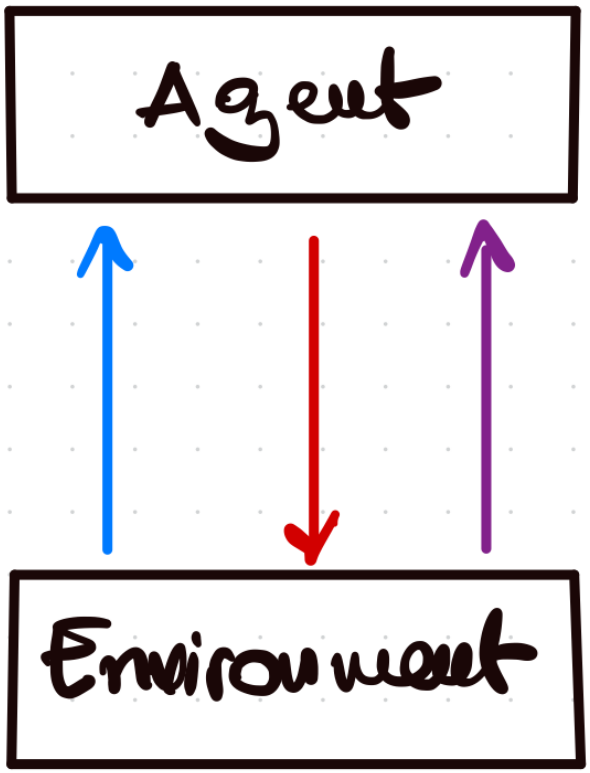
Reinforcement Learning

Multi-Agent Reinforcement Learning (MARL)

aka "Playing with others"

