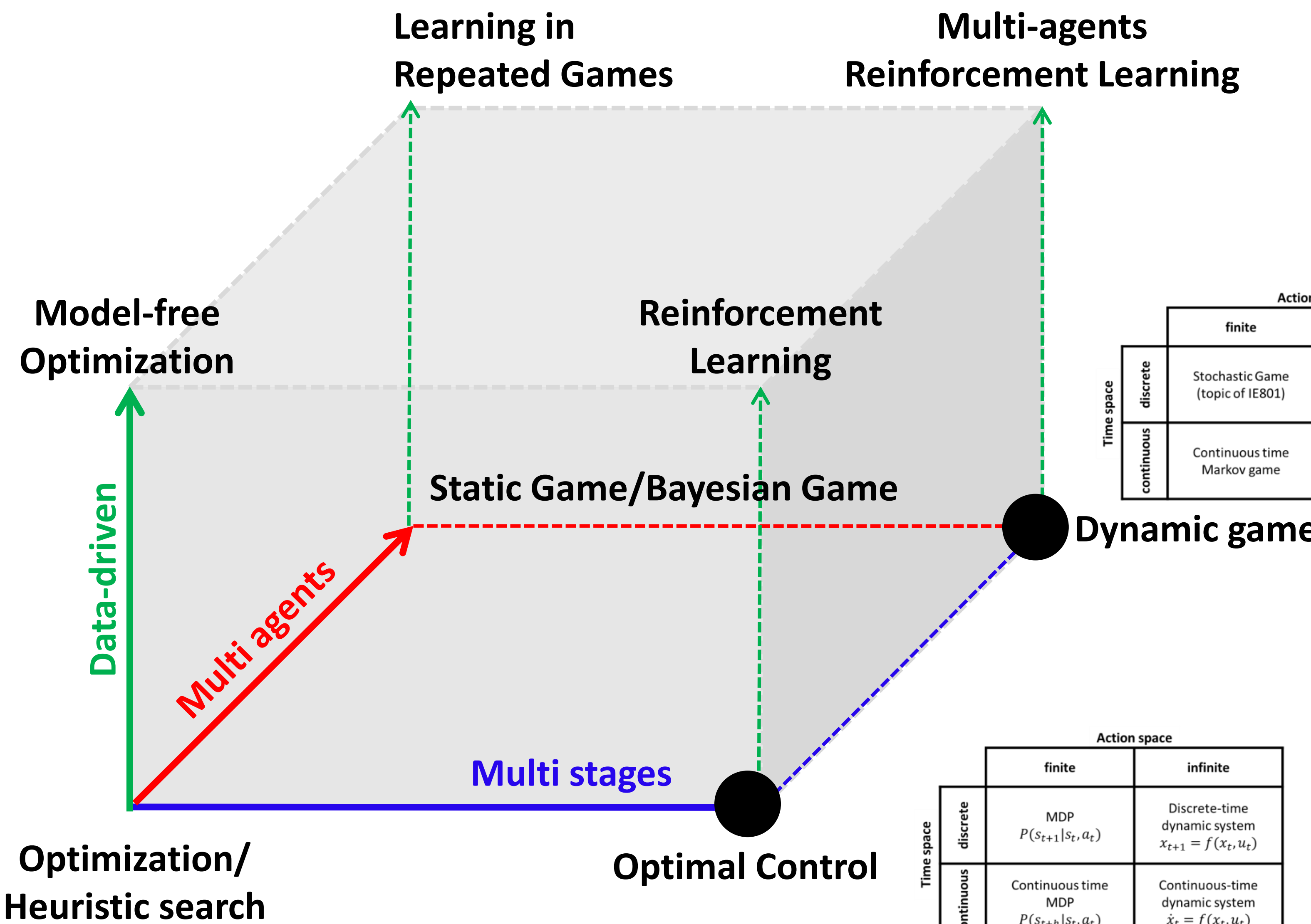
Lecture 19-Stochastic Game Introduction

Optimal Control in Dynamic System



Action space

Time space		finite	infinite
	discrete	Stochastic Game (topic of IE801)	Discrete-time infinite dynamic game
	continuous	Continuous time Markov game	Discrete-time infinite dynamic game (Differential game)

