Lecture 11b: Multiagent RL
CS885 Reinforcement Learning

2025-02-11

Complementary readings:
Caroline Claus and Craig Boutilier (1998) The Dynamics of Reinforcement Learning in Cooperative Multiagent Systems, AAAI.
Michael Littman (1994) Markov games as a framework for multi-agent reinforcement learning, Machine learning proceedings.
Junling Hu and Michael P. Wellman (2003) Nash Q-learning for General-Sum Stochastic Games, JMLR

Pascal Poupart
David R. Cheriton School of Computer Science
CIFAR AI Chair at Vector Institute

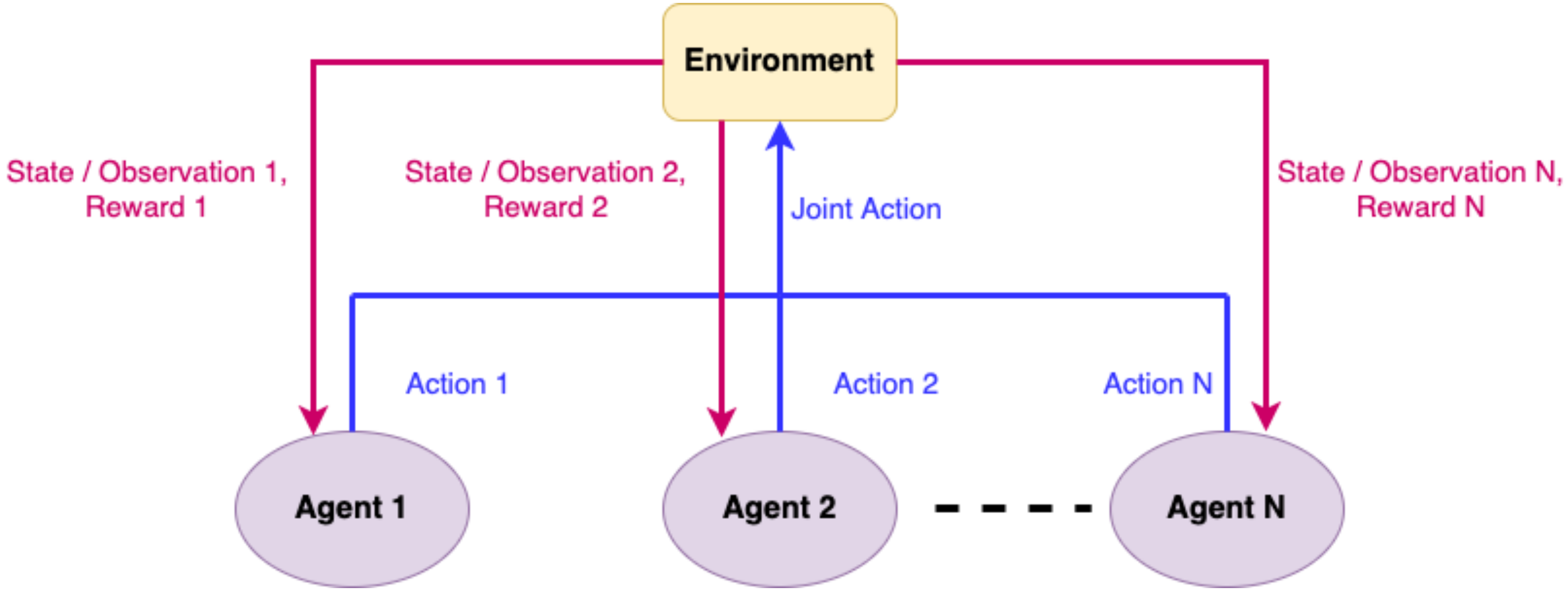
UNIVERSITY OF WATERLOO

Outline

- Stochastic Games
- Multi-agent Reinforcement Learning (MARL)
- Opponent Modelling: Fictitious Play
- Cooperative Stochastic Games
  - Joint Q learning
- Competitive Stochastic Games (Zero-sum games)
  - Minimax Q learning

Multi-agent Reinforcement Learning

Multi-agent Games + Sequential decision making



Newer field with unique challenges and opportunities

Stochastic Games

- (Simultaneously moving) Stochastic Game (N-agent MDP)
  - N: Number of agents
  - S: Shared state space $s \in S$
  - A^j: Action space of agent j
    - $\langle a^1, a^2, \ldots, a^N \rangle \in A^1 \times A^2 \times \cdots \times A^N$
  - R^j: Reward function for agent j: $R^j(s, a^1, \ldots, a^N) = \sum_{r^j} r^j \, Pr(r^j | s, a^1, \ldots, a^N)$
    - Cooperative game: same reward for all agents
    - Competitive game: $\sum_j R^j(s, a^1, \ldots, a^N) = 0$
  - T: Transition function: $Pr(s' | s, a^1, \ldots, a^N)$
  - γ: Discount factor: $0 \leq \gamma \leq 1$
  - Horizon (i.e., # of time steps): h

