多智能体学习专题

第一讲：多智能体矩阵博弈

教师：张启超

中国科学院大学
中国科学院自动化研究所

Spring, 2023

多智能体学习目录

4.18　矩阵博弈求解均衡

介绍强化学习基础知识

https://github.com/zhoubolei/introRL

第2课 MDP，　第4课Q学习，　第5课策略优化

4.23　多智能体强化学习

4.25　多智能体强化学习的应用

本次课程目录

- 背景
  - 多智能体学习基础
  - 多智能体学习的挑战
- 博弈论
  - 矩阵博弈
    - ✓ 计算纳什均衡
    - ✓ 学习最佳对策
  - 演化博弈 <选学>
    - ✓ 演化博弈论
    - ✓ 复制动态方程
- 强化学习基础

课程简介

部分讲义参考以下资源，仅用于教学和交流

1. “Multi-agent AI”，Jun Wang, UCL, https://www.bilibili.com/video/BV1fz4y1S72S?from=search&seid=16756461887437l471

2. “Multi-Agent Machine Learning”, Schwartz, H. M

3. “Multi-agent Systems”，Bo An, Nangyang Technological University https://www.bilibili.com/video/BV16v411v7SF

4. “Advances of Multi-agent Learning”，Yaodong Yang, Huawei R&D UK, https://www.bilibili.com/video/BV14p4y1v7s1

1.1 多智能体学习基础

- 国内外多智能体学习相关人员

多智能体系统研究的历史、现状及挑战

安 波[1,2] 史忠植[2]
[1]新加坡南洋理工大学
[2]中国科学院计算技术研究所

关键词：智能体 多智能体系统 人工智能

多智能体系统(multi-agent systems)由分布式人工智能演变而来，其研究目的是解决大规模、复杂、实时和有不确定信息的现实问题，而这类问题是单个智能体所不能解决的。多智能体系统通常具有自主性、分布性、协调性等特征，并具有自组织能力、学习能力和推理能力。对多智能体系统的研究既包括构建单个智能体的技术，如建模、推理、学习及规划等，也包括使多个智能体协调运行的技术，例如交互通信、协调、合作、协商、调度、冲突消解等。随着互联网技术的发展，多智能体系统中往往会出现一些“自私”的智能体(如电子商务市场的交易方)，因此需要引入博弈论来分析智能体的交互策略，其研究内容包括电子商务、拍卖、机制设计、社会选择理论等。

经过近30年的发展，多智能体系统已经成为国际人工智能领域的前沿和研究热点。在近年来的AAAI人工智能会议(AAAI Conference on Artificial Intelligence)和国际人工智能联合会议(International Joint Conference On Artificial Intelligence, IJCAI)上，

图1 多智能体系统创始人维克托·莱瑟

录用的关于多智能体系统的文章数量一直名列前两位。多智能体系统领域的创始人维克托·莱瑟(Victor Lesser)教授于2009年获得了IJCAI杰出研究奖(IJCAI Award for Research Excellence)。自1995年以来，IJCAI计算机与思维奖(IJCAI Computers and Thought Award)的获奖者有一半以上来自多智能体系统领域。

多智能体系统研究的历史

自1956年约翰·麦卡锡(John McCarthy)在著名的达特茅斯研讨会上提出“人工智能”这一概念后，“智能体”的概念便开始兴起。例如，阿兰·图灵(Alan Turing)提出了用来判断一台机器是否具备人类智能的“图灵测试”。在此测试中，测试者通过监视设备向被测试的实体(也就是我们现在所说的智能体)提问。根据图灵的观点，如果测试者无法区分被测试对象是计算机还是人，那么被测试对象就是智能的。人工智能泰斗马文·明斯基在他的《心智社会》这本书中将智能体描述为实现人类智能的

- 史忠植教授（中科院计算所）：多智能体协同
- 陈小平教授（中国科技大学）：人机交互与多智能体
- 谢广明教授/卢宗青教授（北京大学）：多智能体控制/强化学习理论
- 汪军教授（UCL）：多智能体学习理论
- 安波教授（南洋理工大学）：安全博弈论
- 唐平中教授/张崇洁教授（清华大学）：多智能体拍卖/多智能体学习
- 高阳教授（南京大学）：多智能体强化学习

…

1.1 多智能体学习基础

- 多智能体学习

IJCAI 卓越研究奖: Victor Lesser (2009), Barbara Grosz(2015)

多智能体系统创始人

马萨诸塞大学安姆斯特分校

哈佛大学

自然语言处理与协作式多智能体

IJCAI计算与思想奖:

Sarit Kraus (95), Nicholas Jennings (99), Tuomas Sandholm (03), Peter Stone (07), Vincent Conitzer (11), Ariel Procaccia (15) 均与多智能体系统相关






1.1 多智能体学习基础

- 什么是多智能体系统?

多智能体系统是由分布式人工智能演变而来(Distributed AI)

- A multi-agent system consists of a number of agents, which interact with one another　非静态环境
- In the most general case, agents will be acting on behalf of users with different goals and motivations
  "达成"各自/团队目标
- To successfully interact, they will require the ability to collaborate, cooperate and coordinate with each other, like people do　博弈论

背景

- 什么是多智能体学习？

The study of multi-agent systems in which one or more of the autonomous entities improves automatically through experience[1].

在多智能体系统中引入学习机理

学习目的是什么？

1. 学习均衡
2. 学习协调
3. 学习通信
4. .....

→ 每个智能体优化策略"达成"各自/团队目标

多智能体学习就是指在多个智能体的非静态环境下，以博弈论为指导，结合学习技术来学习每个智能体的交互行为策略，来达成各自/团队的目标。

[1] K. Tuyls and P. Stone: Multiagent Learning Paradigms, 2017.

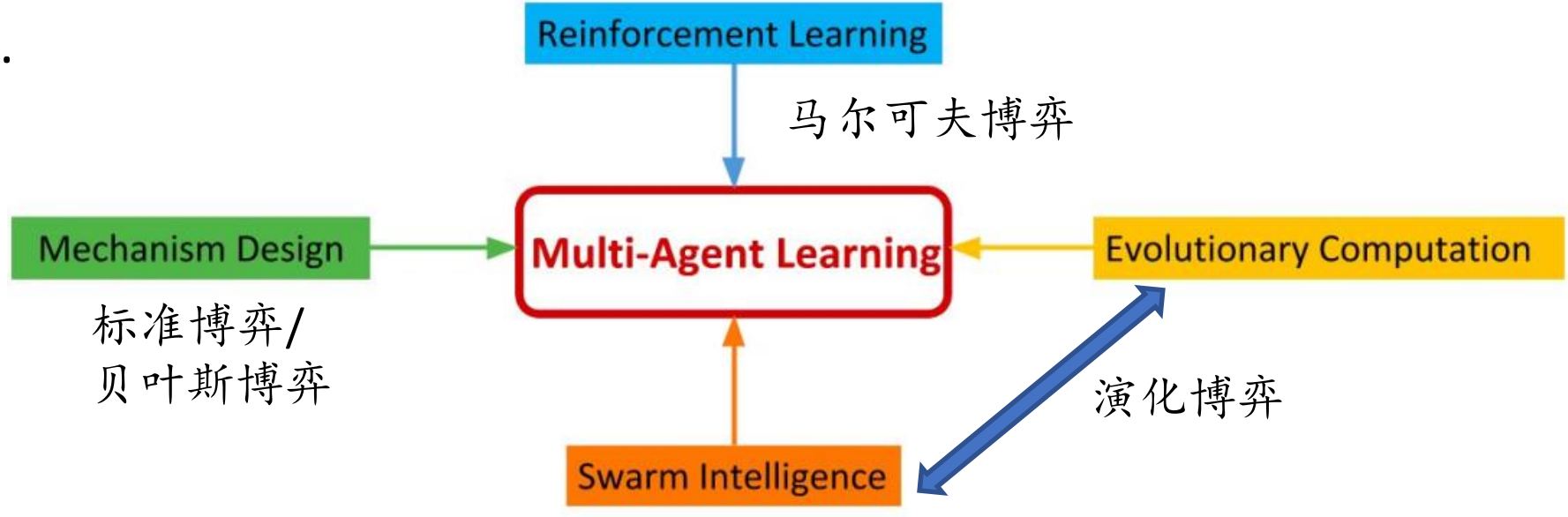
1.2 多智能体学习挑战

然而多智能体学习自身缺乏系统的理论基础

目前涉及的方法有：

- 机理设计
- 演化计算
- 多智能体强化学习
- 群体智能

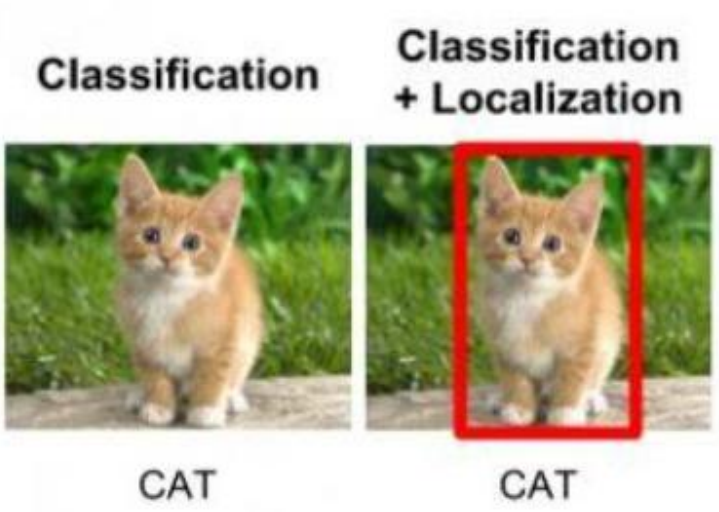
…



1.1 多智能体学习基础

- 多智能体学习

深度学习



数据 → 知识

学习聚焦于感知层面

数据iid

腾讯"绝悟"王者荣耀AI

数据 → 知识 → 决策

交互

时序决策

1.1 多智能体学习基础

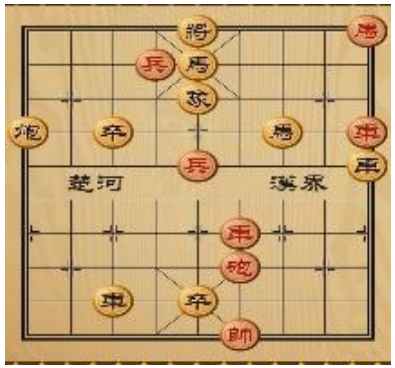
博弈类型

非合作博弈

◆ 合作式博弈
Shared team reward
coordination problem

无人机编队等

◆ 竞争式(零和)博弈
zero-sum games
individual opposing rewards



象棋/围棋/兵棋等

◆ 一般和(混合)博弈
General-sum games

实时对抗类游戏
斗地主等
大国博弈等

1.1 多智能体学习基础

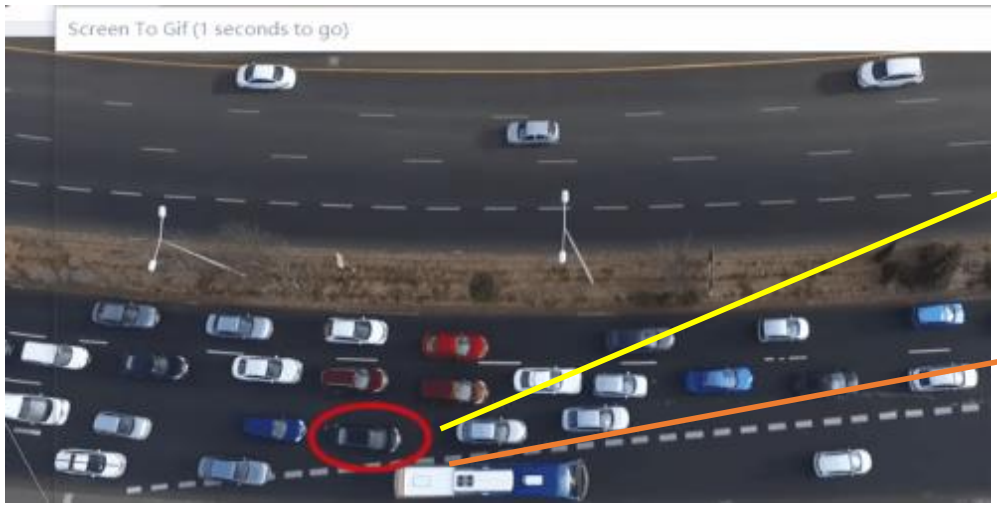
- 多智能体学习面临着单智能体中没有的难题。

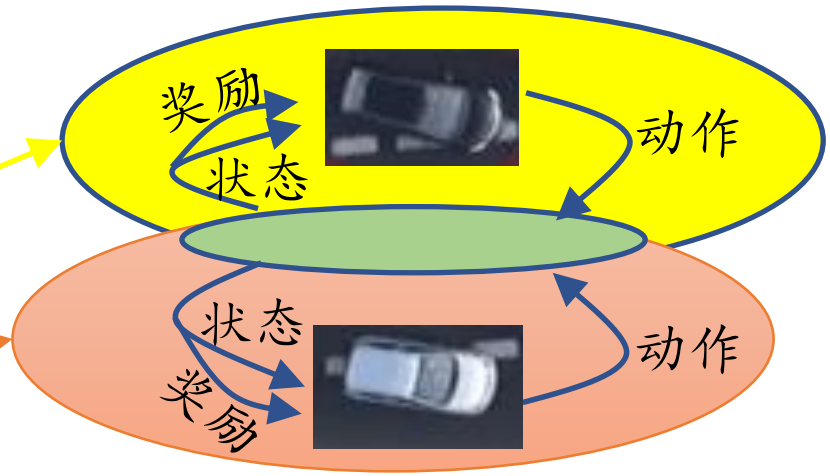
白车 \ 黑车	让行	加速通过
让行	（0，0）	（1，2）
加速通过	（2，1）	（0，0）

1. 纳什均衡难求解；

2. 对于存在多个纳什均衡时，纳什均衡点的选择仍是一个开放问题。

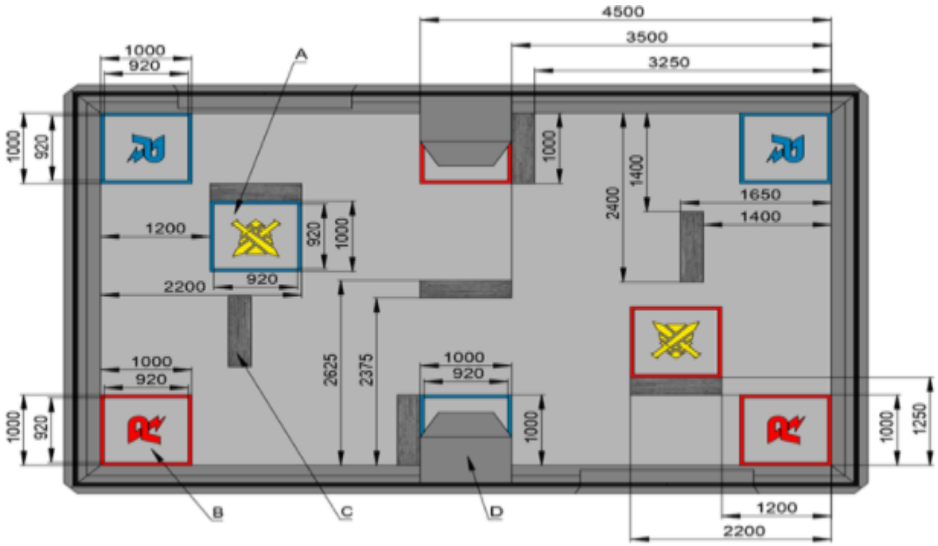
Normal-from game





1.1 多智能体学习基础

ICRA DJI RoboMaster AI Challenge 2021



红方2辆步兵 VS 蓝方2辆步兵

3. 非静态环境其他智能体的行为难以预测；

1.1 多智能体学习基础

中科院自动化所 VS 同济大学

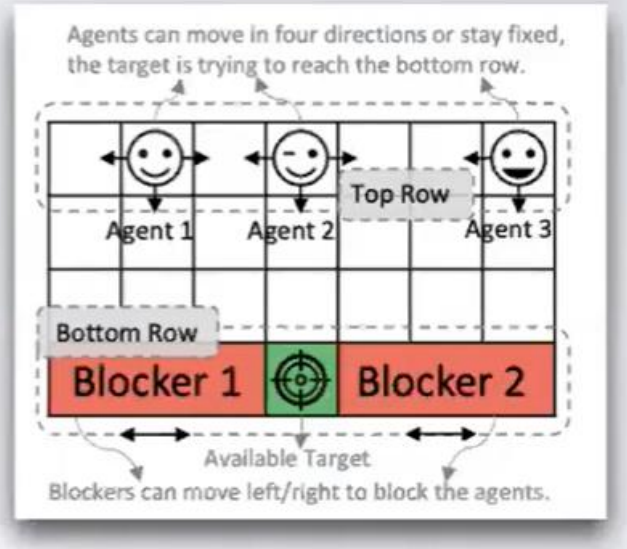
1.2 多智能体学习挑战

4. 随着智能体数量增加存在的规模化问题！

what you think you are doing

Multi-player general-sum games with high-dimensional continuous state-action space

what you are actually doing



Two-player discrete-action game in a grid world.

