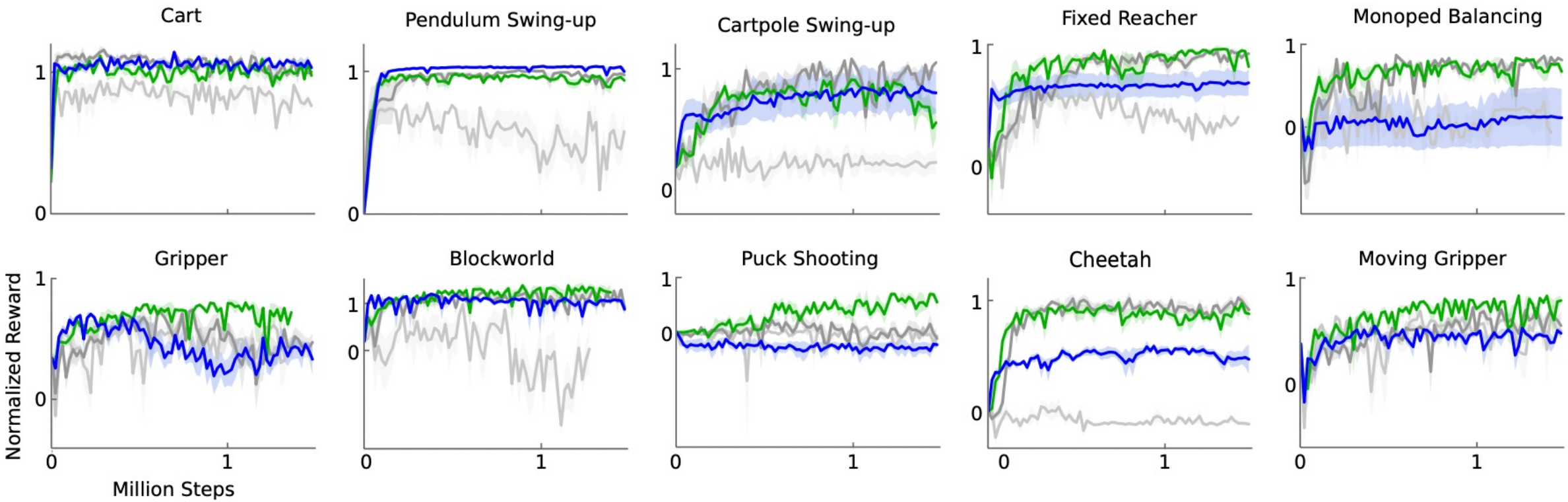
$$g \propto \sum_t \nabla_\theta Q_\phi(s_t, \pi_\theta(s_t, v_t))$$

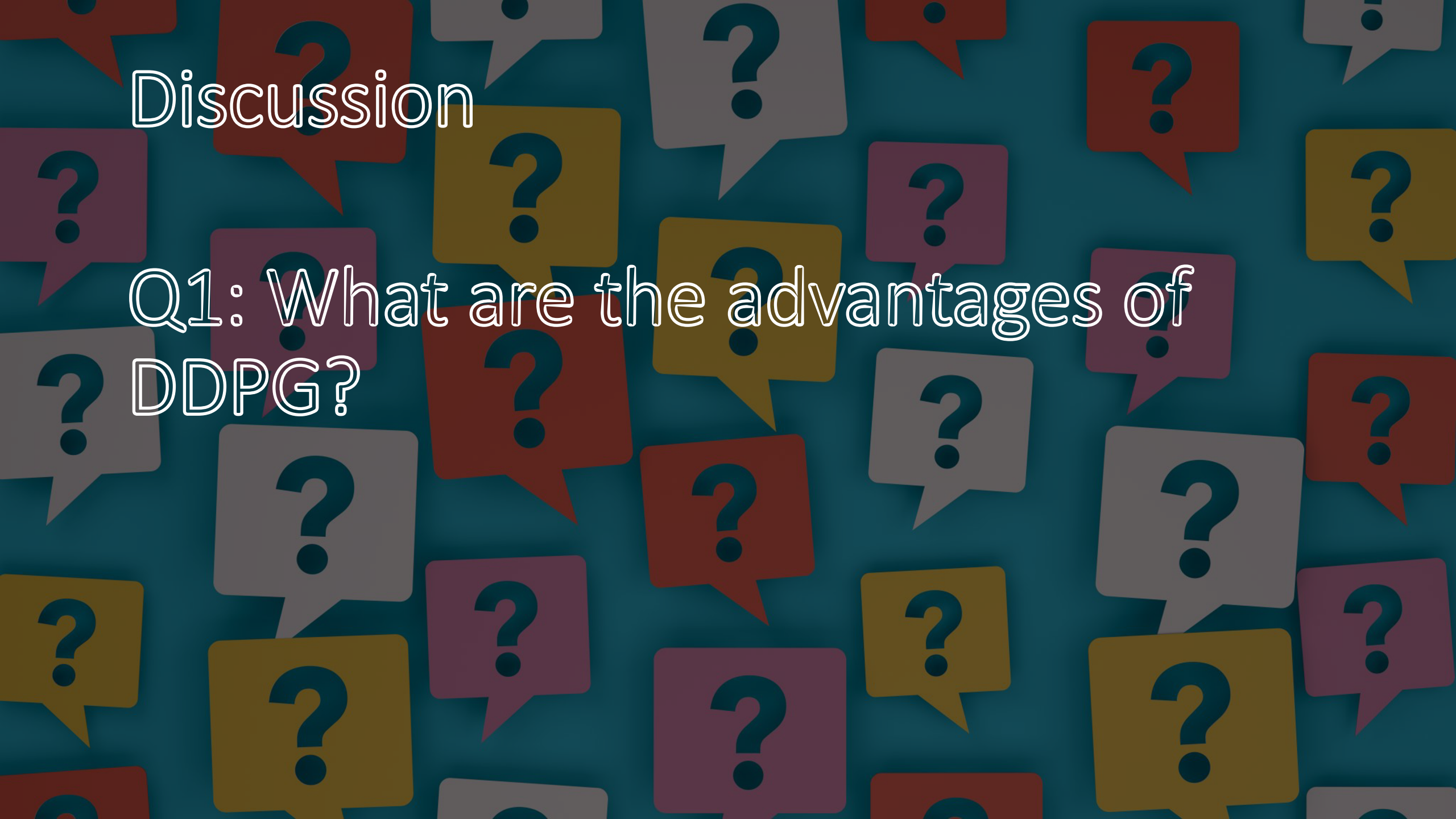
QUIZ 2:

