

中国科学院大学：专业普及课《计算博弈原理与应用》

第二讲：计算博弈基础知识

李凯

kai.li@ia.ac.cn

2020年9月24日



中国科学院大学
University of Chinese Academy of Sciences



自动化研究所
Institute of Automation

本讲提纲

1 博弈表示方法

2 常见博弈类型

3 博弈的解概念

4 课程设计任务



本讲提纲

1 博弈表示方法

2 常见博弈类型

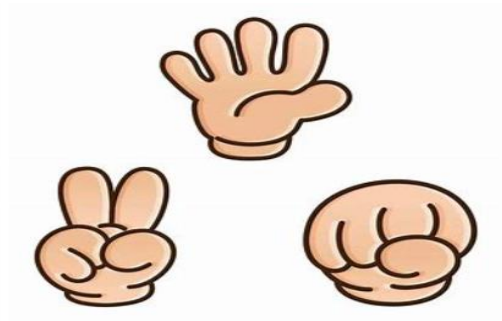
3 博弈的解概念

4 课程设计任务



博弈论 (Game Theory)

- 博弈论的定义
 - 研究互动局势下理性人的策略行为的科学
- 现实世界中的博弈



博弈的基本要素

- **参与人 (Players)** : 是博弈的行为主体, 自然人、智能体、企业、团体等, $N = \{1, 2, \dots, i, \dots, n\}$ 表示参与人集合
 - 石头剪刀布博弈中, $N = \{1, 2\}$
- **策略 (Strategy)** : 参与人 i 的策略集合为 $S_i, s_i \in S_i$, $\mathbf{s} = (s_1, \dots, s_n)$ 为所有参与人的策略组合 (strategy profile), 博弈的策略空间为 $S = \prod_{1 \leq i \leq n} S_i$ (笛卡尔积)
 - $S_i = \{\text{石头}, \text{剪刀}, \text{布}\}$
 - $S = \{(\text{石头}, \text{石头}), (\text{石头}, \text{剪刀}), (\text{石头}, \text{布}), \dots\}$
- **收益 (Utility) 函数**: 又称支付 (Payoff) 函数, 是参与人最关心的, $\mathbf{u} = (u_1, u_2, \dots, u_n), u_i: \mathbf{s} \rightarrow R$
 - $u_1(\text{石头}, \text{剪刀}) = 1$
 - $u_2(\text{石头}, \text{剪刀}) = -1$

纯策略和混合策略

- 参与人在给定情况下只选择一种特定的行动，我们就称该策略为纯（Pure）策略
 - 前面讲的集合 S_i 中的每一个元素就是参与人 i 的一个纯策略
- 参与人在给定情况下以某种概率分布随机选择不同的行动，我们就称该策略为混合（Mixed）策略

混合策略（Mixed Strategy）

假设 i 有 K 个纯策略： $S_i = \{s_{i1}, \dots, s_{iK}\}$ ，概率分布 $\sigma_i = (\sigma_{i1}, \dots, \sigma_{iK})$ 为 i 的一个混合策略， σ_{ik} 是 i 选择纯策略 s_{ik} 的概率

纯策略是混合策略的一个特例

- 对于石头剪刀布博弈来说：
 - 纯策略：只出石头， $\sigma_i = (1, 0, 0)$
 - 混合策略：等概率出石头剪刀布， $\sigma_i = (1/3, 1/3, 1/3)$

混合策略的表示

- 参与人 i 的某一混合策略: $\sigma_i = (\sigma_{i1}, \dots, \sigma_{i2}, \dots, \sigma_{ik})$
- 参与人 i 的混合策略集合: $\Sigma_i, \sigma_i \in \Sigma_i$
- 所有参与人的某一混合策略组合: $\sigma = (\sigma_1, \dots, \sigma_i, \dots, \sigma_n)$
- 博弈的混合策略空间: $\Sigma = \times_i \Sigma_i$
- 参与人 i 的期望收益函数 (Expected Utility) :
 - $v_i(\sigma) = v_i(\sigma_i, \sigma_{-i}) = \sum_{s \in S} p(s) u_i(s)$
 - $p(s) = \prod_j^n \sigma_j(s_j)$
 - 将参与人 i 在所有策略组合下的收益按照该策略组合出现的概率进行加权平均
 - 每个策略组合出现的概率等于各个纯策略概率的乘积
 - σ_{-i} 代表除了参与人 i 之外其他参与人的混合策略组合

用矩阵来表示博弈

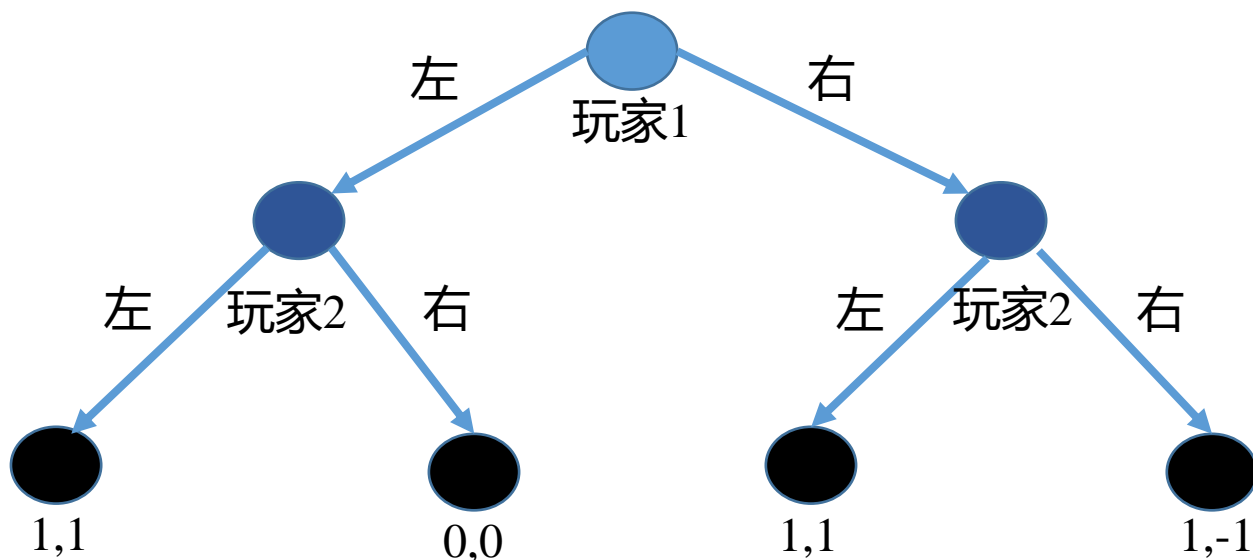
- 通常用来表示参与人**同时选择策略**的博弈，如石头剪刀布、囚徒困境等
- 也称策略式（Strategic-form）表示，或标准式（Normal-form）表示
- 用多维矩阵的方式来表示：
 - 博弈的参与人： $i \in N, N = (1, 2, \dots, n)$ ，决定矩阵维度
 - 每个参与人的纯策略集合： $S_i, i = 1, 2, \dots, n$ ，决定每一维大小
 - 每个参与人的收益： $u_i(s_1, \dots, s_i, \dots, s_n)$ ，决定每一个元素

囚徒A \ 囚徒B	坦白	抵赖
	坦白	抵赖
坦白	2, 2	0, 3
抵赖	3, 0	1, 1

玩家1 \ 玩家2	石头	剪刀	布
	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

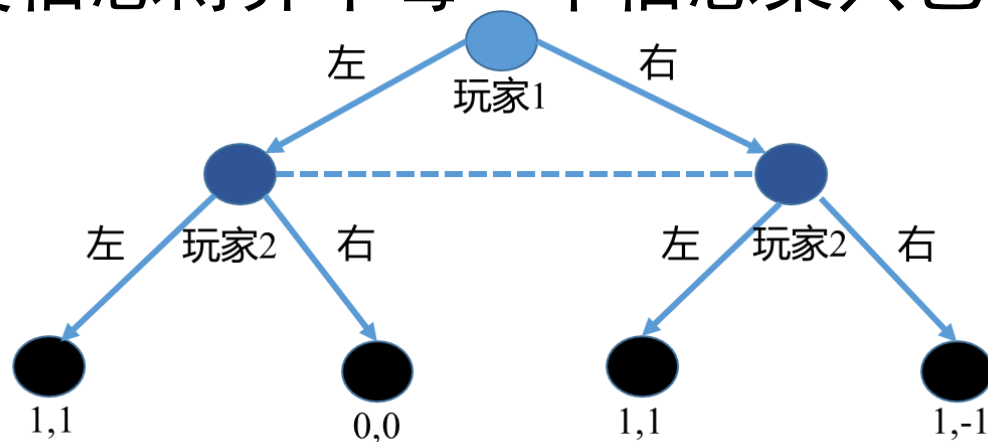
用树来表示博弈

- 通常用来表示参与人**行动有先后顺序**的博弈
- 一般称为博弈的扩展式表示 (Extensive-form)
- 用博弈树来表示：
 - 节点：某一参与者的决策点
 - 边：每一个可选动作就代表一条边
 - 叶子：代表博弈结束，并返回每一个参与者的收益



博弈中的不完美信息

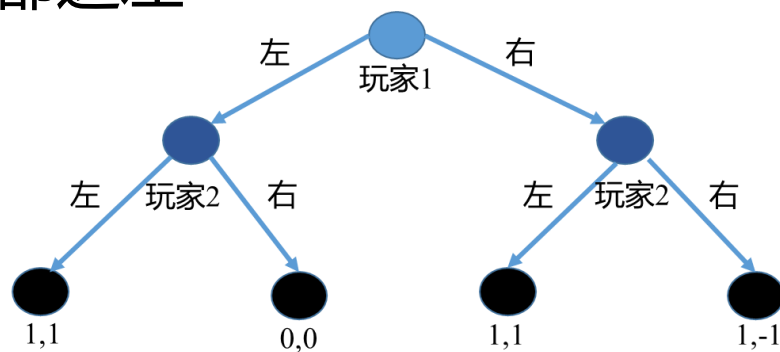
- 不完美信息（Imperfect Information）：一些参与者观测不到其他参与者的动作选择
- 信息集（Information Set, **博弈论中非常重要的概念**）：
 - 1) 集合中的每个节点都是同一个参与人进行决策；2) 参与人知道博弈进入该集合，但是不知道自己具体在哪一个节点；3) 每一个信息集中节点的可选动作都相同
- 每一个信息集对应一个“决策点”
- 完美信息博弈中每一个信息集只包含一个节点