Copyright from Yaodong Yang

1.2 多智能体学习挑战

07年发展情况

Available online at www.sciencedirect.com
ScienceDirect
Artificial Intelligence 171 (2007) 365–377
Artificial Intelligence
www.elsevier.com/locate/artint

If multi-agent learning is the answer, what is the question?

Yoav Shoham *, Rob Powers, Trond Grenager

Department of Computer Science, Stanford University, Stanford, CA 94305, USA

Received 8 November 2005; received in revised form 14 February 2006; accepted 16 February 2006

Available online 30 March 2007

Abstract

The area of learning in multi-agent systems is today one of the most fertile grounds for interaction between game theory and artificial intelligence. We focus on the foundational questions in this interdisciplinary area, and identify several distinct agendas that ought to, we argue, be separated. The goal of this article is to start a discussion in the research community that will result in firmer foundations for the area.[1]

© 2007 Published by Elsevier B.V.

For the field to advance one cannot simply define arbitrary learning strategies, and analyze whether the resulting dynamics converge in certain cases to a Nash equilibrium or some other solution concept of the stage game. This in and of itself is not well motivated.

1.2 多智能体学习挑战

19年发展情况

Multi-Agent Hide and Seek, 2019, OpenAI.

1.2 多智能体学习挑战

1. 非静态环境

2. 均衡难求解

3. 均衡难协调

4. 规模化问题

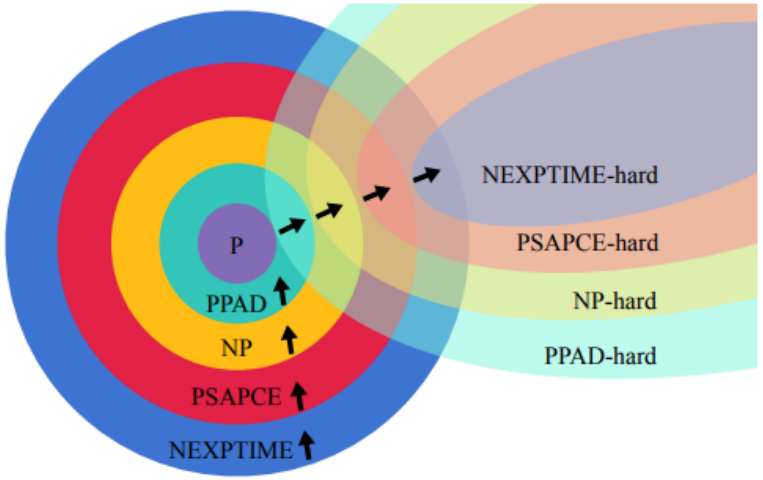


Figure 5: The landscape of different complexity classes. Relevant examples are 1) solving the NE in a two-player zero-sum game, *P*-complete (Neumann, 1928), 2) solving the NE in a general-sum game, *PPAD*-hard (Daskalakis et al., 2009), 3) checking the uniqueness of the NE, *NP*-hard (Conitzer and Sandholm, 2002), 4) checking whether a pure-strategy NE exists in a stochastic game, *PSPACE*-hard (Conitzer and Sandholm, 2008), and 5) solving Dec-POMDP, *NEXPTIME*-hard (Bernstein et al., 2002).

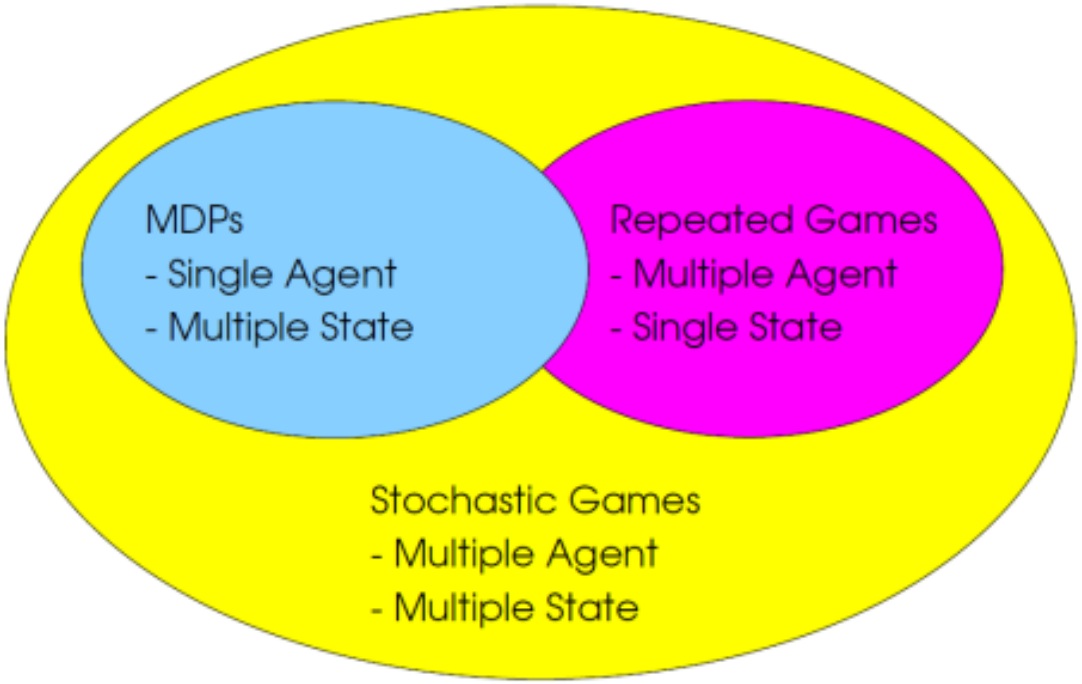
https://arxiv.org/pdf/2011.00583.pdf

本次课程目录

- 背景
  - 多智能体学习基础
  - 多智能体学习的挑战
- 博弈论
  - 矩阵博弈
    - ✓ 计算纳什均衡
    - ✓ 学习最佳对策
  - 演化博弈 <选学>
    - ✓ 演化博弈论
    - ✓ 复制动态方程
- 强化学习基础

矩阵博弈

$< S, A, P, R, \gamma >$



$< S, A, P, R, \gamma, N >$

$< A, R, N >$

一步状态

状态即动作

2.1 矩阵博弈

博弈论研究的是多个理性智能体的交互决策问题。

矩阵博弈的三大元素：

博弈矩阵

- 玩家(智能体)集合 $N = \{1,2,\dots,n\}$
- 动作集合 $A = \{A_1, A_2, \dots, A_n\}$
- 收益（效用）函数 Payoff (utility)

$$u = (u_1, u_2, \dots, u_n)$$
$$u_1: A_1 \times A_2 \to \mathbb{R}$$
$$u_2: A_1 \times A_2 \to \mathbb{R}$$

[2]Myerson, Roger B. (1991). *Game Theory: Analysis of Conflict,* Harvard University Press,

2.1 矩阵博弈

对于经典矩阵博弈问题，一般假设玩家已知：

- 环境是完全已知的
- 效用函数是完全已知的

理性智能体会选择最大化自身收益函数的动作

2.1 矩阵博弈

例1：智猪博弈

		小猪	
		等待	按钮
大猪	等待	0, 0	9, -1
	按钮	4, 4	5, 1

$N = \{1: 大猪, 2: 小猪\}$, $A = \{等待，按钮\}$

例2：囚徒困境

		囚徒2	
		坦白	抵赖
囚徒1	坦白	-4，-4	0，-10
	抵赖	-10，0	-1，-1

$N = \{1: 囚徒1, 2: 囚徒2\}$, $A = \{坦白，抵赖\}$

2.1 矩阵博弈

- 纯策略：确定性动作

  $a_1 \in A_1 = \{坦白, 对抗\}$

  $a_2 \in A_2 = \{坦白, 对抗\}$

	坦白c	抵赖d
坦白c	-4， -4	0， -10
抵赖d	-10， 0	-1， -1

- 混合策略: 纯策略上的概率分布

  $x = (p_c, p_d),\ p_c \in [0,1], p_d \in [0,1], p_c + p_d = 1$

  $y = (q_c, q_d),\ q_c \in [0,1], q_d \in [0,1], q_c + q_d = 1$

- 期望收益 $V_1 = x^T R_1 y$ $V_2 = x^T R_2 y$

  $V_1 = p_c q_c R_1(c,c) + p_c q_d R_1(c,d) + p_d q_c R_1(d,c) + p_d q_d R_1(d,d)$

  $V_2 = p_c q_c R_2(c,c) + p_c q_d R_2(c,d) + p_d q_c R_2(d,c) + p_d q_d R_2(d,d)$

  $x = (0.1, 0.9)$ $y = (0.5, 0.5)$ $V_1 = ?$ $V_2 = ?$

2.1 矩阵博弈

- 优势策略(Dominant Strategy)

无论对方采取什么策略，己方都是最优的策略

玩家2

玩家1		坦白c	抵赖d
	坦白c	-4，-4	0，-10
	抵赖d	-10，0	-1，-1

$-4 > 10$ 坦白

$0 > -1$ 坦白

若玩家2选择坦白

玩家1选择坦白收益为-4，选择抵赖收益为-10，选择坦白

若玩家2选择抵赖

玩家1选择坦白收益为0，选择抵赖收益为-1，选择坦白

对于玩家1而言，坦白就是其优势策略

2.1 矩阵博弈

- 最优反应(Best-response)

  - Given $a_{-i} \in A_1 \times \cdots \times A_{i-1} \times A_{i+1} \times \cdots \times A_n$
  - a_i is best response to a_{-i} $\Leftrightarrow u_i(a_i, a_{-i}) \geq u_i(a_i', a_{-i}), \forall a_i' \in A_i$

	坦白c	抵赖d
坦白c	-4，-4	0，-10
抵赖d	-10，0	-1，-1

针对玩家2坦白的策略，玩家1选择坦白是最优反应

针对玩家2抵赖的策略，玩家1选择坦白是最优反应

对玩家2任意策略，坦白均是最优反应，则坦白为优势策略

a_i is dominant strategy $\Leftrightarrow$ Given any a_{-i}, a_i is best response

2.1 矩阵博弈

对于仅一方存在优势策略的游戏

60%的人选择低消品，40%的人选择奢侈品

如果两家公司进军同一领域，公司1占80%市场份额，公司2占20%市场份额

公司2

公司1	低消品	奢侈品
低消品	0.48, 0.12	0.6，0.4
奢侈品	0.4，0.6	0.32, 0.08

Marketing game

如果均不存在优势策略呢？

公司1存在优势策略：低消品，而公司2并不存在优势策略

公司2会假设公司1选择优势策略，那么它将选择奢侈品策略

2.1 矩阵博弈

- 纳什均衡(Nash Equilibrium)

任何玩家都不能通过独自改变策略而获益的策略组合，
即所有玩家均处于最优反应的策略组合(最优反应不动点)

Definition

- A joint strategy (or strategy profile) $a \in A$ is a Nash Equilibrium $\Leftrightarrow$ a_i is best response to a_{-i} holds for every player i

给定一个策略组合$a = (a_1, a_2, \dots, a_n) \in A_1 \times A_2 \times \cdots \times A_n$
若$V_i(a_1, a_2, \dots a_i, \dots a_n) \geq V_i(a_1, a_2, \dots a_i', \dots a_n), \forall a_i' \in A_i, \forall i \in N$
那么策略组合a是一个纳什均衡。

	坦白c	抵赖d
坦白c	2，2	5，0
抵赖d	0，5	1，1

	猎鹿	猎兔
猎鹿	4，4	0，3
猎兔	3，0	3，3

2.1 矩阵博弈

- 混合策略纳什均衡(Mixed Strategy Nash Equilibrium)

  一个混合策略组合，任何玩家都不能通过独自改变混合策略而使自身期望收益提高

例子:石头剪刀布

玩家1混合策略：$(\frac{1}{3},\frac{1}{3},\frac{1}{3})$

玩家2混合策略：$(\frac{1}{3},\frac{1}{3},\frac{1}{3})$

	石头	剪刀	布
石头	0，0	-1，1	1，-1
剪刀	-1，1	0，0	-1，1
布	1，-1	1，-1	0，0

2.1 矩阵博弈

总结

单个玩家策略

纯策略：确定性动作

混合策略：动作集合上的概率分布

最优反应：给定一组对手策略，$u_i(a_i, a_{-i}) \geq u_i(a_i', a_{-i}), \forall a_i' \in A_i$

优势策略：给定任意一组对手策略，己方都是最优的策略

a_i is dominant strategy ⇔ Given any a_{-i}, a_i is best response

联合策略

纳什均衡：均衡是最优反应的不动点，所有玩家均处于最优反应的策略组合

A joint strategy (or strategy profile) $a \in A$ is a Nash Equilibrium ⇔ a_i is best response to a_{-i} holds for every player i

计算纳什均衡

2.1.1 计算纳什均衡

对于纯策略纳什均衡

		小猪	
		等待	按钮
大猪	等待	0, 0	9, -1
	按钮	4, 4	5, 1

- BR(按钮，等待) = (按钮，等待)
- BR(按钮，按钮) = (等待，等待)
- BR(等待，等待) = (按钮，等待)
- BR(等待，按钮) = (等待，等待)

均衡 (状态、点) 是最优反应的不动点

Theorem (Nash, 1951): Every finite game (finite number of players, finite number of pure strategies) has at least one mixed-strategy Nash equilibrium.