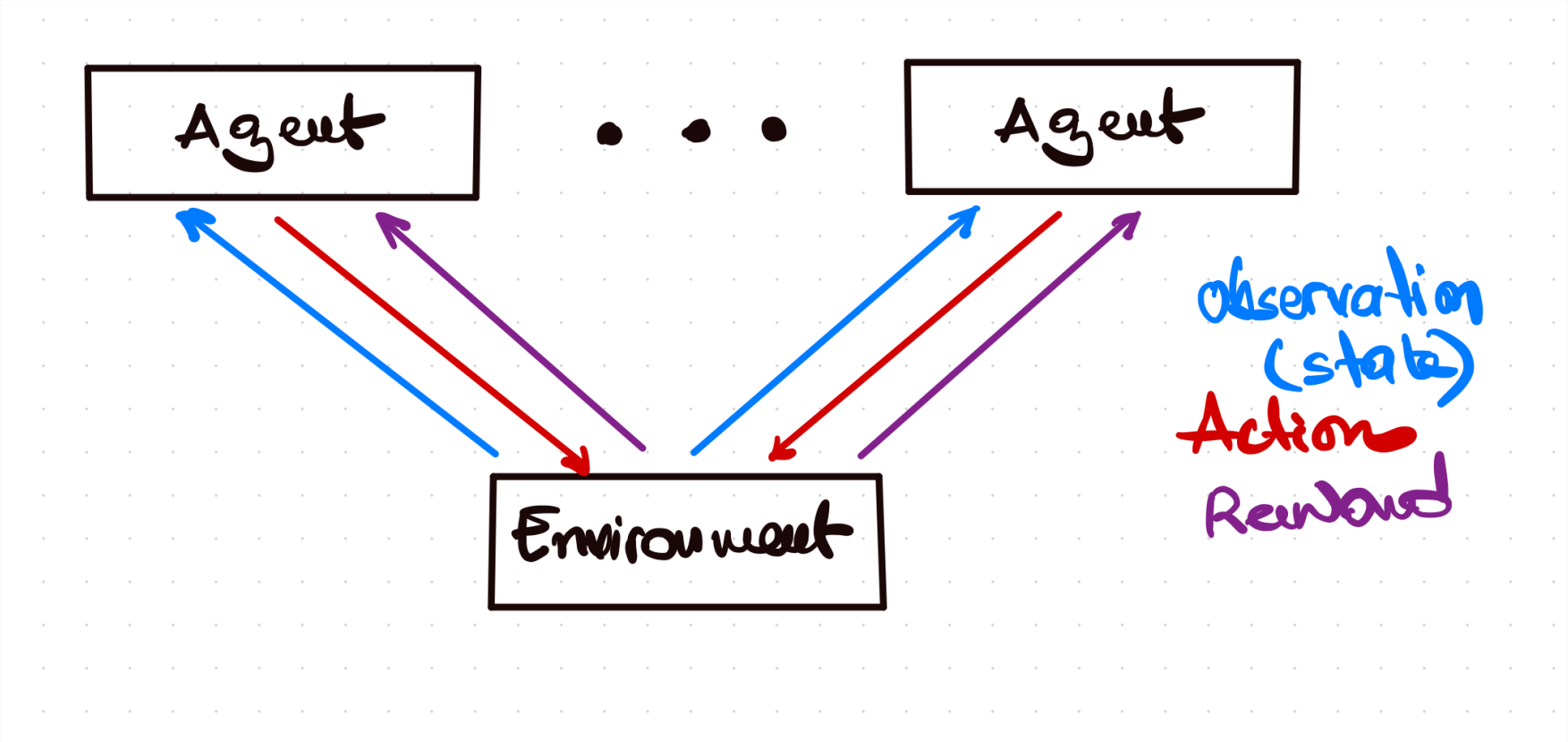
Akka Zemmari

What we have seen so far



observation (state)
Action
Reward

What we will see today



Disclaimer

This course is mainly based on the following references:

- *Multi-Agent Reinforcement Learning: Foundations and Modern Approaches* by Stefano V. Albrecht, Filippos Christianos and Lukas Schäfer.
- The course on RL by Stefano V. Albrecht
- The courses on RL from `huggingface.co`

Introduction to Multi-Agent Reinforcement Learning (MARL)

Introduction to Multi-Agent Reinforcement Learning (MARL)

- Multi-Agent Reinforcement Learning (MARL) involves multiple agents interacting within a shared environment.

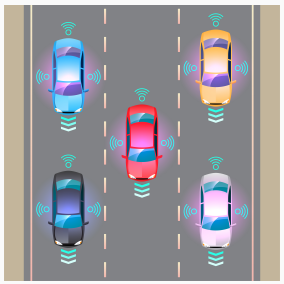
Introduction to Multi-Agent Reinforcement Learning (MARL)

- Multi-Agent Reinforcement Learning (MARL) involves multiple agents interacting within a shared environment.
- Each agent learns to maximize its own reward by adjusting its actions based on interactions with both the environment and other agents.

Introduction to Multi-Agent Reinforcement Learning (MARL)

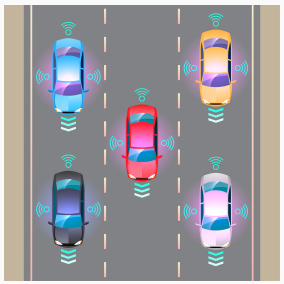
- Multi-Agent Reinforcement Learning (MARL) involves multiple agents interacting within a shared environment.
- Each agent learns to maximize its own reward by adjusting its actions based on interactions with both the environment and other agents.
- MARL combines elements from both Reinforcement Learning (RL) and Game Theory.

MARL systems: Examples[1]



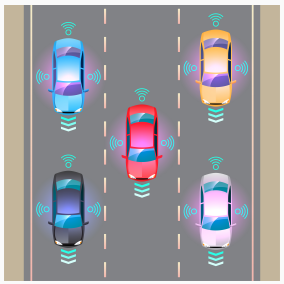
[1] Images from https://huggingface.co

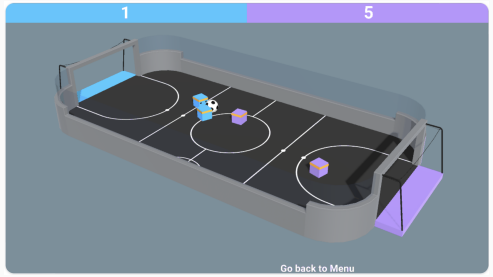
MARL systems: Examples[1]



