Lecture 7： Multi-Agent RL

16th June. 2022

Multi-Agent RL

- What is the multi-agent reinforcement learning(MARL)?
- What is the difference between single-agent reinforcement learning and multi-agent reinforcement learning?
- What is the difficulty in the multi-agent reinforcement learning?
- What are the categories of multi-agent reinforcement learning?

Multi-Agent RL

- What is the multi-agent reinforcement learning(MARL)?
  - MARL addresses the sequential decision-making problem of multiple autonomous agents that operate in a common environment, each of which aims to optimize its own long-term return by interacting with the environment and other agents
  - A group of agents work together to optimize team performance
  - Multiagent systems include a set of autonomous entities(agents) that share a common environment and where each agent can independently perceive environment, act acting to its individual objectives and as a consequence, modify the environment
  - In an multiagent system, agents must compete or cooperate to obtain the best overall results.

Multi-Agent RL

What is the multi-agent reinforcement learning(MARL)?

Robots

Drone Delivery

Games

Autonomous Vehicles

Smart Grids

MALib

Multi-Agent RL

- What is the multi-agent reinforcement learning(MARL)?
- What is the difference between single-agent reinforcement learning and multi-agent reinforcement learning?
- What is the difficulty in the multi-agent reinforcement learning?
- What are the categories of multi-agent reinforcement learning?

Multi-Agent RL

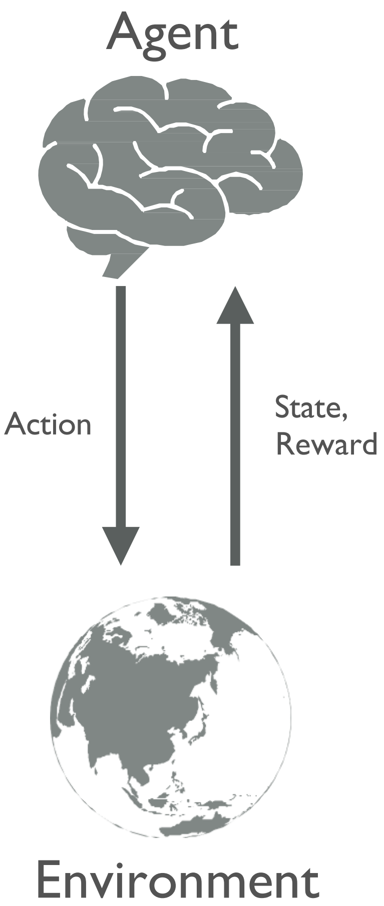
- What is the difference between single-agent reinforcement learning and multi-agent reinforcement learning?
  - Single-agent RL:
    - Only one agent
    - State、local action、single reward
  - Multi-agent RL:
    - At least two agents
    - Local observation、joint action、team reward
    - Agents communicate with each other and interact with environment at the same time.

Multi-Agent RL

- **What is the difference between single-agent reinforcement learning and multi-agent reinforcement learning?**
  - Problem Formulation: single-agent RL:

    Markov Decision Process(MDP) $(S, A, R, T, P_0, \gamma)$

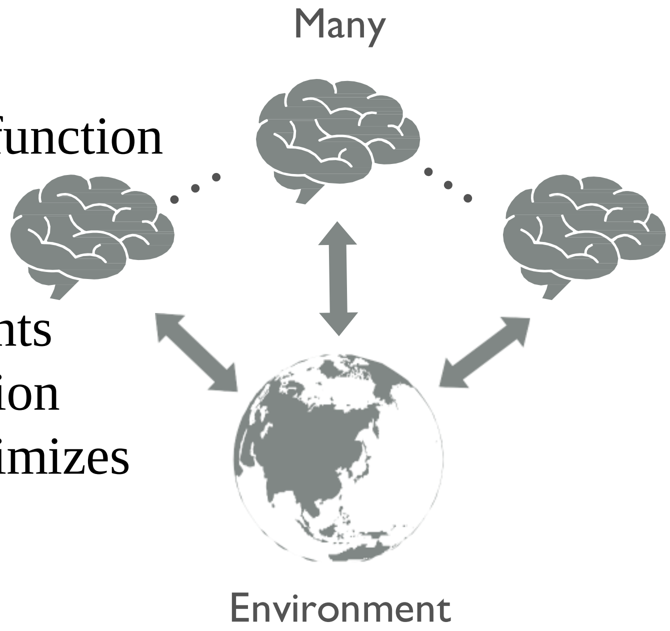
    - S denotes the state space
    - A is the action space
    - $R = R(s, a)$ is the reward function
    - $T: S \times A \times S \to [0,1]$ is the state transition function
    - P_0 is the distribution of the initial state
    - γ is a discount factor
    - Goal: find the optimal policy that maximizes expected reward



Multi-Agent RL

- What is the difference between single-agent reinforcement learning and multi-agent reinforcement learning?
  - Problem Formulation: multi-agent RL:

    Partially Observable Markov Decision Process(POMDP)$(S, A, R, T, P_0, Z, O, n, \gamma)$
    - n agents in the environment
    - S denotes the state space
    - A is the joint action space $A^1 \times \cdots \times A^n$
    - $R = R(S, A)$ is the share reward function
    - $T: S \times A \times S \rightarrow [0,1]$ is the state transition function
    - P_0 is the distribution of the initial state
    - γ is a discount factor
    - Z is the individual observation for each agents
    - $O(s, a): S \times A \rightarrow Z$ is the observation function
    - Goal: find the optimal joint policy that maximizes expected team reward



Multi-Agent RL

- What is the multi-agent reinforcement learning(MARL)?
- What is the difference between single-agent reinforcement learning and multi-agent reinforcement learning?
- What is the difficulty in the multi-agent reinforcement learning?
- What are the categories of multi-agent reinforcement learning?

