MARL week3

2024. 01. 25

Agenda

- MARL background
  - Multi-Agent Reinforcement Learning: Foundations and Modern Approaches
- Multi-Agent Reinforcement Learning(Part I) review
  - Learning and Games Boot Camp, 2022

Multi-Agent System

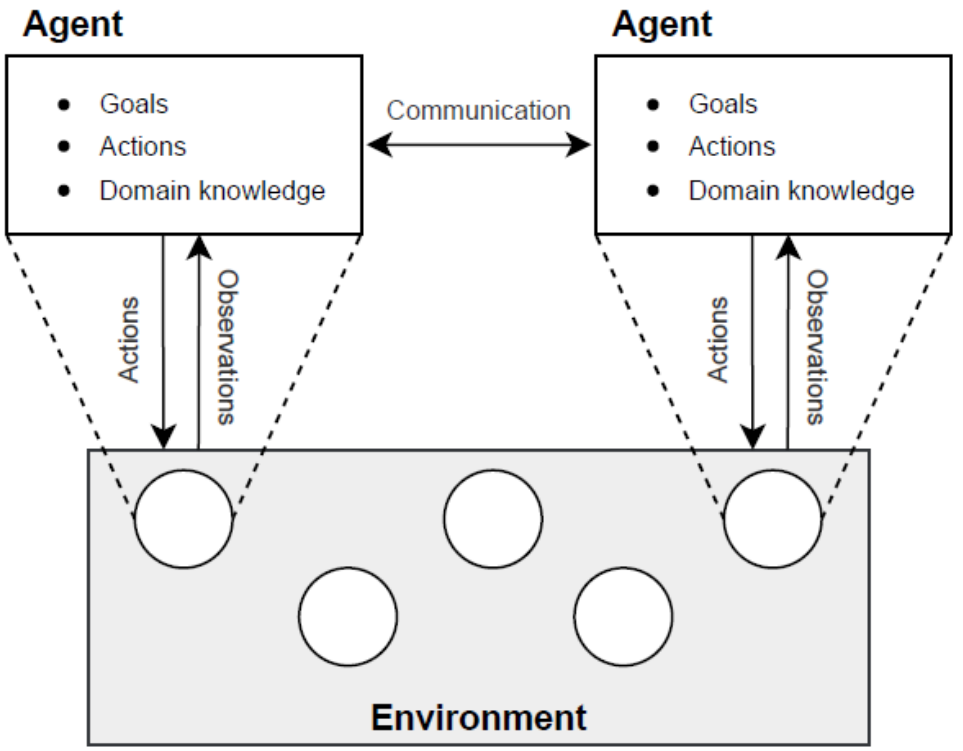


Figure 1.1: Schematic of a multi-agent system. A multi-agent system consists of an environment and multiple decision-making agents (shown as circles inside the environment). The agents can observe information about the environment and take actions to achieve their goals.

Multi-Agent Reinforcement Learning(MARL)

- A set of *n* agents choose individual actions, which together are referred to as the *joint action*.
- The joint action changes the state of the environment according to the environment dynamics.
- The agents receive individual rewards as a result of this change, as well as individual observations about the new environment states.

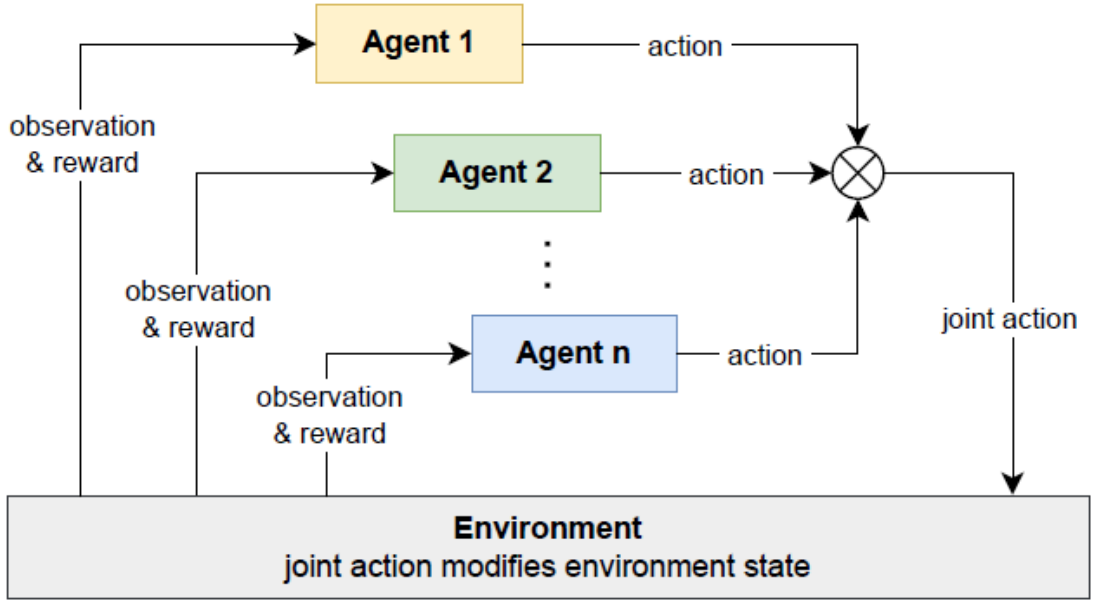


Figure 1.3: Schematic of multi-agent reinforcement learning. A set of *n* agents receive individual observations about the state of the environment, and choose actions to modify the state of the environment. Each agent then receives a scalar reward and a new observation, and the loop repeats.

Multi-Agent Reinforcement Learning(MARL)

Several important use cases in which MARL can have significant benefits over single-agent RL

1. To decompose a large, intractable decision problem into smaller, more tractable decision problem.
2. The Agents may need to learn decentralized policies, where each agent executes its own policy local based on its own observations.