[1] Images from https://huggingface.co

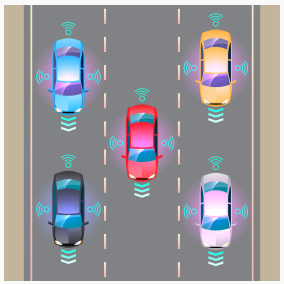
MARL systems: Examples[1]

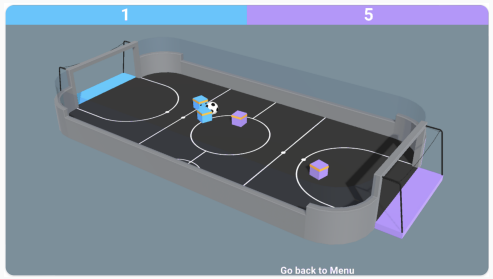




[1] Images from https://huggingface.co

MARL systems: Examples[1]





Questions: What's common in these systems? what's different?

[1] Images from https://huggingface.co

Applications of Multi-Agent Reinforcement Learning

- Autonomous vehicles: Coordination among multiple self-driving cars.
- Robotics: Multiple robots cooperating in tasks like warehouse management.
- Finance: Agents optimizing trading strategies in competitive markets.
- Gaming: Simulations where agents cooperate or compete, e.g., in real-time strategy games.

Challenges in MARL

Challenges in MARL

- **Coordination:** How agents can learn to work together effectively.

Challenges in MARL

- **Coordination:** How agents can learn to work together effectively.
- **Competition:** Learning in competitive scenarios, where one agent's gain might be another's loss.

Challenges in MARL

- **Coordination:** How agents can learn to work together effectively.
- **Competition:** Learning in competitive scenarios, where one agent's gain might be another's loss.
- **Communication:** Effective ways for agents to share information without compromising performance.

Basic Models for Multi-Agent Environments

Basic Models for Multi-Agent Environments

- **Cooperative models:** All agents work together to achieve a shared goal.

Basic Models for Multi-Agent Environments

- **Cooperative models:** All agents work together to achieve a shared goal.
- **Competitive models:** Agents pursue individual goals that may conflict with others' objectives.

Basic Models for Multi-Agent Environments

- **Cooperative models:** All agents work together to achieve a shared goal.
- **Competitive models:** Agents pursue individual goals that may conflict with others' objectives.
- **Mixed models:** Some agents cooperate while others compete, leading to complex dynamics.

Basic Models for Multi-Agent Environments

- **Cooperative models:** All agents work together to achieve a shared goal.
- **Competitive models:** Agents pursue individual goals that may conflict with others' objectives.
- **Mixed models:** Some agents cooperate while others compete, leading to complex dynamics.
- Policies in MARL can be independent or dependent, influencing how agents learn in shared environments.

Multi-Agent Models

Standard models for multi-agent interaction:

- **Normal-form game**
- **Repeated game**
- **Stochastic game**

Normal-form game

Normal-form game consists of:

Normal-form game

Normal-form game consists of:

- A set of agents $N = \{1, 2, \ldots, n\}$.

Normal-form game

Normal-form game consists of:

- A set of agents $N = \{1, 2, \ldots, n\}$.
- For each agent $i \in N$:
  - A set of actions $A_i = \{a_{i,1}, a_{i,2}, \ldots, a_{i,m_i}\}$.
  - A utility function :

$$u_i : A = A_1 \times A_2 \times \ldots \times A_n \to \mathbb{R}.$$

Normal-form game

Normal-form game consists of:

- A set of agents $N = \{1, 2, \ldots, n\}$.
- For each agent $i \in N$:
  - A set of actions $A_i = \{a_{i,1}, a_{i,2}, \ldots, a_{i,m_i}\}$.
  - A utility function :

$$u_i : A = A_1 \times A_2 \times \ldots \times A_n \to \mathbb{R}.$$

Each agent i uses its policy $\pi_i : A_i \to [0, 1]$ to choose an action a_i with probability $\pi_i(a_i)$, and receives a payoff (reward) $u_i(a_1, a_2, \ldots, a_n)$.