2.1 矩阵博弈

求解纳什均衡是非常有挑战性的问题。

- Two-player general-sum normal-form game:
  - Compute NE → PPAD-Hard
  - Count number of NE → #P-Hard
  - Check uniqueness of NE → NP-Hard
  - Guaranteed payoff for one player → NP-Hard
  - Guaranteed sum of agents payoffs → NP-Hard
  - Check action inclusion / exclusion in NE → NP-Hard

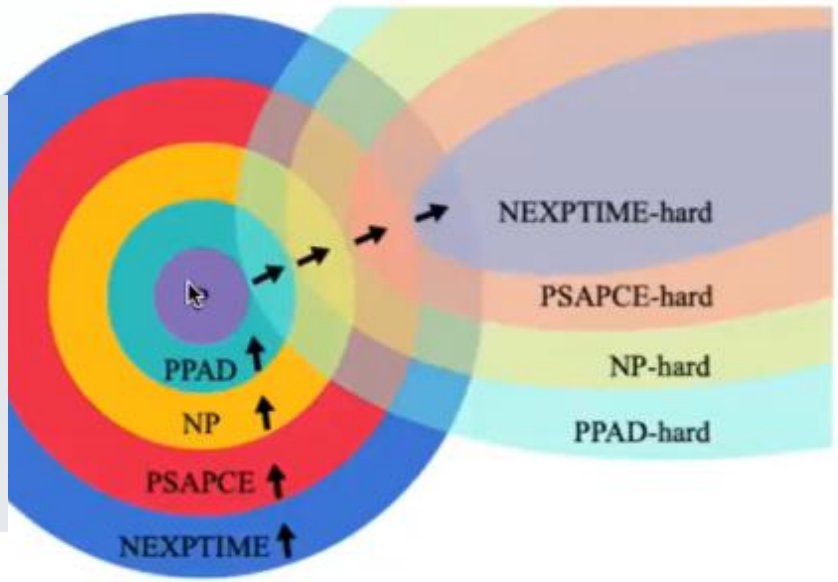


Figure 1.5: Landscape of different complexity classes. Relevant examples are: 1) solving NE in two-player zero-sum game is *P* (Neumann, 1928). 2) solving NE in two-player general-sum game is *PPAD*-hard (Daskalakis et al., 2009). solving NE in three-player zero-sum game is also *PPAD*-hard (Daskalakis and Papadimitriou, 2005). 3) checking the uniqueness of NE is *NP*-hard (Conitzer and Sandholm, 2002). 4) checking whether pure-strategy NE exists in stochastic game is *PSPACE*-hard (Conitzer and Sandholm, 2008). 5) solving Dec-POMDP is *NEXPTIME*-hard (Bernstein et al., 2002).

双人零和博弈可计算求解

多人零和博弈/一般和博弈难以得到解析解

Copyright from Yaodong Yang

2.1.1 计算纳什均衡

计算混合策略纳什均衡

1. 直接计算双人双动作零和博弈的纳什均衡
2. 利用线性规划可以求解双人多动作零和博弈的纳什均衡
3. 利用Lemke-Howson算法可以求解双人一般和标准博弈的纳什均衡

2.1.1 计算纳什均衡

双人零和博弈

玩家2 / 玩家1	第1列	第2列
第1行	-2，2	3，-3
第2行	3，-3	-4，4

如何得到混合策略纳什均衡？

令玩家1选择第1行动作的概率为x_1，则第2行动作的概率为1-x_1

令玩家2选择第1列动作的概率为y_1，则第2列动作的概率为1-y_1

2.1.1 计算纳什均衡

双人零和博弈

玩家2 / 玩家1	第1列	第2列
第1行	-2，2	3，-3
第2行	3，-3	-4，4

假设玩家2选择第1列的概率y_1，使得玩家1选择第1行所获得的收益大于第2行所获得的收益，

玩家1选择第1行收益 $-2y_1 + 3(1-y_1)$

玩家1选择第2行收益 $3y_1 - 4(1-y_1)$

那么玩家1毫无疑问将选择第1行动作，但这并非是混合策略纳什均衡解。

$$-2y_1 + 3(1-y_1) = 3y_1 - 4(1-y_1)$$

2.1.1 计算纳什均衡

双人零和博弈

玩家2 / 玩家1	第1列	第2列
第1行	-2，2	3，-3
第2行	3，-3	-4，4

假设玩家1选择第1列的概率x_1，使得玩家2选择第1列所获得的收益大于第2列所获得的收益，

玩家2选择第1列收益 $2x_1 - 3(1-x_1)$

玩家2选择第2列收益 $-3x_1 + 4(1-x_1)$

那么玩家2毫无疑问将选择第1列动作，但这并非是混合策略纳什均衡解。

$$2x_1 - 3(1-x_1) = -3x_1 + 4(1-x_1)$$

2.1.1 计算纳什均衡

双人零和博弈

玩家2 / 玩家1	第1列	第2列
第1行	-2，2	3，-3
第2行	3，-3	-4，4

可以计算出混合策略纳什均衡

$2x_1 - 3(1-x_1)$
$=-3x_1 + 4(1-x_1)$
$\Rightarrow x_1 = 7/12 \Rightarrow$

玩家1的混合策略为：$\left\{\frac{7}{12}, \frac{5}{12}\right\}$

$-2y_1 + 3(1-y_1)$
$=3y_1 - 4(1-y_1)$
$\Rightarrow y_1 = 7/12 \Rightarrow$

玩家2的混合策略为：$\left\{\frac{7}{12}, \frac{5}{12}\right\}$

2.1.1 计算纳什均衡

双人零和博弈

玩家2 / 玩家1	第1列	第2列
第1行	-2，2	3，-3
第2行	3，-3	-4，4

玩家1的混合策略为：

$$\left\{\frac{7}{12}, \frac{5}{12}\right\}$$

玩家2的混合策略为：

$$\left\{\frac{7}{12}, \frac{5}{12}\right\}$$

玩家1的期望收益为 $V_1 = x^T R_1 y$

$$x_1y_1(-2) + x_1(1\text{-}y_1)3 + (1\text{-}x_1)\,y_1 3 + (1\text{-}x_1)\,(1\text{-}y_1)(\text{-}4) = 1/12$$

玩家2的期望收益为 $V_2 = x^T R_2 y$

$$x_1y_1 2 + x_1(1\text{-}y_1)(\text{-}3) + (1\text{-}x_1)\,y_1(\text{-}3) + (1\text{-}x_1)\,(1\text{-}y_1)(+4) = \text{-}1/12$$

2.1.1 计算纳什均衡

利用线性规划求解双人零和博弈均衡

2.1.1 计算纳什均衡

利用线性规划(LP)来求解纳什均衡

玩家2 / 玩家1	第1列	第2列
第1行	-2，2	3，-3
第2行	3，-3	-4，4

$$R_1 = \begin{bmatrix} -2 & 3 \\ 3 & -4 \end{bmatrix}$$

$$V_1 = x^T R_1 y$$

$$V_2 = y^T R_2^T x$$

$$R_1 = -R_2$$

每个玩家最大化各自的期望收益

$$\max_x \quad x^T R_1 y$$

$$\text{s.t.} \quad 1^T x = 1, x_i \geq 0$$
$$1^T y = 1, y_i \geq 0$$

$$\max_y \quad y^T R_2^T x$$

$$\text{s.t.} \quad 1^T x = 1, x_i \geq 0$$
$$1^T y = 1, y_i \geq 0$$

2.1.1 计算纳什均衡

利用线性规划(LP)来求解纳什均衡

玩家2 / 玩家1	第1列	第2列
第1行	-2，2	3，-3
第2行	3，-3	-4，4

如果玩家1宣称自己的策略为$<x_1, x_2>$, 则玩家2的期望回报为：

$$V_2(a_1) = 2x_1 - 3x_2$$

$$V_2(a_2) = -3x_1 + 4x_2$$

则玩家2对于玩家1策略$<x_1, x_2>$的最优反应为$\max(2x_1 - 3x_2, -3x_1 + 4x_2)$

由于是零和博弈，$\max(2x_1 - 3x_2, -3x_1 + 4x_2) = \min(-2x_1 + 3x_2, 3x_1 - 4x_2)$

对于玩家1策略的目标为：

$$(x_1, x_2) = \text{argmax}_{(x_1,x_2)} \min(-2x_1 + 3x_2, 3x_1 - 4x_2)$$

2.1.1 计算纳什均衡

利用线性规划(LP)来求解纳什均衡

玩家2 / 玩家1	第1列	第2列
第1行	-2，2	3，-3
第2行	3，-3	-4，4

玩家1的策略通过求解如下线性规划问题：

$$\max_x \ V_1$$

$$\text{s.t.} \ -2x_1 + 3x_2 \geq V_1,$$
$$3x_1 - 4x_2 \geq V_1,$$
$$x_1 + x_2 = 1, x_i \geq 0$$

$$\Rightarrow$$

$$\max_x \ V_1$$

$$\text{s.t.} \quad x^T R_1 \geq V_1 1^T,$$
$$1^T x = 1, x_i \geq 0$$

2.1.1 计算纳什均衡

利用线性规划来求解纳什均衡

$$\max_x \min_y \quad x^T R_1 y$$

$$\text{s.t.} \quad 1^T x = 1, x_i \geq 0$$
$$1^T y = 1, y_i \geq 0$$

在对手对抗的最坏情况下最大化自身的期望回报

行玩家的LP问题

$$\max_x \ V_1$$

$$\text{s.t.} \quad x^T R_1 \geq V_1 1^T, \quad 1^T x = 1, x_i \geq 0 \qquad \text{(LP1)}$$

列玩家的LP问题

$$\max_y \ V_2$$

$$\text{s.t.} \quad R_2 y \geq V_2 1, \quad y^T 1 = 1, y_i \geq 0 \qquad \text{(LP2)}$$

2.1.1 计算纳什均衡

定理

如果(x, V_1)对于(LP1)是最优，(y, V_2)对于(LP2)是最优，则(x, y)为零和博弈$(R_1, -R_1)$的纳什均衡，同时行列玩家的期望收益分别为V_1和$V_2 = -V_1$.

$$\max_x \ V_1$$

$$\text{s.t.} \quad x^T R_1 \geq V_1 1^T, \quad 1^T x = 1, x_i \geq 0 \quad \text{(LP1)}$$

对偶问题

$$\min_y \ V_1'$$

$$\text{s.t.} \quad -y^T R_1^T + V_1' 1^T \geq 0, \quad y^T 1 = 1, y_i \geq 0 \quad \text{(LP3)}$$

由于(LP3)为(LP1)的对偶，根据LP的强对偶定理可知：

如果(x, V_1)对于(LP1)是最优，(y, V_1')对于(LP3)是最优，则$V_1 = V_1'$

2.1.1 计算纳什均衡

列玩家的LP问题

$$\min_{y} V_1'$$

$$\text{s.t. } -y^T R_1^T + V_1' 1^T \geq 0, \text{ (LP3)}$$
$$y^T 1 = 1, y_i \geq 0$$

$$V_2 = -V_1'$$
$$R_1 = -R_2$$

$$\max_{y} V_2$$

$$\text{s.t. } R_2 y \geq V_2 1, \quad \text{(LP2)}$$
$$y^T 1 = 1, y_i \geq 0$$

如果(x, V_1)对于(LP1)是最优，(y, V_2)对于(LP2)是最优，则 $V_1 = -V_2$

$$x^T R_1 \geq V_1 1^T \Rightarrow x^T R_1 y \geq V_1$$

$$R_2 y \geq V_2 1 \Rightarrow x'^T R_2 y \geq V_2 \Rightarrow x'^T R_1 y \leq V_1$$

对于列玩家策略y，行玩家策略x是其最优反应。

2.1.1 计算纳什均衡

利用线性规划来求解纳什均衡

玩家1 \ 玩家2	第1列	第2列
第1行	-2，2	3，-3
第2行	3，-3	-4，4

$$R_1 = \begin{bmatrix} -2 & 3 \\ 3 & -4 \end{bmatrix}$$

$$\max_{\pi_1} \min_{\pi_2} V_1$$

假设玩家1的策略为x=<x_1, x_2>，收益值V_1

$$\max_{x_1, x_2} V_1$$

$$x_1 r_{11} + x_2 r_{21} \geq V_1$$
$$x_1 r_{12} + x_2 r_{22} \geq V_1$$
$$x_1 + x_2 = 1, x_1, x_2 > 0$$

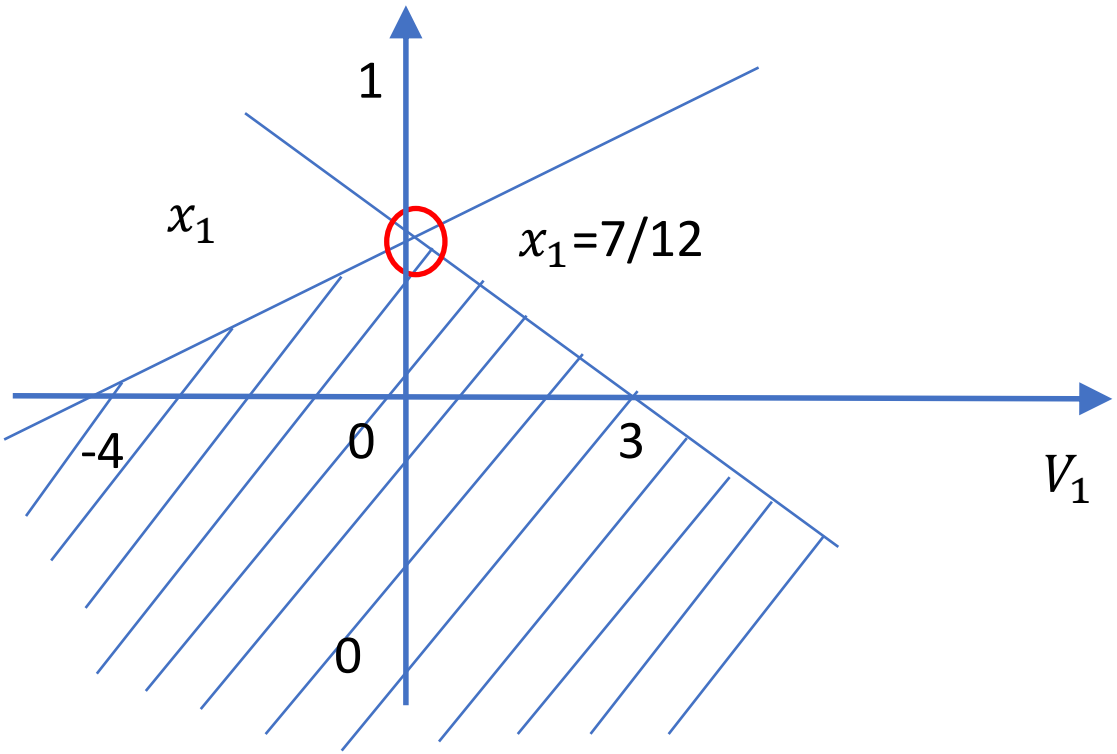
2.1.1 计算纳什均衡

利用线性规划来求解纳什均衡

$$x_1 r_{11} + x_2 r_{21} \geq V_1$$
$$x_1 r_{12} + x_2 r_{22} \geq V_1$$
$$x_1 + x_2 = 1, x_1, x_2 > 0$$

$$\Rightarrow$$

$$-5x_1 + 3 \geq V_1$$
$$7x_1 - 4 \geq V_1$$
$$0 \leq x_1 \leq 1$$



玩家1的混合策略为：

$$\left\{\frac{7}{12}, \frac{5}{12}\right\}$$

V_1=1/12

2.1.1 计算纳什均衡

利用线性规划来求解纳什均衡

假设玩家2的策略为<y_1, y_2>，收益值V_2

$$\max_{y_1, y_2} V_2$$
$$y_1(-r_{11}) + y_2(-r_{12}) \geq V_2$$
$$y_1(-r_{21}) + y_2(-r_{22)} \geq V_2$$
$$y_1 + y_2 = 1, y_1, y_2 > 0$$