Multi-Agent RL

□ What is the difficulty in the multi-agent reinforcement learning?

- Non-stationarity:
  - an agent observes not only the outcomes of its own action but also the behavior of other agents
  - Learning among the agents is complex because all agents potentially interact with each other and learn concurrently
- Partial observability:
  - The agents only capture partial information about the environment before making decision
- Dimension catastrophic:
  - Joint action space and Joint state space
  - Large-scale multi-agent decision-making、
- Credit assignment:
  - Lazy agent
- Sample efficiency、Exploration and Exploitation、complex mixed environment, etc.

Multi-Agent RL

What is the difficulty in the multi-agent reinforcement learning?

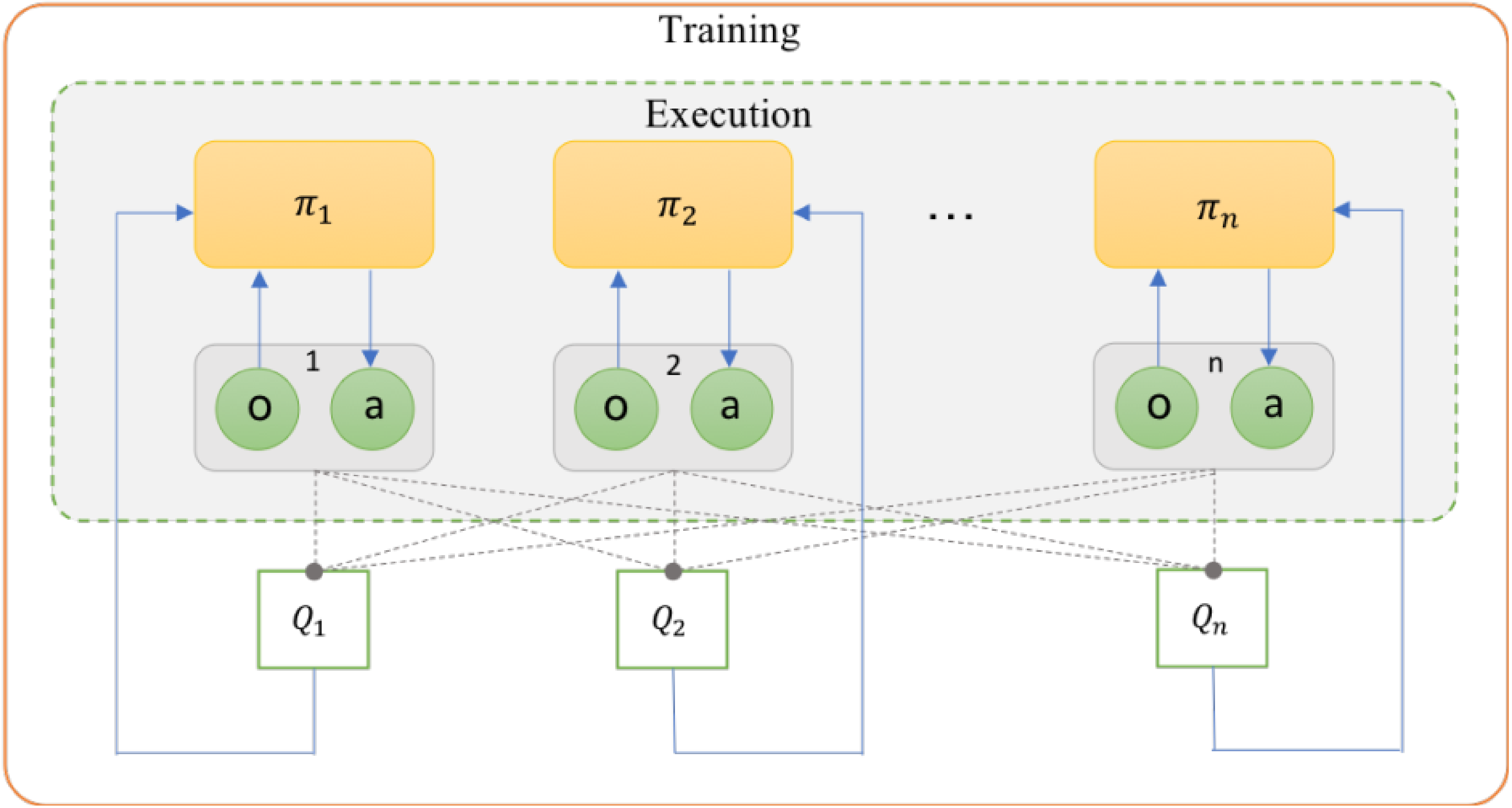
Solution

- Non-stationarity:
  - Centralized Train and Decentralized execution(CTDE), e.g. MADDPG
  - Communication, e.g. CommNet
- Partial observability:
  - RNN、GRU、LSTM, e.g. DRQN
- Dimension catastrophic:
  - CTDE, e.g. VDN、QMIX
  - Mean-Field, e.g. MFAC
- Credit assignment:
  - Counterfactual mechanisms, e.g. COMA
- Exploration and Exploitation:
  - Reward shaping(intrinsic reward、novelty)
  - UCB
  - Influence(mutual information)