玩家2不知道玩家1选了左还是右

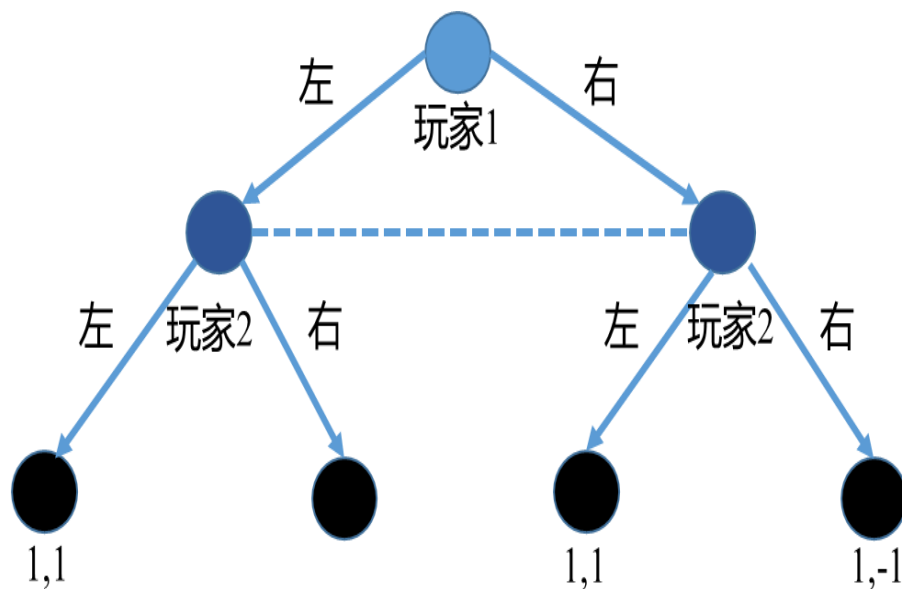
扩展式博弈中的纯策略集合

- 扩展式博弈中的一个纯策略描述了参与者**在其所有决策点（信息集）的动作选择**
- 每一个参与人 i 的纯策略集合 S_i :
 - $S_1 = \{\text{左}, \text{右}\}$, 玩家1有两个纯策略
 - 玩家2有两个决策点, 每个决策点2种动作选择 \rightarrow 4个纯策略
 - $S_2 = \{\text{左左}, \text{左右}, \text{右左}, \text{右右}\}$, 玩家2有4个纯策略
 - 代表什么含义?
 - 左左: 不管玩家1如何选择玩家2都选左
 - 左右: 1左2也左, 1右2也右
 - 右左: 1左2右, 1右2左
 - 右右: 不管1如何, 2都右



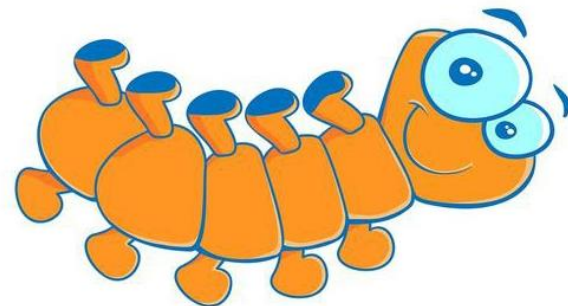
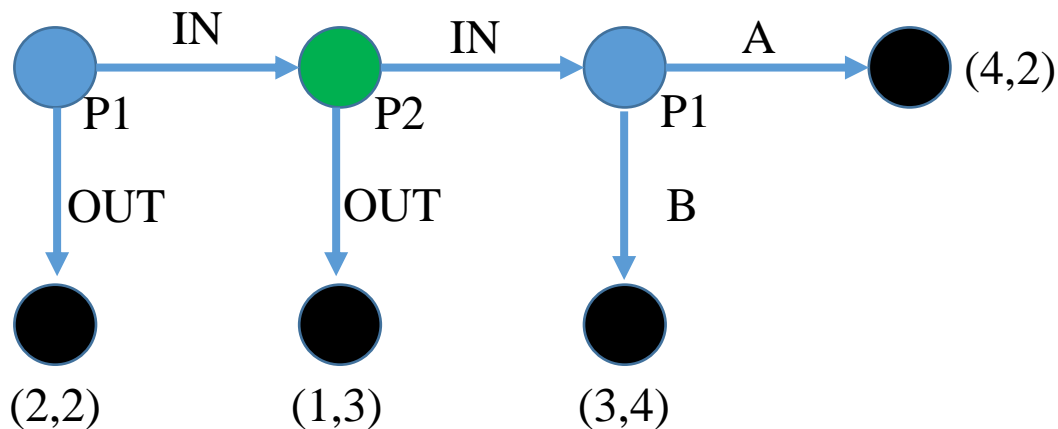
扩展式博弈中的纯策略集合

- 不完美信息条件下，每一个参与人 i 的纯策略集合 S_i ：
 - $S_1 = \{\text{左}, \text{右}\}$ ，玩家1有2个纯策略
 - $S_2 = \{\text{左}, \text{右}\}$ ，玩家2也有2个纯策略
 - 玩家2为什么只有2个纯策略？这和上个例子有什么不同？
 - 因为玩家2只有1个决策点（信息集），该信息集下只有2个动作可供选择，因此玩家2只有两个纯策略



扩展式博弈中的纯策略集合：练习

• 蜈蚣博弈 (The Centipede Game)



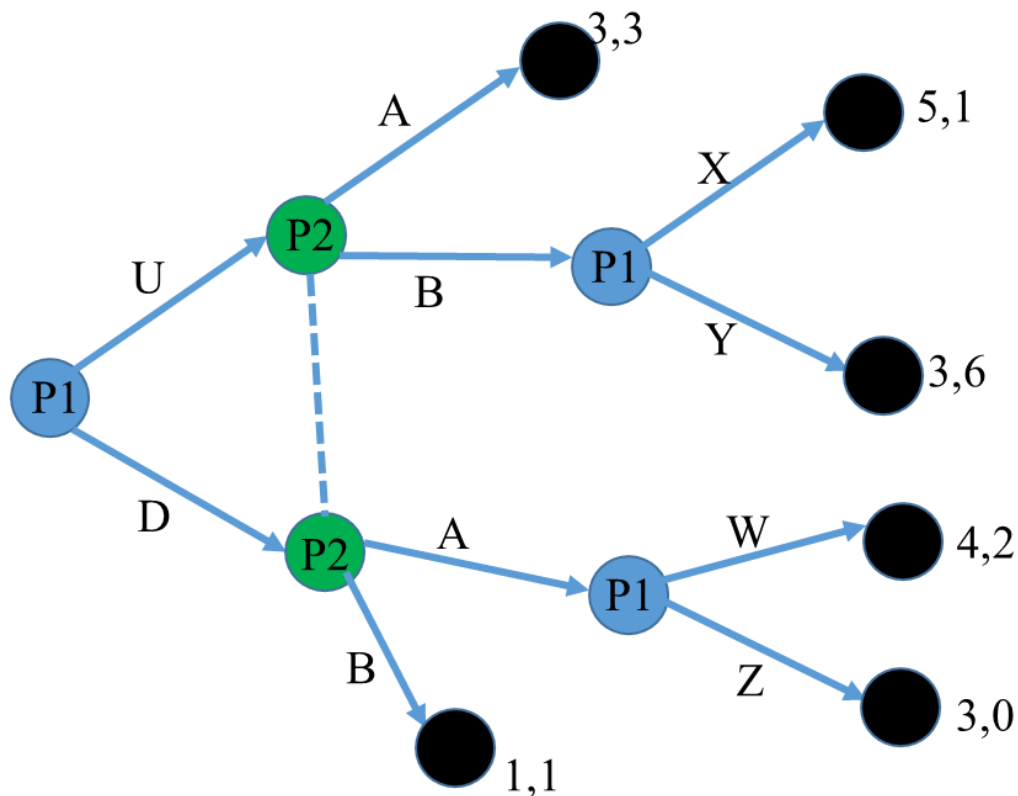
- $S_2 = \{\text{IN}, \text{OUT}\}$

- $S_1 = \{\text{IN A}, \text{IN B}, \text{OUT A}, \text{OUT B}\}$

- 很多人会有疑问：P1选了OUT之后博弈就结束了，为什么还要指定后续的动作？
- 回忆定义：扩展式博弈的一个纯策略描述了参与者在其**所有**决策点（信息集）的动作选择，不能只指定部分决策点的选择！

扩展式博弈中的纯策略集合：练习

- 不完美信息博弈




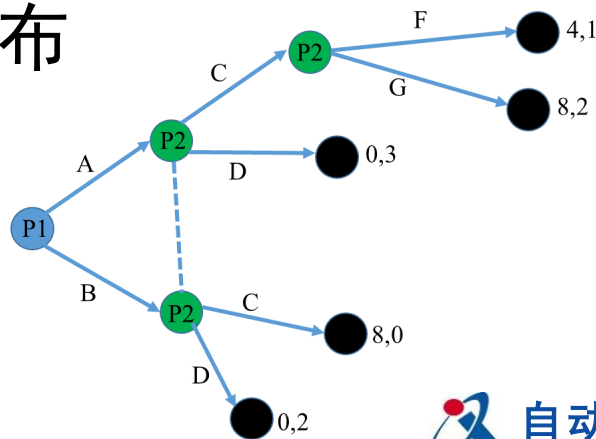
- $S_1 = \{UXW, UXZ, UYW, UYZ, DXW, DXZ, DYW, DYZ\}$

- $S_2 = \{A, B\}$, 玩家2只有1个决策点！

- 与矩阵式博弈相同：

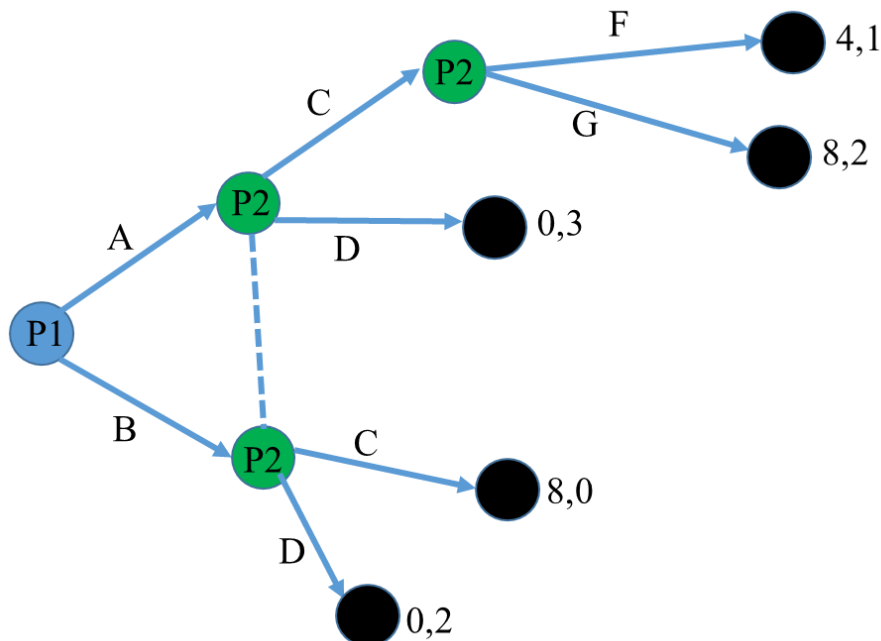
假设 i 有 K 个纯策略: $S_i = \{s_{i1}, \dots, s_{iK}\}$, 概率分布 $\sigma_i = (\sigma_{i1}, \dots, \sigma_{iK})$ 为 i 的一个混合策略, σ_{ik} 是 i 选择纯策略 s_{ik} 的概率