$R_2 = -R_1$

玩家2的混合策略为：

$$\left\{\frac{7}{12}, \frac{5}{12}\right\}$$

V_2=-1/12

2.1.1 计算纳什均衡

作业练习：利用线性规划来求解纳什均衡

玩家2 / 玩家1	石头	剪刀	布
石头	0，0	1，-1	-1，1
剪刀	-1，1	0，0	1，-1
布	1，-1	-1，1	0，0

猜拳(石头-剪刀-布)游戏

玩家2想最大化自身期望收益，等同于最小化玩家1的期望收益。

2.1.1 计算纳什均衡

利用线性规划来求解纳什均衡

玩家1的优化问题

$$\max_{\{x_1, x_2, x_3\}} V_1$$

$$x_1 r_{11} + x_2 r_{21} + x_3 r_{31} \geq V_1$$
$$x_1 r_{12} + x_2 r_{22} + x_3 r_{32} \geq V_1$$
$$x_1 r_{13} + x_2 r_{23} + x_3 r_{33} \geq V_1$$
$$x_1 + x_2 + x_3 = 1,\ x_1, x_2, x_3 > 0$$

$$V_2 = -V_1$$

玩家2的优化问题

$$\min_{\{y_1, y_2, y_3\}} V_1$$

$$y_1 r_{11} + y_2 r_{12} + y_3 r_{13} \leq V_1$$
$$y_1 r_{21} + y_2 r_{22} + y_3 r_{23} \leq V_1$$
$$y_1 r_{31} + y_2 r_{32} + y_3 r_{33} \leq V_1$$
$$y_1 + y_2 + y_3 = 1,\ y_1, y_2, y_3 > 0$$

2.1.1 计算纳什均衡

总结：

利用线性规划来求解纳什均衡，两个玩家的线性规划是独立的，并不考虑对手的行为。

只满足收敛性，不满足合理性。收敛性指玩家策略可以收敛到混合策略纳什均衡。

LP方法是否可以用于求解双人一般和博弈问题？

由于两个玩家的收益矩阵不再满足零和关系，LP难以求解一般和博弈

2.1.1 计算纳什均衡

在多智能体强化学习算法中的合理性与收敛性：

合理性（rationality）是指在对手使用一个恒定策略的情况下，当前智能体能够学习并收敛到一个相对于对手策略的最优反应。

收敛性（convergence）是指在其他智能体也在学习其策略时，当前智能体能够学习并收敛到一个稳定策略(纳什均衡)。

2.1.1 计算纳什均衡

利用*Lemke–Howson*求解双人一般和博弈均衡

（选学）

2.1.1 计算纳什均衡

双人一般和博弈

对应前面x和y

针对双人一般和矩阵博弈(A_1, A_2)，混合策略(s_1^*, s_2^*)为一组混合纳什均衡，当且仅当存在一组(U_1^*, U_2^*)使得$(s_1^*, s_2^*, U_1^*, U_2^*)$为下述双线性规划问题的解。

$$\text{maxmize } \{s_1^T A_1 s_2 + s_1^T A_2 s_2 - U_1 - U_2\}$$

$$\text{s.t.} \quad A_1 s_2 \le U_1 1_n \qquad A_2^T s_1 \le U_2 1_m$$
$$\sum_j s_1^j = 1 \qquad \sum_k s_2^k = 1$$
$$\forall j \in \mathbf{A}_1,\ s_1^j \ge 0 \qquad \forall k \in \mathbf{A}_2,\ s_2^k \ge 0$$

2.1.1 计算纳什均衡

可以将上述问题转化为不含优化目标函数的线性互补问题(LCP，linear complementarity problem)

$$\sum_{k\in A_2} u_1(a_1^j, a_2^k)\cdot s_2^k + r_1^j = U_1^* \qquad \forall j \in \mathbf{A}_1$$

$$\sum_{j\in A_1} u_2(a_1^j, a_2^k)\cdot s_1^j + r_2^k = U_2^* \qquad \forall k \in \mathbf{A}_2$$

松弛变量

$$\sum_{j\in A_1} s_1^j = 1, \quad \sum_{k\in A_2} s_2^k = 1$$

$$s_1^j \geq 0, \quad s_2^k \geq 0$$

$$r_1^j \geq 0, \quad r_2^k \geq 0$$

互补条件，非线性约束

$$r_1^j \cdot s_1^j = 0, \quad r_2^k \cdot s_2^k = 0$$

2.1.1 计算纳什均衡

互补条件的约束的意义：

1. 不会使得U_i^*取无限大的值
2. 任意一个玩家以正概率选择其动作，则松弛变量应为0

LCP问题没有目标函数，是约束满足问题，而非优化问题。

2.1.1 计算纳什均衡

定义: 玩家i的混合策略s_i满足

- 玩家i选择的动作a_i^j 概率为0
- 或者其他玩家$-i$的动作a_{-i}^j是玩家i混合策略s_i的最优反应

则称该策略是被标记的，给予该策略标记$L(s_1) \subseteq a_i^j \cup a_{-i}^j$

若一对策略(s_1, s_2)是完全标记的，即满足 $L(s_1) \cup L(s_2) = A_1 \cup A_2$ 那么它就是双人一般和博弈的一组混合策略纳什均衡。

$$\sum_{k \in A_2} u_1(a_1^j, a_2^k) \cdot s_2^k + r_1^j = U_1^* \qquad \forall j \in \mathbf{A}_1$$

$$\sum_{j \in A_1} u_2(a_1^j, a_2^k) \cdot s_1^j + r_2^k = U_2^* \qquad \forall k \in \mathbf{A}_2$$

$$r_1^j \cdot s_1^j = 0, \quad r_2^k \cdot s_2^k = 0$$

2.1.1 计算纳什均衡

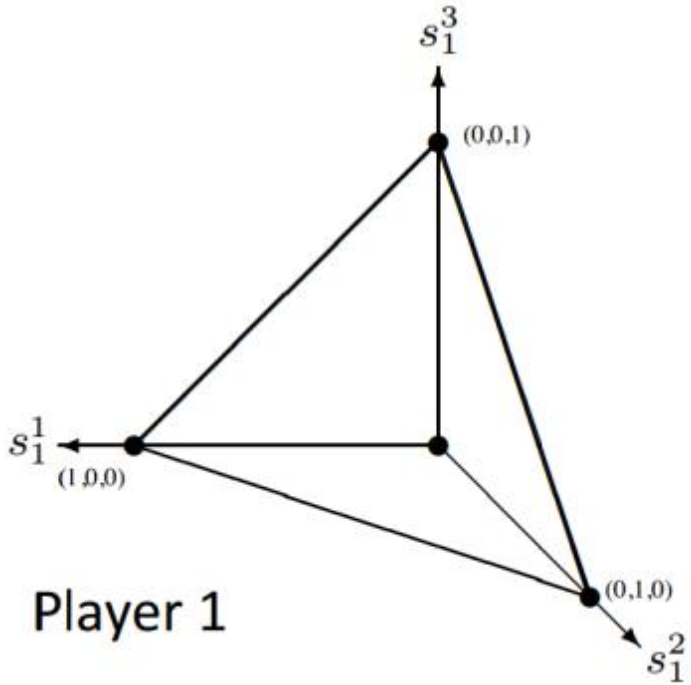
Lemke–Howson algorithm

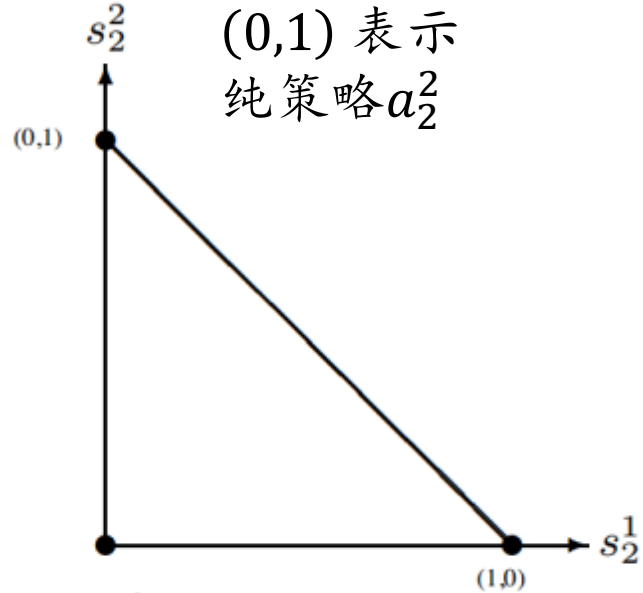
求一般和博弈问题的混合策略纳什均衡，等价于求LCP问题的解。

我们用图示的形式来解释

玩家2

玩家1		
	0, 1	6, 0
	2, 0	5, 2
	3, 4	3, 3





(0,1) 表示纯策略a_2^2

每一个坐标轴上的点对应一个玩家的纯策略

2.1.1 计算纳什均衡

第1步：先找到玩家i相对于玩家j动作的被标记混合策略s_i集合；

第2步：再确认完全标记的一对策略，即纳什均衡策略。

最优反应

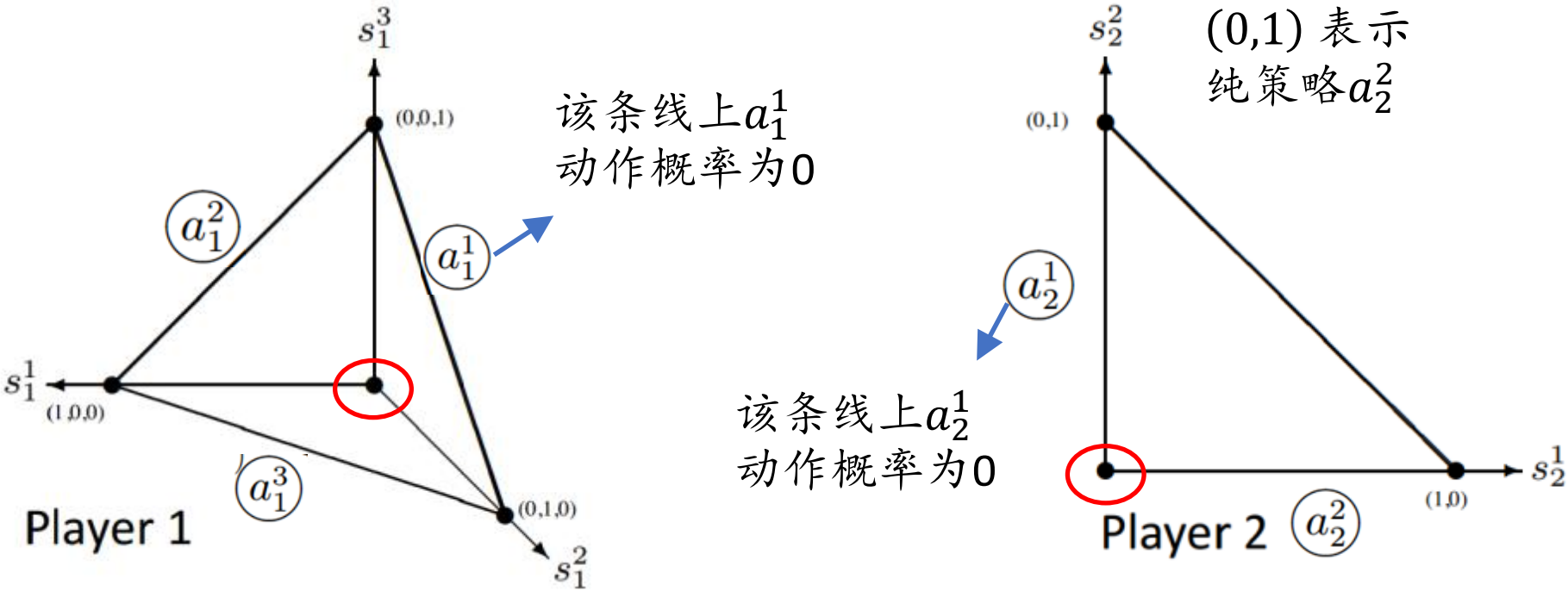
- Given $a_{-i} \in A_1 \times \cdots \times A_{i-1} \times A_{i+1} \times \cdots \times A_n$
- a_i is best response to $a_{-i} \Leftrightarrow u_i(a_i, a_{-i}) \geq u_i(a_i', a_{-i}), \forall a_i' \in A_i$

