This lecture will be recorded!!!

Welcome to

DS595
Reinforcement Learning

Prof. Yanhua Li

Time: 6:00pm –8:50pm W
Zoom Lecture
Spring 2022

Project 3 leader board

Top	Date	Name	Score
1	4/6/2022	Hongchao Zhang	128
2	4/6/2022	Puru Upadhyay	82
3	4/6/2022	Khai Yi Chin	81
3	4/6/2022	Karter Krueger	81
5	4/6/2022	Sailesh Rajagopalan	78
5	4/6/2022	Steven Hyland	78
7	4/6/2022	Yiran Fang	74
8	4/6/2022	Zhentian Qian	67
9	4/6/2022	Anujay Sharma	66

Project 4 Progressive report due today

- https://github.com/yingxue-zhang/DS595-RL-Projects/tree/master/Project4
- *Important Dates:*
- Progressive report: April 13, 2022, *Up to 5 pages, to be submitted to Canvas discussion board.*
- Final Project:
  - Monday 4/25/2022 team project report is due
  - Wed 4/27/2022 Virtual Poster Session

This Lecture

- Advanced RL Techniques
  - Sparse Reward
    - Reward shaping, Curiosity module
    - Curriculum learning,
    - Hierarchical RL
- Multi-Agent RL
  - Intro
  - Multi-Agent RL principle
  - Multi-Agent Deep Deterministic Policy Gradient (MA-DDPG)
- Project #4 progress update

	Reinforcement Learning	Inverse Reinforcement Learning
Single Agent	**Tabular representation of reward** *Model-based control* *Model-free control* *(MC, SARSA, Q-Learning)*	**Linear reward function learning** Imitation learning Apprenticeship learning Inverse reinforcement learning MaxEnt IRL MaxCausalEnt IRL MaxRelEnt IRL
	Function representation of reward *1. Linear value function approx (MC, SARSA, Q-Learning)* *2. Value function approximation (Deep Q-Learning, Double DQN, prioritized DQN, Dueling DQN)* *3. Policy function approximation (Policy gradient*, PPO, TRPO) 4. Actor-Critic methods (A2C, A3C, Pathwise Derivative PG) Advanced topics in RL（Sparse Rewards）	**Non-linear reward function learning** Generative adversarial imitation learning (GAIL) Adversarial inverse reinforcement learning (AIRL)
	Review of Deep Learning *As bases for non-linear function approximation (used in 2-4).*	Review of Generative Adversarial nets As bases for non-linear IRL
Multiple Agents	**Multi-Agent Reinforcement Learning** Multi-agent Actor-Critic etc.	**Multi-Agent Inverse Reinforcement Learning** MA-GAIL MA-AIRL AMA-GAIL

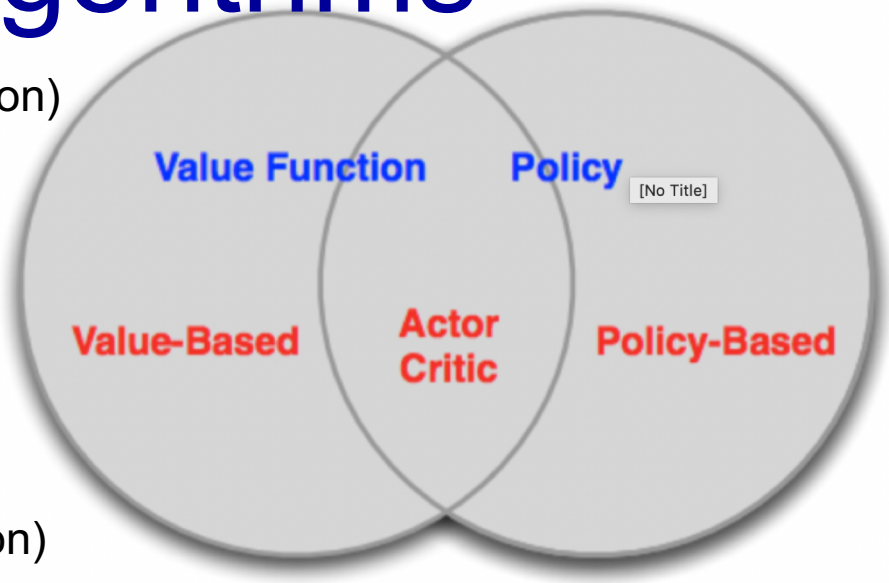
Applications

This Lecture

- Advanced RL Techniques
  - Sparse Reward
    - Reward shaping, Curiosity module
    - Curriculum learning,
    - Hierarchical RL
- Multi-Agent RL
  - Intro
  - Multi-Agent RL principle
  - Multi-Agent Deep Deterministic Policy Gradient (MA-DDPG)
- Project #4 progress update

Model-free RL Algorithms

- Value-based (Learned Value Function)

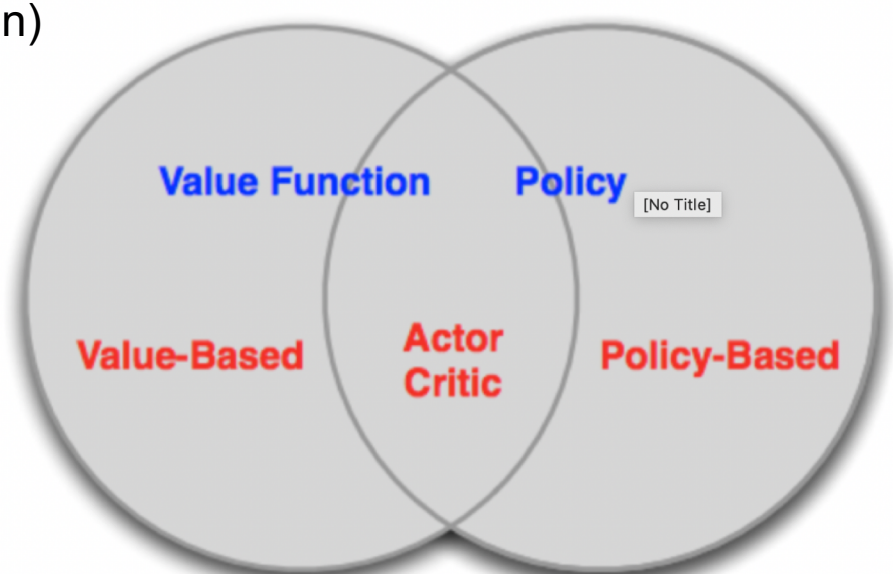


- Policy-based (Learned Policy Function)

- Actor-Critic (Learned both Value and Policy Functions)

Model-free RL Algorithms

- Value-based (Learned Value Function)
  - Deep Q-Learning (DQN)
  - Double DQN
  - Dueling DQN
  - Prioritized DQN Multi-step DQN,
  - Noisy net DQN
  - Distributional DQN
  - DQN for continuous action space
- Policy-based (Learned Policy Function)
  - Basic Policy Gradient Algorithm
  - REINFORCE
  - Vanilla, PPO, TRPO, PPO2
- Actor-Critic (Learned both Value and Policy Functions)
  - A2C
  - A3C
  - Pathwise Derivative Policy Gradient



Sparse Reward

Reward Shaping

Reward Shaping

Take "Play",

$r_{t+1} = 1, r_{t+100} = -100$

Take "Study",

~~$r_{t+1} = -1$~~, $r_{t+100} = 100$

$r_{t+1} = 1$

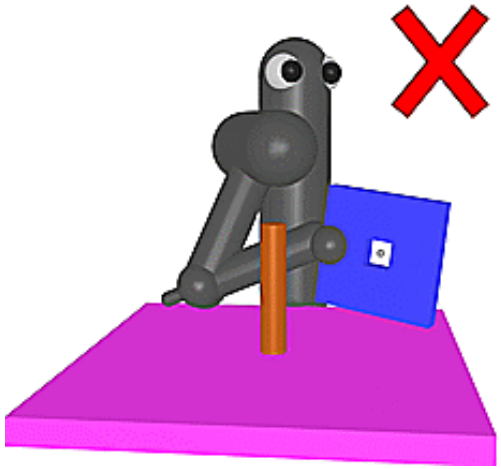
Reward Shaping

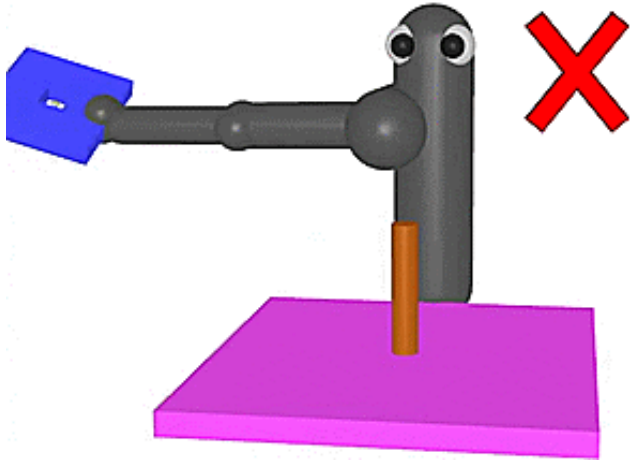
VizDoom

https://openreview.net/forum?id=Hk3mPK5gg&noteId=Hk3mPK5gg

Parameters	Description	FlatMap | CIGTrack1
living	Penalize agent who just lives	-0.008 / action
health_loss	Penalize health decrement	-0.05 / unit
ammo_loss	Penalize ammunition decrement	-0.04 / unit
health_pickup	Reward for medkit pickup	0.04 / unit
ammo_pickup	Reward for ammunition pickup	0.15 / unit
dist_penalty	Penalize the agent when it stays	-0.03 / action
dist_reward	Reward the agent when it moves	9e-5 / unit distance