- 玩家1的纯策略集合 $S_1 = \{A, B\}$ ，它的一个混合策略 σ_1 为 S_1 上的某一概率分布
 - 玩家2的纯策略集合 $S_2 = \{CF, CG, DF, DG\}$ ，它的一个混合策略 σ_2 同样为 S_2 上的某一概率分布
- 



扩展式博弈中的行为策略

- 玩家 i 的行为策略（Behavioral Strategy）：
 - 为玩家 i 每一个决策点指定一个概率分布
 - 玩家1的行为策略： α 的概率选A动作，以 $1 - \alpha$ 的概率选B动作
 - 玩家2的行为策略：第一个决策点， β 概率选C， $1 - \beta$ 概率选D，第二个决策点， γ 概率选F， $1 - \gamma$ 概率选G



扩展式博弈中纯、混合、行为策略的直观理解

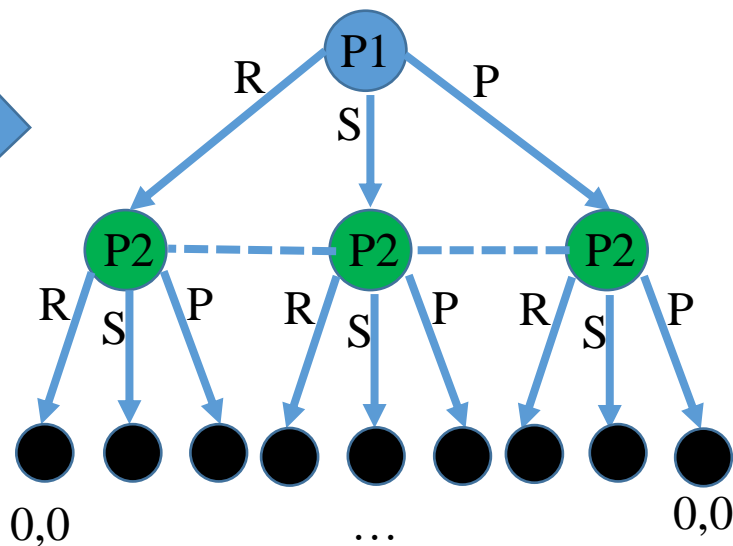
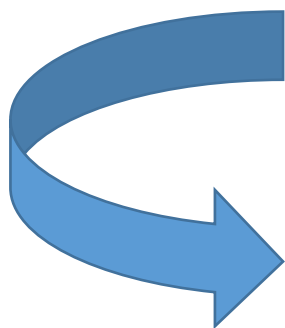
- 假设博弈中某玩家有5个决策点（信息集），那么一个纯策略就是一本书，这本书有5页，每一页说明了在该决策点会选择的动作。因此当采取这个纯策略时，就相当于拿出这本书，在遇到每个决策点时翻开对应的页数根据书上写的选择动作。所有可能的纯策略组成的集合就是一个书柜，里面堆满了不一样的书，每本书对应一个纯策略
- **混合策略**：一个mixed strategy就是随机在这个书柜里面抽取书，也就是说对每一本书给定一个抽取的概率，使其加总起来等于1
- **行为策略**：一个behavior strategy 是一本不一样的书，这本书里的每一页不再要求只能指定一个动作，而是要指定在这个决策点选择每个动作的概率
- 行为策略更加直观且高效
- **Kuhn's Theorem**：扩展式博弈满足一定条件时（Perfect Recall），对于每一个混合策略，存在一个行为策略与之等价

矩阵式表示 \leftrightarrow 扩展式表示

- 博弈的矩阵式表示和扩展式表示可以互相转化
- 矩阵式转化为扩展式
 - 利用信息集来表示动作同时进行

玩家1 \ 玩家2	石头	剪刀	布
	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

石头剪刀布的矩阵式表示

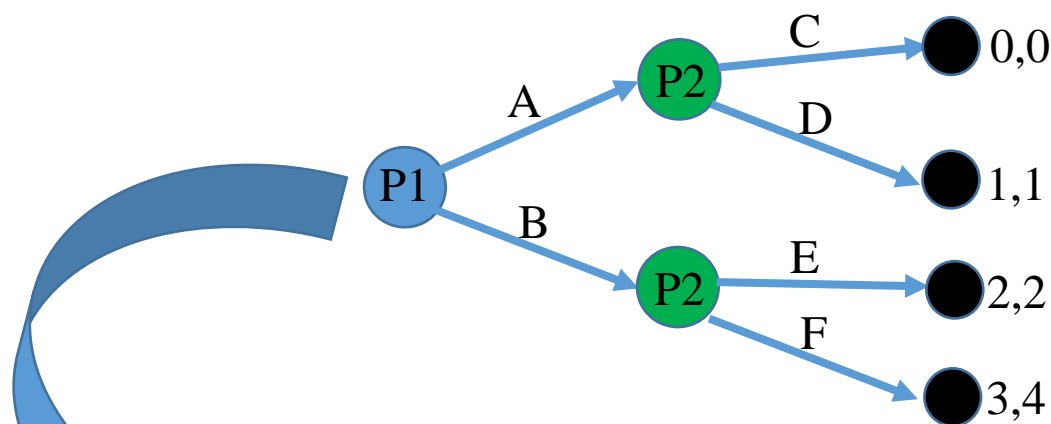


石头剪刀布的扩展式表示

矩阵式表示 \leftrightarrow 扩展式表示

• 扩展式转化为矩阵式

- 首先写出所有玩家的纯策略集合，集合大小确定了矩阵每一维度的大小，然后用收益函数进行填空



扩展式表示

$$S_1 = \{A, B\}$$

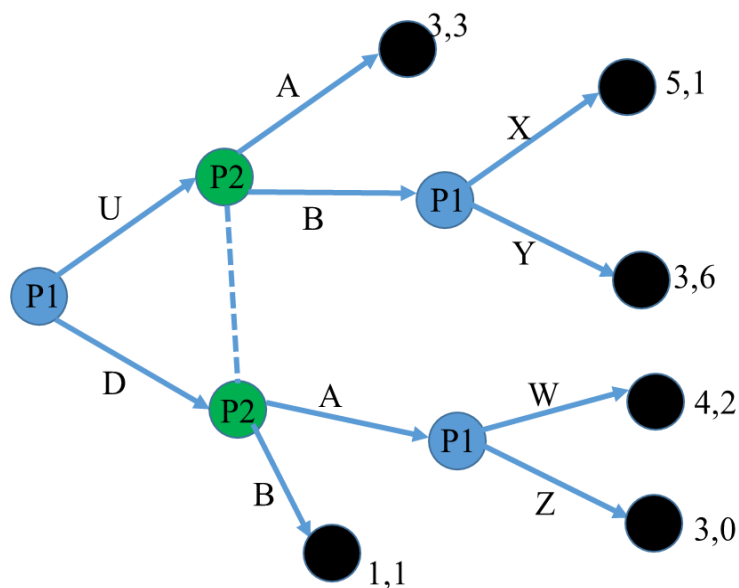
$$S_2 = \{CE, CF, DE, DF\}$$

	CE	CF	DE	DF
A	0,0	0,0	1,1	1,1
B	2,2	3,4	2,2	3,4

矩阵式表示

矩阵式表示 \leftrightarrow 扩展式表示

• 扩展式转化为矩阵式



扩展式表示

$S_1 = \{UXW, UXZ, UYW, UYZ, DXW, DXZ, DYW, DYZ\}$

$S_2 = \{A, B\}$

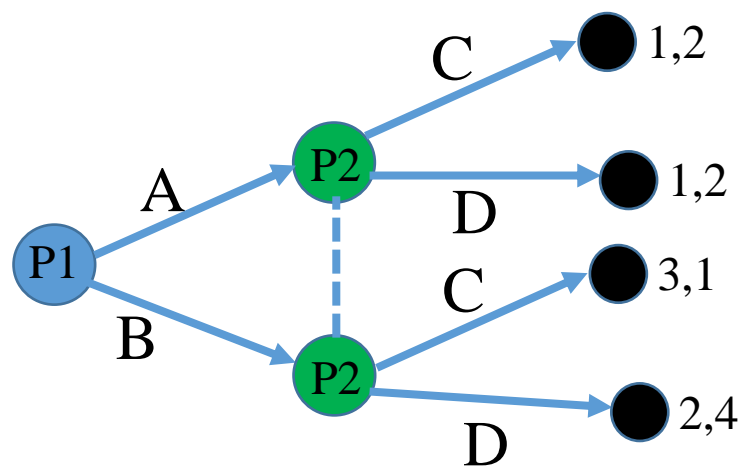
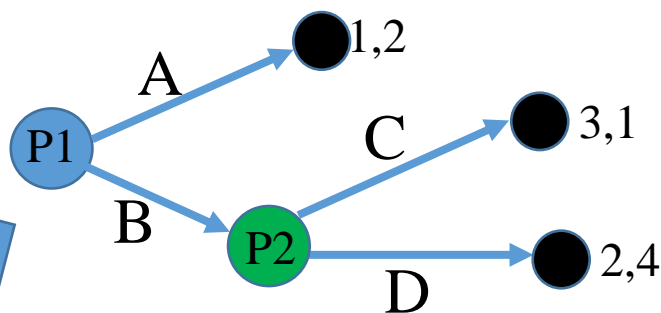
	A	B
UXW	3,3	5,1
UXZ
UYW
UYZ
DXW
DXZ
DYW
DYZ	3,0	1,1

矩阵式表示

矩阵式表示 \leftrightarrow 扩展式表示

- 扩展式转化为矩阵式是唯一的
- 矩阵式转化为扩展式可能不唯一！

	C	D
A	1,2	1,2
B	3,1	2,4



博弈表示方法小结

- 参与人、策略空间、收益函数
- 矩阵表示
 - 纯策略
 - 混合策略
- 扩展式表示
 - 不完美信息
 - 信息集
 - 纯策略
 - 混合策略
 - 行为策略
- 矩阵式表示 \leftrightarrow 扩展式表示
 - 转化是否唯一