Thus, the expected reward for agent i is:

$$U_i(\pi_1, \pi_2, \ldots, \pi_n) = \sum_{a \in A} u_i(a) \prod_{j=1}^{n} \pi_j(a_j).$$

Example: Prisoner's Dilemma

- Two prisoners are arrested for a crime and interrogated separately.
- Each prisoner can either confess (C) or remain silent (S).

Example: Prisoner's Dilemma

- Two prisoners are arrested for a crime and interrogated separately.
- Each prisoner can either confess (C) or remain silent (S).

Reward matrix for the prisoners:

	Prisoner 2: C	**Prisoner 2: S**
Prisoner 1: C	(-1, -1)	(-5, 0)
Prisoner 1: S	(0, -5)	(-3, -3)

Example: Rock-Paper-Scissors

- Two players, three actions: Rock, Paper, Scissors.
- Reck beats Scissors, Scissors beats Paper, Paper beats Rock.

Example: Rock-Paper-Scissors

- Two players, three actions: Rock, Paper, Scissors.
- Reck beats Scissors, Scissors beats Paper, Paper beats Rock.

Reward matrix for the player:

	Player 2: Rock	**Player 2: Paper**	**Player 2: Scissors**
Player 1: Rock	(0, 0)	(-1, 1)	(1, -1)
Player 1: Paper	(1, -1)	(0, 0)	(-1, 1)
Player 1: Scissors	(-1, 1)	(1, -1)	(0, 0)

Repeated game

- Normal-form game is single interaction $\rightarrow$ no experience

Repeated game

- Normal-form game is single interaction $\rightarrow$ no experience

Repeated game:

- Repeat the same normal-form game for time steps $t = 0, 1, 2, 3, \cdots$
- At time t, each agent i:
  - selects policy π_i^t,
  - samples action a_i^t with probability $\pi_i^t(a_i^t)$,
  - receives reward $u_i(a^t)$, where $a^t = (a_1^t, a_2^t, \ldots, a_n^t)$.

Repeated game

- Normal-form game is single interaction $\rightarrow$ no experience

Repeated game:

- Repeat the same normal-form game for time steps $t = 0, 1, 2, 3, \cdots$
- At time t, each agent i:
  - selects policy π_i^t,
  - samples action a_i^t with probability $\pi_i^t(a_i^t)$,
  - receives reward $u_i(a^t)$, where $a^t = (a_1^t, a_2^t, \ldots, a_n^t)$.
- Learning: modify policy π_i^t based on history $H^t = (a^0, a^1, \ldots, a^{t-1})$.

Stochastic game

- Agents interact in a dynamic environment with uncertain outcomes.

Stochastic game

- Agents interact in a dynamic environment with uncertain outcomes.

Stochastic game:

- At time t, each agent i:
  - selects policy π_i^t,
  - samples action a_i^t with probability $\pi_i^t(a_i^t)$,
  - receives reward $u_i(a^t)$, where $a^t = (a_1^t, a_2^t, \ldots, a_n^t)$.
- The environment is stochastic, with transition probabilities $P(s'|s, a)$.
- Learning: modify policy π_i^t based on history $H^t = (a^0, a^1, \ldots, a^{t-1})$.