Unknown models and unknown policies of other agents

- Policy (strategy) for agent i: $\pi^i : S \rightarrow \Omega(A^i)$
- Goal: Find optimal policy such that $\boldsymbol{\pi}^* = \{\pi_1^*, \ldots, \pi_N^*\}$,

  where $\pi_i^* = \underset{\pi^i}{\operatorname{argmax}} \sum_{t=0}^{h} \gamma^t \mathbb{E}_{\boldsymbol{\pi}}[r_t^i(s, \boldsymbol{a})]$, where $\boldsymbol{a} \triangleq \{a^1, \ldots, a^N\}$ and $\boldsymbol{\pi} \triangleq \{\pi^1, \ldots, \pi^N\}$

Playing a stochastic game

- Players choose their actions at the same time
  - No communication with other agents
  - No observation of other player's actions

- Each player chooses a strategy π^i which is a mapping from states to actions and can be either
  - Mixed strategy: Distribution over actions for at least one state
  - Pure strategy: One action with prob 100% for all states

- At each state, all agents face a stage game (normal form game) with the Q values of the current state and joint action of each player being the utility for that player

- The stochastic game can be thought of as a repeated normal form game with a state representation

Solution Concept

- In MARL, a solution often corresponds to some equilibrium of the stochastic game

- The most common solution concept is the Nash equilibrium

- Let us define a value function for the multi-agent setting

$$V_{\boldsymbol{\pi}}^{j}(s) \triangleq \sum_{t=0}^{\infty} \gamma^{t} \mathbb{E}_{\boldsymbol{\pi}}[r_t^j | s_o = s, \boldsymbol{\pi}]$$

- Nash equilibrium under the stochastic game satisfies

$$V^{j}_{\left(\pi_*^j, \boldsymbol{\pi}_*^{-j}\right)}(s) \geq V^{j}_{\left(\pi^j, \boldsymbol{\pi}_*^{-j}\right)}(s). \quad \forall s \in S; \forall j; \forall \pi^j \neq \pi_*^j$$

UNIVERSITY OF WATERLOO

Independent learning

- Naive approach: Apply the single agent Q-learning directly
- Each agent would update its Q-values using the Bellman update:

$$Q^j(s,a^j) \leftarrow Q^j(s,a^j) + \alpha(r^j + \gamma \max_{a'^j} Q^j(s',a'^j) - Q^j(s,a^j))$$

- Each agent assumes that the other agent(s) are part of the environment

- Advantage: Simple approach, easy to apply
- Disadvantages:
  - Might not work well against opponents playing complex strategies
  - Non-stationary transition and reward models
  - No convergence guarantees

Opponent Modelling

- Note that an agent's response requires knowledge of other agent's actions
- This is a simultaneously move game where each agent does not know what the other agents will do
- So each agent should maintain a belief over other agents actions at current state
- Maintaining a belief over the actions of other agents is called opponent modelling

- Techniques for Opponent Modelling:
  - Fictitious Play
  - Gradient Based Methods
  - Solving Unique Equilibrium (for each stage game)
  - Bayesian Approaches

Fictitious Play

- Each agent assumes that all opponents are playing a stationary mixed strategy
- Agents maintain a count of number of times another agent performs an action

$$n_t^i(s, a^j) \leftarrow 1 + n_{t-1}^i(s, a^j), \forall j, \forall i$$

- Agents update their belief about this strategy at each state according to

$$Pr_t^i\left(a^j \middle| s\right) = \frac{n_t^i(s,a^j)}{\sum_{a'^j} n_t^i(s,a'^j)}$$

- Agents calculate best responses according to this belief

UNIVERSITY OF WATERLOO

Joint Q learning

JointQlearning(s, Q)

Repeat

Repeat for each agent i

Select (e.g., epsilon greedy, Boltzmann exploration) and execute a^i

Observe s', r^i and $\boldsymbol{a}^{-i}$, where $\boldsymbol{a}^{-i} = \{a^1, \dots, a^{i-1}, a^{i+1}, \dots, a^N\}$

Update counts: $n(s, \boldsymbol{a}) \leftarrow n(s, \boldsymbol{a}) + 1,\ n^i(s, a^j) \leftarrow 1 + n^i(s, a^j), \forall j$

Sample others' actions: $\hat{a}'^j \sim Pr^i(a'_j | s') = \frac{n^i(s', a'^j)}{\sum_{a'} n^i(s', a'^j)}\ \forall j \neq i$

Learning rate: $\alpha \leftarrow 1/n(s, \boldsymbol{a})$

Update Q-value:

$$Q^i(s, a^i, \boldsymbol{a}^{-i}) \leftarrow Q^i(s, a^i, \boldsymbol{a}^{-i}) + \alpha(r^i + \gamma \max_{a'^i} Q^i(s', a'^i, \hat{a}'^1, \dots, \hat{a}'^N) - Q^i(s, a^i, \boldsymbol{a}^{-i}))$$