Multi-Agent RL

- **What is the difficulty in the multi-agent reinforcement learning?**

  Centralized Train and Decentralized execution(CTDE)



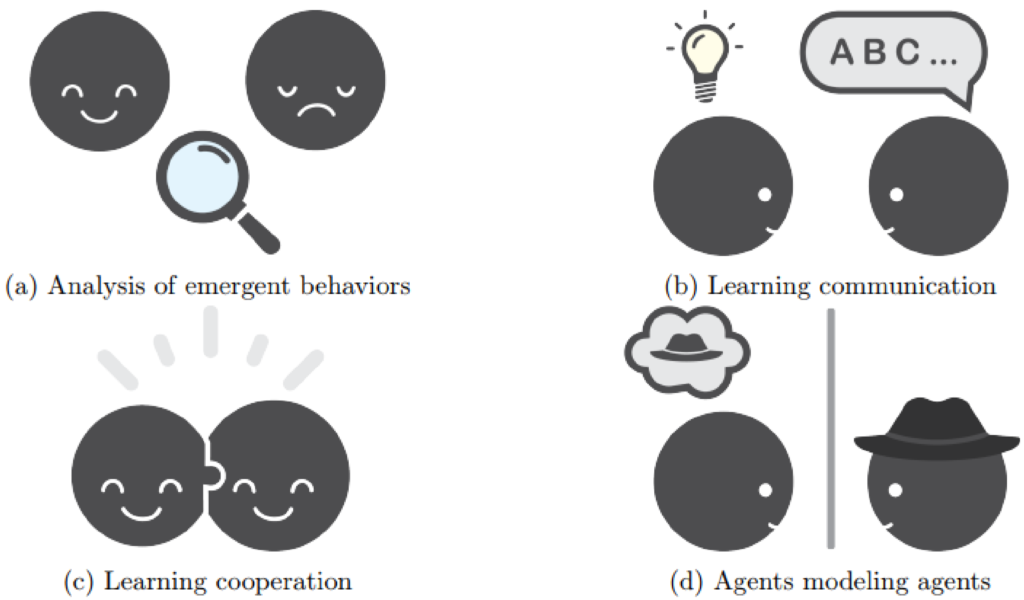
Multi-Agent RL

- What is the multi-agent reinforcement learning(MARL)?
- What is the difference between single-agent reinforcement learning and multi-agent reinforcement learning?
- What is the difficulty in the multi-agent reinforcement learning?
- What are the categories of multi-agent reinforcement learning?

Multi-Agent RL

What are the categories of multi-agent reinforcement learning?

- Analysis of emergent learning:
  - Simply using Single-agent RL algorithm in multi-agent scenarios
- Learning communication:
  - Learning communication protocols among agents
- Learning cooperation:
  - Learning to cooperate using only actions and local observation
- Agents modeling agents:
  - Reasoning about others