本讲提纲

1 博弈表示方法

2 常见博弈类型

3 博弈的解概念

4 课程设计任务



常见博弈类型

信息 \ 行动次序	静态	动态
	完全信息 静态博弈	完全信息 动态博弈
完全信息		
不完全信息	不完全信息 静态博弈	不完全信息 动态博弈



重复博弈
Repeated Games

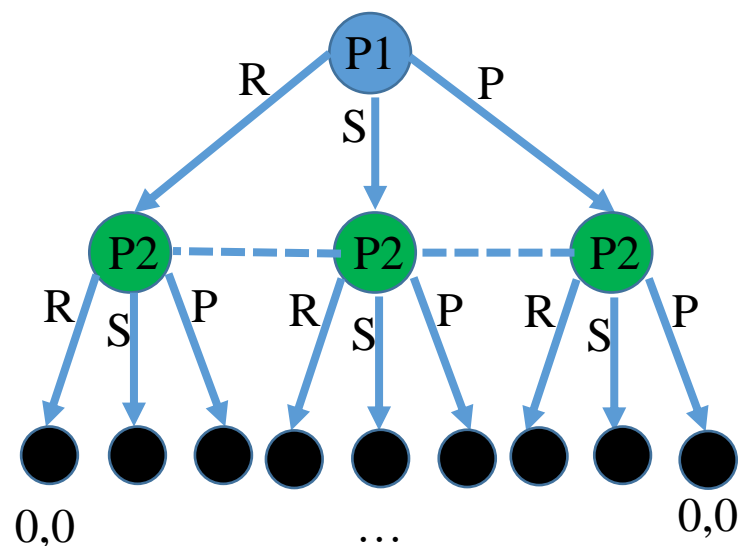


随机博弈
Stochastic Game

静态博弈

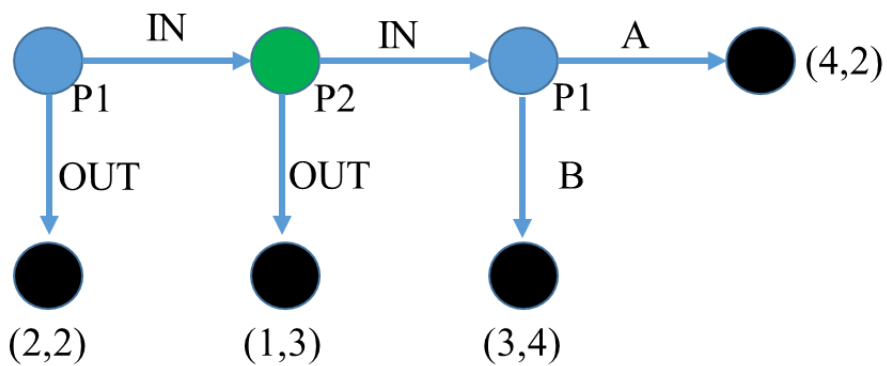
- 博弈中参与者同时采取行动，或者尽管参与者行动有先后顺序，但后行动的不知道先行行动的采取的是什么行动，如石头剪刀布、囚徒困境等
- 一般用矩阵式表示
- 也可用扩展式表示，用信息集表示某方动作不可观测
- 可以认为是一种不完美信息博弈

玩家1 \ 玩家2			
	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

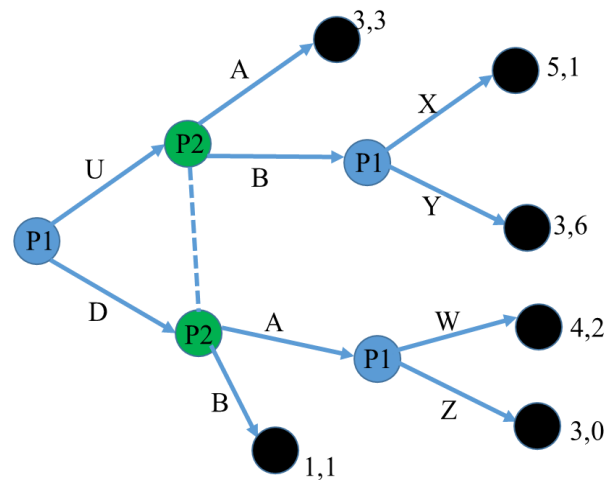


动态博弈

- 博弈参与人的行动有先后顺序，后行动者可以观察到先行动者的完全或部分动作，并据此作出相应的策略选择
- 一般用扩展式表示
- 扩展式可以转化为矩阵式，但是不够高效
- 同样用信息集表示某方动作不可观测



完美信息，可观测到所有动作

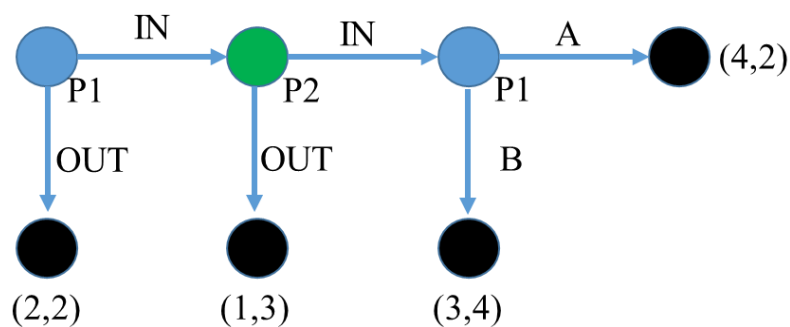


不完美信息，部分动作不可观测

完全信息博弈

- 博弈的所有参与者都对博弈各方的各种情况下的收益完全知晓
- 完全信息静态博弈：完全信息+静态博弈
- 完全信息动态博弈：完全信息+动态博弈

玩家1 \ 玩家2			
	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

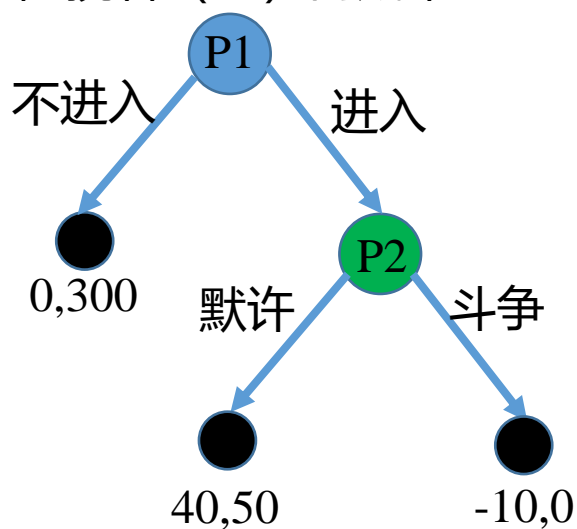


不完全信息博弈

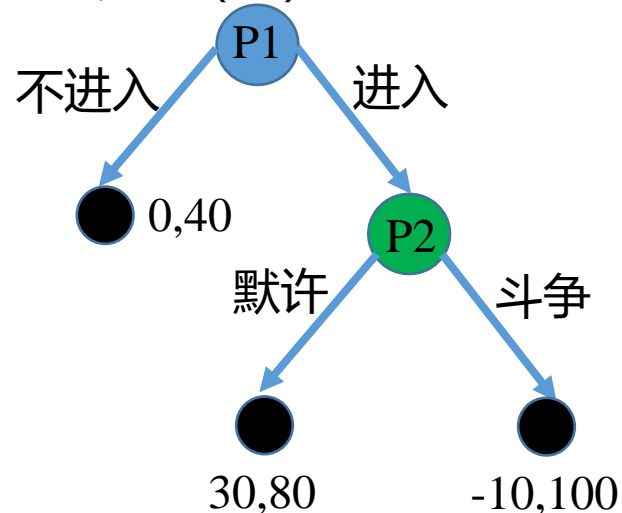
- 至少有一个参与人有私有信息，而其他人没有该信息，该私有信息称为参与人的类型（type）
- 私有信息的存在导致博弈的其它参与者对最终的收益不完全知晓
- 也称为贝叶斯博弈（Bayesian Game）

进入者 \ 阻挠者	高成本情况		低成本情况	
	默许	斗争	默许	斗争
进入	40, 50	-10, 0	30, 80	-10, 100
不进入	0, 300	0, 300	0, 40	0, 400

阻挠者 (P2) 高成本



阻挠者 (P2) 低成本

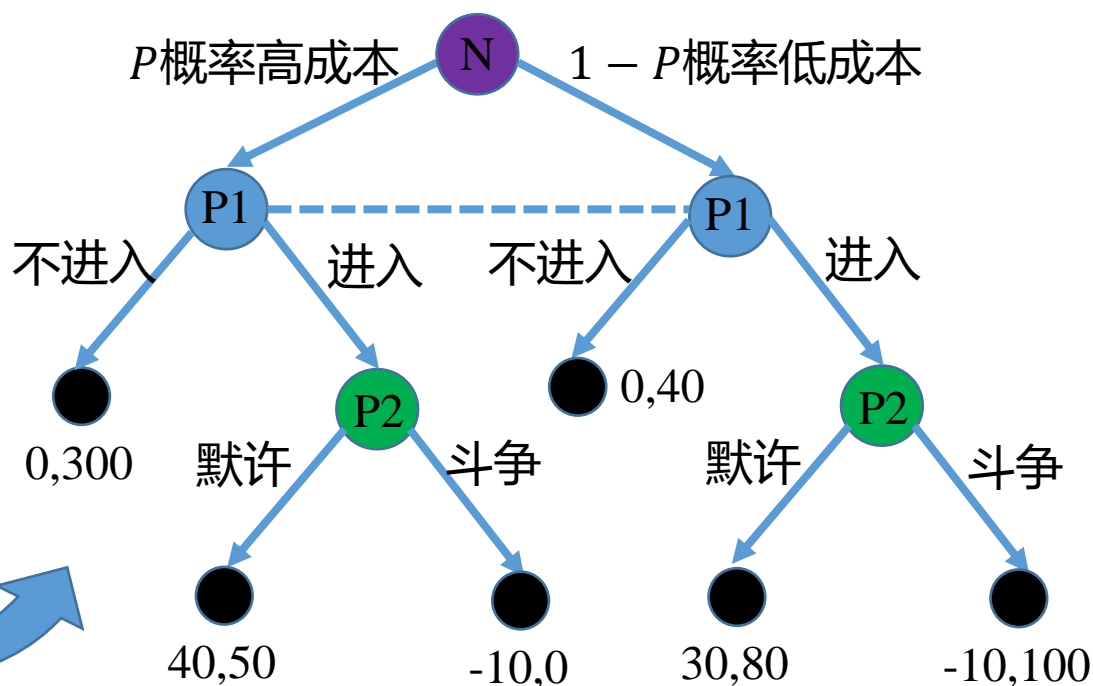


海萨尼转换

- 海萨尼（Harsanyi）转换：将不完全信息博弈转换为完全不完备信息博弈
 - 通过引入一个虚拟的参与人：自然（Nature）
 - 自然决定类型出现的概率，自然没有收益函数

阻挠者 进入者	高成本情况		低成本情况	
	默许	斗争	默许	斗争
进入	40, 50	-10, 0	30, 80	-10, 100
不进入	0, 300	0, 300	0, 40	0, 400

海萨尼转换



完全信息/完美信息

- 完全信息：对所有局中人的收益函数都完全了解
- 完美信息：对所有局中人已有行动完全了解
- 海萨尼转换将不完全信息转换为完全 imperfect 信息博弈
- 举例说明：
 - 完全并且完美信息博弈：围棋、象棋
 - 不完全但完美信息博弈：德州扑克（海萨尼转换之前）
 - 完全但不完美信息博弈：德州扑克（海萨尼转换之后）
 - 不完全不完美信息博弈：战争
- https://en.wikipedia.org/wiki/Perfect_information
- https://en.wikipedia.org/wiki/Complete_information

重复博弈 (Repeated Games)

- 前面讲述的博弈都只进行一次
- 重复博弈，同样结构的博弈重复多次或无限次：
 - 每次博弈称为阶段博弈 (Stage Games)
 - 多轮囚徒困境 (Iterated Prisoners' Dilemma)
- 重复博弈中可采取的策略更为丰富多样
 - 以牙还牙策略 (Tit-for-tat)：复制对手上次的动作选择

囚徒A \ 囚徒B	坦白	抵赖
	坦白	抵赖
坦白	2,2	0,3
抵赖	3,0	1,1

	1	2	3	4
P1	坦白	坦白	坦白	...
P2	坦白	坦白	坦白	...