$s \leftarrow s'$

Convergence of Tabular Joint Q learning

- If the game is finite (finite agents and finite number of strategies for each agent), then fictitious play will converge to true response of opponent(s) in the limit in self-play
- Self-play: All agents learn using the same algorithm
- Joint Q-learning converges to Nash Q-values in a cooperative stochastic game if
  - Every state is visited infinitely often (e.g., epsilon greedy or Boltzmann exploration)
  - The learning rate α is decreased fast enough, but not too fast (sufficient conditions for α):

$$(1)\ \sum_n \alpha_n \rightarrow \infty \qquad (2)\ \sum_n (\alpha_n)^2 < \infty$$

- In cooperative stochastic games, the Nash Q-values are unique (guaranteed unique equilibrium)

Cooperative Stochastic Games

- Cooperative stochastic game: same reward function for all agents
- Equilibrium for cooperative stochastic games is the Pareto dominating (Nash) equilibrium
  - Nash equilibrium: $\forall i, a_i,\ R_i(a_i^*, a_{-i}^*) \geq R_i(a_i, a_{-i}^*)$
  - Pareto dominating: $\forall i\ R_i(a^*) \geq R_i(a'^*)$
- There exists a unique Pareto dominating (Nash) equilibrium

		Bob: Baseball	Bob: Soccer
Alice	Baseball	2,2	0,0
Alice	Soccer	0,0	1,1

Competitive Stochastic Games

- The equilibrium in the case of competitive stochastic games is the min-max Nash equilibrium
- Each stage game of this stochastic game faces a zero-sum game
- There exists a unique min-max (Nash) equilibrium in utilities
- Optimal min-max value function

$$V_*^j(s) = \max_{a^j} \min_{a^{-j}} [r^j(s, a^j, a^{-j}) + \gamma \sum_{s'} Pr(s'|s, a^j, a^{-j}) V_*^j(s')]$$

- For a competitive stochastic game there exists a unique min-max value function and hence a unique min-max Q-function

UNIVERSITY OF WATERLOO

Learning in competitive stochastic games

- Algorithm: Minimax Q-Learning
- Q-values for each agent j are over joint actions: $Q^j(s, a^j, a^{-j})$
  - s = state
  - a^j = action
  - a^{-j} = opponent action
- Instead of playing the best $Q^j(s, a^j, a^{-j})$ play min-max Q

$$Q^j(s, a^j, a^{-j}) \leftarrow (1-\alpha) Q^j(s, a^j, a^{-j}) + \alpha(r^j + \gamma V^j(s'))$$

$$V^j(s') \leftarrow \max_{a^j} \min_{a^{-j}} Q^j(s', a^j, a^{-j})$$

Minimax Q learning

Minimax Qlearning

Repeat

Repeat for each agent

Select and execute action a^j

Observe s', a^{-j} and r

Update counts: $n(s, \boldsymbol{a}) \leftarrow n(s, \boldsymbol{a}) + 1$

Learning rate: $\alpha \leftarrow \frac{1}{n(s,\boldsymbol{a})}$

Update Q-value:

$$Q_*^j(s, a^j, a^{-j}) \leftarrow (1 - \alpha) Q_*^j(s, a^j, a^{-j}) + \alpha\left(r^j + \gamma \max_{a'^j} \min_{a'^{-j}} Q_*^j(s', a'^j, a'^{-j})\right)$$

$s \leftarrow s'$

Convergence of Minimax Tabular Q learning

- Convergence in self-play
- Minimax Q-learning converges to min-max equilibrium in competitive game if:
  - Every state is visited infinitely often (e.g. epsilon-greedy or Boltzmann exploration)
  - The learning rate α is decreased fast enough, but not too fast (sufficient conditions for α):

$$(1)\ \sum_n \alpha_n \to \infty \qquad (2)\ \sum_n (\alpha_n)^2 < \infty$$

- In a competitive stochastic games, the Nash Q-values are unique (guaranteed unique min-max equilibrium point in utilities)

Opponent Modelling

- In a competitive game rational agents always take a min-max action
- There is no requirement for a separate opponent modelling strategy in self-play

- However:
  - Other agents could use different algorithms
  - Computing the min-max action can be time consuming

- Alternative: Fictitious play
  - Fact: Fictitious play also converges in competitive zero-sum games
  - Fact: Fictitious play converges to the min-max action in self-play

(Mixed) Stochastic Games/ General-sum Stochastic Game

- Rewards for each agent can be arbitrary
  - Rewards are not the same for all agent (i.e., not cooperative)
  - They do not sum to 0 (i.e., not entirely competitive)
- Objective for agent: Find the optimal policy for best response
- What should be the solution concept?
  - There could be multiple Nash equilibria
  - Nash theorem: at-least one mixed strategy Nash equilibrium exists
- Area of research
  - Various solution concepts
  - Various forms of opponent modeling