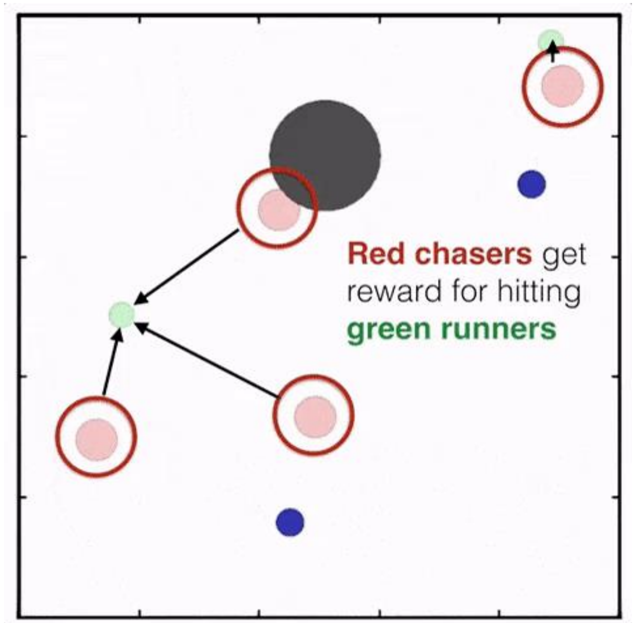


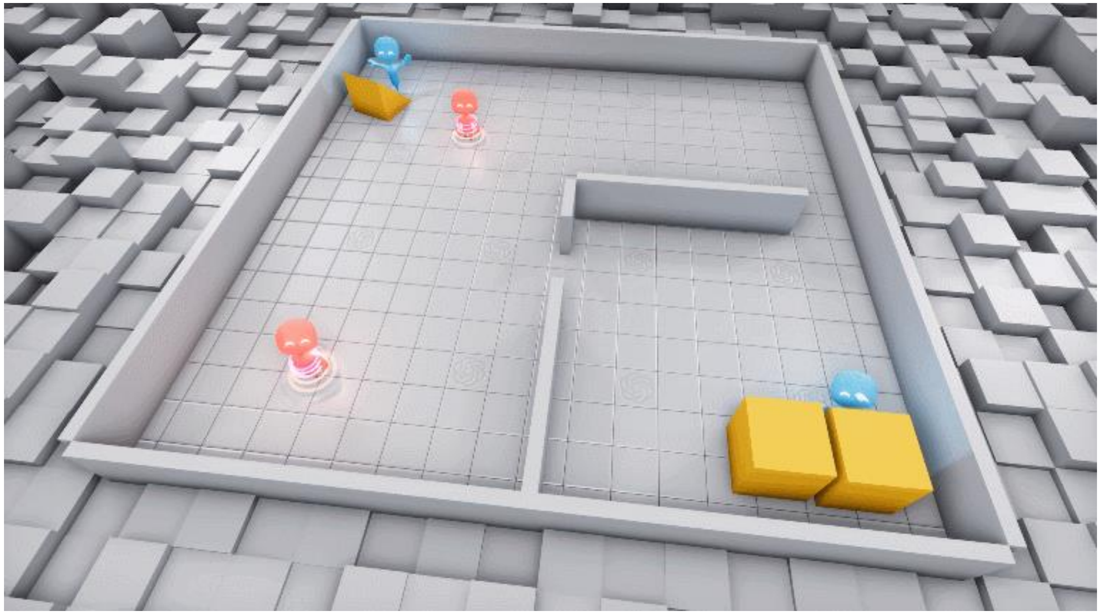
(a) Analysis of emergent behaviors
(b) Learning communication
(c) Learning cooperation
(d) Agents modeling agents

Multi-Agent RL

What are the types of multi-agent reinforcement learning?

- Three major settings: cooperative, competitive, mixed scenarios
  - Cooperative： working together and coordinating their actions、maximizing a shared team reward
  - Competitive： self-interested(maximizing an individual reward)、opposite rewards、 zero-sum games
  - Mixed scenarios： general-sum games





Multi-Agent RL-Learning Cooperation

MADDPG

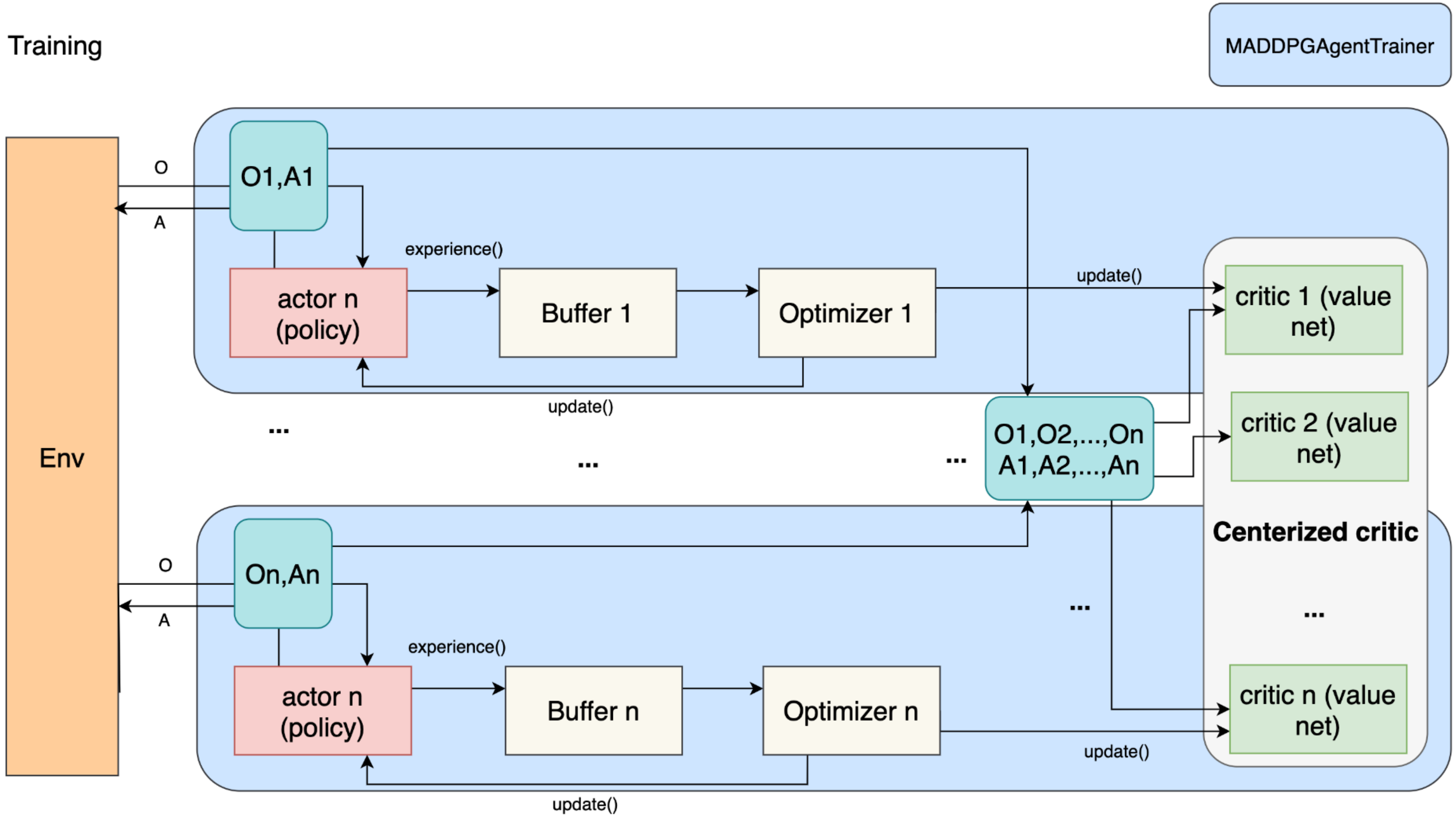
- Challenge
  - Non-stationarity of the environment
  - Policy gradient suffers from a variance that increases as the number of agents grows
- Main idea
  - Leads to learned policies that only use local information at execution time and allow the policies to use extra information to ease training
  - Does not assume a differentiable model of the environment dynamics or any particular structure on the communication method between agents
  - Is applicable not only to cooperative interaction but to competitive or mixed interaction involving both physical and communicative behavior