Reward Shaping





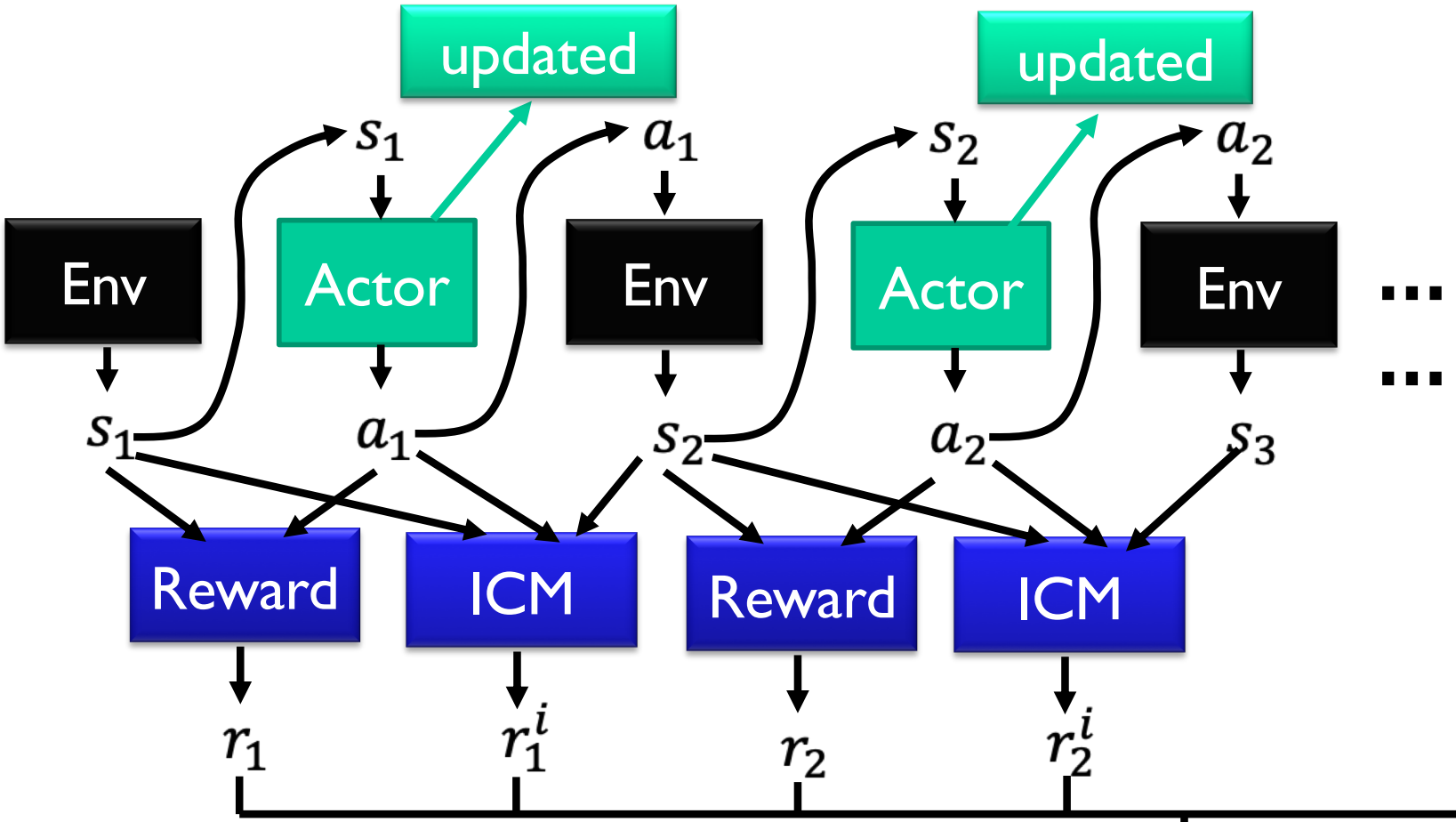
Get reward, when closer

Need domain knowledge

https://openreview.net/pdf?id=Hk3mPK5gg

Curiosity

https://arxiv.org/abs/1705.05363

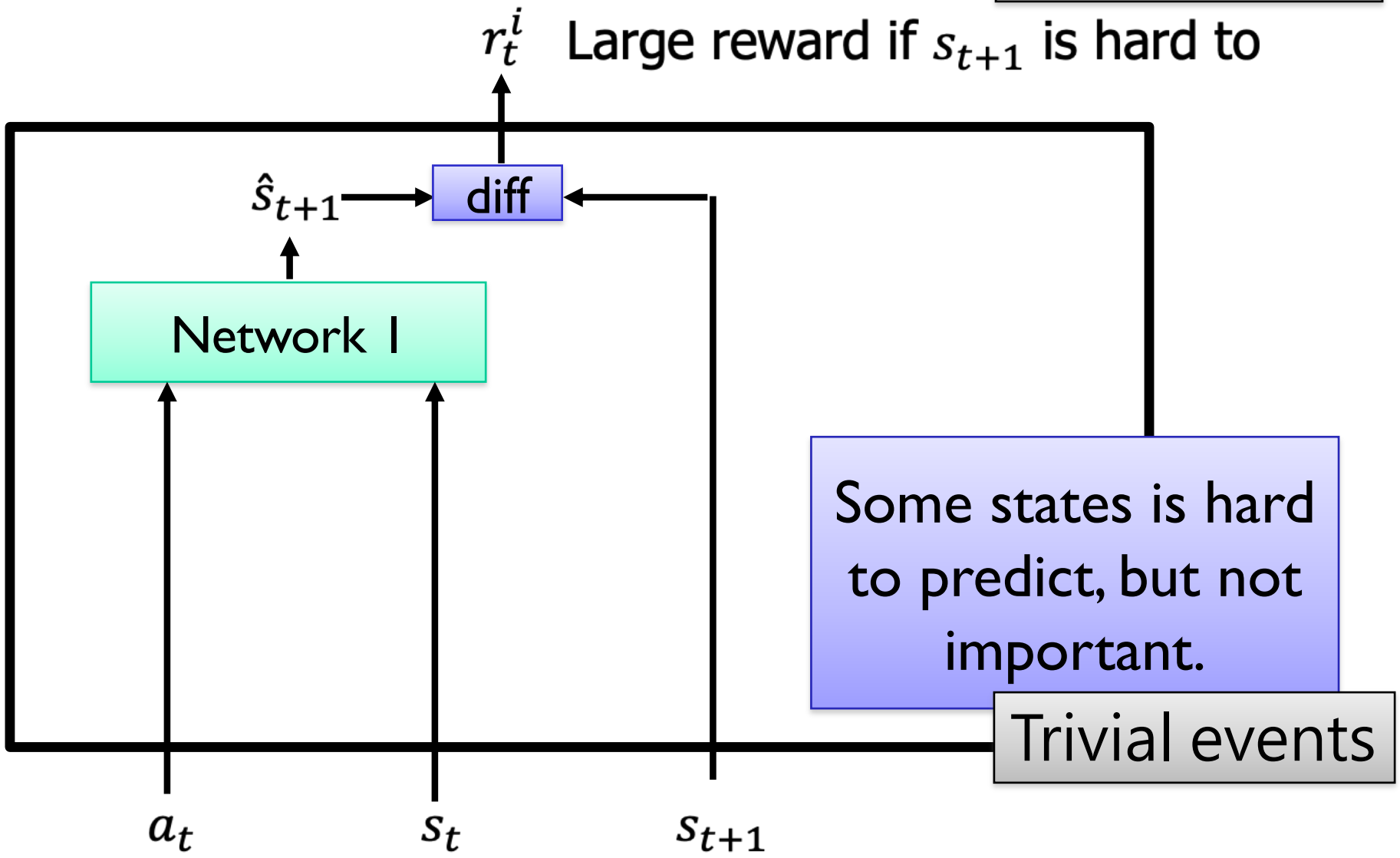


ICM = intrinsic curiosity module