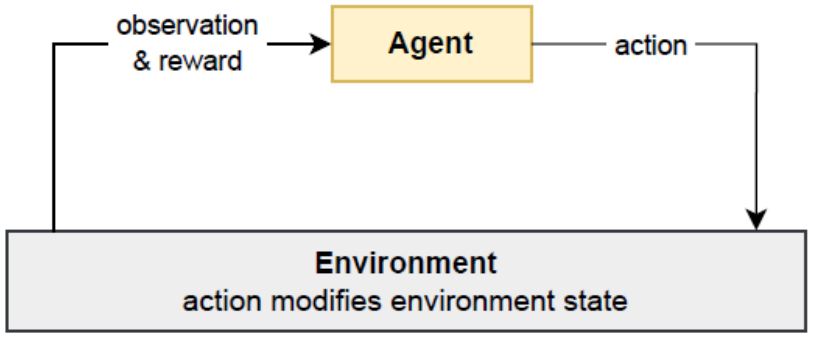


Figure 2.2: Basic reinforcement learning loop for a single-agent system.

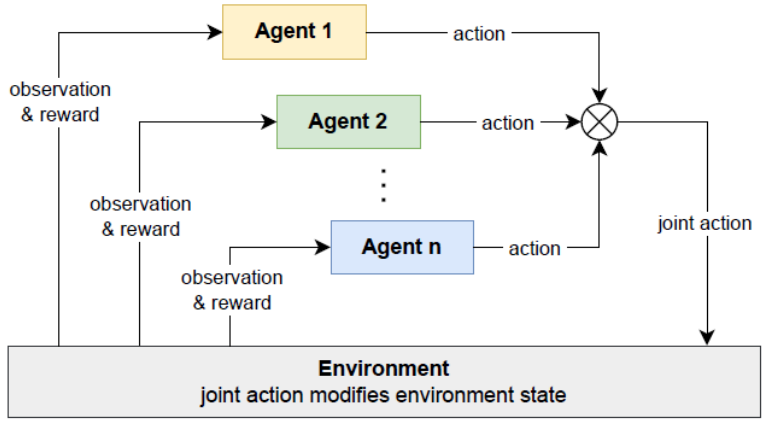


Figure 1.3: Schematic of multi-agent reinforcement learning. A set of n agents receive individual observations about the state of the environment, and choose actions to modify the state of the environment. Each agent then receives a scalar reward and a new observation, and the loop repeats.

MARL algorithms can be categorized!

- Agent's rewards (e.g. fully cooperative, competitive, or mixed)
- What type of solution concept the algorithm is designed to achieve (e.g. Nash equilibrium)
- What agents can observe about their environment
- Algorithms can also be categorized based on assumptions made during the learning of agent policies(Training) versus assumption made after learning(Execution)
  - Centralized training and Execution
    - both stages have access to some centrally shared information
  - Decentralized training and execution
    - no centrally shared information, agent's policy only use local information of that agent
  - Centralized training with Decentralized execution
    - Centralization is feasible during training, while producing policies that can be executed in a fully decentralized way.

MARL algorithms can be categorized!

Dimension	Questions
Size	How many agents exist in the environment? Is the number of agents fixed, or can it change? How many states and actions does the environment specify? Are states/actions discrete or continuous? Are actions defined as single values or multi-valued vectors? *(Chapter 3)*
Knowledge	Do agents know what actions are available to themselves and to other agents? Do they know their own reward functions, and the reward functions of other agents? Do agents know the state transition probabilities of the environment? *(Chapter 3)*
Observability	What can agents observe about their environment? Can agents observe the full environment state, or are their observations partial and noisy? Can they observe the actions and/or the rewards of other agents? *(Chapter 3)*
Rewards	Are the agents opponents, with zero-sum rewards? Or are agents teammates, with common (shared) rewards? Or do agents have to compete and cooperate in some way? *(Chapter 3)*
Objective	Is the agents' goal to learn an equilibrium joint policy? What type of equilibrium? Is performance during learning important, or only the final learned policies? Is the goal to perform well against certain classes of other agents? *(Chapters 4 and 5)*
Centralisation & Communication	Can agents coordinate their actions via a central controller or coordinator? Or do agents learn fully independent policies? Can agents share/communicate information during learning and/or after learning? Is the communication channel reliable, or noisy and unreliable? *(Chapters 3, 5, 6 and 9)*

Figure 1.4: Dimensions in MARL and relevant book chapters for further details.

Main Challenges of MARL

- **Non-stationarity caused by learning agents**
  - Caused by the continually changing policies of the agents during their learning processes
  - Could lead moving target problem
- **Optimality of policies and equilibrium selection**
  - When are the policies of agents in a multi-agent system *optimal*?
  - The returns of one agent's policy depend on the other agent's policies
    → need sophisticated notions of optimality
- **Multi-agent credit assignment**
  - Determining *whose* action contributed to the reward
- **Scaling in number of agents**
  - The total number of possible action combinations between agents may grow exponentially with the number of agents
  - It is common to use a number of agents between 2 and 10.

How to define MARL problem?

- **Basic game model**
  - Define concepts such as states, actions, observations, and reward in a multi-agent environment
  - Represent interaction processes in a multi-agent system(normal-form game, stochastic games, …)
- **Solution concepts**
  - Define what is means to solve these game models: what is means for agents to act optimally
  - Equilibrium-type solution(minmax, Nash, ..)