2.1.1 计算纳什均衡

第1步：先找到玩家i相对于玩家j动作的被标记混合策略s_i集合；

首先我们在图中对两个玩家定义两个虚拟原点，注意我们并不靠率这两个虚拟原点的策略组合，因为不满足概率和为1的约束

先将几个纯策略点连线，找出该线上**概率为0的动作**，并标记



2.1.1 计算纳什均衡

第1步：先找到玩家i相对于玩家j动作的被标记混合策略s_i集合；

- 找出满足**被标记条件第2条即最优反应的混合策略**

若玩家1的动作a_1^1为玩家2混合策略的最优反应，应满足

$$6a_2^2 \geq 2a_2^1 + 5a_2^2$$

$$6a_2^2 \geq 3a_2^1 + 3a_2^2$$

$$a_2^1 + a_2^2 = 1$$

将不等式改变为等式求解：

解得 $a_2^1 = \frac{1}{3}, a_2^2 = \frac{2}{3}$

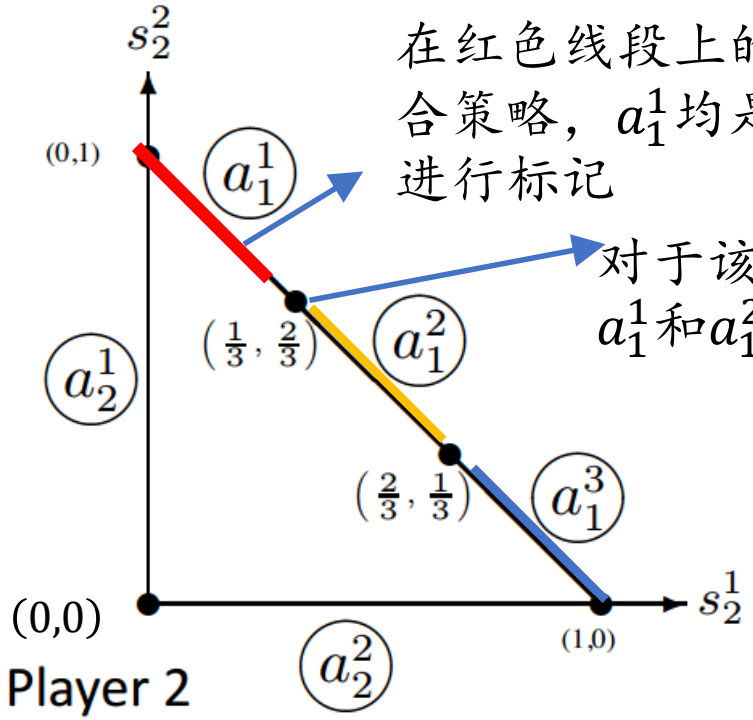
最优反应

以玩家2为例

	a_2^1	a_2^2
a_1^1	0, 1	6, 0
a_1^2	2, 0	5, 2
a_1^3	3, 4	3, 3

2.1.1 计算纳什均衡

第1步：先找到玩家i相对于玩家j动作的被标记混合策略s_i集合；



在红色线段上的玩家2的任意混合策略，a_1^1均是其最优反应，进行标记

对于该点的混合策略，a_1^1和a_1^2均是其最优反应

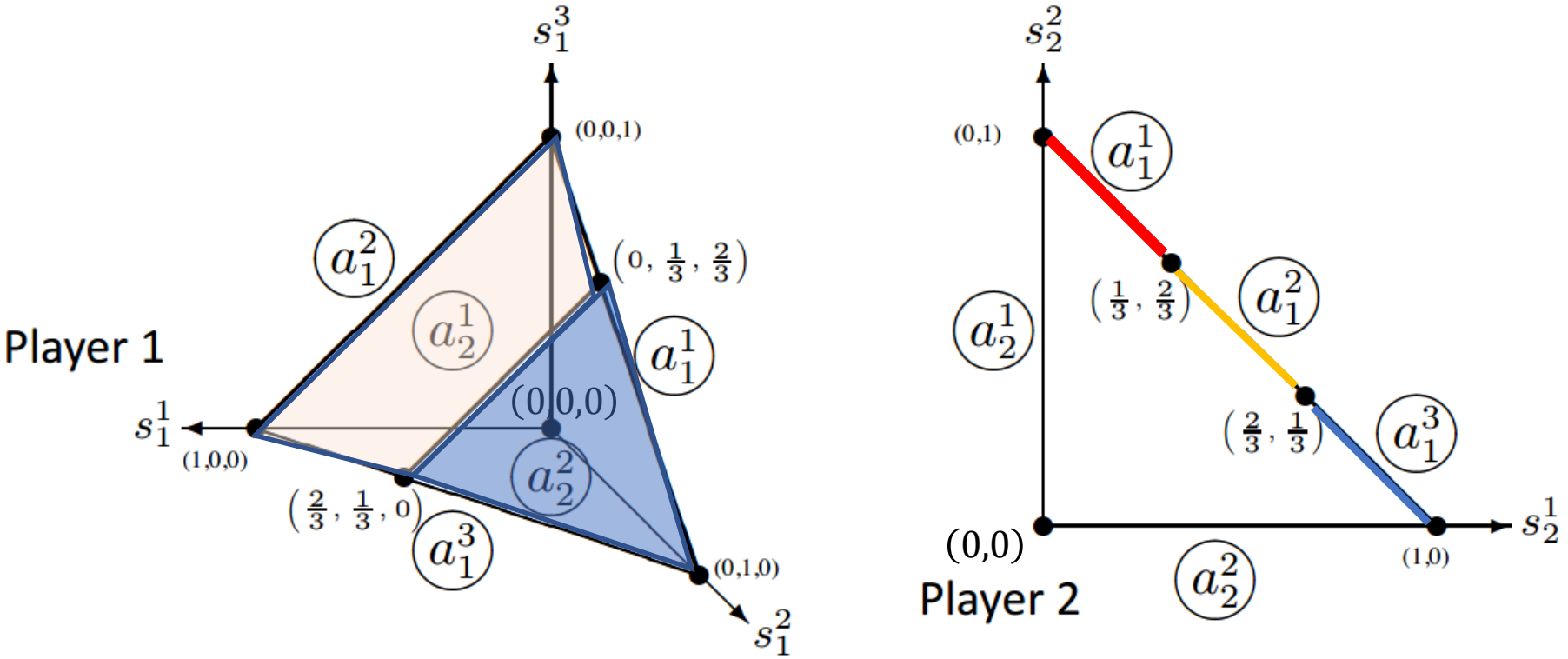
标记策略

a_1^1	0, 1	6, 0
a_1^2	2, 0	5, 2
a_1^3	3, 4	3, 3

对于玩家2红线表示的由(0,1)到(1/3,2/3)混合策略，动作a_1^1为玩家1的最优反应，在图上进行标记

2.1.1 计算纳什均衡

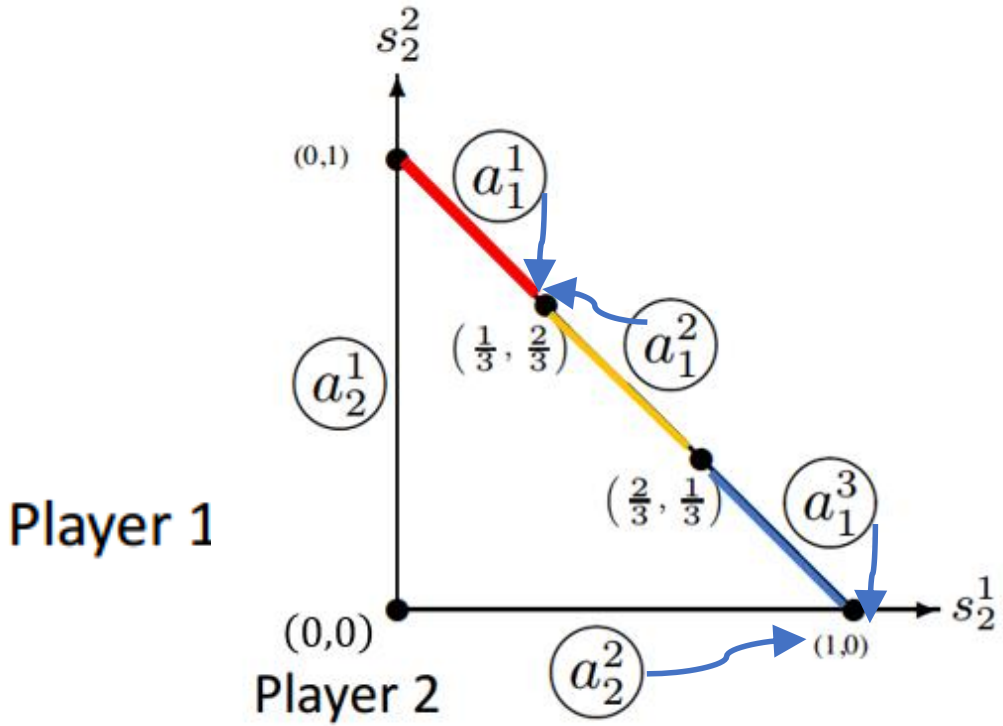
第1步：先找到玩家i相对于玩家j动作的被标记混合策略s_i集合；

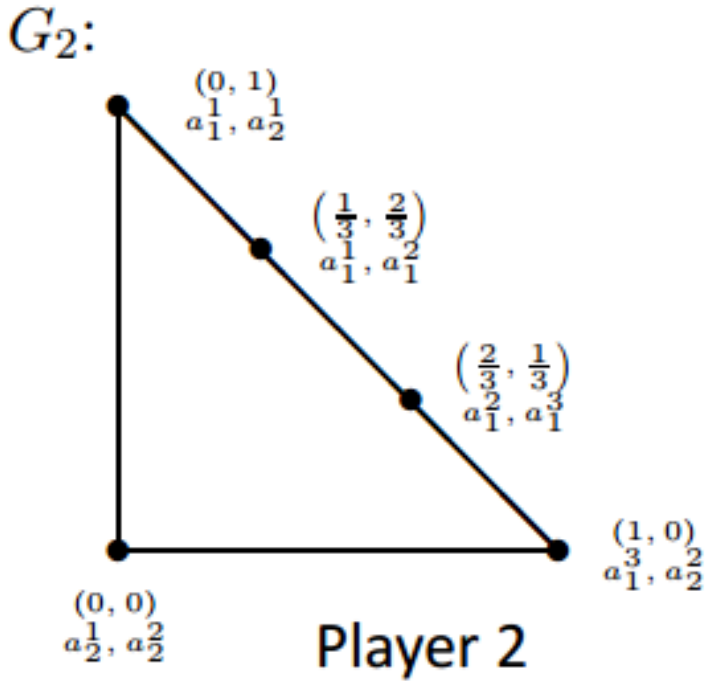


对于玩家1按照同样的步骤在图上标记出满足标记条件的策略

2.1.1 计算纳什均衡

第1步：对于玩家2在图中进行标记 $L(s_2)$





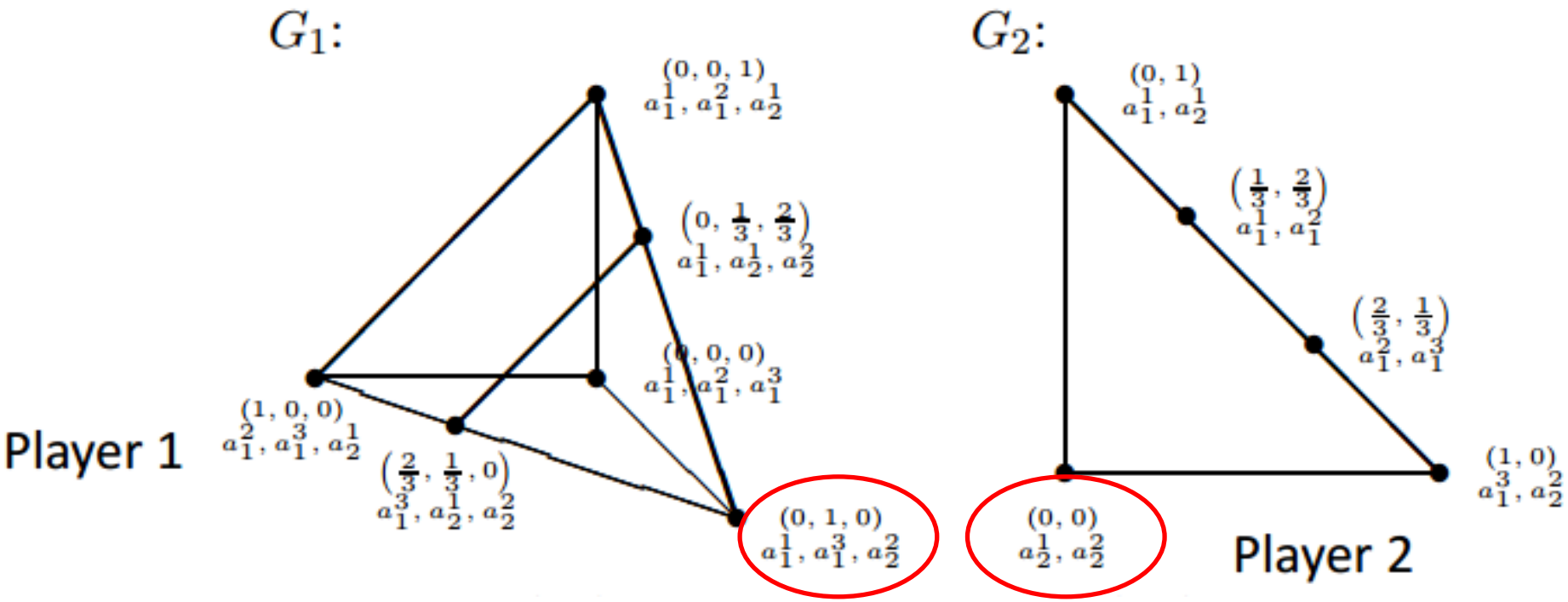
实际上，就是对两个线段相交的点标记上线段上已标记的动作

比如混合策略$\{\frac{1}{3},\frac{2}{3}\}$这个点，被标记的动作就是$a_1^1 \cup a_1^2$

代表的是玩家1的动作a_1^1和a_1^2均是玩家2混合策略$\{\frac{1}{3},\frac{2}{3}\}$的最优反应动作

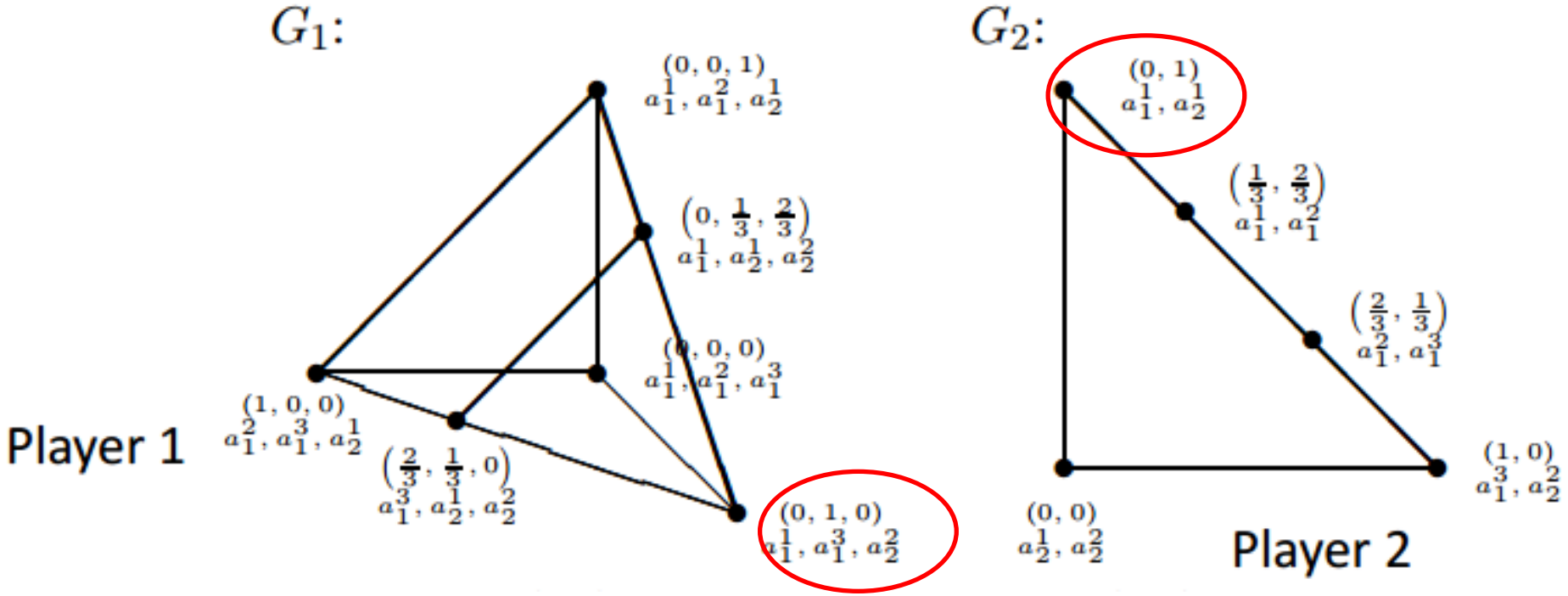
2.1.1 计算纳什均衡

第2步：确认完全标记的一对策略，　$L(s_1) \cup L(s_2) = A_1 \cup A_2$



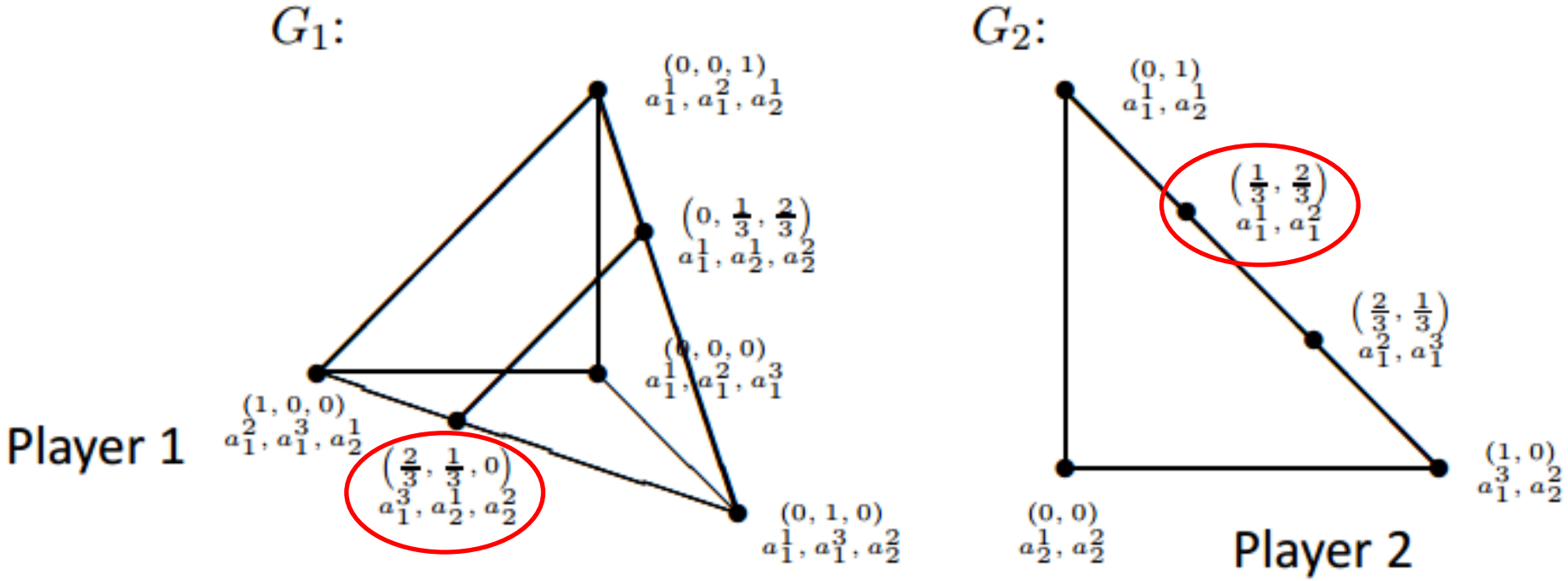
1. 在两个图的原点开始$(0,0)$，　$(0,0,0)$
2. 在G_1改变策略a_1,由原点$(0,0,0)$沿边到新点$(0,1,0)$,发现缺少动作a_1^2,同时存在重复标记动作a_2^2

2.1.1 计算纳什均衡



3. 在G_2改变策略a_2，由原点$(0,0)$沿边移动到新点$(0,1)$，由于另外的节点$(1,0)$也存在重复标记动作a_2^2

2.1.1 计算纳什均衡



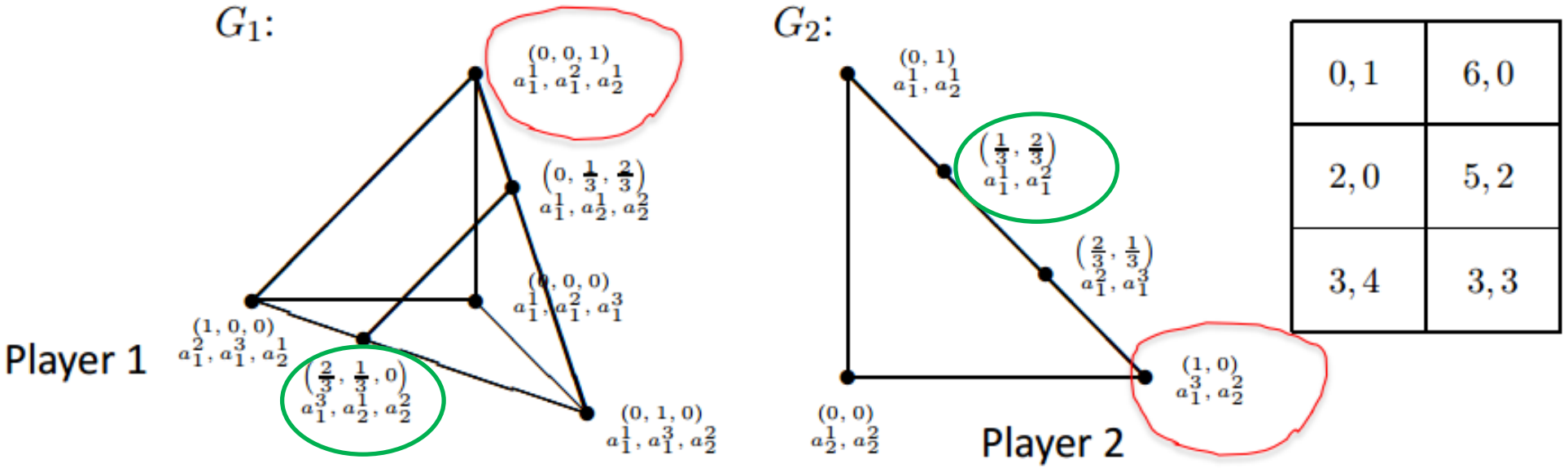
4.在G_1改变策略a_1， 由$(0,1,0)$点移动到新点$(2/3,1/3,0)$

5.在G_2改变策略a_2， 由$(0,1)$点移动到新点$(1/3,2/3)$

a_1^3, a_2^1, a_2^2 $(2/3,1/3,0)$ a_1^1, a_1^2 $(1/3,2/3)$

$L(s_1) \cup L(s_2) = A_1 \cup A_2$ s_1 $\{1/3, 2/3, 2/3\}$ s_2 $\{1/3, 0\}$

2.1.1 计算纳什均衡



Lemke–Howson algorithm 目的就是搜索完全标记的策略对

针对该问题，可以找到3组完全标记的策略对，分别为

[(0, 0, 1), (1, 0)]，　[(0, 1/3, 2/3) , (2/3, 1/3)] ,
[(2/3, 1/3,0) , (1/3, 2/3)]

2.1.1 计算纳什均衡

总结：

1. 对于双人两个动作矩阵博弈，计算纳什均衡的方法

2. 利用线性规划LP求解双人零和博弈的纳什均衡解

3. 利用Lemke–Howson算法求解双人一般和博弈的纳什均衡解

学习最佳对策(最优反应)

2.1.2 学习最佳对策

1. 梯度上升(GA)算法

可用于双人两个动作的一般和博弈(A, B)

$$A=\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \qquad B=\begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix}$$