$$R(\tau) = \sum_{t=1}^{T} r_t + r_t^i$$

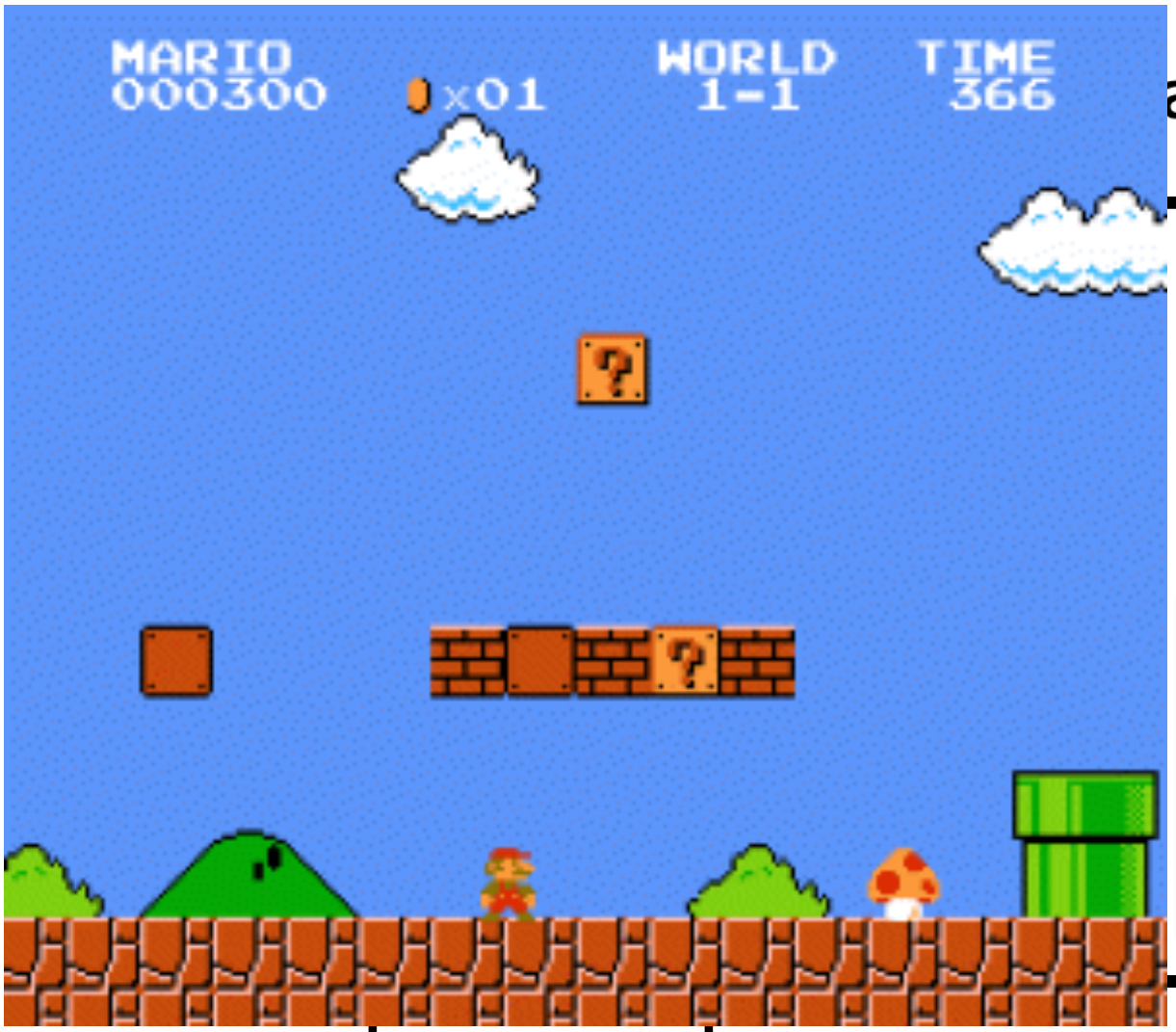
Intrinsic Curiosity Module

Encourage exploration



Intrinsic Curiosity Module

Encourage exploration



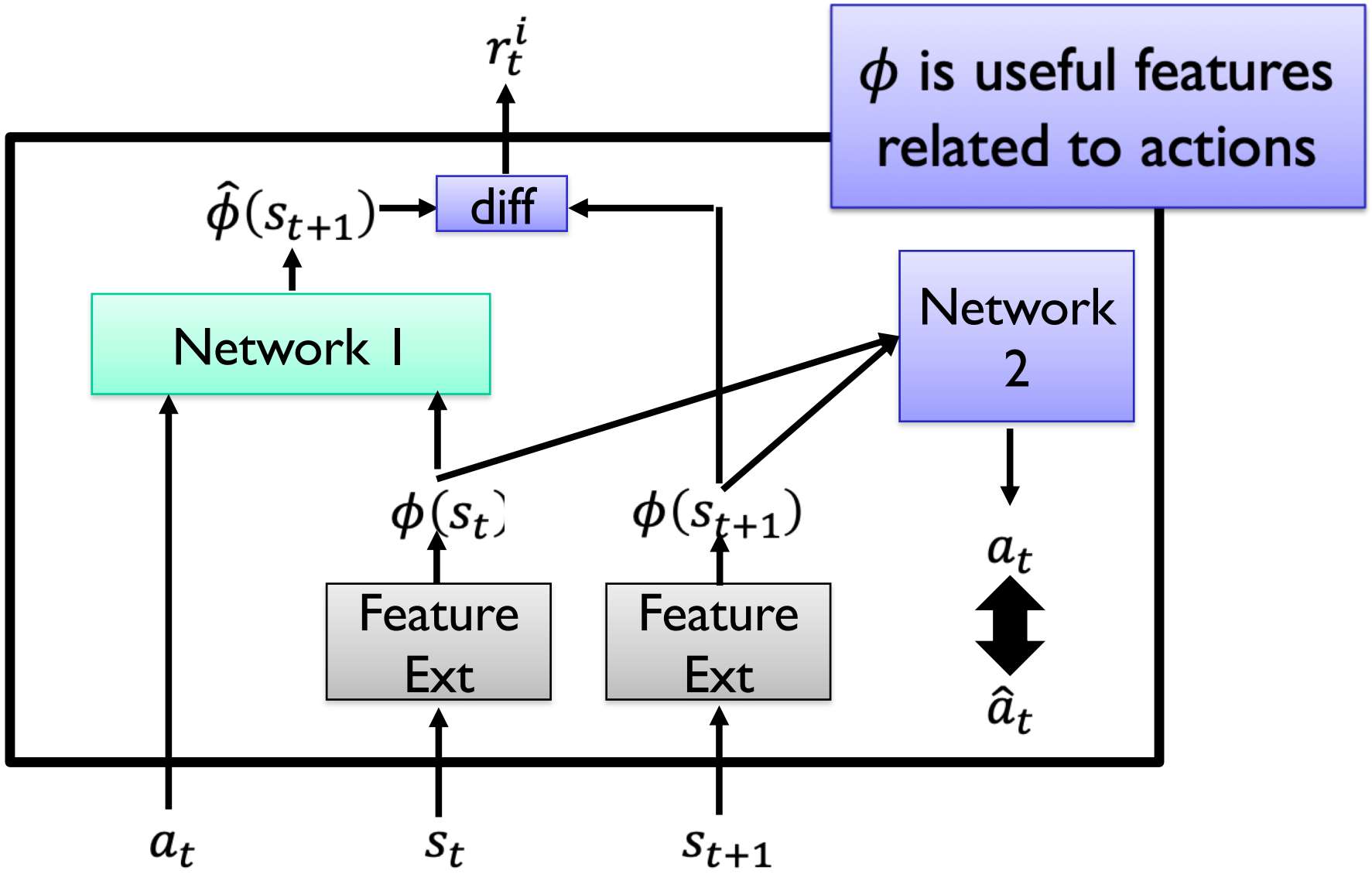
ard if s_{t+1} is hard to

Some states is hard to predict, but not important.