Lowe, Ryan, et al. "Multi-agent actor-critic for mixed cooperative-competitive environments." *Advances in neural information processing systems* 30 (2017).

Multi-Agent RL-Learning Cooperation

MADDPG

- Simple extension of actor-critic policy gradient methods where critic is augmented with extra information about the policies of other agents, while the actor only has access to local information



Multi-Agent RL-Learning Cooperation

MADDPG

- The gradient of the expected return for agent i:

$$\nabla_{\theta_i} J(\theta_i) = \mathbb{E}_{s\sim p^{\boldsymbol{\mu}}, a_i \sim \boldsymbol{\pi}_i}\left[\nabla_{\theta_i} \log \boldsymbol{\pi}_i(a_i \mid o_i) Q_i^{\boldsymbol{\pi}}(\mathbf{x}, a_1, \ldots, a_N)\right], \mathbf{x} = (o_1, \ldots, o_n)$$

- Deterministic policies:

$$\nabla_{\theta_i} J(\boldsymbol{\mu}_i) = \mathbb{E}_{\mathbf{x}, a\sim \mathcal{D}}\left[\nabla_{\theta_i} \boldsymbol{\mu}_i(a_i \mid o_i) \nabla_{a_i} Q_i^{\boldsymbol{\mu}}(\mathbf{x}, a_1, \ldots, a_N)\Big|_{a_i = \boldsymbol{\mu}_i(o_i)}\right]$$

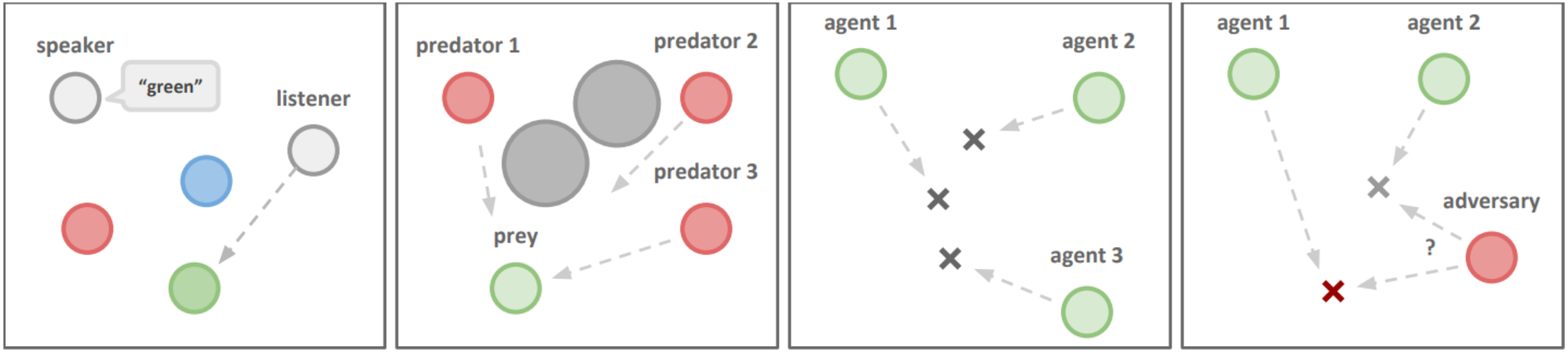
$$\mathcal{L}(\theta_i) = \mathbb{E}_{\mathbf{x}, a, r, \mathbf{x}'}\left[(Q_i^{\boldsymbol{\mu}}(\mathbf{x}, a_1, \ldots, a_N) - y)^2\right], y = r_i + \gamma Q_i^{\boldsymbol{\mu}'}(\mathbf{x}', a_1', \ldots, a_N') \mid a_j' = \boldsymbol{\mu}_j'(o_j)$$

Lowe, Ryan, et al. "Multi-agent actor-critic for mixed cooperative-competitive environments." *Advances in neural information processing systems* 30 (2017).