P2的收益: $2 + 2\gamma + 2\gamma^2 + \dots = \frac{2}{1-\gamma}$
 γ 是衰减因子 (Discount Factors)

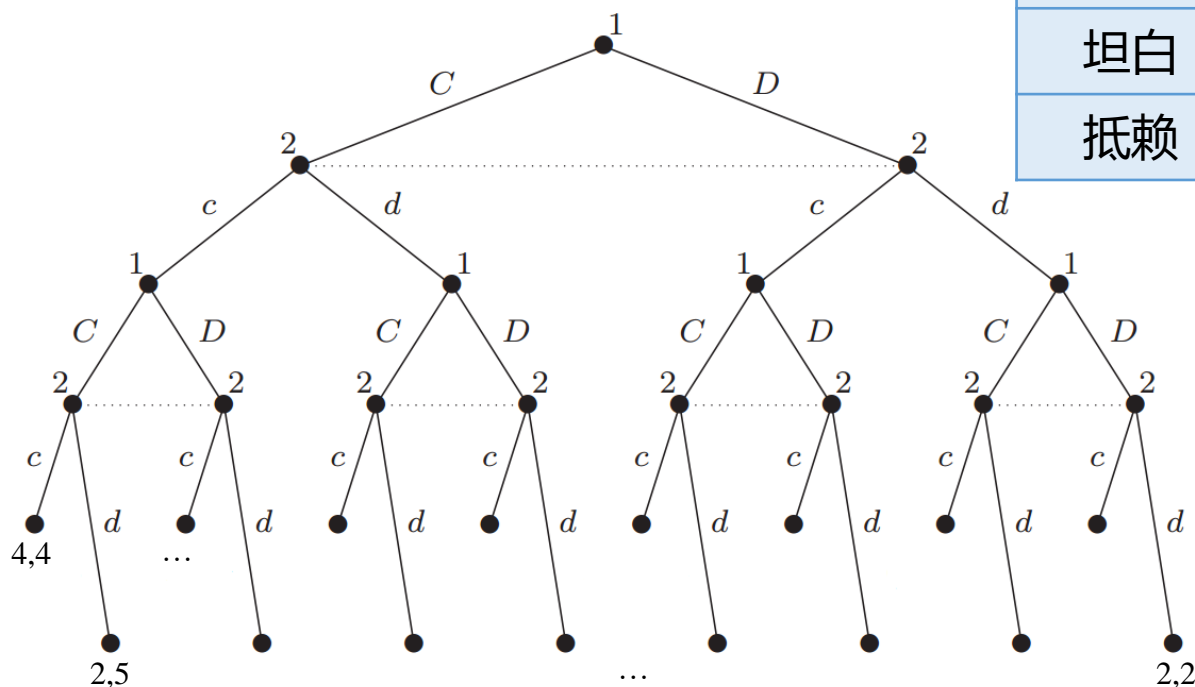
	1	2	3	4
P1	坦白	抵赖	坦白	...
P2	抵赖	坦白	抵赖	...

P2的收益: $3 + 0 + 3\gamma^2 + 0\gamma^3 + \dots = \frac{3}{1-\gamma^2}$

重复博弈

- 多轮囚徒困境的扩展式表示（以两轮为例）
 - 每个阶段博弈中看不到对手动作
 - 但是下一个阶段博弈开始时可以看到对手上次的动作
 - 各玩家收益为每轮博弈收益之和

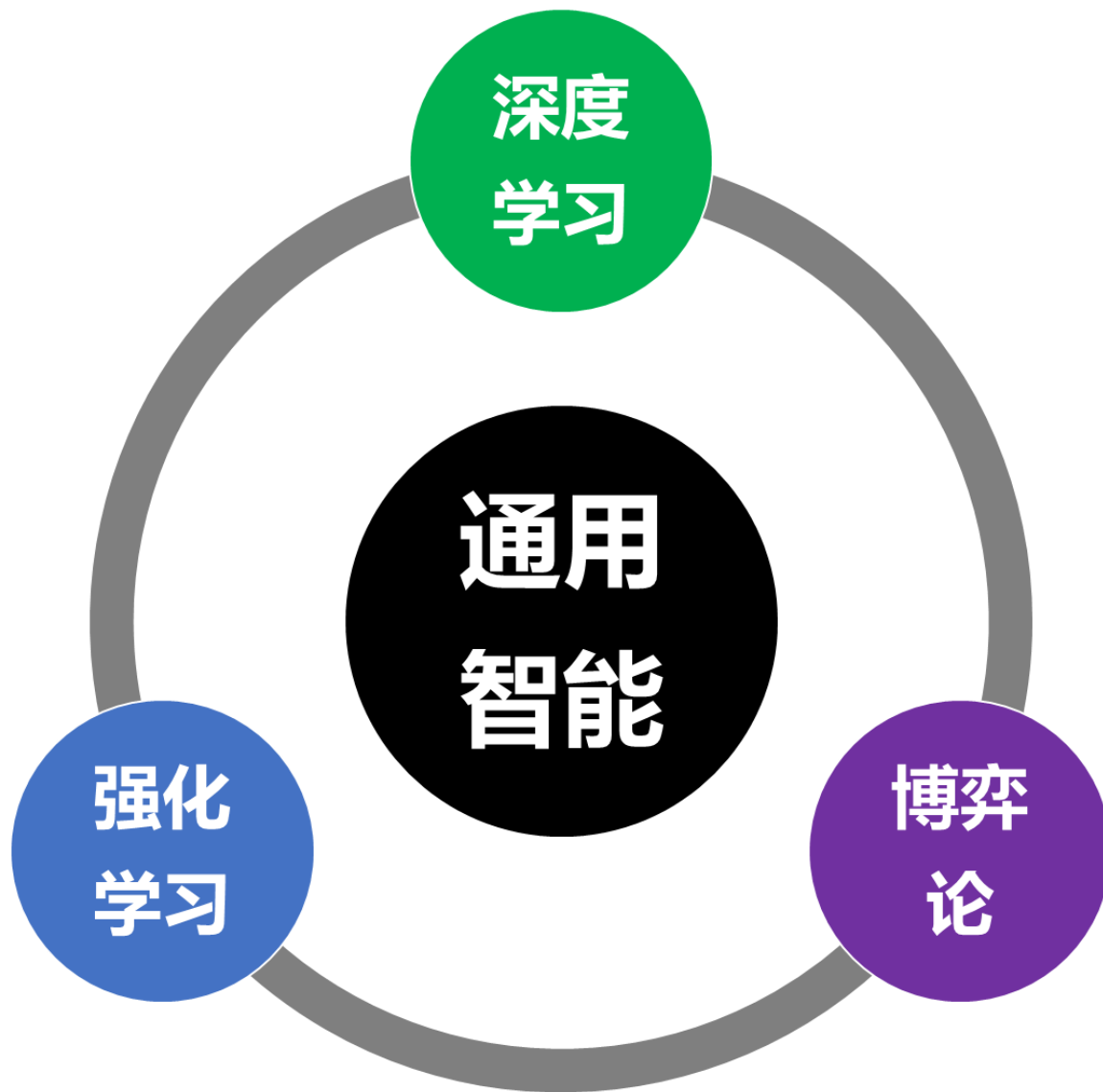
囚徒A \ 囚徒B	坦白	抵赖
	坦白	抵赖
坦白	2,2	0,3
抵赖	3,0	1,1



随机博弈 (Stochastic Games)

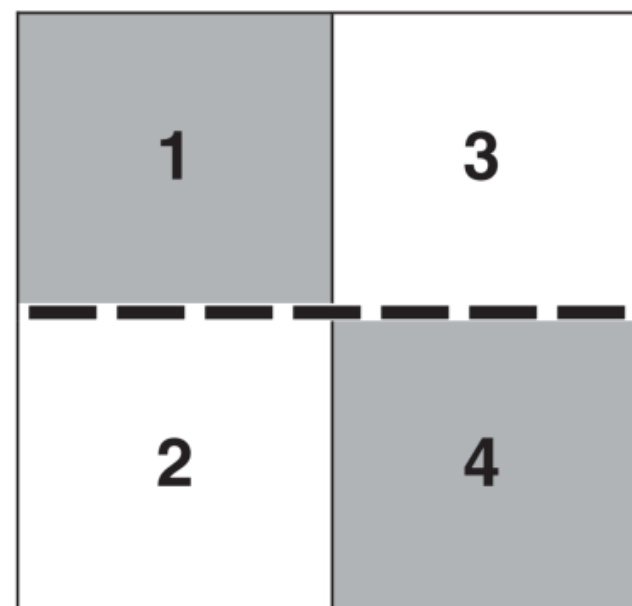
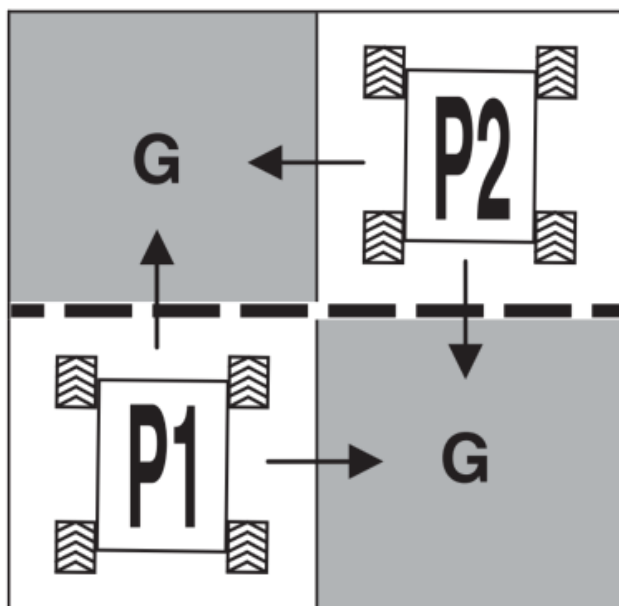
- 随着博弈进行环境发生随机的状态转移 → 随机博弈
- 随机博弈可以看作是一个五元组 (Q, N, A, P, R) :
 - Q 是状态集合, $N = \{1, 2, \dots, i, \dots, n\}$ 是玩家集合
 - $A = A_1 \times \dots \times A_n$, A_i 是玩家 i 的动作空间
 - $P: Q \times A \times Q \rightarrow [0, 1]$ 是状态转移函数
 - $R = r_1, \dots, r_n$, $r_i: Q \times A \rightarrow \mathbb{R}$ 是玩家 i 的奖励
 - 重复博弈是随机博弈的特例, $|Q| = 1$
 - 随机博弈是单智能体强化学习中马尔科夫决策过程 (Markov Decision Process, MDP) 的推广, $n = 1$
 - 随机博弈是多智能体强化学习的基础
 - 连接博弈论和多智能体强化学习的桥梁