Trivial events

a_t s_t s_{t+1}

Intrinsic Curiosity Module



This Lecture

- Advanced RL Techniques
  - Sparse Reward
    - Reward shaping, Curiosity module
    - Curriculum learning,
    - Hierarchical RL
- Multi-Agent RL
  - Intro
  - Multi-Agent RL principle
  - Multi-Agent Deep Deterministic Policy Gradient (MA-DDPG)
- Project #4 progress update

Sparse Reward

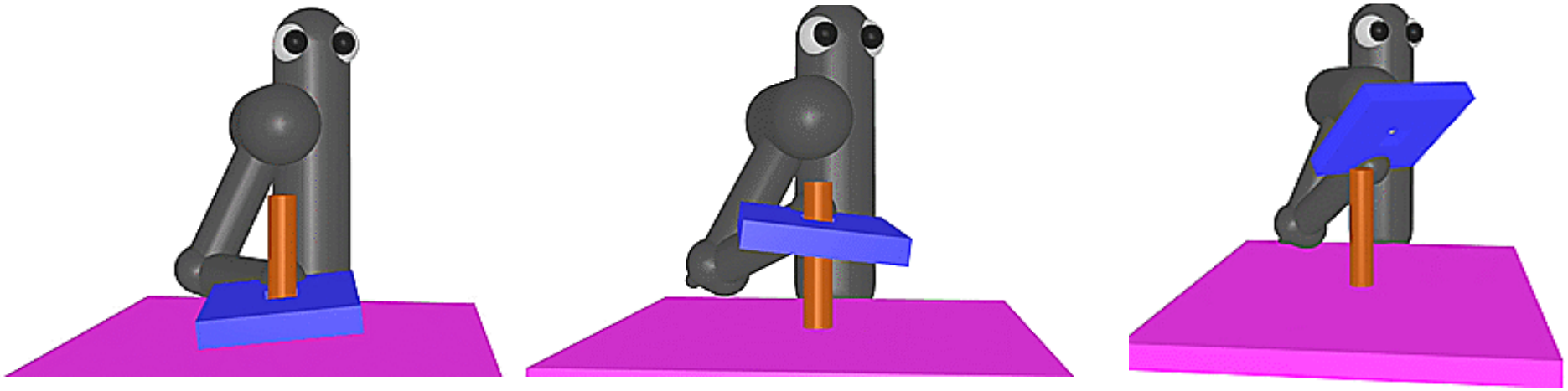
Curriculum Learning

Curriculum Learning

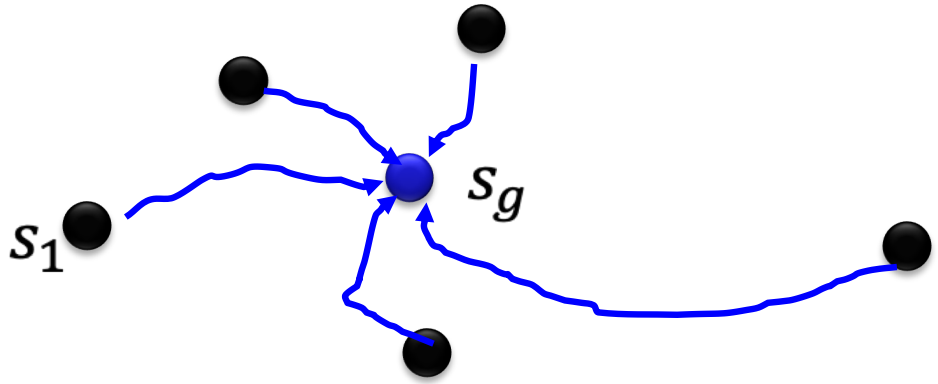
- Starting from simple training examples, and then becoming harder and harder.

Facebook wins the VizDoom competition

	Class 0	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Class 7
Speed	0.2	0.2	0.4	0.4	0.6	0.8	0.8	1.0
Health	40	40	40	60	60	60	80	100

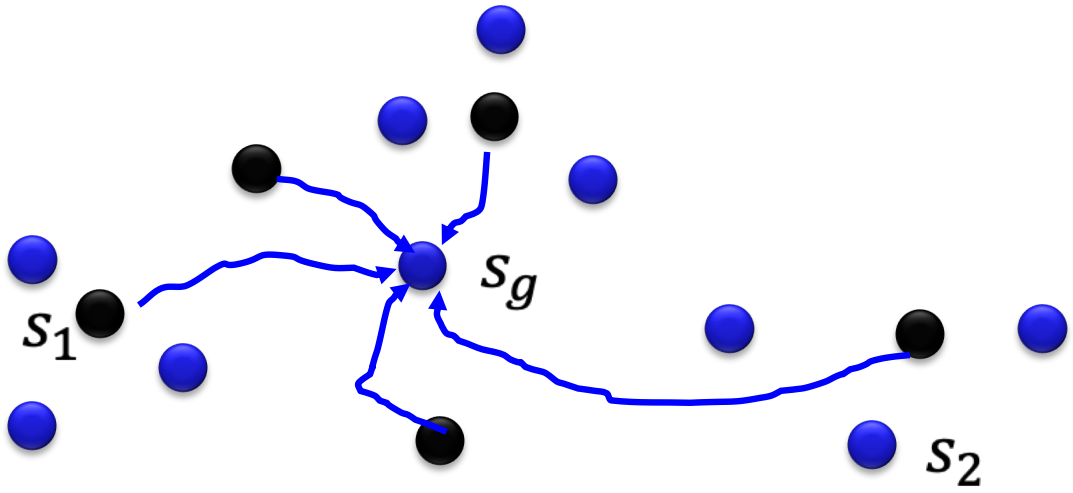


Reverse Curriculum Generation



- Given a goal state s_g.
- Sample some states s_1 "close" to s_g
- Start from states s_1, each trajectory has reward $\mathrm{R}(s_1)$

Reverse Curriculum Generation



- Delete s_1 whose reward is too large (already learned) or too small (too difficult at this moment)
- Sample s_2 from s_1, start from s_2

This Lecture

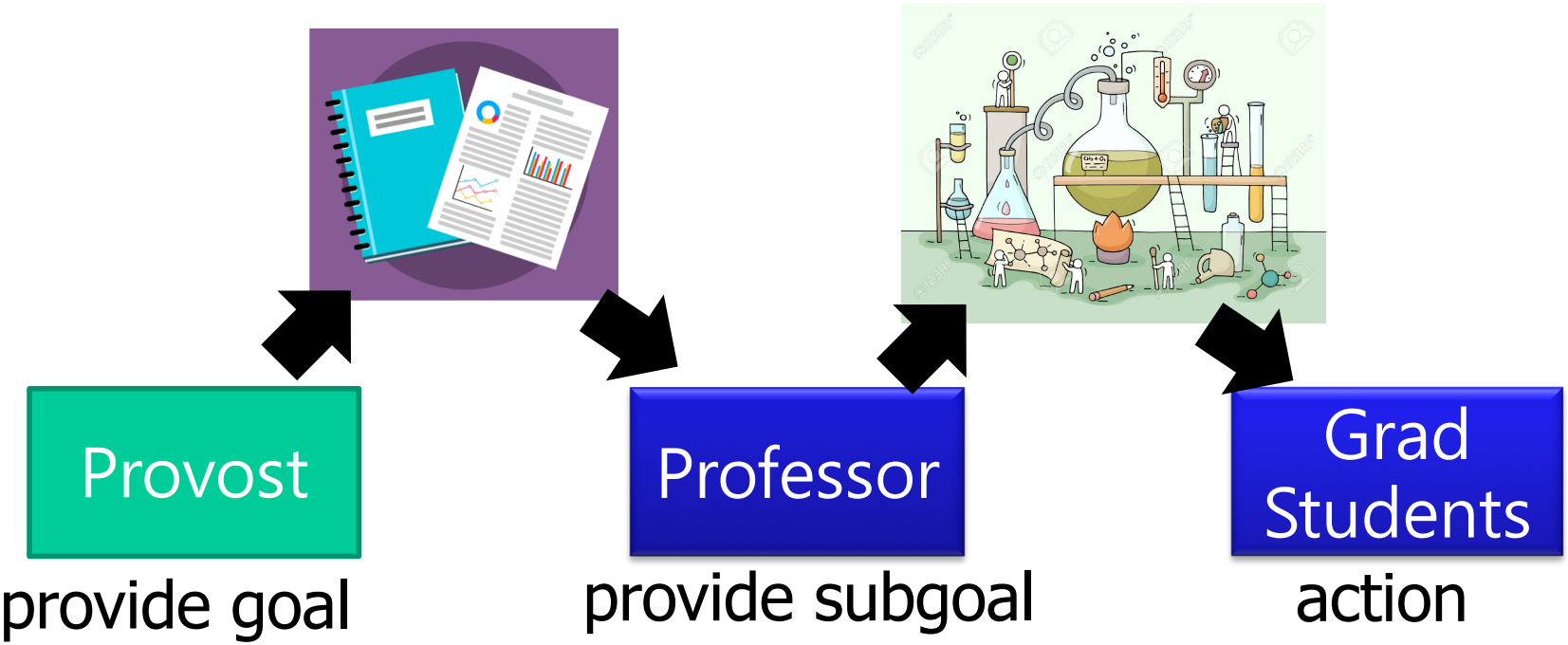
- Advanced RL Techniques
  - Sparse Reward
    - Reward shaping, Curiosity module
    - Curriculum learning,
    - Hierarchical RL
- Multi-Agent RL
  - Intro
  - Multi-Agent RL principle
  - Multi-Agent Deep Deterministic Policy Gradient (MA-DDPG)
- Project #4 progress update

Sparse Reward

Hierarchical Reinforcement Learning

Hierarchical RL

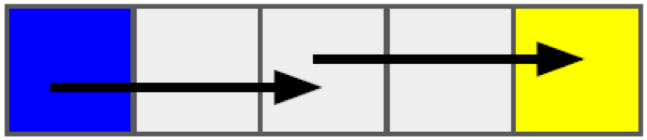
This is a fake example.
Don't take it seriously.