Action space

Time space		finite	infinite
	discrete	MDP $P(s_{t+1} \mid s_t, a_t)$	Discrete-time dynamic system $x_{t+1} = f(x_t, u_t)$
	continuous	Continuous time MDP $P(s_{t+h} \mid s_t, a_t)$	Continuous-time dynamic system $\dot{x}_t = f(x_t, u_t)$

Optimal Control in Dynamic System

Reinforcement Learning

Action space

Time space		finite	infinite
	discrete	MDP $P(s_{t+1} \mid s_t, a_t)$	Discrete-time dynamic system $x_{t+1} = f(x_t, u_t)$
	continuous	Continuous time MDP $P(s_{t+h} \mid s_t, a_t)$	Continuous-time dynamic system $\dot{x}_t = f(x_t, u_t)$

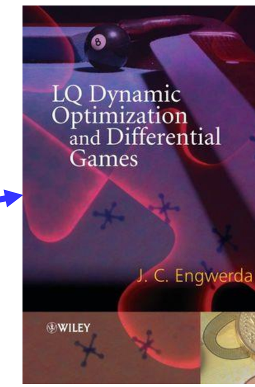
Will be briefly discussed later

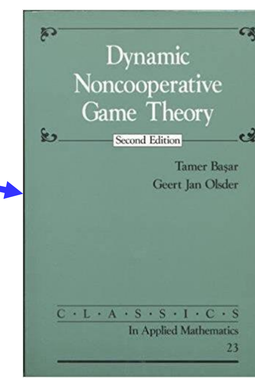
Extension of Optimal Control to Dynamic Game

Multi-Agent Reinforcement Learning

Time space		Action space: finite	Action space: infinite
	discrete	Stochastic Game (topic of IE801)	Discrete-time infinite dynamic game
	continuous	Continuous time Markov game	Discrete-time infinite dynamic game (Differential game)

Will be briefly discussed later





Motivations

- What if we didn't always repeat back to the same stage game?
- A stochastic game is a generalization of repeated games
  - agents repeatedly play games from a set of normal-form games
  - the game played at any iteration depends on the previous game played and on the actions taken by all agents in that game
- A stochastic game is a generalized Markov decision process
  - there are multiple players one reward function for each agent
  - the state transition function and reward functions depend on the action choices of all the game participants

Formal Definition

Definition (Stochastic game)

A stochastic game is a tuple (N, S, A, R, T), where

- N is a finite set of n players
- S is a finite set of states (stage games),
- $A = A_1 \times \cdots \times A_n$, where A_i is a finite set of actions available to player i,
- $T : S \times A \times S \longmapsto [0,1]$ is the transition probability function; $T(s, a, s')$ is the probability of transitioning from state s to state s' after joint action a,
- $R = r_1 \dots, r_n$, where $r_i : S \times A \longmapsto \mathbb{R}$ is a real-valued payoff function for player i

- In a discounted stochastic game, the objective of each player is to maximize the discounted sum of rewards, with discount factor $\gamma \in [0,1)$.
- Let π_i be the strategy of player i. For a given initial state s, player i tries to maximize

$$V_i(s, \pi_1, \dots, \pi_i, \dots, \pi_n) = \sum_{t=0}^{\infty} \gamma^t E\left[r_{i,t} \middle| \pi_1, \dots, \pi_i, \dots, \pi_n, s_0 = s\right]$$

  - The accumulated rewards also depends on the strategy of other agents

Formal Definition

- All agents $(1, \dots, n)$ share the joint state s
- The transition equation is similar to the Markov Decision Process decision transition:

MDP : $\sum_{s'} T(s, a, s') = 1 \; \forall s \in S, \forall a \in A$

SG: $\sum_{s'} T(s, a_1, \dots, a_i, \dots, a_n, s') = 1 \; \forall s \in S, \forall a_i \in A_i, i = (1, \dots, n)$

- Reward function r_i for agent i depends on the current joint state s, the joint action $a = (a_1, \dots, a_n)$, and the next joint future state s'