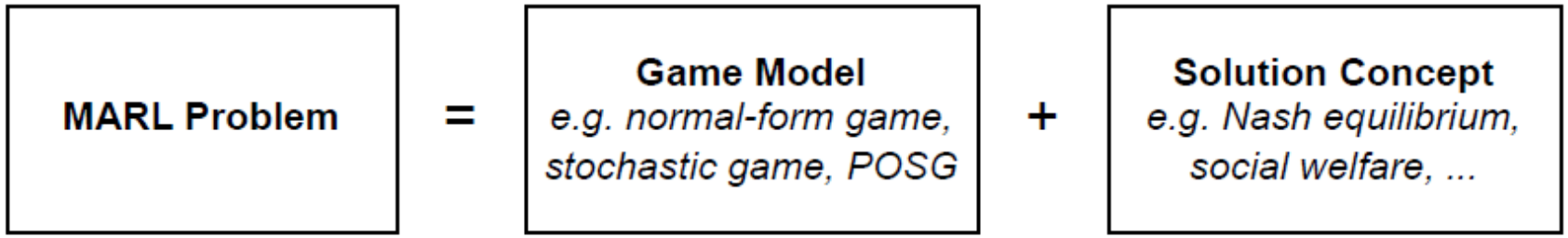
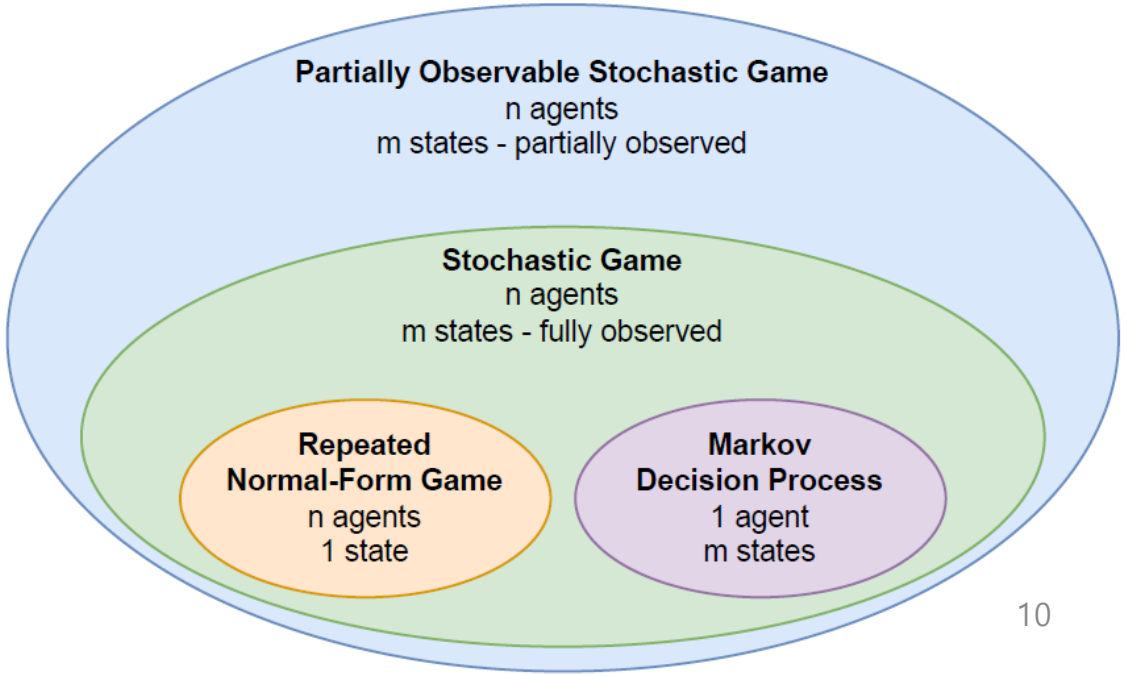


Figure 4.1: A MARL problem is defined by the combination of a game model which defines the mechanics of the multi-agent system and interactions, and a solution concept which specifies the desired properties of the joint policy to be learned. (See also Figure 2.1, page 20.)

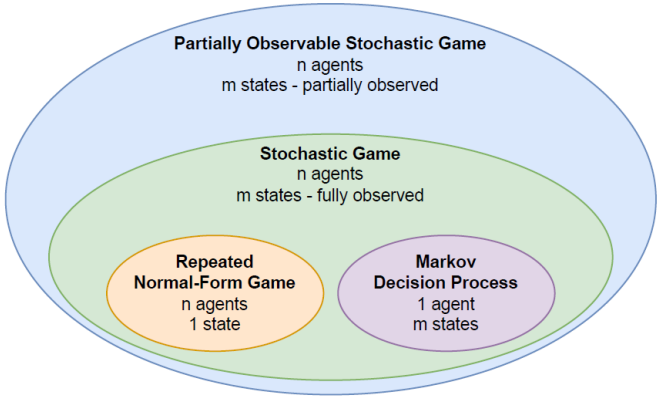
Games: Model of Multi-Agent Interaction

- **Normal-Form Game**
  - Multiple agents but there is no evolving environment state
- **Stochastic Game(=Markov game)**
  - Define states that change over time as a result of the agent's actions and probabilistic state transitions
- **Partially Observable Stochastic Game**
  - Agent' do not directly observe the full environment
  - But, observe incomplete or noisy information about the environment

MARL Problem	=	Game Model *e.g. normal-form game, stochastic game, POSG*	+	Solution Concept *e.g. Nash equilibrium, social welfare, ...*



Games: Model of Multi-Agent Interaction



- **Normal-Form Game**
  - single interaction between two or more agents

Definition 2 (Normal-form game) *A normal-form game consists of:*

- *Finite set of agents* $I = \{1, ..., n\}$
- *For each agent* $i \in I$*:*
  - *Finite set of actions* A_i
  - *Reward function* $\mathcal{R}_i : A \to \mathbb{R}$*, where* $A = A_1 \times ... \times A_n$

- Matrix game
  = normal-form games for two agents

probability $\pi_i(a_i)$ given by its policy. The resulting actions of all agents form a *joint action*, $a = (a_1, ..., a_n)$. Finally, each agent i receives a reward based on its reward function and the joint action, $r_i = \mathcal{R}_i(a)$.