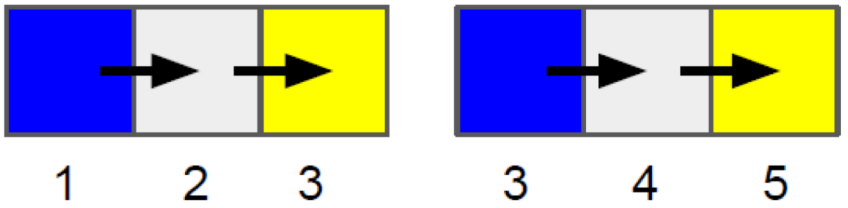
- If lower agent cannot achieve the goal, the upper agent would get penalty.

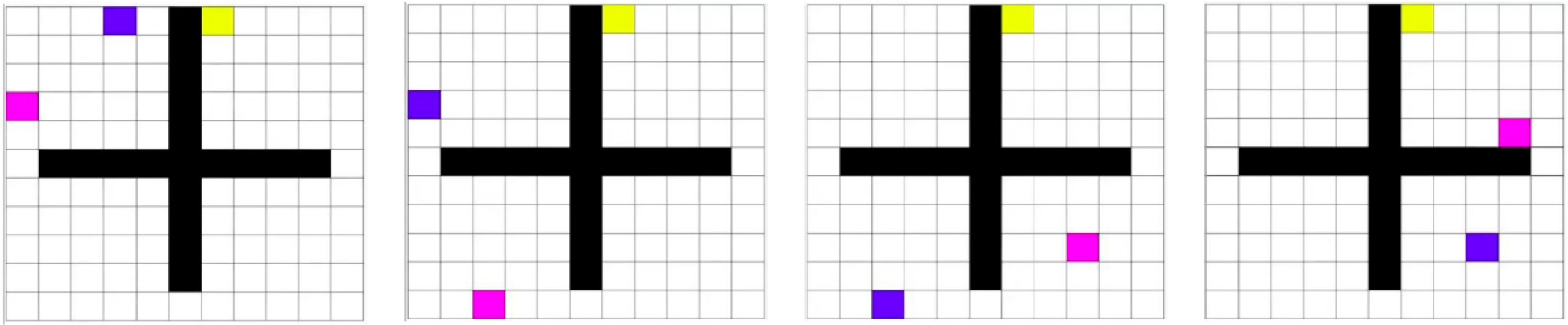
https://arxiv.org/abs/1805.08180

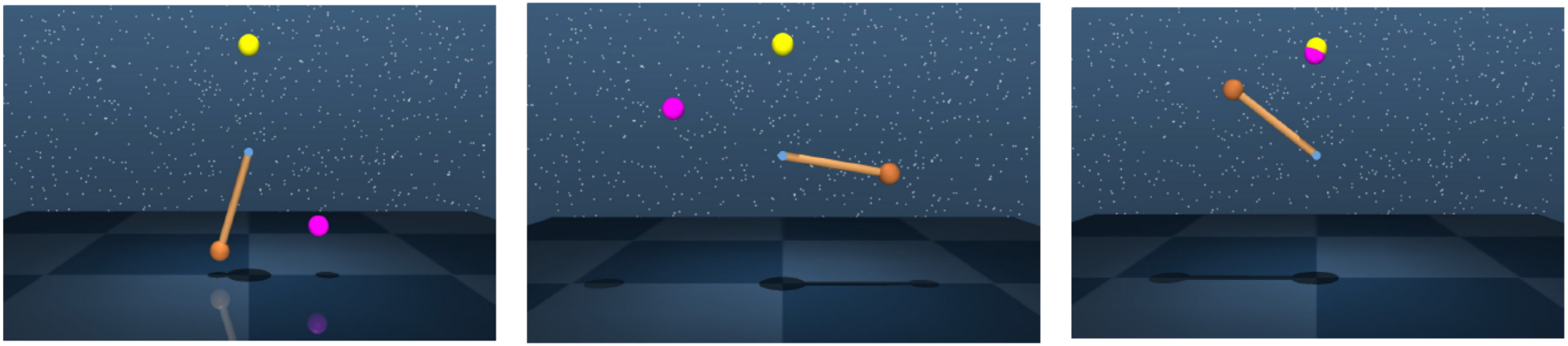
High-Level



Low-Level







https://arxiv.org/abs/1805.08180

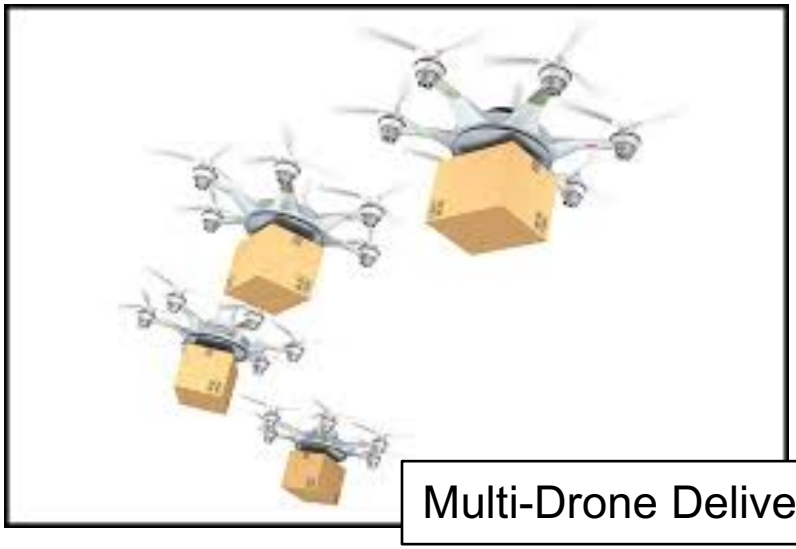
This Lecture

- Advanced RL Techniques
  - Sparse Reward
    - Reward shaping, Curiosity module
    - Curriculum learning,
    - Hierarchical RL
- Multi-Agent RL
  - Intro
  - Multi-Agent RL principle
  - Multi-Agent Deep Deterministic Policy Gradient (MA-DDPG)
- Project #4 progress update

Multi-Agent Systems are going to be everywhere

Home Robots

Autonomous Vehicles



Multi-Drone Delivery

Smart energy grid

New advances in Multi-Agent RL research - Hide-and-Seek

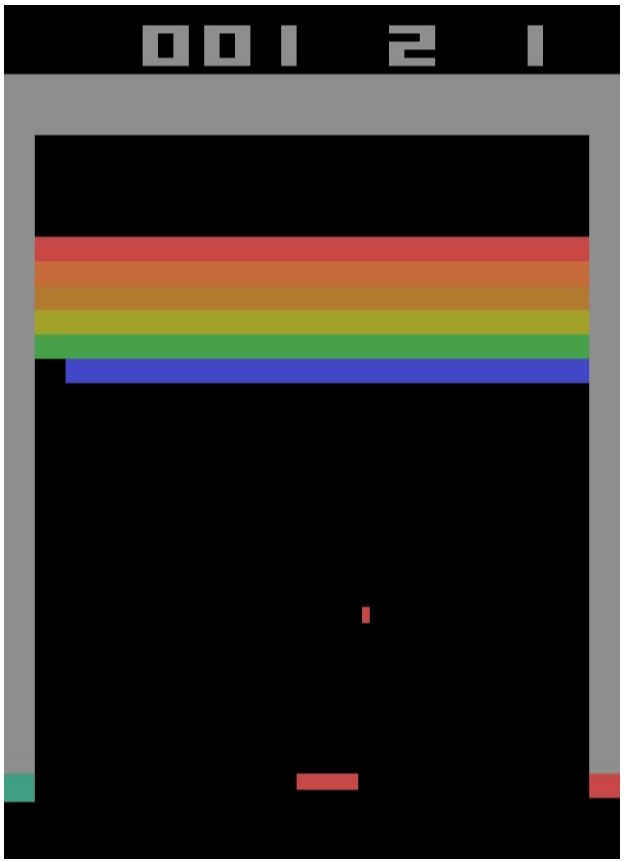
https://openai.com/blog/emergent-tool-use/ for more details.
Just released on 9/17/2019 by OpenAI.