$$\{x, 1-x\} \qquad \{y, 1-y\}$$

$$V_1 = xya_{11}+x(1-y)a_{12} + (1-x)ya_{21} + (1-x)(1-y)a_{22}$$

$$V_2 = xyb_{11}+x(1-y)b_{12} + (1-x)yb_{21} + (1-x)(1-y)b_{22}$$

2.1.2 学习最佳对策

1. 梯度上升算法

$\max V_1$ 需要已知对手策略 $\max V_2$

梯度

$$\partial V_1 / \partial x = y(a_{11} - a_{12} - a_{21} + a_{22}) + (a_{12} - a_{22})$$

$$\partial V_2 / \partial y = x(b_{11} - b_{12} - b_{21} + b_{22}) + (b_{21} - b_{22})$$

需要已知收益矩阵

更新

$$x_{k+1} = x_k + \alpha \left. \partial V_1(x_k, y_k) \right/ \partial x_k$$

$$y_{k+1} = y_k + \alpha \left. \partial V_2(x_k, y_k) \right/ \partial y_k$$

2.1.2 学习最佳对策

1. 梯度上升算法

收敛性定理

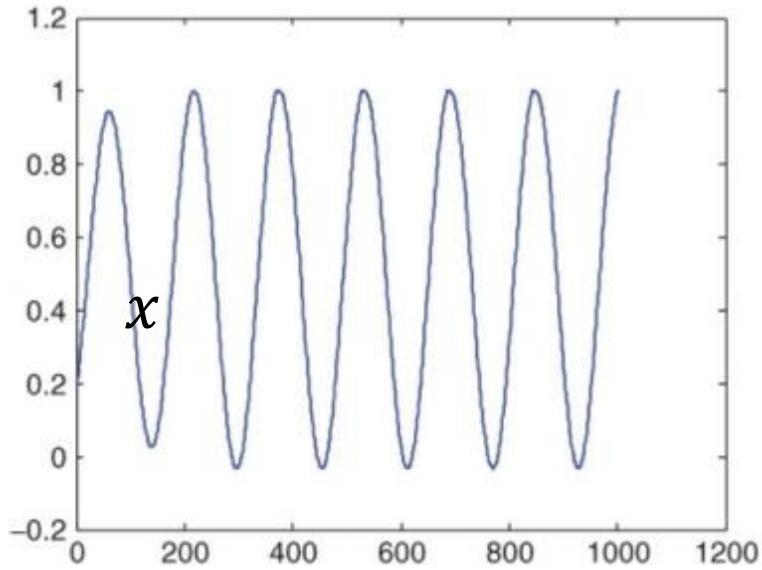
Theorem 3.1

If both players follow infinitesmal gradient ascent (IGA), where $\eta \to 0$, then their strategies will converge to a Nash equilibrium, or the average payoffs over time will converge in the limit to the expected payoffs of a Nash equilibrium.

如果玩家都执行GA算法，当 $\lim_{t\to\infty} \alpha \to 0$，则各自策略将收敛于纳什均衡，或整个过程内的平均回报将随时间收敛于纳什均衡期望回报。

学习率α不好选择

利用GA训练猜硬币游戏，策略难以收敛



2.1.2 学习最佳对策

2. WoLF-IGA (win or learn fast 快速取胜无穷小梯度上升)

玩家获胜时，让学习率较小，
玩家落败时，让学习率较大。

同样具有理论收敛性保障

$$x_{k+1} = x_k + \alpha\eta_1(k)\left.\frac{\partial V_1(x_k, y_k)}{\partial x_k}\right/$$

$$y_{k+1} = y_k + \alpha\eta_2(k)\left.\frac{\partial V_2(x_k, y_k)}{\partial y_k}\right/$$

$$\eta_1(k)=\begin{cases}\eta_{min}, & \text{如果} V_1(x_k, y_k) > V_1(x^*, y_k)\\ \eta_{max}, & \text{其他}\end{cases}$$

需要估计

$$\eta_2(k)=\begin{cases}\eta_{min}, & \text{如果} V_2(x_k, y_k) > V_2(x_k, y^*)\\ \eta_{max}, & \text{其他}\end{cases}$$

(x^*, y^*)为均衡策略，$V_1(x^*, y_k)$需要估计

出现这种情况的主要原因是：y_k是逐渐逼近纳什均衡的，x^*是y^*的最优反应，但不一定是y_k的

2.1.2 学习最佳对策

3. PHC 策略爬山算法

无需已知玩家执行过程中其他玩家的当前策略

1. 初始化：

学习率α, 超参数$\delta \in (0,1]$,

折扣因子γ，初始策略$\pi(s,a) \leftarrow \frac{1}{|A_i|}$，初始$Q(s,a) \leftarrow 0$

2. 迭代：

(a) 根据状态s, 利用探索策略$\pi(s,a)$选择动作a

(b) 观测reward收益r和下一时刻状态s',更新Q

$$Q(s,a) \leftarrow (1-\alpha)Q(s,a)+\alpha\left(r+\gamma \max_{a'} Q(s',a')\right)$$

(c) 更新策略$\pi(s,a)$，将其约束在一个合理的概率分布

$$\pi(s,a) \leftarrow \pi(s,a)+\begin{cases}\delta & \text{如果} a=\operatorname{argmax}_{a'} Q(s,a') \\ \frac{-\delta}{|A_i|-1}, & \text{其他}\end{cases}$$

给予使Q值最大的动作更高的选择概率

2.1.2 学习最佳对策

3. PHC 策略爬山算法

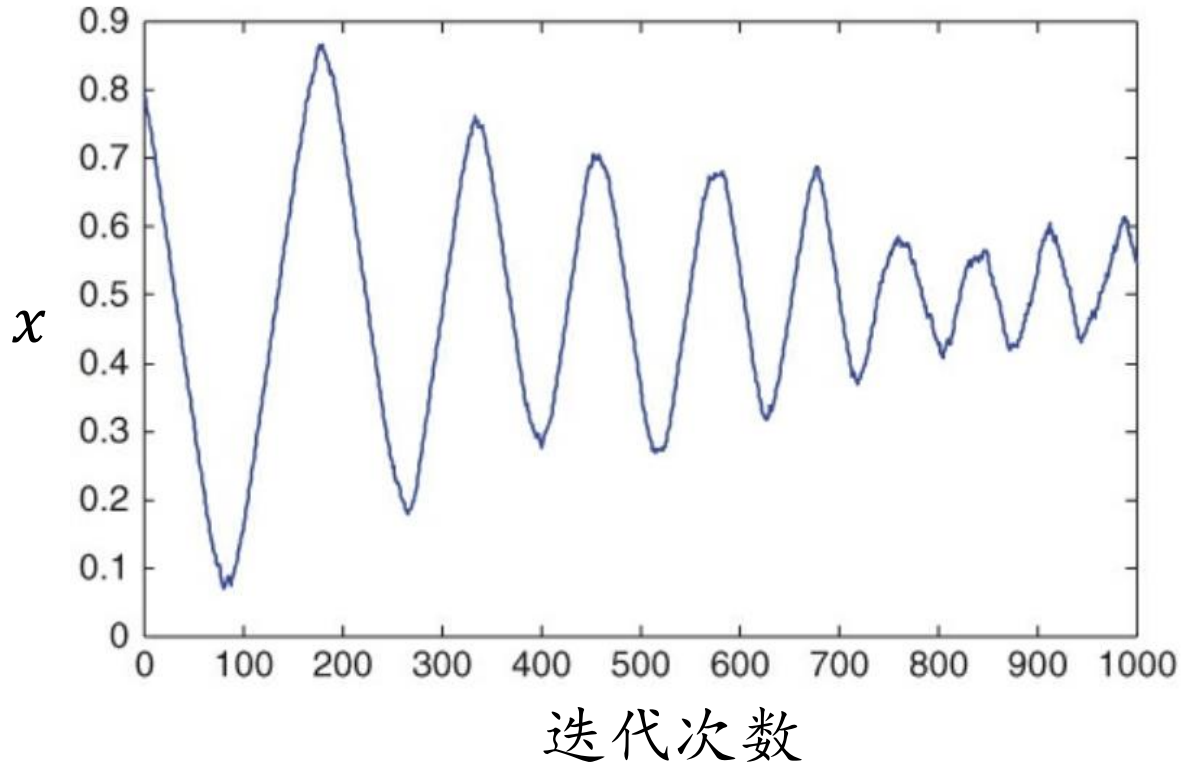
PHC的收敛性与Q学习一致，当其他玩家策略固定，可收敛；当其他玩家也在学习时，算法无法保障收敛性。