随机博弈 (Stochastic Games)



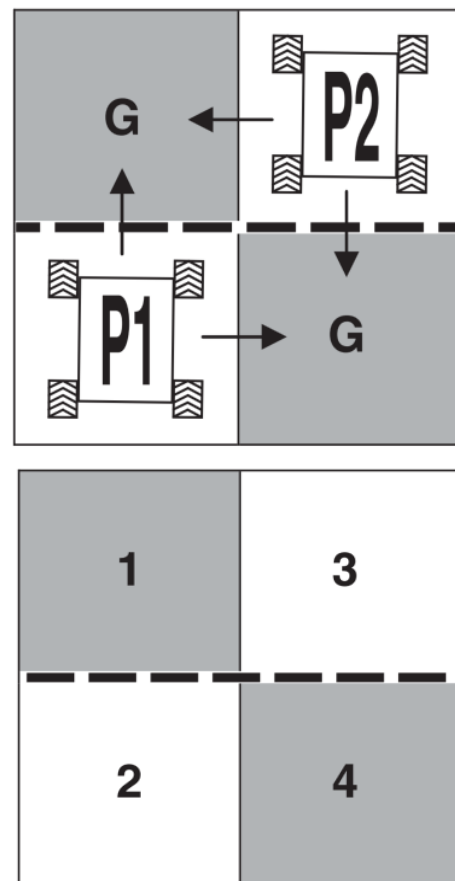
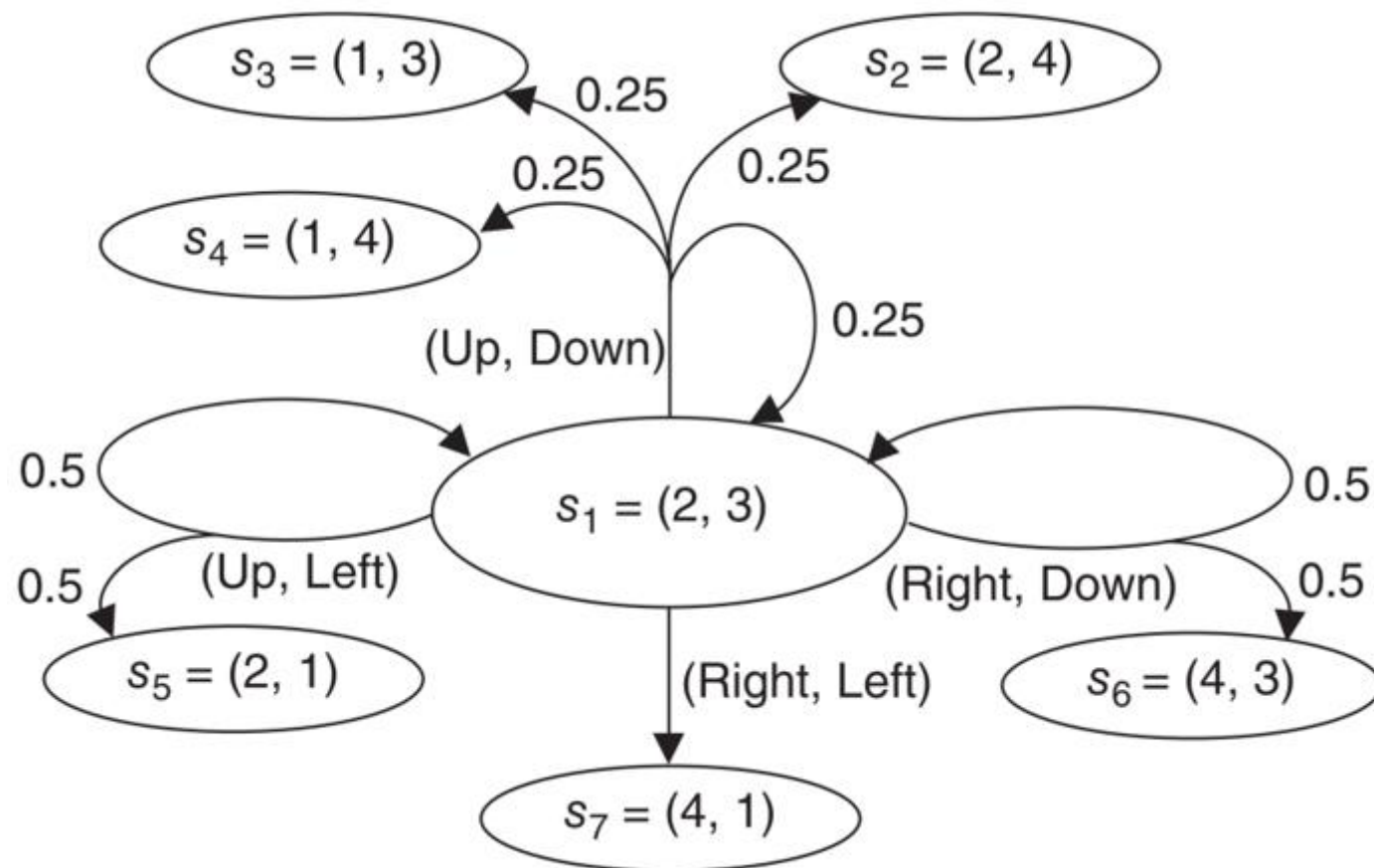
随机博弈例子

- 两小车都想到G位置，任意小车到达G则游戏结束
- P1两个动作，P2两个动作
- 虚线是障碍，只有50%的概率能通过
- 如果双方进入同一个cell，双方直接复位到原始位置



随机博弈例子

• 状态转移示意图：



常见博弈类型小结

- 静态博弈
 - 矩阵式表示，也可用扩展式
- 动态博弈
 - 扩展式表示，也可转化为矩阵式但不高效
- 完全信息博弈
- 不完全信息博弈
 - 海萨尼转换
- 完美信息和完全信息
- 重复博弈
- 随机博弈
 - 多智能体强化学习问题的基础性描述工具

本讲提纲

1 博弈表示方法

2 常见博弈类型

3 博弈的解概念

4 课程设计任务



博弈的解概念

- 我们讲述了博弈表示方法以及常见的博弈类型
- 接下来，我们将对博弈进行分析
- 博弈论中的解概念（Solution Concepts）：
 - 指定了博弈各玩家采取的策略以及最终的收益
 - 可以从不同的角度得到不同的解：从局外人的角度、从当事人的角度等
 - 对于许多博弈，同一类型的解概念可能得到多个解，存在如何选择的问题
 - 得到的解可能存在不合理之处，需要进行精炼（Refinement）
- 常见的解概念：帕累托最优（Pareto Optimality）、纳什均衡（Nash equilibrium）等

帕累托最优

- 从一个客观公正局外人的角度来看，博弈中是否有一些结果优于其他结果？

帕累托占优 (Pareto Domination)

策略组合 $s = (s_i, \dots, s_n)$ 帕累托占优 $s' = (s'_i, \dots, s'_n)$ ：
如果对于任何玩家 $i \in N$ 来说， $u_i(s) \geq u_i(s')$

帕累托最优 (Pareto Optimality)

策略组合 $s = (s_i, \dots, s_n)$ 是帕累托最优：
如果不存在其他策略组合 s' 帕累托占优 s

- 不存在其他策略组合增长一方收益的同时而不损害他方

帕累托最优：举例

- 帕累托最优：不存在其他策略组合增长一方收益的同时而不损害他方

3,3	0,2
2,0	1,1

3,3	1,4
4,1	2,2

1,-1	-1,1
-1,1	1,-1

- 在两人零和博弈中，所有的策略组合都是帕累托最优的！

纳什均衡 (Nash equilibrium)

- 接下来，我们从玩家自身角度出发分析博弈
- 纳什均衡是博弈论中最具影响力的解概念

最优反应 (Best Response)

玩家 $i \in N$ 对 s_{-i} 的最优反应是 s_i^* ：
 s_i^* 满足对任意 s_i , $u_i(s_i^*, s_{-i}) \geq u_i(s_i, s_{-i})$

纳什均衡 (Nash Equilibrium)

一个策略组合 $s = (s_i, \dots, s_n)$ 是纳什均衡：
如果对任意玩家 $i \in N$ 来说, s_i 都是 s_{-i} 的最优反应

- 纳什均衡是一个稳定状态，每个玩家都没有动机改变自己的策略
- 上述两个定义同样适用于混合策略
- 称作纯策略纳什均衡与混合策略纳什均衡

纯策略纳什均衡求法

- 固定一方的策略，求取另一方的最优反应，最优反应的交集就是纯策略纳什均衡

囚徒B		坦白	抵赖
囚徒A	坦白	2,2	0,3
	抵赖	3,0	1,1

0,0	0,0	0,0	0,0
0,0	0,0	-1,1	-1,1
0,0	1,-1	0,0	-1,1
0,0	1,-1	1,-1	0,0

混合策略纳什均衡求法

- 右图不存在纯策略纳什均衡

纳什均衡存在性定理 (Nash, 1950)