MDP : $r(s, a, s')$

SG: $r_i(s, a_1, \dots, a_i, \dots, a_n, s')$

Formal Definition

- In a discounted stochastic game, the objective of each player is to maximize the discounted sum of rewards, with discount factor $\gamma \in [0,1)$.
- Let π_i be the strategy of player i. For a given initial state s, player i tries to maximize

$$V_i(s, \pi_1, \ldots, \pi_i, \ldots, \pi_n) = \sum_{t=0}^{\infty} \gamma^t E\left[r_{i,t} | \pi_1, \ldots, \pi_i, \ldots, \pi_n, s_0 = s\right]$$

  - The accumulated rewards also depends on the strategy of other agents

Remarks

- The strategy space of the agents is the same in all games
  - ➢ The difference between the games is only in the payoff function

- The payoff of a player is assigned at each state (or stage game)

- Before, a history was just a sequence of actions
  - But now we have action profiles rather than individual actions, and each profile has several possible outcomes
  - Thus a history is a sequence $h_t = (q_0, a_0, q_1, a_1, \ldots, a_{t-1}, q_t)$, where t is the number of stages

- How to aggregate the payoffs from multiple states? The two most commonly used aggregation methods are:
  - Future discounted reward
  - Average reward

Strategies

- What is a pure strategy?
  - pick an action conditional on every possible history
  - of course, mixtures over these pure strategies are possible too!
- Some interesting restricted classes of strategies:
  - behavioral strategy: $s_i(h_t, a_{i_j})$ returns the probability of playing action a_{i_j} for history h_t.
    - the substantive assumption here is that mixing takes place at each history independently, not once at the beginning of the game
  - Markov strategy: s_i is a behavioral strategy in which $s_i(h_t, a_{i_j}) = s_i(h'_t, a_{i_j})$ if $q_t = q'_t$, where q_t and q'_t are the final states of h_t and h'_t, respectively.
    - for a given time t, the distribution over actions only depends on the current state
  - stationary strategy: s_i is a Markov strategy in which $s_i(h_{t_1}, a_{i_j}) = s_i(h'_{t_2}, a_{i_j})$ if $q_{t_1} = q'_{t_2}$, where q_{t_1} and $q'_{t_2}$are the final states of h_{t_1} and h'_{t_2}, respectively.
    - No dependence even on t

Multi Agent Q-learning Template

MultiQ(StochastiGame, f, γ, α, T)

Inputs	equilibrium selection function f
	discounting factor γ
	learning rate α
	total training time T
Outputs	state − value functions V_i^*
	action − value functions Q_i^*
Initialize	$s, a_1, \ldots, a_n$ and $Q_1, \ldots, Q_n$

for $t = 1{:}T$

1. select actions $a_1, \ldots, a_n$ in state s
2. observe rewards $r_1, \ldots, r_n$ and next state s'
3. for $i = 1$ to n (for each agent)
   (a) $V_i(s') = f_i(Q_1(s', a), \ldots, Q_n(s', a))$
   (b) $Q_i(s, a) = (1 - \alpha_i)Q_i(s, a) + \alpha_i[r_i + \gamma V_i(s')]$
4. agent choose actions action $a_1', \ldots, a_n'$
5. $s = s', a_1 = a_1', \ldots, a_n = a_n'$
6. adjust learning rate $\alpha = (\alpha_1, \ldots, \alpha_n)$

Multi Agent Q-learning Template

equilibrium selection function f : $V_i(s') = f_i(Q_1(s', a), \ldots, Q_n(s', a))$

- We going to study the following equilibrium concept:
  - Value function based (Bellman function based)
    - Single agent Q-learning
    - Independent Q learning by multiple agents
    - Minmax-Q learning (Littman 1994)
    - Nash-Q learning (Hu and Wellman 1998)
    - Friend-or-Foe Q learning (Littman 2001)
    - Correlated Q learning (Greenwald and Hall 2003)
  - Policy gradient methods (direct search for policy)
    - Wind-or-Learn-Fast Policy Hill Climbing (WOLF-PHC) (Policy gradient method)