Multi-Agent RL-Learning Cooperation

MADDPG

- Multiagent-particle-environment
  - Cooperative Communication
  - Predator-Prey
  - Cooperative Navigation
  - Physical Deception
- The environments are publicly available: https://github.com/openai/multiagent-particle-envs

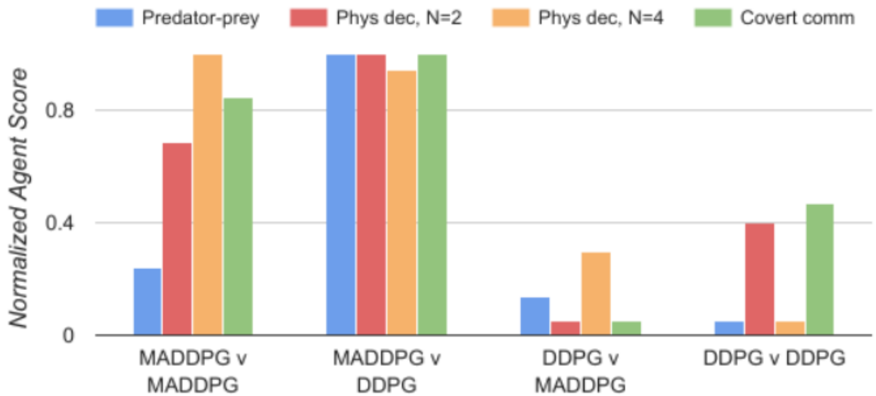


Lowe, Ryan, et al. "Multi-agent actor-critic for mixed cooperative-competitive environments." *Advances in neural information processing systems* 30 (2017).

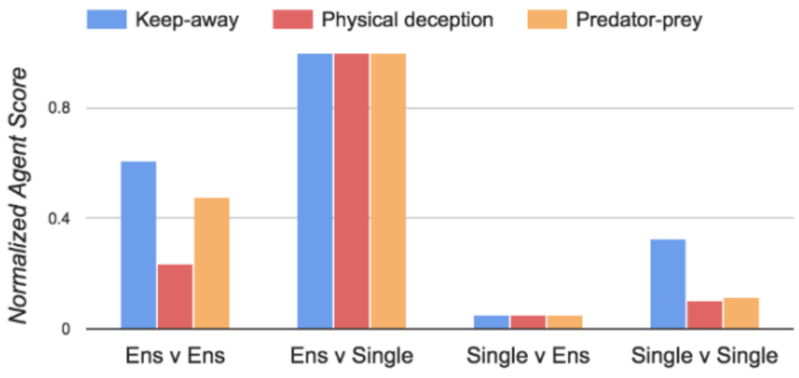
Multi-Agent RL-Learning Cooperation

MADDPG

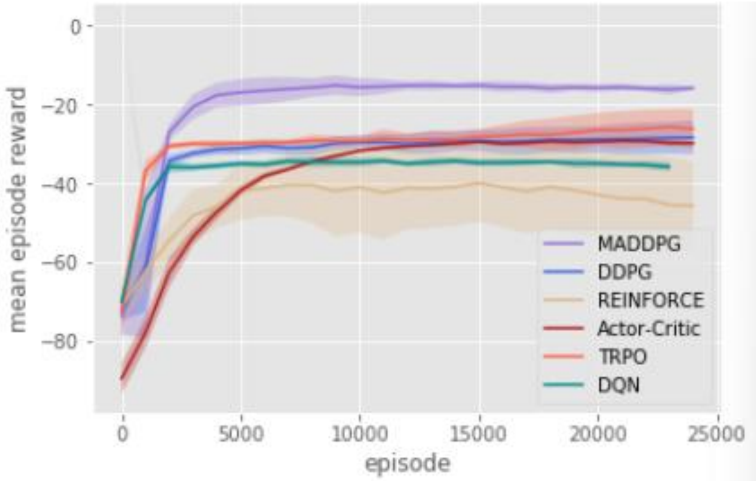
- Performance



Comparison of MADDPG and DDPG (Agent v Adversary)



Comparison of Ensemble and Single Policies (Agent v Adversary)



Lowe, Ryan, et al. "Multi-agent actor-critic for mixed cooperative-competitive environments." *Advances in neural information processing systems* 30 (2017).

Multi-Agent RL-Learning Cooperation

VDN

- Challenge
  - Lazy agent: one agent learns a useful policy, but a second agent is discouraged from learning because its exploration would hinder the first agent and lead to worse team reward.
  - Large-scale multi-agent scenarios.
- Main idea
  - Training individual agents with a novel value decomposition network architecture, which learns to decompose the team value function into agent-wise value functions
  - The value decomposition network aims to learn an optimal linear value decomposition from the team reward signal, by back-propagating the total Q gradient through deep neural networks representing the individual component value functions.