Normal-form games can be classified based on the relationship between the reward functions of agents:

- In a *zero-sum* game, the sum of the agents' rewards is always 0, i.e. $\sum_{i \in I} \mathcal{R}_i(a) = 0$ for all $a \in A$.[2] In zero-sum games with two agents, i and j,[3] one agent's reward function is simply the negative of the other agent's reward function, i.e. $\mathcal{R}_i = -\mathcal{R}_j$.
- In a *common-reward* game, all agents receive the same reward, i.e. $\mathcal{R}_i = \mathcal{R}_j$ for all $i, j \in I$.
- In a *general-sum* game, there are no restrictions on the relationship of reward functions.

Games: Model of Multi-Agent Interaction

- **Repeated Normal-Form Game**
  - Normal-form game + sequence(time)
  - Repeating the same normal-form game for a finite or infinite number of times
  - The policies of agents can be conditioned on the history of past joint actions chosen by the agents

action $a_i^t \in A_i$ with probability given by its policy, $\pi_i(a_i^t \mid h^t)$.
conditioned on the *joint-action history*, $h^t = (a^0, \ldots, a^{t-1})$,

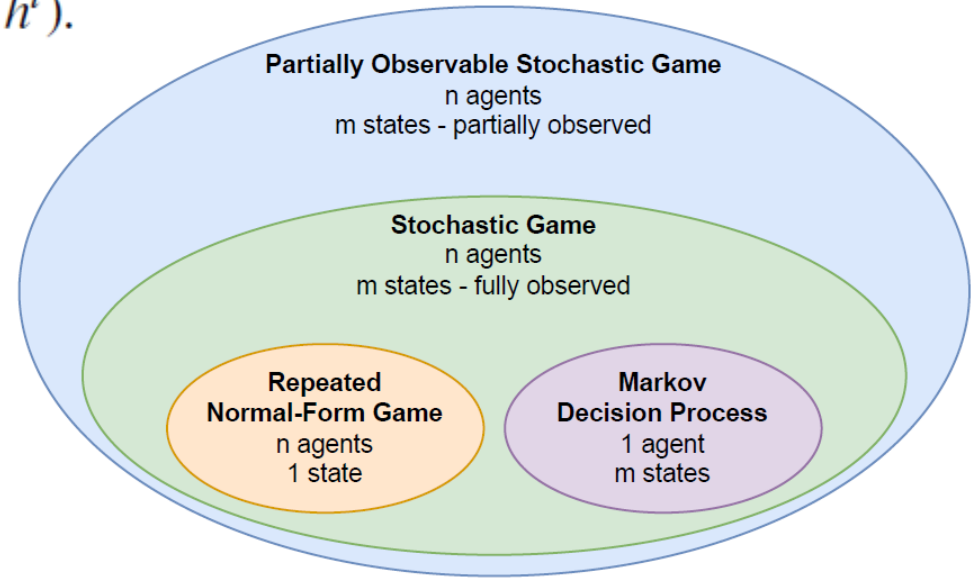


Figure 3.1: Hierarchy of game models used in this book. Partially observable stochastic games (POSGs) include stochastic games as a special case in which the states and agents' chosen actions are fully observable by all agents. Stochastic games include repeated normal-form games as a special case in which there is only a single environment state, and they include Markov decision processes (MDPs) as a special case in which there is only a single agent.

Games: Model of Multi-Agent Interaction

- **Stochastic Game(=Markov game)**
  - Define a state-based environment in which the state evolves over time based on the agent's actions and probabilistic state transitions.

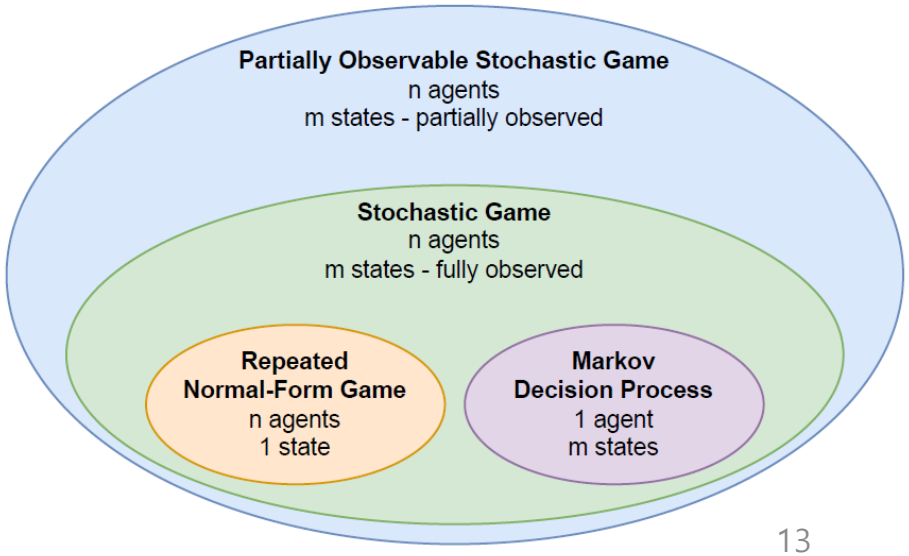
Definition 3 (Stochastic game) *A stochastic game consists of:*

- *Finite set of agents* $I = \{1, ..., n\}$
- *Finite set of states* S, *with subset of terminal states* $\bar{S} \subset S$
- *For each agent* $i \in I$:
  - *Finite set of actions* A_i
  - *Reward function* $\mathcal{R}_i : S \times A \times S \to \mathbb{R}$, *where* $A = A_1 \times ... \times A_n$
- *State transition probability function* $\mathcal{T} : S \times A \times S \to [0, 1]$ *such that*

$$\forall s \in S, a \in A : \sum_{s' \in S} \mathcal{T}(s, a, s') = 1 \qquad (3.1)$$