有限博弈至少存在一个纯/混合纳什均衡

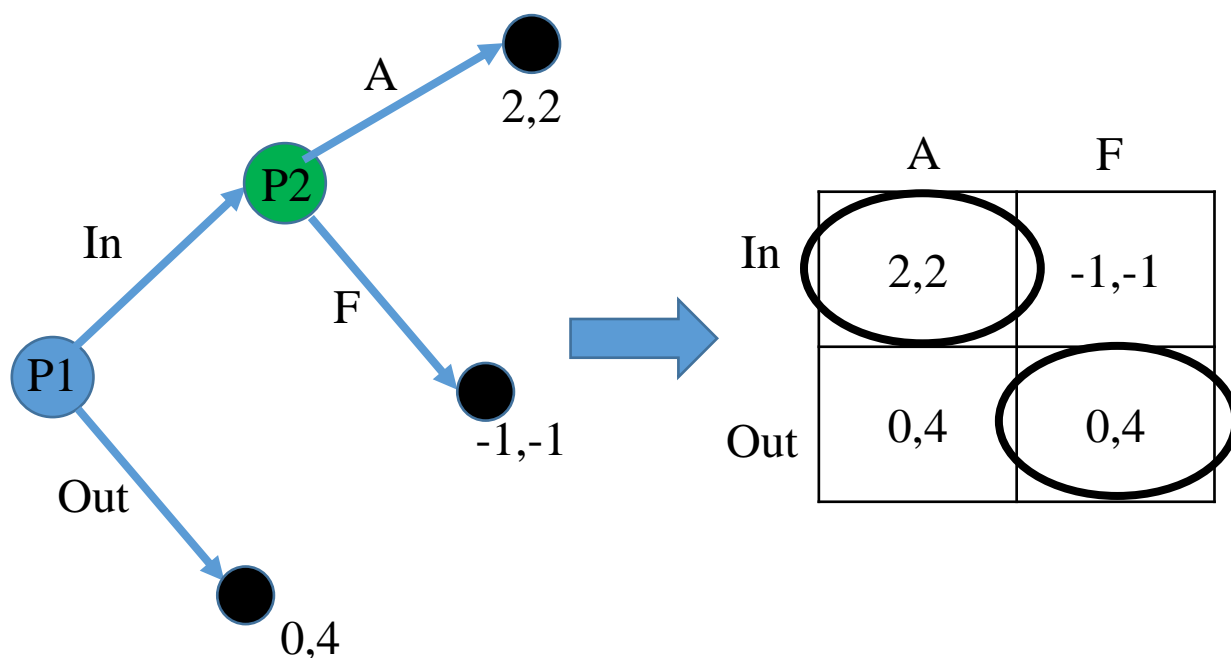
	h	t
h	1, -1	-1, 1
t	-1, 1	1, -1

- 因此存在混合策略纳什均衡

- 假设玩家1以 p 的概率选h, $1 - p$ 的概率选t
- 玩家2以 q 的概率选h, $1 - q$ 的概率选t
- 既然玩家1要采取混合策略, 那么选h或t对他的收益是一样的
- 如果不一样, 玩家1肯定会选收益更大的纯策略!
- 是玩家2的混合策略导致玩家1认为h和t的收益一样
- 玩家1选h或t的收益都是 q 的函数, 通过两者相等求取 q
- $q \times 1 + (1 - q) \times (-1) = q \times (-1) + (1 - q) \times 1 \rightarrow q = 0.5$
- 同理, 可得 $p = 0.5$

完全信息动态博弈中的纳什均衡

- 完全信息动态博弈一般用扩展式表示
- 扩展式表示可以转化为矩阵式表示
- 然后用前面的方法求纳什均衡，So Easy! But...

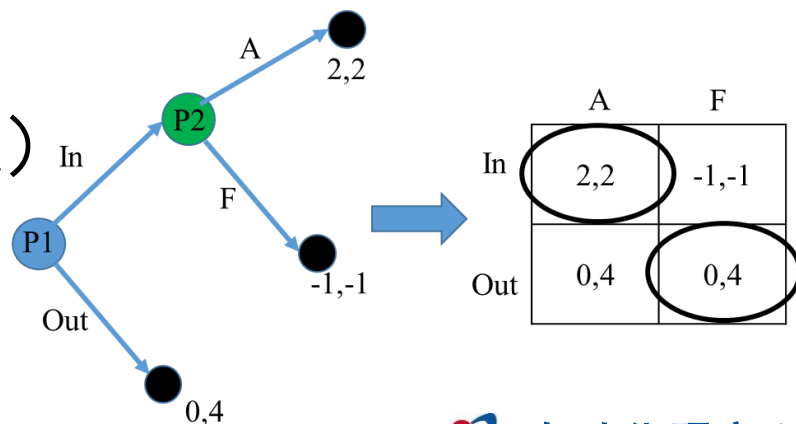


这两个纳什均衡都合理吗？

完全信息动态博弈中的纳什均衡

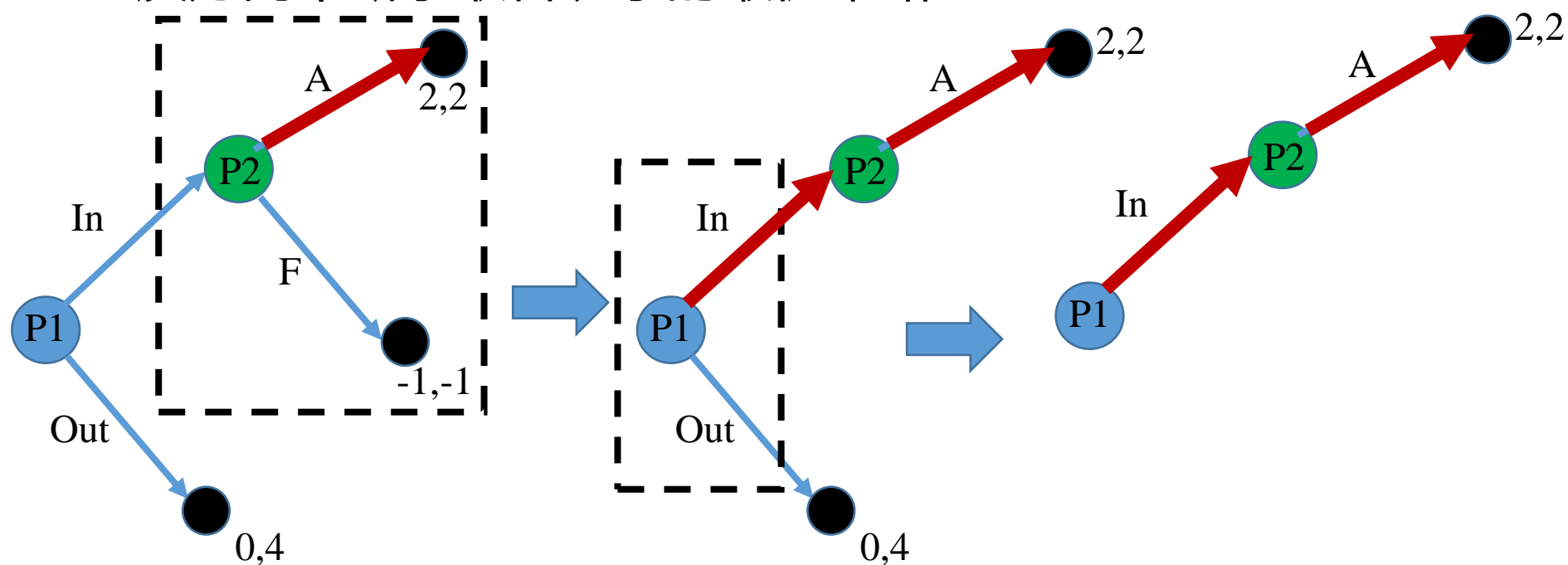
- IN/A合理，Out/F不合理
- Out/F的看似合理之处：
 - Out/F：假设P1选了In，接下来P2选F导致P1收益 $-1 < 0$ ，因此P1没有动机更改策略
 - Out/F：P1选Out，P2即使选A，P2收益不变，因此P2也没有动机更改策略，看起来Out/F似乎是一个合理的纳什均衡策略
- Out/F的不合理之处：
 - 但是，假设P1选In，P2会选F吗？不会，P2会选A！P2选F是不理性的

- 不可信的威胁（Incredible Threat）



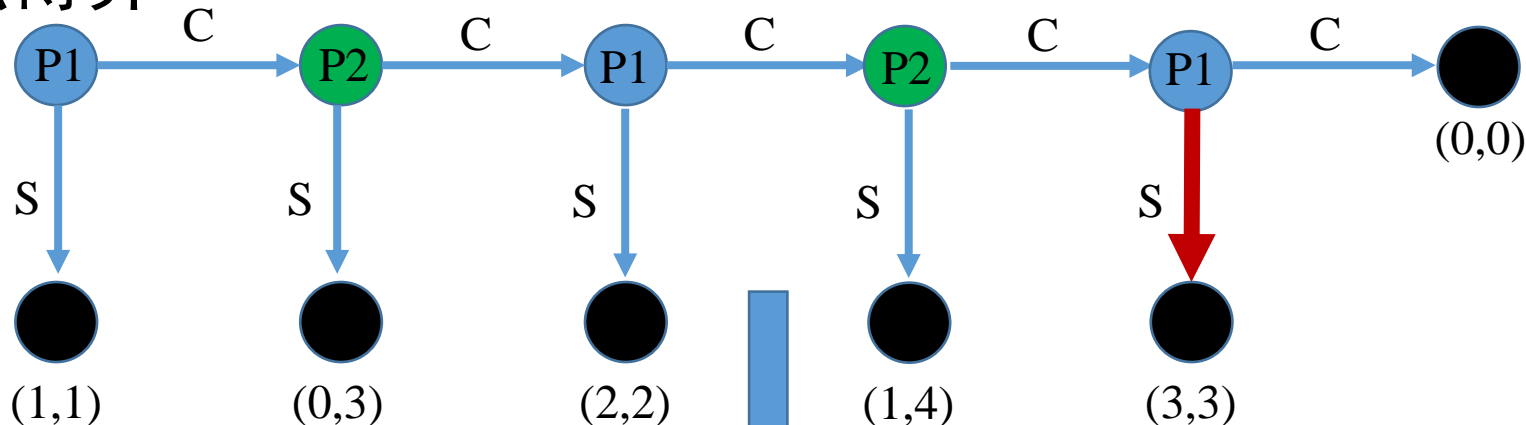
完全信息动态博弈中的纳什均衡

- 子博弈精炼纳什均衡（Subgame Perfect Equilibrium）
 - 子博弈可以粗略看作是博弈树的一棵子树
 - 每个子博弈都要满足纳什均衡条件
- 用逆向归纳法（Backward Induction）求解
 - 反方向不断求取各玩家的最优策略