Sunehag, Peter, et al. "Value-decomposition networks for cooperative multi-agent learning." *arXiv preprint arXiv:1706.05296* (2017).

Multi-Agent RL-Learning Cooperation

VDN

- Architecture

SUM

Joint action-value function can be additively decomposed into value functions across agents:

$$Q((h^1,\ldots,h^d),(a^1,\ldots,a^d)) \approx \sum_{i=1}^{d} \tilde{Q}_i(h^i,a^i)$$

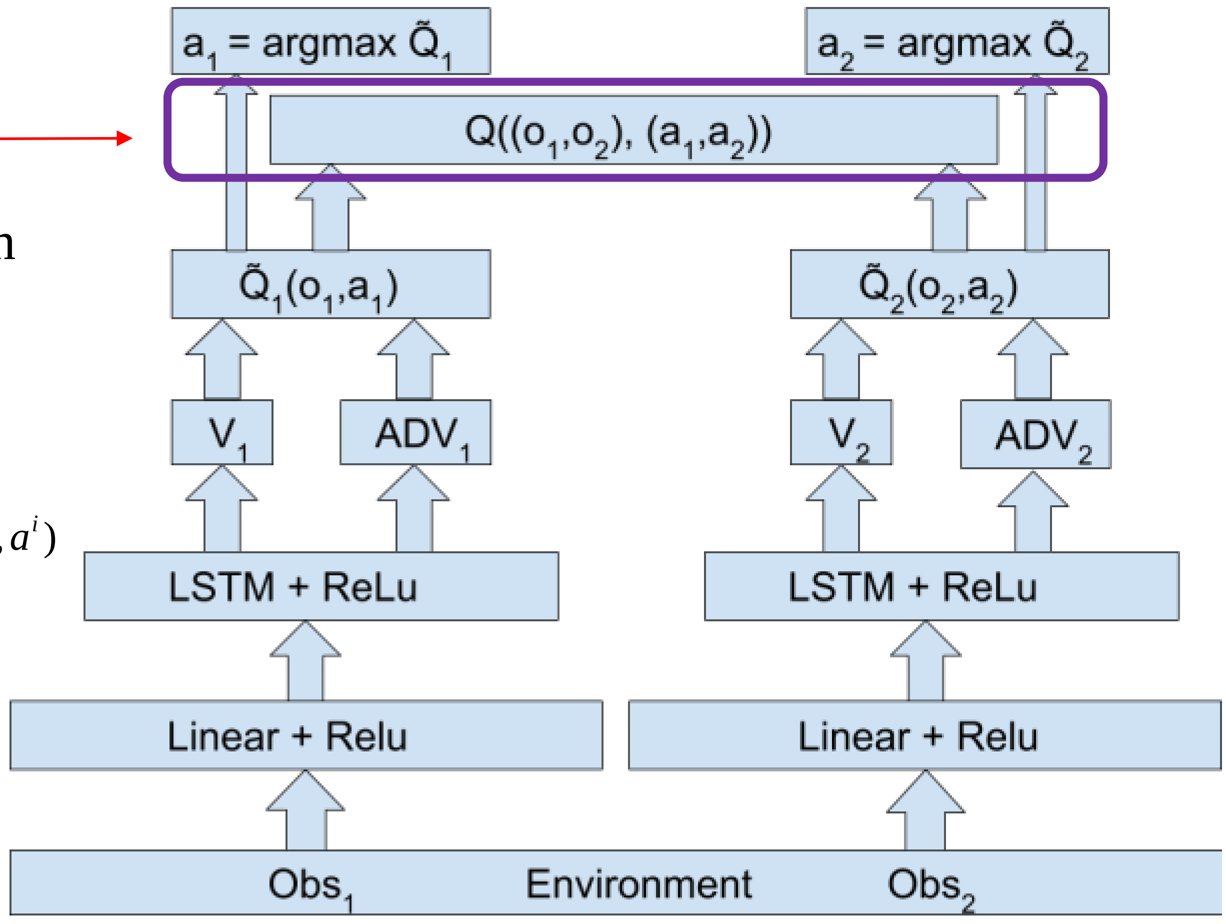


Figure 15: Value-Decomposition Individual Architecture

Sunehag, Peter, et al. "Value-decomposition networks for cooperative multi-agent learning." *arXiv preprint arXiv:1706.05296* (2017).