- *Initial state distribution* $\mu : S \to [0, 1]$ *such that*

$$\sum_{s \in S} \mu(s) = 1 \quad \text{and} \quad \forall s \in \bar{S} : \mu(s) = 0 \qquad (3.2)$$

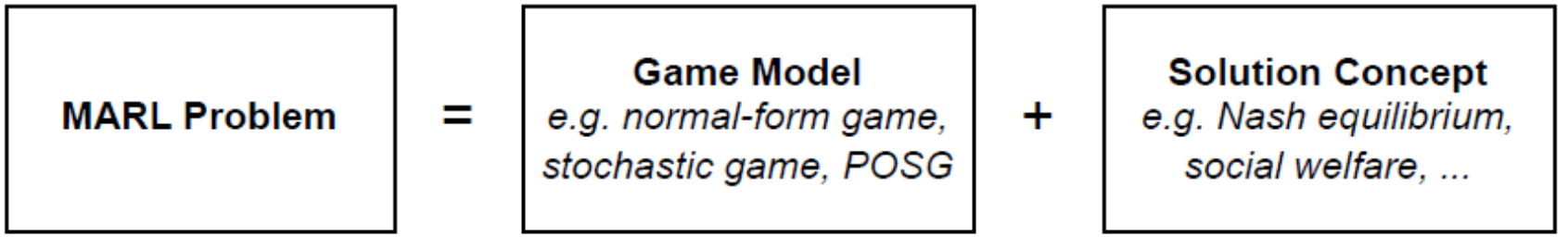


Solution Concepts for games

- What does it mean for agents to interact *optimally* in a multi-agent system?
  == What is a *solution* to a game?

- For common-reward games, in which all agents receive the same reward,
  Solution is to maximize the expected return received by all agents.
- What if the agent have differing reward?

- General equilibrium solution: minmax equilibrium, Nash equilibrium, correlated equilibrium
- Solution refinements: Pareto-optimality, social welfare and fairness, no-regret



Multiagent Reinforcement Learning

Chi Jin

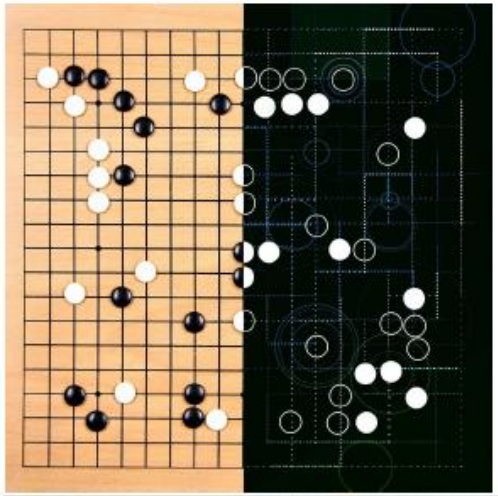
Princeton University.

Slides: on my homepage

Blog post: `yubai.org/blog/marl_theory.html`

Interesting Problems

Go



Hide-and-seek

Starcraft 2

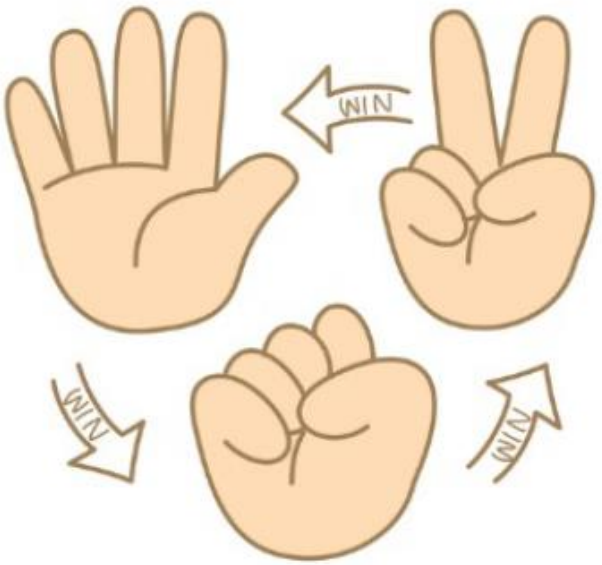
Texas poker

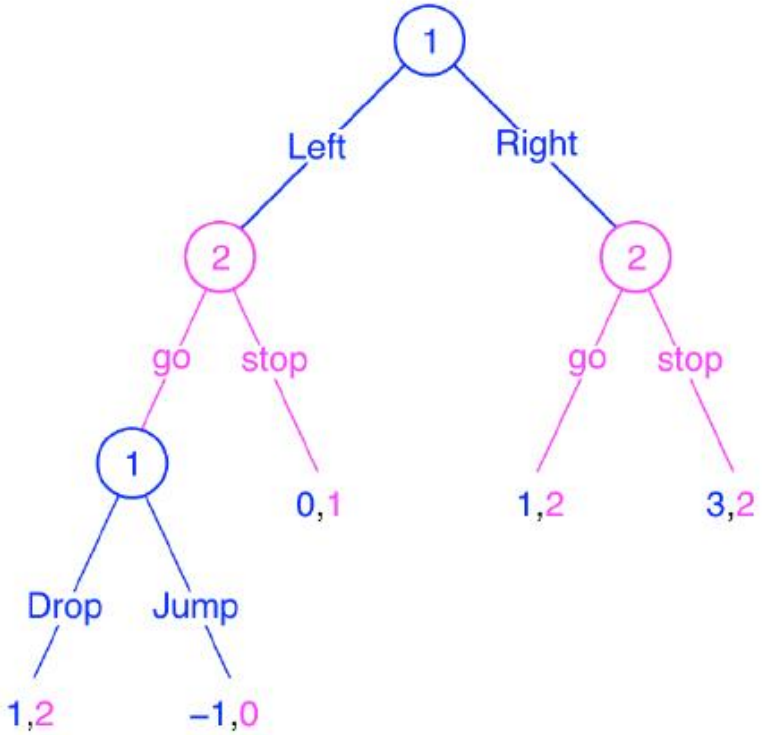
Multiagent **Games** + **Sequential** decision making