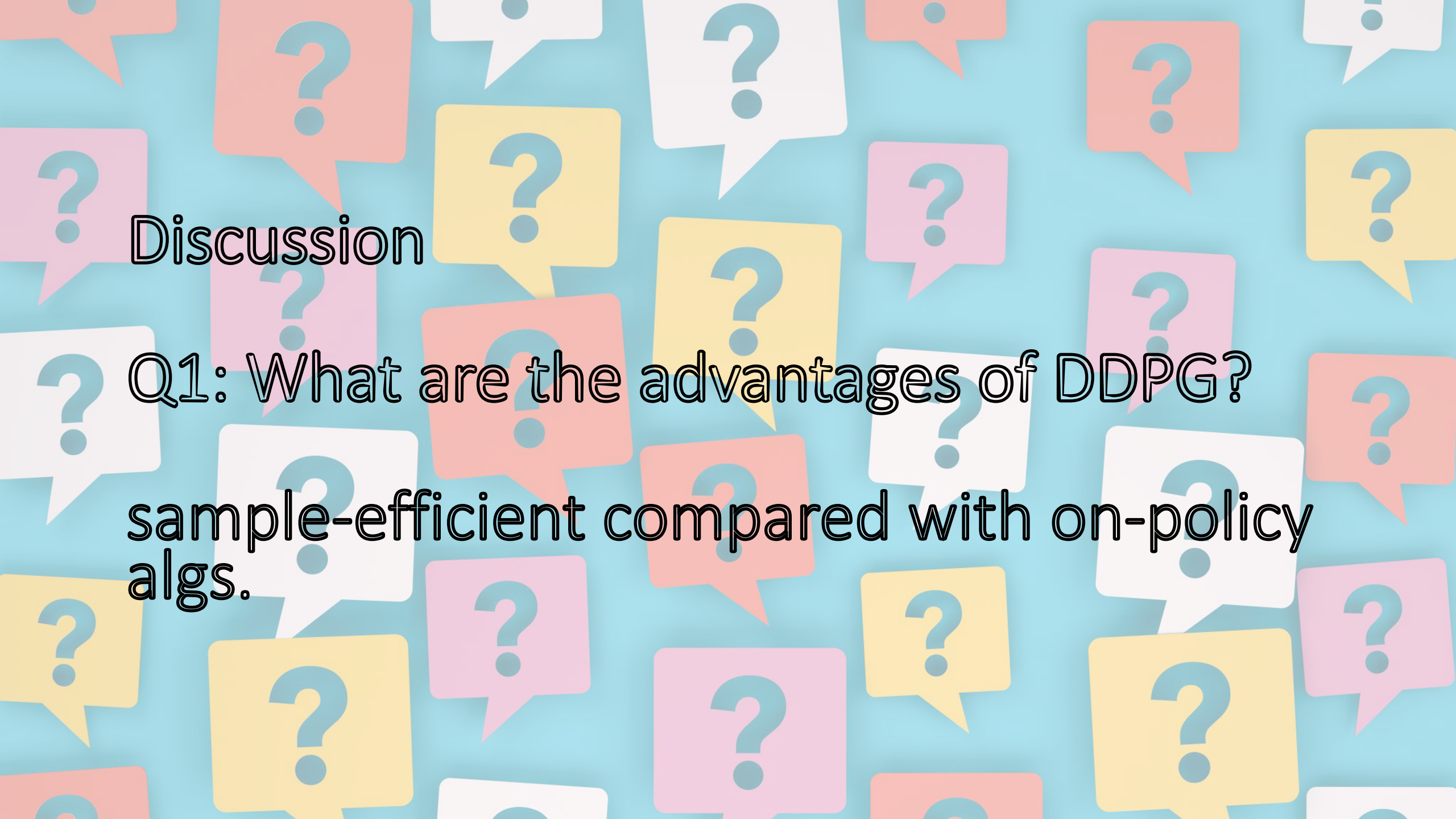
2.2 Why we want to use the trick?