Multi-Agent RL-Learning Cooperation

VDN

- Performance

Agent	V.	S.	Id	L.	H.	C.
1						
2	✓					
3	✓	✓				
4	✓	✓	✓			
5	✓	✓	✓	✓		
6	✓	✓	✓		✓	
7	✓	✓	✓	✓	✓	
8	✓					✓
9						✓

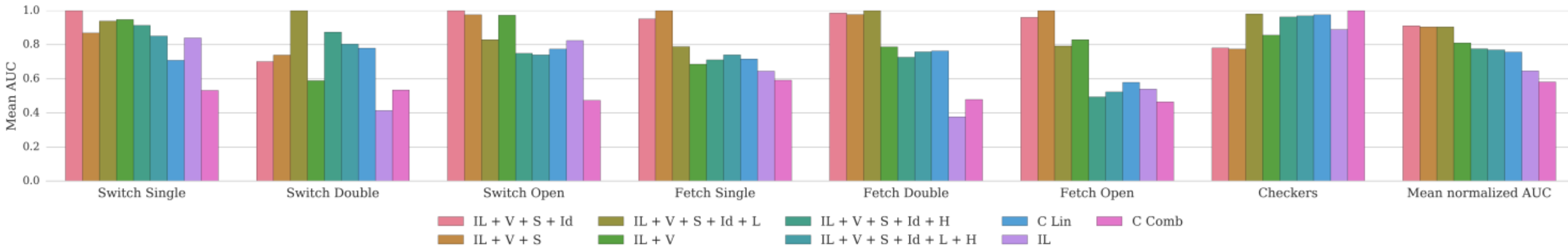
Table 1: Agent architectures. V is value decomposition, S means shared weights and an invariant network, Id means role info was provided, L stands for lower-level communication, H for higher-level communication and C for centralization. These architectures were selected to show the advantages of the independent agent with value-decomposition and to study the benefits of additional enhancements added in a logical sequence.

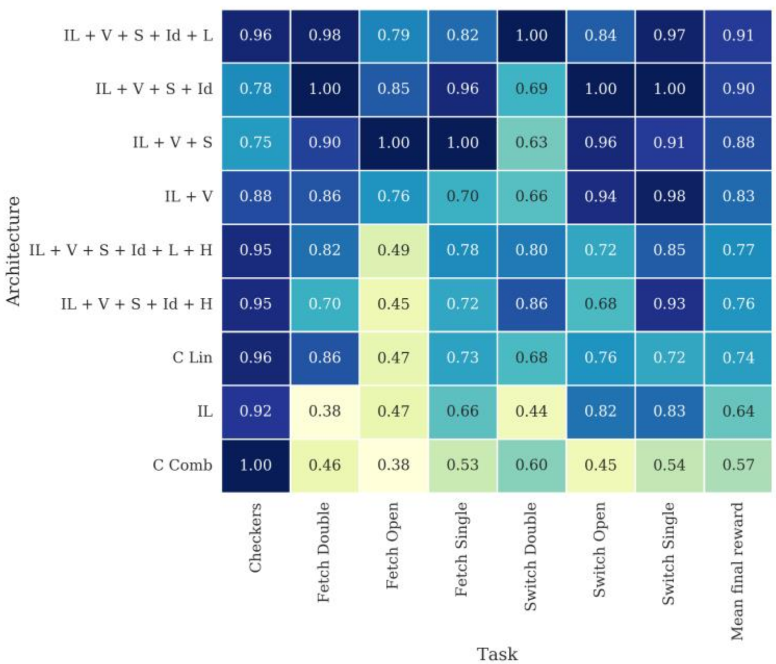
Sunehag, Peter, et al. "Value-decomposition networks for cooperative multi-agent learning." *arXiv preprint arXiv:1706.05296* (2017).

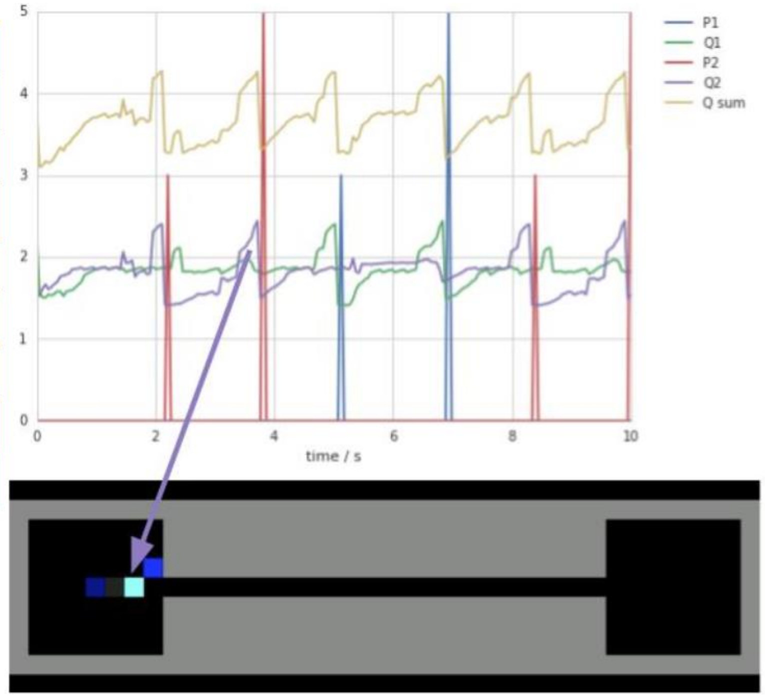
Multi-Agent RL-Learning Cooperation

VDN

- Performance







Sunehag, Peter, et al. "Value-decomposition networks for cooperative multi-agent learning." *arXiv preprint arXiv:1706.05296* (2017).