(at least 2 agents)

Classical Game Theory

All players just play simultaneously
→ there's no sequential information



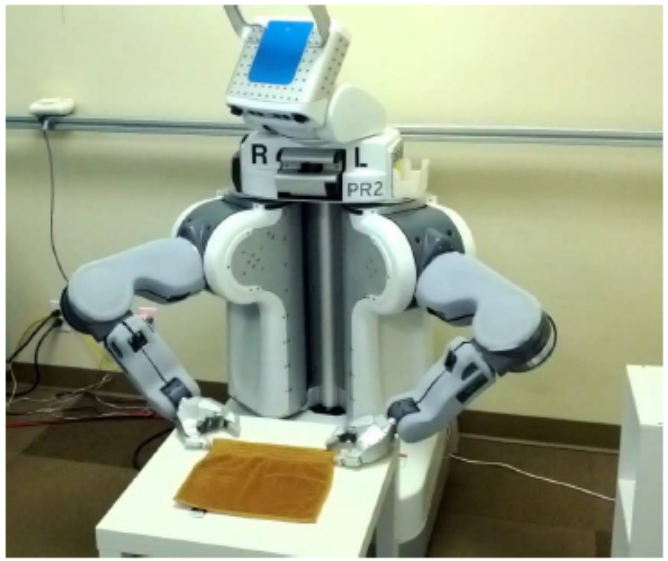


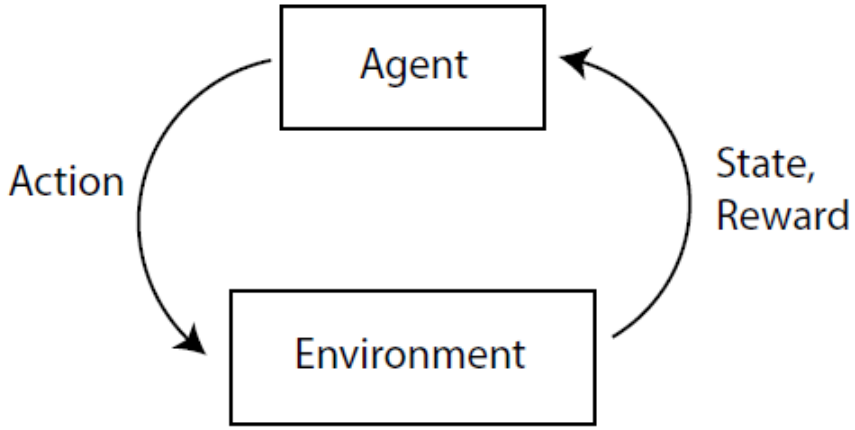
- Normal-form games, Extensive-form games, …

each player takes actions.
It can grow exponentially with steps

Limitation → They don't handle **sequential games** with **long horizon** efficiently.

Single-agent Reinforcement Learning





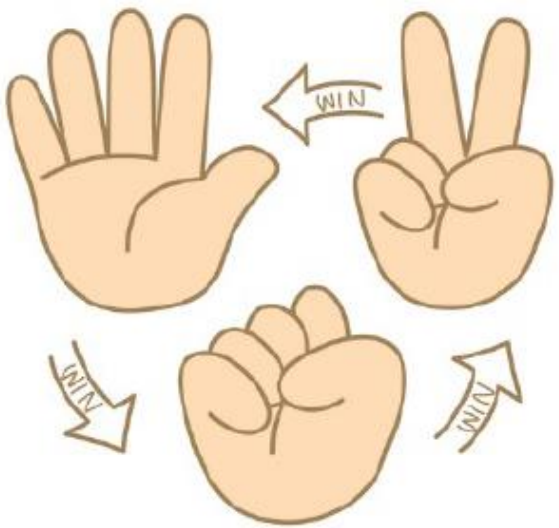
- Goal: find the best policy within a fixed environment.

Limitation → Opponents in MARL are not fixed, and can be **adaptive**!