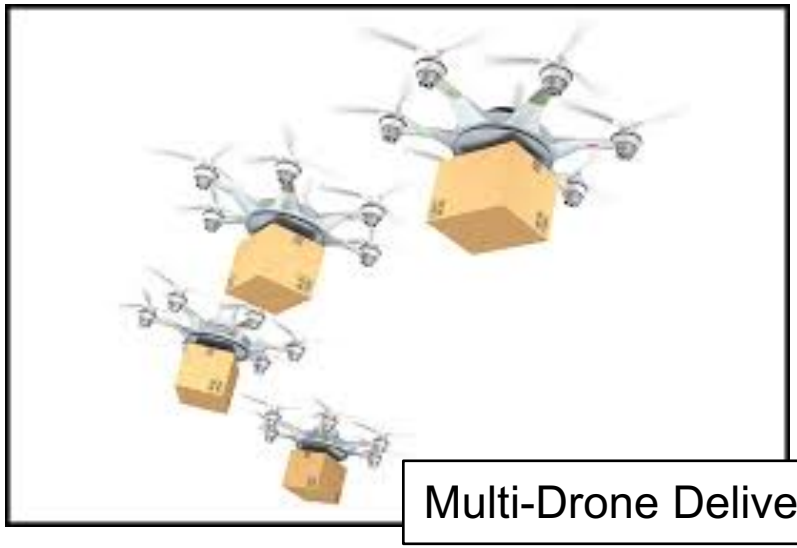
New advances in Multi-Agent RL research - DOTA

https://openai.com/blog/openai-five/ for more details.
Released by OpenAI on 6/25/2018.

Multi-Agent Reinforcement Learning: Naïve Methods

- Each agent learns independently, by viewing all other agents and the system as the environment.





Multi-Drone Delivery

Multi-Agent Reinforcement Learning: Naïve Methods

Each individual agent runs an RL algorithm, separately:

- Policy Gradient
- DQN
- Actor-Critic

Issues with the Naïve Methods

- Single agent
  - Model: Markov Decision Process (MDPs)
  - Goal: Maximize reward
  - Steady state: Maximized reward
  - Agents dependency: Independent
- Multiple agents
  - Model: Markov Games (MGs)
  - Goal: Maximize its own/or team reward
  - Steady state: Nash Equilibrium
  - Agents dependency: no agents can achieve a higher expected reward by unilaterally changing its own policy

Dependency across agents are missing!

This Lecture

- Advanced RL Techniques
  - Sparse Reward
    - Reward shaping, Curiosity module
    - Curriculum learning,
    - Hierarchical RL
- Multi-Agent RL
  - Intro
  - Multi-Agent RL principle
  - Multi-Agent Deep Deterministic Policy Gradient (MA-DDPG)
- Project #4 progress update

MDPs vs MGs

- MDP: $<S, A, P, \gamma, R>$
  - State set: S
  - *Action set:* A
  - *Transition Probability:* $P(s'|s, a)$
  - *Discount factor* γ
  - *Reward function:* R

    $R : S \times A \longmapsto \mathbb{R}$

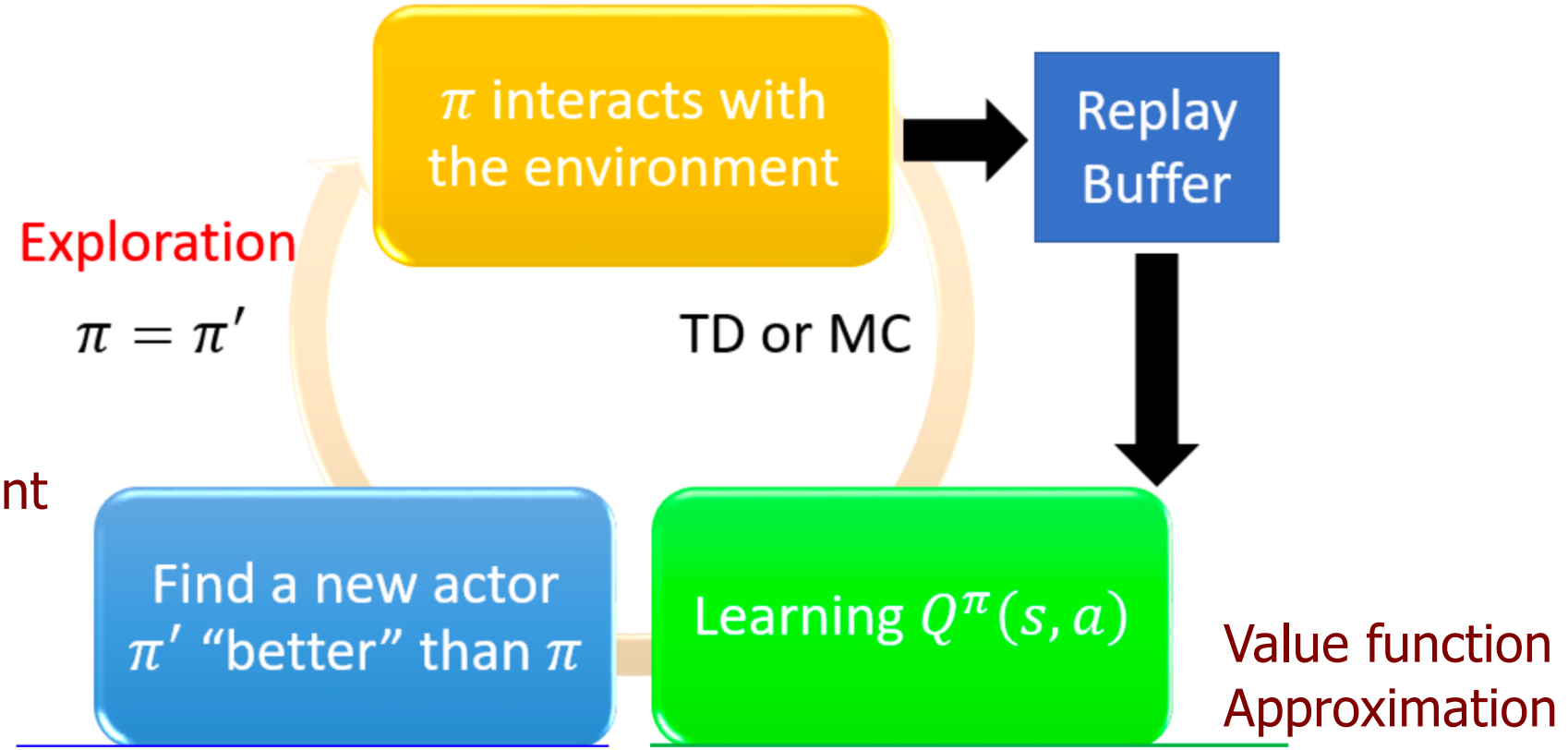
- MG: $<S, \{A_i\}, P, \gamma, \{R_i\}>$
  - State set: S *(joint states)*
  - *Action set:* $\{A_1,\ldots,A_N\}$
  - *Transition Probability:* $P(s'|s, a_1,\ldots,a_N)$
  - *Discount factor* γ
  - *Reward function:* $\{R_1,\ldots,R_N\}$

    $R_i : S \times A_i \longmapsto \mathbb{R}$

A2C algorithm

Updating each $\boldsymbol{w}$ by DQN

$$\Delta \boldsymbol{w} = -\alpha\left(r + \gamma \max_{a'} \hat{Q}(s', a'; \boldsymbol{w}) - \hat{Q}(s, a; \boldsymbol{w})\right)\nabla_{\boldsymbol{w}} \hat{Q}(s, a; \boldsymbol{w})$$



Policy Gradient

Updating θ by policy gradient

V(s) net:

$$\nabla \bar{R}_\theta = \frac{1}{N}\sum_{n=1}^{N}\sum_{t=1}^{\tau_n}\left(r^n{}_t + V(s^n{}_{t+1}) - V(s^n{}_t)\right)\nabla_\theta \log \pi(a^n{}_t|s^n{}_t)$$

Q(s,a) net:

$$\nabla \bar{R}_\theta = \frac{1}{N}\sum_{n=1}^{N}\sum_{t=1}^{\tau_n}\left(r^n{}_t + \max_{a^n{}_{t+1}} Q(s^n{}_{t+1}, a^n{}_{t+1}) - \max_{a^n{}_t} Q(s^n{}_t, a^n{}_t)\right)\nabla_\theta \log \pi(a^n{}_t|s^n{}_t)$$

Multi-Agent RL (Principle)