猜硬币游戏

$$\alpha = \frac{1}{10 + 0.0001t},$$

$$\varepsilon = \frac{0.5}{1 + 0.0001t},$$

δ=0.0001



2.1.2 学习最佳对策

4. Wolf-PHC

1. 初始化：

学习率$\alpha, \delta_w < \delta_l \in (0,1]$，折扣因子$\gamma$，

初始策略$\pi(s,a) \leftarrow \frac{1}{|A_i|}$，初始$\mathrm{Q(s,a)} \leftarrow 0$，$C(s) \leftarrow 0$

2. 迭代：

(a) 根据状态s，利用探索策略$\pi(s,a)$选择动作a

(b) 观测reward收益r和下一时刻状态s'，更新Q

$$Q(\mathrm{s,a}) \leftarrow (1-\alpha)Q(\mathrm{s,a}) + \alpha\left(r + \gamma \max_{a'} Q(s',a')\right)$$

PHC中需要与最佳对策的值估计进行对比；
Wolf-PHC通过估计平均策略，若当前策略下的Q值优于平均策略，则认为win，否则learn fast

2.1.2 学习最佳对策

4. Wolf-PHC

2. 迭代：

(a) 根据状态s,利用探索策略$\pi(s,a)$选择动作a

(b) 观测reward收益r和下一时刻状态s',更新Q

$$Q(\mathrm{s},\mathrm{a}) \leftarrow (1-\alpha)Q(\mathrm{s},\mathrm{a}) + \alpha\left(r + \gamma \max_{a'} Q(s',a')\right)$$

(c) 估计平均策略$\bar{\pi}(.)$ $C(s) \leftarrow C(s) + 1$

$$\forall a' \in A_i, \bar{\pi}(s,a') \leftarrow \bar{\pi}(s,a') + \frac{1}{C(s)}\left(\pi(s,a') - \bar{\pi}(s,a')\right)$$

(d) 更新策略$\pi(s,a)$，其约束在一个合理的概率分布

$$\pi(s,a) \leftarrow \pi(s,a) + \Delta_{sa}, \quad \Delta_{sa} = \begin{cases} -\delta_{sa} \text{ 如果 } a \neq \operatorname{argmax}_{a'} Q(s,a') \\ \sum_{a' \neq a} \delta_{sa'}, \text{ 其他} \end{cases}$$

给予使Q值最大的动作更高的选择概率

$$\delta_{sa} = \min\left(\pi(s,a), \frac{\delta}{|A_i| - 1}\right)$$

对比当前策略与平均策略的期望收益

$$\delta = \begin{cases} \delta_w \text{ 如果} \sum_{a'} \pi(s,a')Q(s,a') > \sum_{a'} \bar{\pi}(s,a')Q(s,a') \\ \delta_l, \text{其他} \end{cases}$$

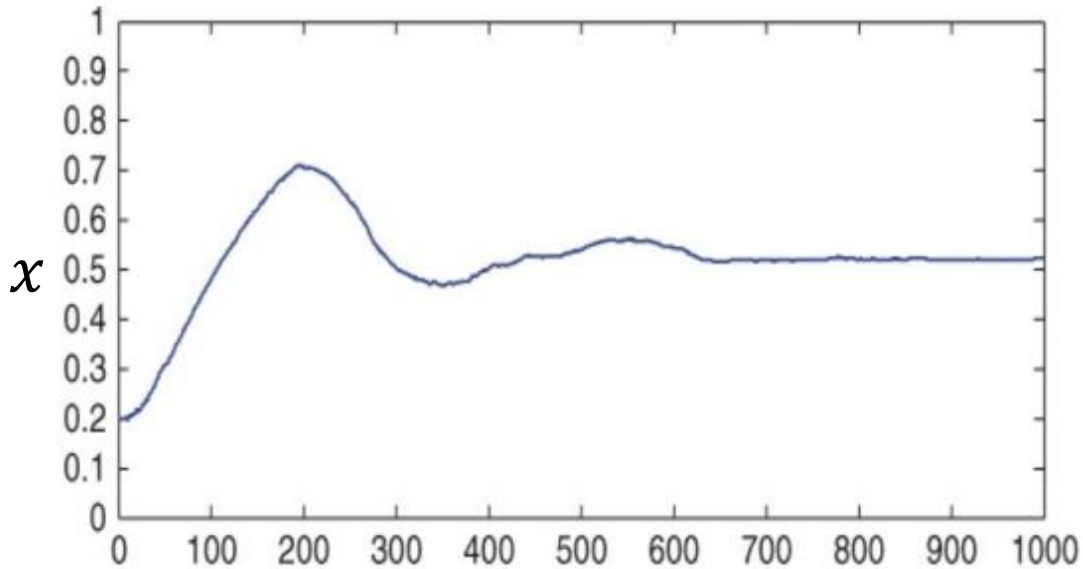
2.1.2 学习最佳对策

4. Wolf-PHC

收敛性

对于双人两个动作的一般和博弈，WoLF-PHC算法在两个玩家同时学习时实际算法具备较好的收敛性。

利用Wolf-PHC训练猜硬币游戏，策略收敛



2.1.2 学习最佳对策

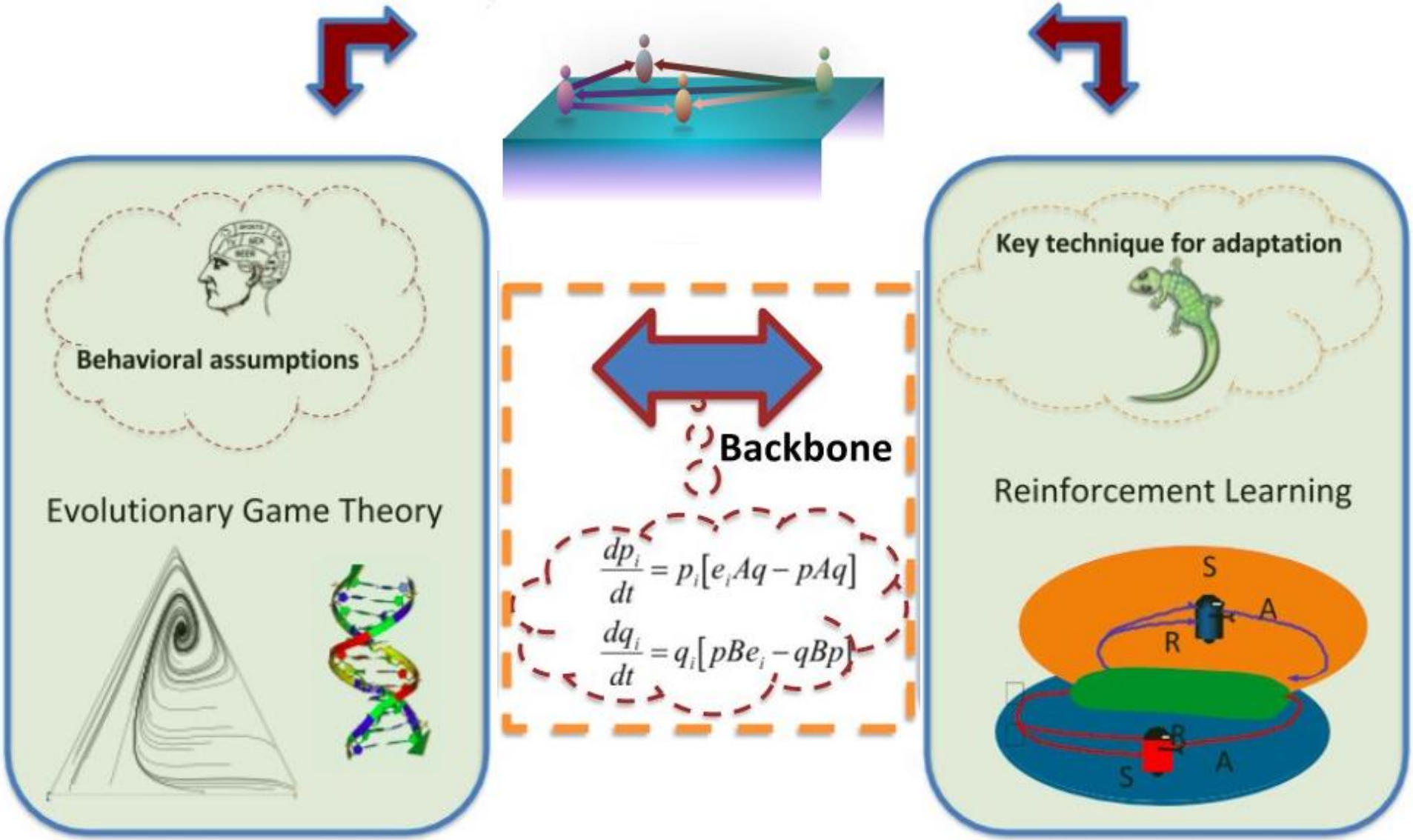
总结

梯度上升算法	需要已知收益矩阵和各自策略	实际任务中两玩家学习收敛困难
快速取胜无穷小梯度上升	需要已知收益矩阵和各自策略	实际任务中两玩家学习收敛困难
策略爬山算法	不需要已知收益矩阵和各自策略	实际任务中两玩家学习收敛困难
快速取胜策略爬山算法	不需要已知收益矩阵和各自策略	实际任务中两玩家学习具有较好收敛效果

课程目录

- 背景
  - 多智能体学习基础
  - 多智能体学习的挑战
- 博弈论
  - 标准博弈论
    - ✓ 计算纳什均衡
    - ✓ 学习最佳对策
  - 演化博弈 <选学>
    - ✓ 演化博弈论
    - ✓ 复制动态方程
- 强化学习基础

2.2.1 演化博弈论

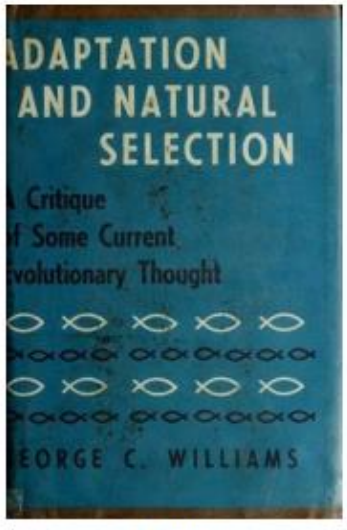


D Balduzzi, T Graepel, et al.， Tutorial:Multi-Agent Learning , ICML, 2019. (DeepMind)

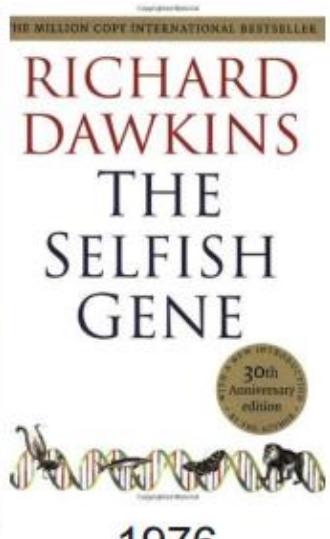
2.2.1 演化博弈论

达尔文的自然选择学说

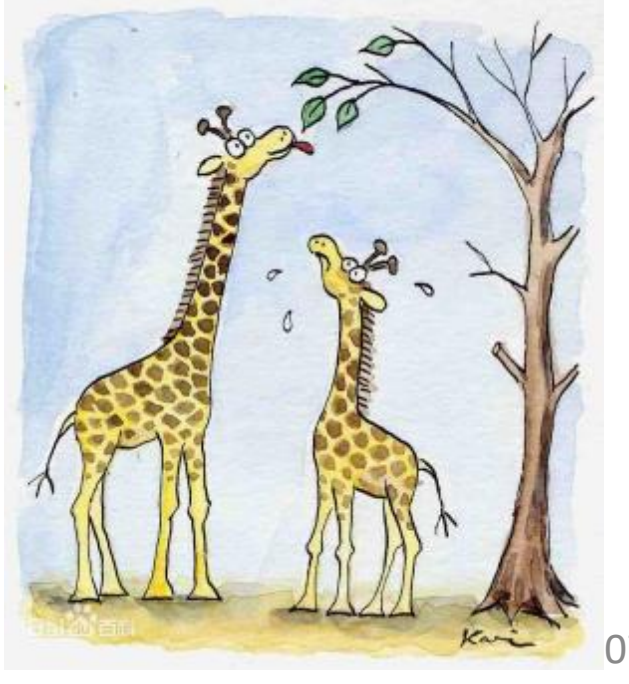
- 一个有机体的基因在很大程度上决定了它在给定环境中的适应性
- 每一代基因交叉和变异，更具适应性的生物体会繁衍更多后代
- 通过自然选择，这使得具备更好适应性的基因增加了它们在群体中的代表性
- 生物演化模型，“自然选择，适者生存”



1966



1976



2.2.1 演化博弈论

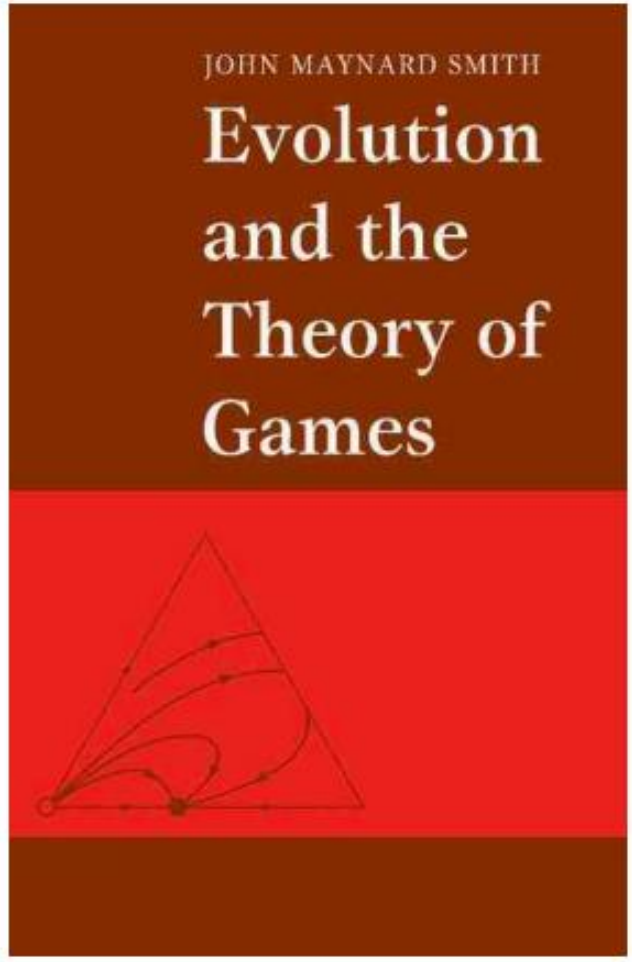
- 1973年生物学家John Maynard Smith 和数学家George R.Price展示了博弈论是如何应用于动物行为的