e.g. change their strategy

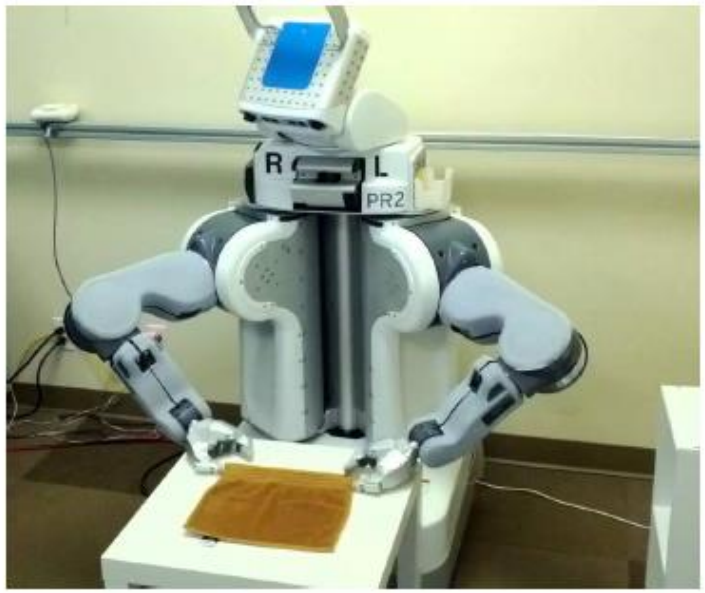
Multiagent Reinforcement Learning

Game theory



+

Reinforcement learning

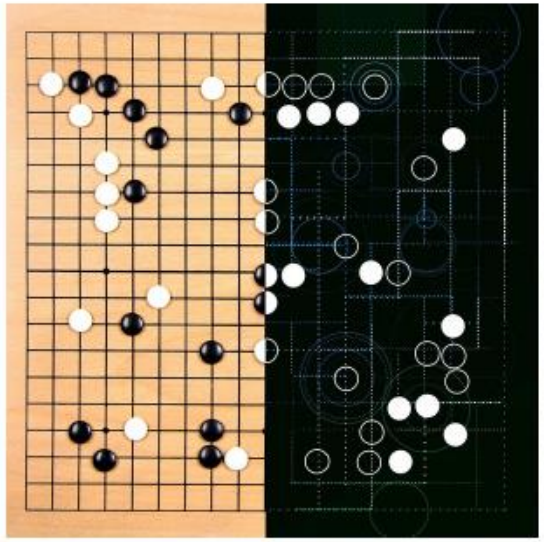


A newer and less developed field, with its own unique challenges and opportunities.

Main Question

Can we establish a solid theoretical foundation for MARL?

Efficiency



Sample efficiency and computational efficiency

Focus하는 이유 → AlphaGo Zero: trained on $\geq 10^7$ games, and took ≥ 1 month.

Statistics + Computer Science

Outline

Part I

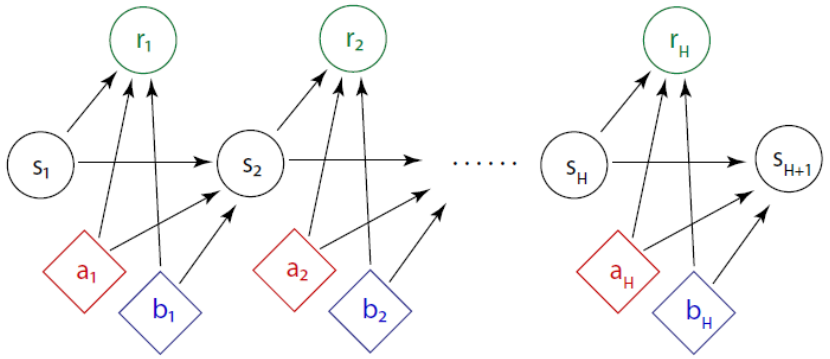
- Formulation and Objectives → Normal-form game, extensive form game으로 충분하지 X
- Direct Combinations of Game Theory & Single-agent RL → Commonly used in practice

- Two-player Zero-sum Games (Markov game)
- Multiplayer General-sum Games
- Advanced Topics : function approximation, partial obserability

Formulation and Objectives

Markov Games (Stochastic Games)

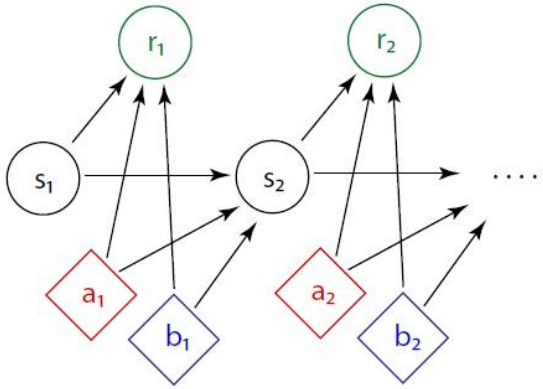
: Markov decision process의 multiagent scenario화



Two-player zero-sum Markov Game $(\mathcal{S}, \mathcal{A}, \mathcal{B}, \mathbb{P}, r, H)$ [Shapley 1953].

- $\mathcal{S}$: set of states; $\mathcal{A}, \mathcal{B}$: set of actions for the max-player/the min-player.
- $\mathbb{P}_h(s_{h+1}|s_h, a_h, b_h)$: transition probability.
- $r_h(s_h, a_h, b_h) \in [0, 1]$: reward for the max-player (loss for the min-player).
- H: horizon/the length of the game.

Interaction Protocol



Interaction protocol

Environment samples initial state s_1.
for step $h = 1, \ldots, H$,
two agents take their own actions (a_h, b_h) simultaneously.
both agents receive their own immediate reward $\pm r_h(s_h, a_h, b_h)$.
environment transitions to the next state $s_{h+1} \sim \mathbb{P}_h(\cdot|s_h, a_h, b_h)$.

- 서로 상대방이 어떤 action을 취하는 지 a_h, b_h를 볼 수 있음

Our Setup

→ 필요한 game model에 대한 정의

In this talk, we mostly focus on fully observable tabular Markov games.

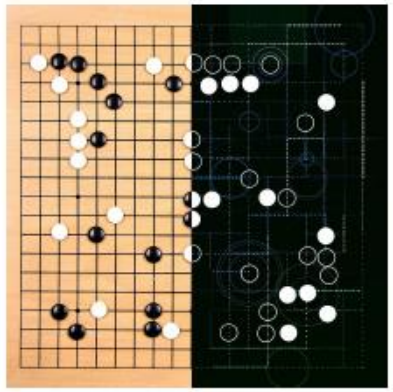
- Fully observable: joint actions and states are revealed to both agents.
  ↔ partial observable
- Tabular: the size of $\mathcal{S}, \mathcal{A}, \mathcal{B}$ is finite and small.

serve as a **foundation** for more advanced setups in the future

Solution Concepts

What's ultimate goal of MARL?

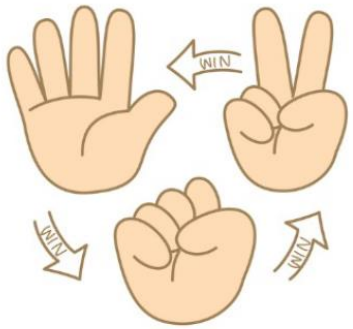
In case of single agent, goal is to maximize cumulative reward


What policy is good?

- Beat the world champion by a large margin?
- Beat all players by a large margin?