to increase learning stability




Cart
Pendulum Swing-up
Cartpole Swing-up
Fixed Reacher
Monoped Balancing
Gripper
Blockworld
Puck Shooting
Cheetah
Moving Gripper
Normalized Reward
Million Steps
0
1


Discussion
Q1: What are the advantages of DDPG?

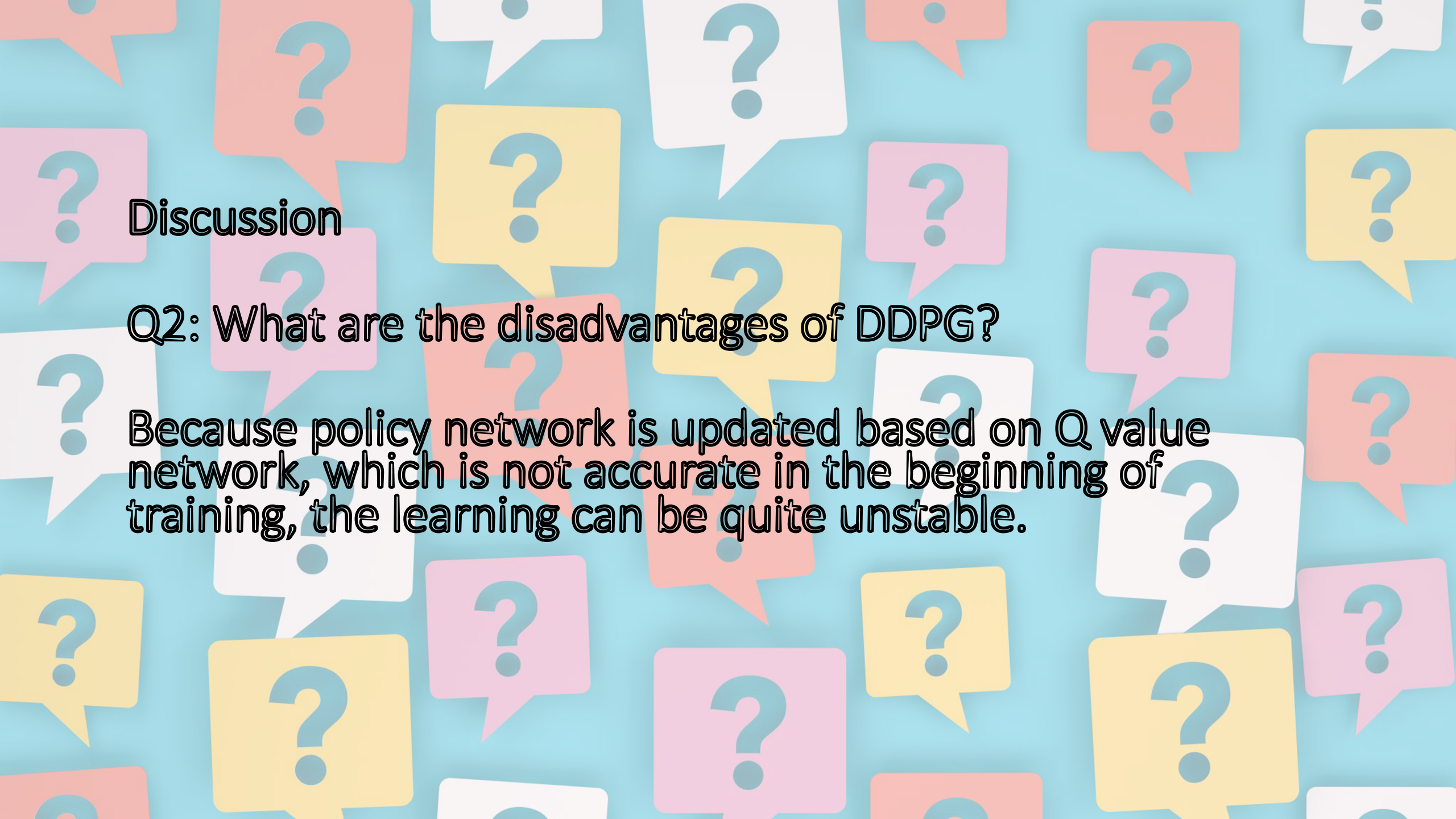


Discussion

Q1: What are the advantages of DDPG?

sample-efficient compared with on-policy algs.


Discussion
Q2: What are the disadvantages of DDPG?



Discussion

Q2: What are the disadvantages of DDPG?

Because policy network is updated based on Q value network, which is not accurate in the beginning of training, the learning can be quite unstable.