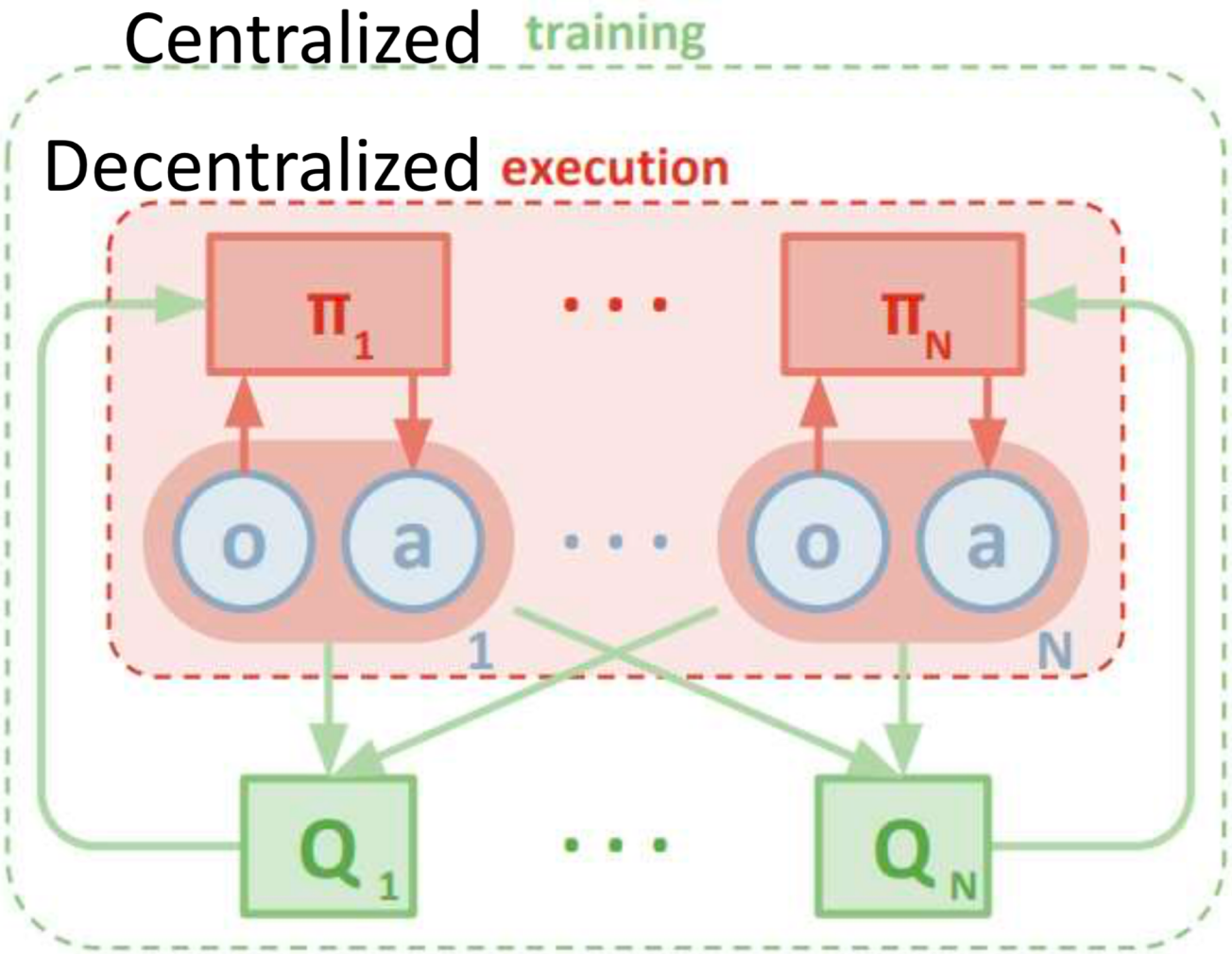
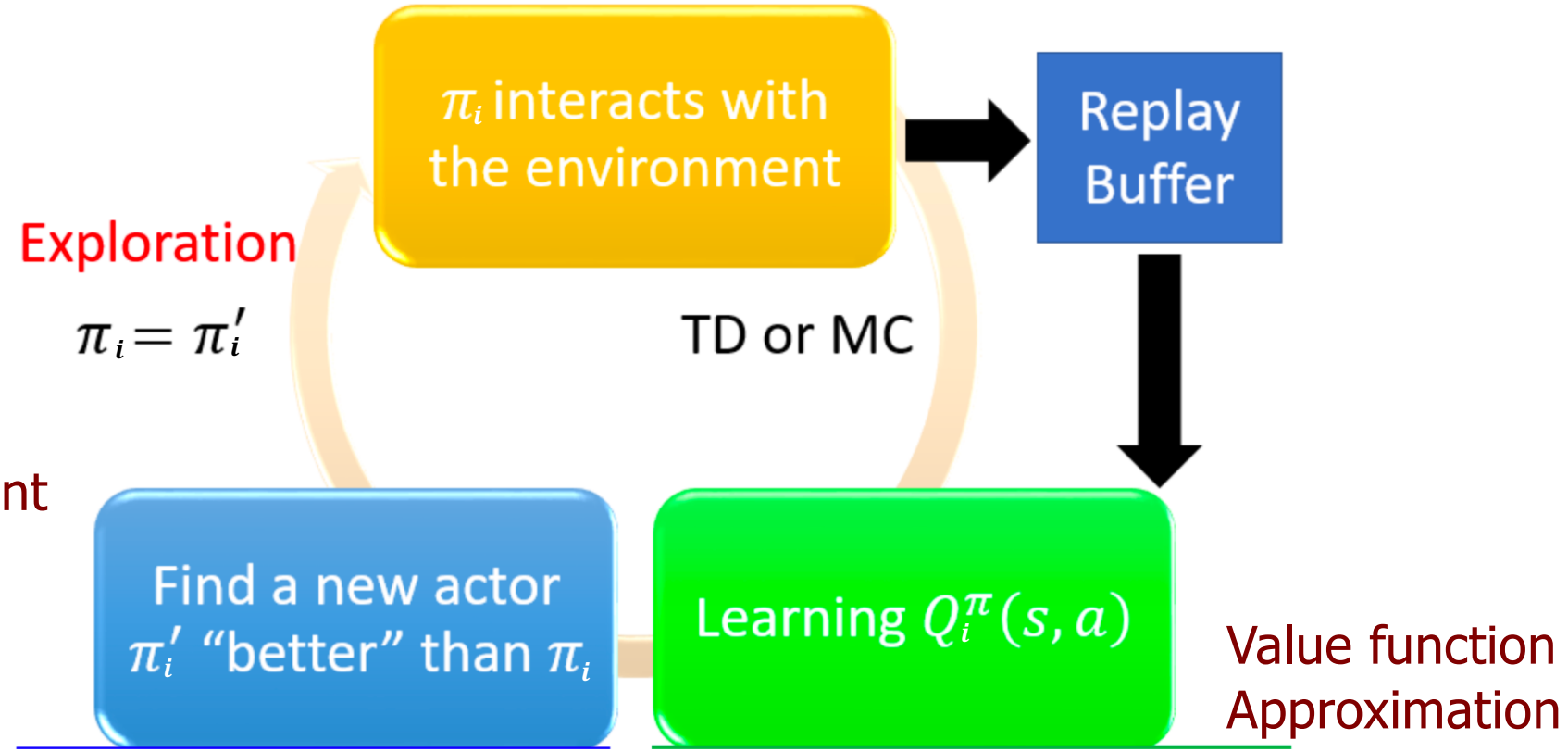


Figure Source: https://medium.com/brillio-data-science/improving-openai-multi-agent-actor-critic-rl-algorithm-27719f3cafd4

Multi-Agent RL

Updating each w_i by DQN

$$\Delta w_i = -\alpha\left(r + \gamma \max_{a'_i} \hat{Q}_i(s', a'; w_i) - \hat{Q}(s, a; w_i)\right)\nabla_{w_i}\hat{Q}_i(s, a; w_i)$$



Updating each θ_i by policy gradient

$$\nabla \bar{R}_{\theta_i} = \frac{1}{N}\sum_{n=1}^{N}\sum_{t=1}^{\tau_n}\left(r^n{}_t + \max_{a^n{}_{i,t+1}} Q_i(s^n{}_{t+1}, \boldsymbol{a}^n{}_{t+1}) - \max_{a^n{}_{i,t}} Q_i(s^n{}_t, \boldsymbol{a}^n{}_t)\right)\nabla_\theta \log \pi_i(a^n{}_{i,t}|s^n{}_t)$$