Best Responses

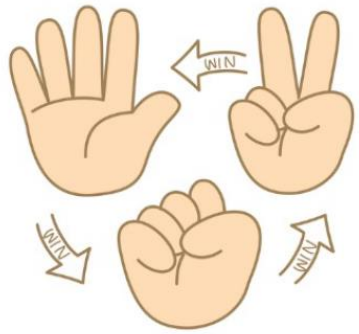


The policy that best exploits the opponent's policy.

$$\mathrm{BR}(\pi_2) := \operatorname*{argmax}_{\pi_1} V^{\pi_1,\pi_2}$$

Good against a fixed opponent, but can be bad against others.

Nash Equilibria II



The optimal strategy if always facing best responses.

"We may not win by a large margin, but no one beats us."

Objective: find ϵ-approximate Nash equilibria $(\hat{\pi}_1, \hat{\pi}_2)$ using a small number of samples with mild dependency on S, A_1, A_2, ϵ, H.

$$\max_{\pi_1} V^{\pi_1,\hat{\pi}_2} - \min_{\pi_2} V^{\hat{\pi}_1,\pi_2} \le \epsilon.$$

Challenges

To name a few:

- Large size of policy space:

  $\Omega((1/\epsilon)^{HSA})$ Markov policies in the tabular setting

- Nash equilibrium policy is Markov, but the best response may not be.

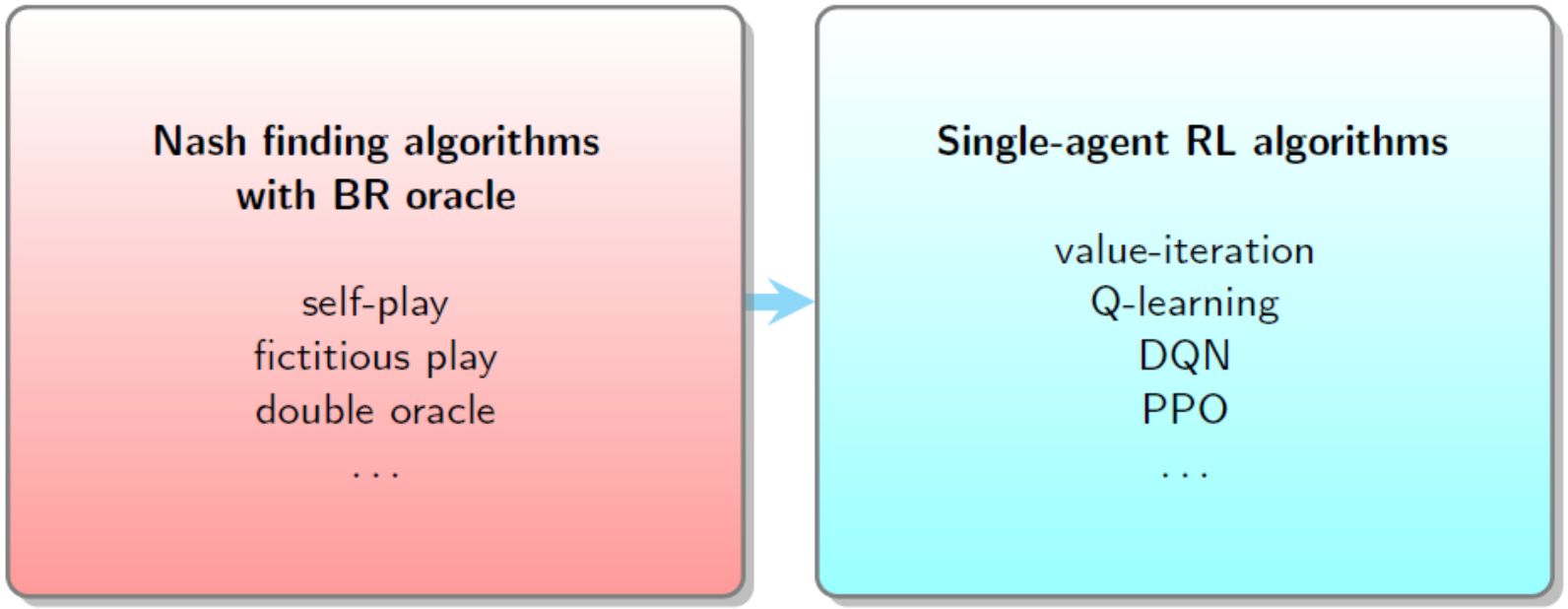
- MGs do not allow efficient no-regret learning [Bai, **Jin**, Yu, 2020].

$$\max_{\pi_1} \sum_{t=1}^{T} V_1^{\pi_1 \times \pi_2^t} - \sum_{t=1}^{T} V_1^{\pi_1^t \times \pi_2^t} \le \mathrm{poly}(H, S, A, B) T^{1-\alpha}.$$

Direct Combinations

Fixed opponent를 environment로 인식

Key observation: given a fixed opponent, computing best response (BR) is a single-agent RL problem.



commonly used in practice.

Drawbacks of Direct Combinations

self-play
fictitious play
double oracle

- Algorithms are designed based on black-box usage of single-agent RL, which does not exploit the detailed structure of MGs.
- Converting a MG into a norm-form game gives a number of action $A = (1/\epsilon)^{HSA'}$.
- Finding BR is NOT a easy single-agent RL problem:
  - When the min-player deploys a fixed **non-Markovian** policy, the game is NOT an MDP from the perspective of the max-player.
  - Existing single-agent RL results do not apply.

Reference

- Multi-Agent Reinforcement Learning(Part I), Chi Jin(Princeton University), Learning and Games Boot Camp, https://simons.berkeley.edu/talks/multi-agent-reinforcement-learning-part-i
- Multi-Agent Reinforcement Learning: Foundations and Mordern Approaches, Stefano V. Albrecht, Filippos Christianos, Lukas Schäfer, https://www.marl-book.com/