- 将博弈论应用于动物的想法在当时看起来很奇怪

  博弈论的假设是玩家是绝对理性的，
  而动物往往很难认为是理性的

- Maynard Smith对传统博弈论做出了三点关键改变

  策略
  智能体交互
  均衡

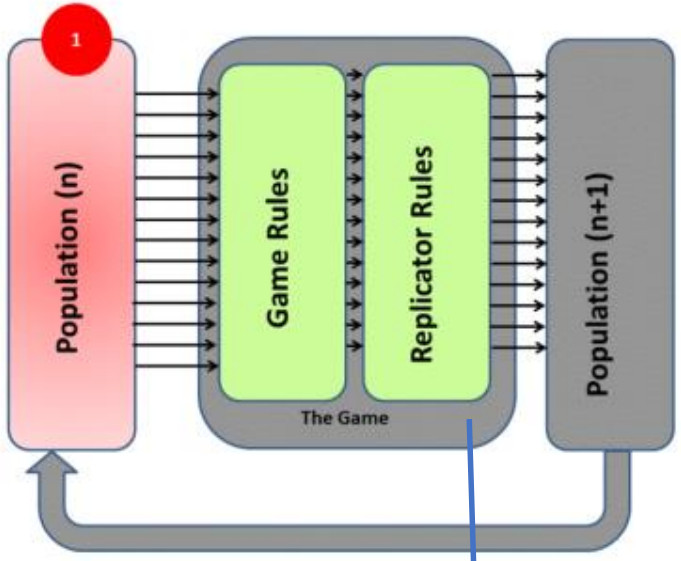


Maynard Smith's 1982 的经典著作

2.2.1 演化博弈论

博弈论

- 每个玩家(个体)选择动作
- 收益依赖所有玩家(个体)的动作
- 对于其他玩家(个体)可能选择什么动作的推理是同一时期发生的



Img source: https://en.wikipedia.org/wiki/Evolutionary_game_theory

演化博弈论

- 对于没有任何玩家(个体)选择动作的情况，演化博弈仍然适用
- 动作可能是无意识发生的
- 什么样的行为会在群体中持续存在?

在生物进化论中，每种生物在繁殖下一代时，都会出现基因的变异。若这种变异是有利于这种生物更好的生活的，那么这种有利变异就会通过环境的筛选，以“适者生存”的方式保留下来。

2.2.1 演化博弈论

核心观点

1. 一种基因代表一类个体，种群行为涉及多种个体间的相互作用

2. 一个个体的适应度取决于它是如何与其他个体相互作用的，无法单独评估单个个体的适应度

3. 个体的适应度必须在其生存的整个种群的背景下进行评估，适应度高的个体繁殖成功率高

类比博弈论

1. 基因—动作　　（遗传基因--策略）
2. 个体的适应度　（相当于收益）
3. 适应度依赖与之交互的个体的策略（相当于收益矩阵）

2.2.1 演化博弈论

举例：甲虫

- 每只甲虫的适应度都依赖于其获取的食物
- 发生变异
  - 变异的甲虫体型变大
  - 大甲虫需要更多食物

那么会发生什么事情？

  - 大甲虫需要更多食物
  - 导致其在种群中适应度变低？
  - 这种变异将会随时间消亡？

2.2.1 演化博弈论

假设甲虫互相竞争食物

大甲虫能有效地获得高于平均水平的食物

假设食物竞争是在一对甲虫间进行的

- 小甲虫vs 小甲虫：平分食物 （5，5）
- 大甲虫vs 小甲虫：大甲虫获得大部分食物 （8，1）
- 大甲虫vs 大甲虫：平分食物，但大甲虫只能获得较少的适应度收益 （3，3）

需要维持体内大量的新陈代谢

2.2.1 演化博弈论

Body-size 博弈

	小甲虫	大甲虫
小甲虫	5，5	1，8
大甲虫	8，1	3，3

甲虫无法主动选择其体型大小

考虑较长时间内，在进化的力量下，随着时间以种群演化的形式发生的策略变化。

2.2.1 演化博弈论

演化稳定策略Evolutionary Stable Strategies (ESS)

- 纳什均衡的概念不再适用

  因为没有个体可以主观改变他们的基因

- 我们期望得到演化稳定策略

定义:如果种群中的每个个体都使用一种策略，同时任何使用不同策略的小群体入侵者都会在几代之后灭绝，那么这种策略就是演化稳定的

一种基因决定的策略，一旦在种群中普遍起来，这种策略就会持续下去

2.2.1 演化博弈论

“小”是一个演化稳定策略么？

	小甲虫	大甲虫
小甲虫	5，5	1，8
大甲虫	8，1	3，3

假设每只甲虫都与其他甲虫随机重复配对

种群数量足够大，两只甲虫之间不会重复相遇

甲虫的适应度=获取食物的平均适应度=繁殖成功率

- 假设种群中有$1-\varepsilon$部分个体是小甲虫，ε(为一个很小的值)为大甲虫，可以认为是，大甲虫入侵了小甲虫种群。
- 在随机相互作用下，小/大甲虫的适应度是多少？

2.2.1 演化博弈论

	小甲虫	大甲虫
小甲虫	5，5	1，8
大甲虫	8，1	3，3

小甲虫

$1-\varepsilon$的概率遇到另一只小甲虫，得到食物 5；
ε的概率遇到一只大甲虫，得到食物 1；
小甲虫的适应度： $5(1-\varepsilon)+1\varepsilon=5-4\varepsilon$

大甲虫

$1-\varepsilon$的概率遇到另一只小甲虫，得到收益8；
ε的概率遇到一只大甲虫，得到收益3；
大甲虫适应度： $8(1-\varepsilon)+3\varepsilon=8-5\varepsilon$

2.2.1 演化博弈论

小甲虫的适应度为5-4ε
大甲虫的适应度为8-5ε

对于小概率ε，大甲虫的适应度超过了小甲虫
因此“小”并不是一个演化稳定策略

“大”是不是一个演化稳定策略？

假设小甲虫以概率ε入侵大甲虫种群

小甲虫的适应度为$5\varepsilon + 1-\varepsilon=1+4\varepsilon$
大甲虫的适应度为$8\varepsilon+3(1-\varepsilon)=3+5\varepsilon$

因此“大”是一个演化稳定策略

2.2.1 演化博弈论

举例：甲虫

◆ 少量大甲虫入侵由小甲虫组成的种群

◆ 在绝大多数竞争中小甲虫仅获得少量食物

◆ 小甲虫的种群无法赶走大甲虫，使其灭绝

因此，“小”并不是演化稳定的策略

2.2.1 演化博弈论

◆ 相反，一个大甲虫的种群可以有效抑制小甲虫的入侵

◆ 大甲虫的适应性非常好

- 一方面大甲虫很小概率遇到大甲虫
- 在大多数竞争中大甲虫占据绝对优势

“大”是演化稳定的策略

	小甲虫	大甲虫
小甲虫	5，5	1，8
大甲虫	8，1	3，3

与传统博弈论中玩家可以改变自身策略不同，甲虫并不能改变其自身体型的大小，但是经过多代演化可以达到相似的效果。

2.2.1 演化博弈论

在什么条件下一个策略是演化稳定的呢？

	A	B
A	a，a	b，c
B	c，b	d，d

种群的生物小概率ε为B，$1-\varepsilon$为A

对于基因A的生物而言：

- 很小概率ε遇到B，获得收益b
- 很大概率1-ε遇到A，获得收益a

A的适应度为： $a(1-\varepsilon)+b\varepsilon$

B的适应度为： $c(1-\varepsilon)+d\varepsilon$

2.2.1 演化博弈论

A是演化稳定的，如果对于所有的ε均满足

$$a(1-\varepsilon)+b\varepsilon > c(1-\varepsilon)+d\varepsilon$$

由于ε较小，因此当a>c时，上式满足；
当a=c时，若b>d，则上式满足。

针对双人双策略的对称矩阵博弈，A是演化稳定的，当满足

- a>c， 或
- {a=c， b>d}

2.2.1 演化博弈论

	A	B
A	a，a	b，c
B	c，b	d，d

- a>c

  A与A对抗至少要优于A与B对抗

  否则，入侵者B将比A在种群中的适应度更高

- {a=c， b>d}

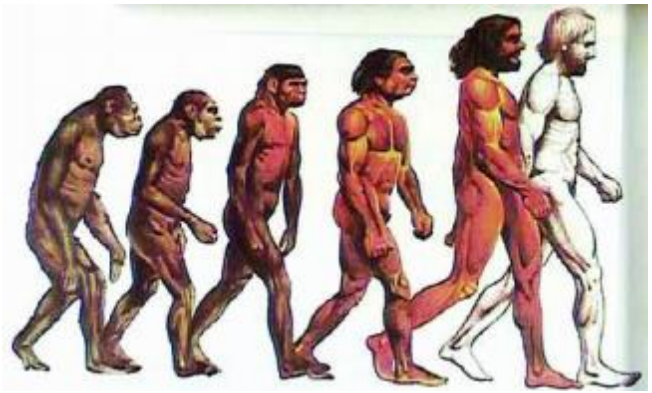
  当A和B分别与A对抗时，性能相当，
  那么需要A与B对抗的优势要强于B与B相互对抗的优势

  否则，B将会压制A，使得A无法演化稳定

2.2.2 复制动态方程

复制动态模拟个体间频繁交互的种群演化，寻找稳定状态

	A	B
A	a，a	b，c
B	c，b	d，d



种群中A的概率为p，B的概率为$1-p$

$$u_1 = ap + b(1-p)$$

$$u_2 = cp + d(1-p)$$

$$\bar{u} = pu_1 + (1-p)u_2$$

复制动态方程为

$$F(p) = \frac{dp}{dt} = p(u_1 - \bar{u})$$

$$= p(1-p)[p(a-c) + (1-p)(b-d)]$$

当$F(p) = 0$时，复制动态稳定状态为p^*=0，p^*=1，$p^* = \frac{(b-d)}{a-b-c+d}$

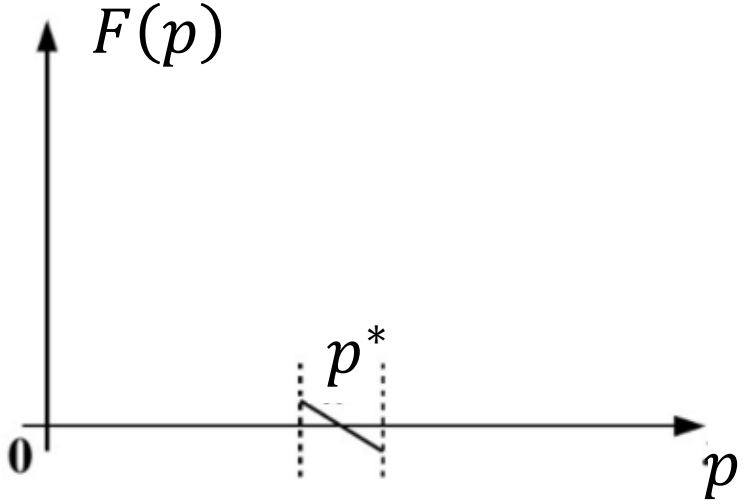
2.2.2 复制动态方程

双人对称博弈

稳定性定理

若$p < p^*$, 为使$p \to p^*$, 应满足$F(p) > 0$
若$p > p^*$, 为使$p \to p^*$, 应满足$F(p) < 0$

$F'(p^*) < 0, p^*$为ESS



2.2.1复制动态方程

鹰鸽博弈

鹰鸽博弈是英国生物学家约翰·梅纳德·史密斯提出的一个博弈论模型，用来解释自然界中的鹰与鸽子两个物种的进化与共存现象

种群中的每只鸟可以选择行为：

1. 好斗的老鹰
2. 温和的鸽子

	鹰	鸽
鹰	-2，-2	6，0
鸽	0，6	3，3

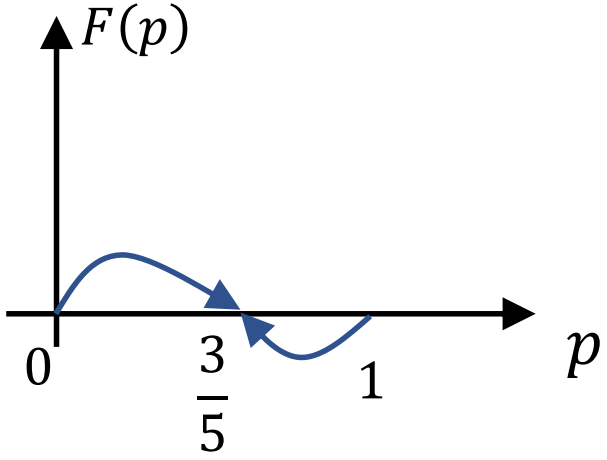
- 总食物为6，战斗的代价为-5
- 当两只鹰同时发现食物的时候，天性决定它们一定要战斗，最后会两败俱伤。二者的收益都是 -2。
- 当两只鸽子相遇的时候，天性要求它们共同分享食物，各自收益都是 3。
- 当鹰和鸽子相遇，鸽子会逃走，鹰独得全部食物，故鹰的收益是 6，鸽子的收益是 0。

2.2.2 复制动态方程

举例

	鹰	鸽
鹰	-2，-2	6，0
鸽	0，6	3，3

p^*=0，p^*=1，$p^*=\frac{(b-d)}{a-b-c+d}=\frac{3}{5}$



$\sigma=\left(\frac{3}{5},\frac{2}{5}\right)$ 是该博弈的ESS

演化博弈

- 演化博弈与标准博弈的区别
- 演化稳定策略
- 利用复制动态方程求解演化稳定策略
- 蚁群算法
- 粒子群算法

...

总结与回顾

纳什均衡

- 简单计算纳什均衡(双人零和博弈，两个动作)
- LP计算纳什均衡 (双人零和博弈，多个动作)
- 利用L-H算法计算纳什均衡(双人一般和博弈)

最佳对策

- 梯度上升 / 快速取胜无穷小梯度上升
- 策略爬山 / 快速取胜策略爬山

演化博弈

- 演化博弈论
- 复制动态方程