Stochastic game

Agents interact in shared environment

- Environment has states, and actions have effect on state
- Agents choose actions based on observed state

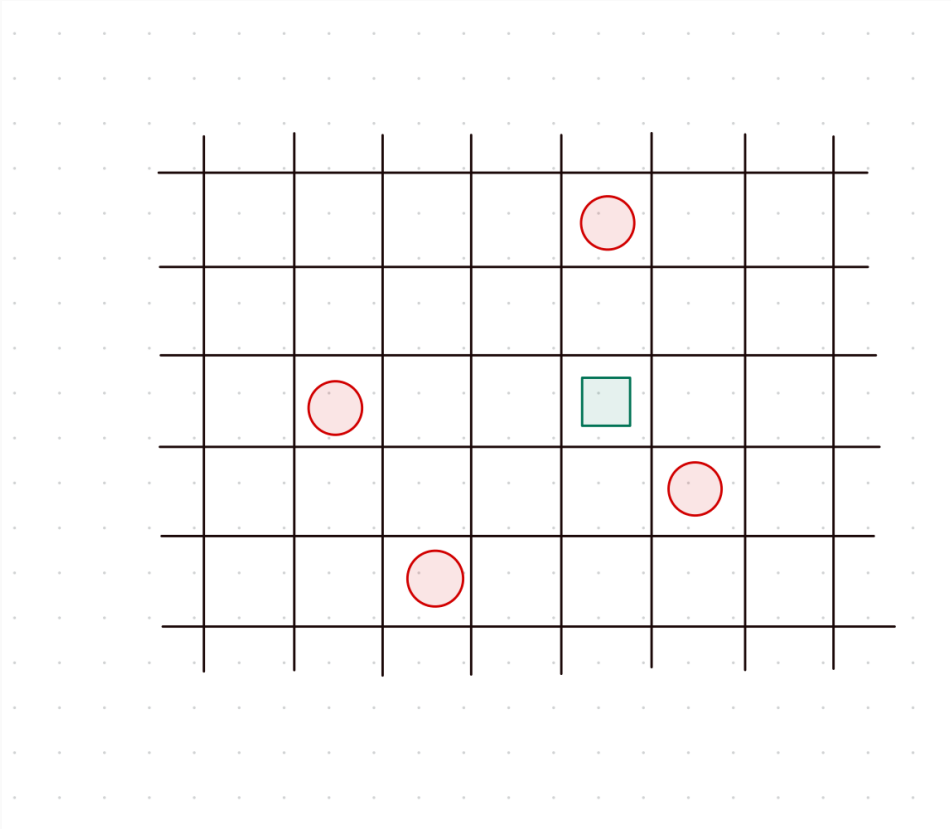
Stochastic game

Agents interact in shared environment

- Environment has states, and actions have effect on state
- Agents choose actions based on observed state

Example: Predator-prey

- Predator agents must catch prey agents.
- States: positions of agents in the environment.
- Actions: up, down, left, right.



Stochastic game (or Markov game)

Definition (Stochastic game)

A stochastic game consists of:

- Finite set of agents $N = \{1, 2, \ldots, n\}$.
- Finite set of states S, with subset of terminal states $\bar{S} \subset S$.
- For each agent $i \in N$:
  - Finite set of actions A_i.
  - Reward function $\mathcal{R}_\rangle : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$, where $A = A_1 \times A_2 \times \ldots \times A_n$.
- State transition probabilities $P : S \times A \times S \to [0, 1]$ s.t.: $\forall s \in S, a \in A : \sum_{s' \in S} P(s, a, s') = 1$.
- Initial state distribution $\mu : S \to [0, 1]$ s.t.: $\sum_{s \in S} \mu(s) = 1$ and $\forall s \in \bar{S} : \mu(s) = 0$.

Stochastic game (or Markov game)

Definition (Stochastic game)

A stochastic game consists of:

- Finite set of agents $N = \{1, 2, \ldots, n\}$.
- Finite set of states S, with subset of terminal states $\bar{S} \subset S$.
- For each agent $i \in N$:
  - Finite set of actions A_i.
  - Reward function $\mathcal{R}_\rangle : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$, where $A = A_1 \times A_2 \times \ldots \times A_n$.
- State transition probabilities $P : S \times A \times S \to [0, 1]$ s.t.: $\forall s \in S, a \in A : \sum_{s' \in S} P(s, a, s') = 1$.
- Initial state distribution $\mu : S \to [0, 1]$ s.t.: $\sum_{s \in S} \mu(s) = 1$ and $\forall s \in \bar{S} : \mu(s) = 0$.