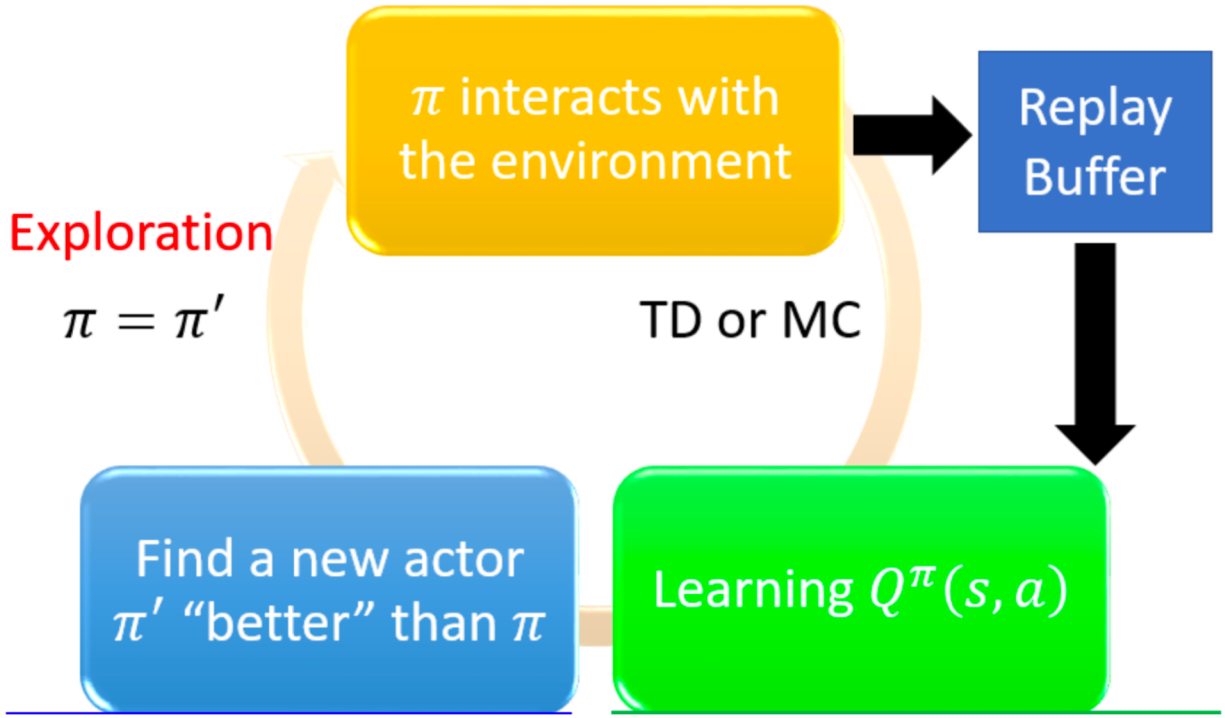
This Lecture

- Advanced RL Techniques
  - Sparse Reward
    - Reward shaping, Curiosity module
    - Curriculum learning,
    - Hierarchical RL
- Multi-Agent RL
  - Intro
  - Multi-Agent RL principle
  - Multi-Agent Deep Deterministic Policy Gradient (MA-DDPG)
- Project #4 progress update

DDPG: Deep Deterministic Policy Gradient

Updating each $\boldsymbol{w}$ by DQN

$$\Delta \boldsymbol{w} = -\alpha\left(r + \gamma \max_{a'} \hat{Q}(s', a'; \boldsymbol{w}) - \hat{Q}(s, a; \boldsymbol{w})\right)\nabla_{\boldsymbol{w}} \hat{Q}(s, a; \boldsymbol{w})$$



Stochastic

$\pi(a^n{}_t | s^n{}_t)$

$a^n{}_t = \mu(s^n{}_t)$

Deterministic

Updating θ by policy gradient

$$\nabla \bar{R}_\theta = \frac{1}{N}\sum_{n=1}^{N}\sum_{t=1}^{\tau_n}\left(r^n{}_t + \max_{a^n{}_{t+1}} Q(s^n{}_{t+1}, a^n{}_{t+1}) - \max_{a^n{}_t} Q(s^n{}_t, a^n{}_t)\right)\nabla_\theta \log \pi(a^n{}_t | s^n{}_t)$$

with $\bar{R}_\theta = \mathbb{E}_{s\sim P, a\sim\pi}[Q(s,a) - V(s)] = \mathbb{E}_{s\sim P, a\sim\pi}[A(s,a)]$.

$\nabla \bar{R}_\theta = \frac{1}{N}\sum_{n=1}^{N}\sum_{t=1}^{\tau_n} \nabla_{a^n{}_t} Q(s^n{}_t, a^n{}_t)\nabla_\theta \mu_\theta(s^n{}_t)$, with $\bar{R}_\theta = \mathbb{E}_{s\sim P, a\sim\mu}[Q(s,a)]$

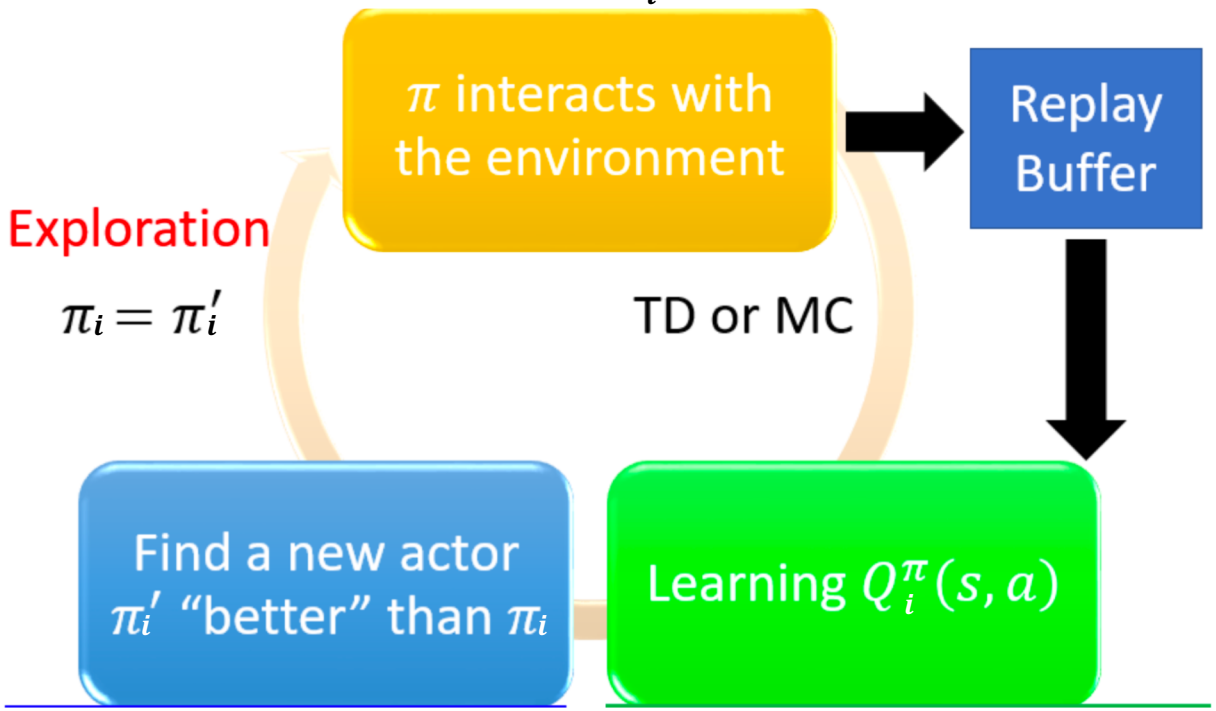
https://arxiv.org/pdf/1509.02971.pdf

Multi-Agent DDPG

By OpenAI

Updating each $\boldsymbol{w_i}$ by DQN

$$\Delta \boldsymbol{w_i} = -\alpha\left(r + \gamma \max_{\boldsymbol{a'_i}} \hat{Q}_i(s', \boldsymbol{a'}; \boldsymbol{w_i}) - \hat{Q}(s, \boldsymbol{a}; \boldsymbol{w_i})\right) \nabla_{\boldsymbol{w_i}} \hat{Q}_i(s, \boldsymbol{a}; \boldsymbol{w_i})$$



Updating each θ_i by MA-DDPG

$$\nabla \bar{R}_{\theta_i} = \frac{1}{N} \sum_{n=1}^{N} \sum_{t=1}^{\tau_n} \nabla_{a^n{}_{i,t}} Q_i\left(s^n{}_t, \boldsymbol{a}^n{}_{i,t}\right) \nabla_{\theta_i} \mu_i\left(s^n{}_t\right)$$

This Lecture

- Advanced RL Techniques
  - Sparse Reward
    - Reward shaping, Curiosity module
    - Curriculum learning,
    - Hierarchical RL
- Multi-Agent RL
  - Intro
  - Multi-Agent RL principle
  - Multi-Agent Deep Deterministic Policy Gradient (MA-DDPG)
- Project #4 progress update

	Reinforcement Learning	Inverse Reinforcement Learning
Single Agent	**Tabular representation of reward** *Model-based control* *Model-free control* *(MC, SARSA, Q-Learning)*	**Linear reward function learning** Imitation learning Apprenticeship learning Inverse reinforcement learning MaxEnt IRL MaxCausalEnt IRL MaxRelEnt IRL
	Function representation of reward *1. Linear value function approx (MC, SARSA, Q-Learning)* *2. Value function approximation (Deep Q-Learning, Double DQN, prioritized DQN, Dueling DQN)* *3. Policy function approximation (Policy gradient*, PPO, TRPO) 4. Actor-Critic methods (A2C, A3C, Pathwise Derivative PG) Advanced topics in RL（Sparse Rewards）	**Non-linear reward function learning** Generative adversarial imitation learning (GAIL) Adversarial inverse reinforcement learning (AIRL)
	Review of Deep Learning *As bases for non-linear function approximation (used in 2-4).*	Review of Generative Adversarial nets As bases for non-linear IRL
Multiple Agents	**Multi-Agent Reinforcement Learning** Multi-agent Actor-Critic etc.	**Multi-Agent Inverse Reinforcement Learning** MA-GAIL MA-AIRL AMA-GAIL

Applications

Questions?