Remark
It's a generalization of Markov Decision Process (MDP) to multiple agents.

Stochastic game

- Game starts in initial state $s_0 \in S$.
- At time t, each agent i:
  - Observes current state s_t.
  - Chooses action a_i^t with probability $\pi_i(s_t, a_i^t)$.
  - Receives reward $u_i(s_t, a_1^t, \ldots, a_n^t)$.
- Game transitions into next state s_{t+1} with probability $P(s_t, a, s_{t+1})$.
- Repeat T times or until terminal state is reached.

$\Rightarrow$ Learning is now based on state-action history

$H_t = (s_0, a_0, s_1, a_1, \ldots, s_t)$

Stochastic game - Expected return

Given policy profile $\pi = (\pi_1, \pi_2, \ldots, \pi_n)$, what is expected return to agent i in state s?

Stochastic game - Expected return

Given policy profile $\pi = (\pi_1, \pi_2, \ldots, \pi_n)$, what is expected return to agent i in state s?

Analogous to Bellman equation in MDP:

$$U_i(s, \pi) = \sum_{a \in A} \left(\prod_{j \in N} \pi_j(s, a_j) \right) \left[u_i(s, a) + \gamma \sum_{s' \in S} P(s, a, s') U_i(s', \pi) \right]$$

Partially Observable Stochastic Games

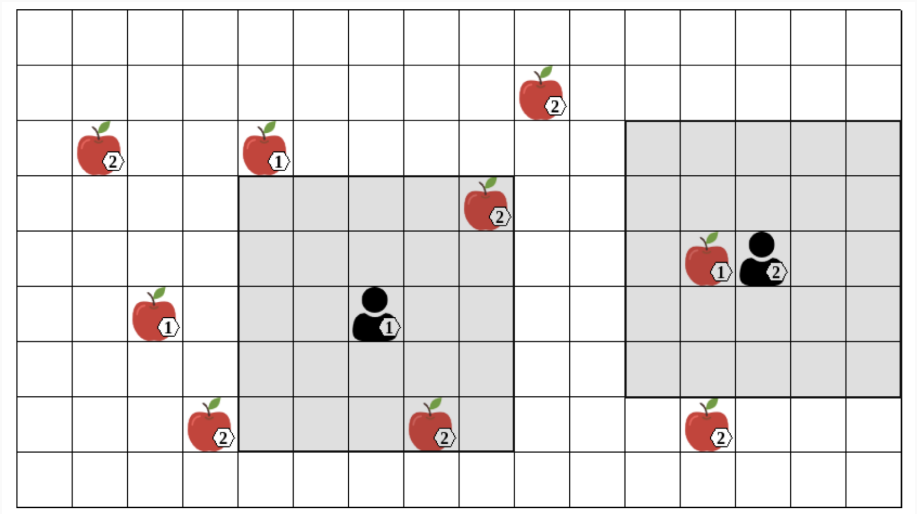
Definition (Partially observable stochastic game)
A partially observable stochastic game (POSG) is defined by the same elements of a stochastic game and additionally defines for each agent $i \in I$:

- Finite set of observations O_i.
- Observation function $\mathcal{O}_i : A \times S \times O_i \to [0, 1]$ such that

$$\forall a \in A, s \in S : \sum_{o_i \in O_i} \mathcal{O}_i(a, s, o_i) = 1$$

Stochastic game vs POSG

- In stochastic games the agents, can directly observe the environment state and the chosen actions of all agents
- In a POSG, the agents receive "observations" that carry some incomplete information about the environment state and agents' actions.



Solving Games

Simple (naive) approach:

For each agent i, consider the other agents as part of the environment.

Whiteboard time!

Solving Games

If agent rewards differ, $u_i \neq u_j$, what should π optimise?

Many solution concepts exist:

- Minimax solution
- Nash/correlated equilibrium
- Pareto-optimality
- Social welfare & fairness
- No-regret
- Targeted optimality & safety

Solving Games

Intuitions: see Whiteboard

Datails: see next session