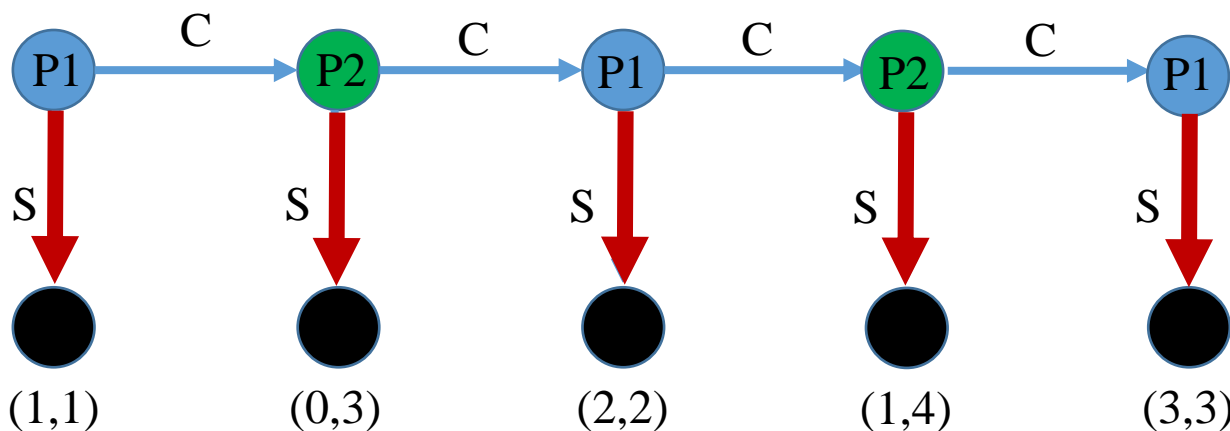


逆向递归法：更多例子

蜈蚣博弈



子博弈精炼纳什均衡解
 $s_1 = \{SSS\}$
 $s_2 = \{SS\}$



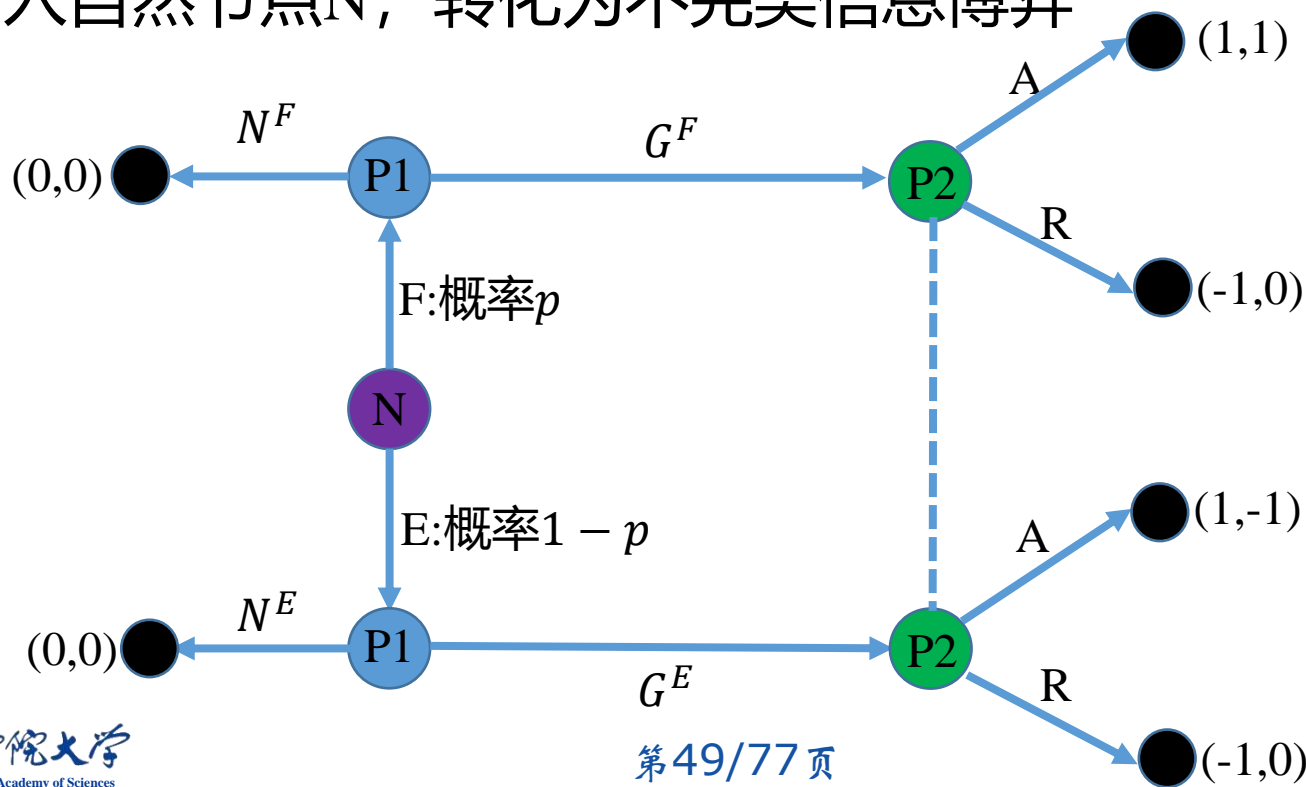
回忆：策略描述了参与者在所有决策点的动作选择！

不完全信息博弈中的纳什均衡

- 回忆：海萨尼转换

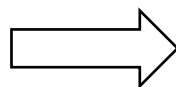
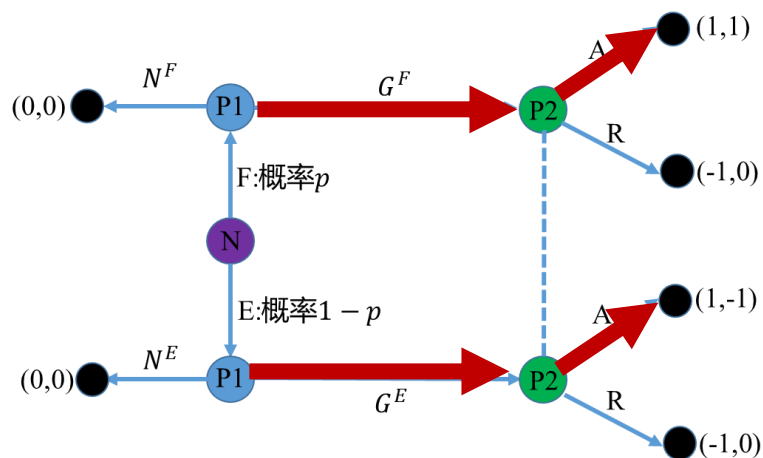
- 送礼物博弈：

- P1有两种类型：F友好E敌对，P1知道自己的类型，P2不知道
- G：送礼物，N：不送，A：接受，R：拒绝
- 引入自然节点N，转化为不完美信息博弈



不完全信息博弈中的纳什均衡

- 转化为矩阵表示（也称Bayesian Normal Form）：
 - $s_1 = \{G^F G^E, G^F N^E, N^F G^E, N^F N^E\}$
 - $s_2 = \{A, R\}$
 - $(G^F G^E, A)$, P1的期望收益: $p \times 1 + (1 - p) \times 1 = 1$, P2的期望收益 $p \times 1 + (1 - p) \times (-1) = 2p - 1$
 - 其他同理



	A	R
$G^F G^E$	$1, 2p - 1$	$-1, 0$
$G^F N^E$	p, p	$-p, 0$
$N^F G^E$	$1 - p, p - 1$	$p - 1, 0$
$N^F N^E$	$0, 0$	$0, 0$

不完全信息博弈中的纳什均衡

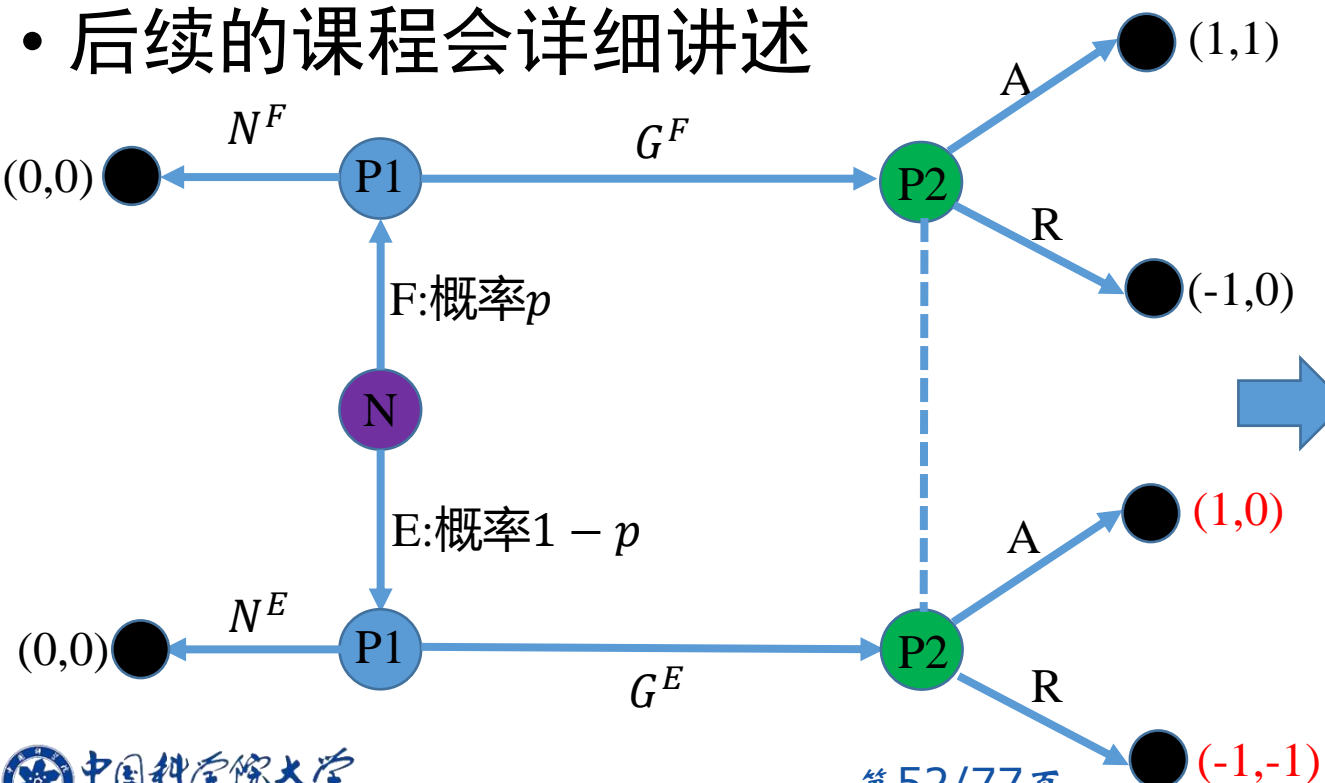
- 如果 $p \geq 0.5, 2p - 1 \geq 0$: 两个贝叶斯均衡
- 如果 $p < 0.5, 2p - 1 < 0$: 一个贝叶斯均衡

	A	R
$G^F G^E$	<div><div>1</div><div>$2p - 1$</div></div>	$-1, 0$
$G^F N^E$	p, p	$-p, 0$
$N^F G^E$	$1 - p, p - 1$	$p - 1, 0$
$N^F N^E$	$0, 0$	<div><div>0</div><div>0</div></div>

	A	R
$G^F G^E$	<div>1</div> $2p - 1$	$-1, 0$
$G^F N^E$	p, p	$-p, 0$
$N^F G^E$	$1 - p, p - 1$	$p - 1, 0$
$N^F N^E$	$0, 0$	<div><div>0</div><div>0</div></div>

不完全信息博弈中的纳什均衡

- 类比纳什均衡在完全信息动态博弈中的不合理性→子博弈精炼纳什均衡
- 贝叶斯均衡在不完全信息动态博弈中同样可能存在不合理性→精炼贝叶斯均衡 (Perfect Bayesian Equilibrium)
- 后续的课程会详细讲述



两个贝叶斯纳什均衡:
 $(G^F G^E, A)$
 $(N^F N^E, R)$
第二个均衡不合理

纳什均衡小结

- 最优反应，纳什均衡定义
- 完全信息静态博弈：纳什均衡
 - 纯策略：划线法，混合策略：不能区分好坏
- 完全信息动态博弈：子博弈精炼纳什均衡
 - 逆向归纳法，反着算
- 不完全信息静态博弈：贝叶斯均衡
 - 转化成Bayesian Normal Form
- 不完全信息动态博弈：精炼贝叶斯均衡
 - 删除不合理的均衡解
 - 后续的课程详细讲述

极小极大值定理 (Minimax Theorem)

• 两人零和博弈

两人零和博弈

玩家1 \ 玩家2	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

非两人零和博弈

囚徒A \ 囚徒B	坦白	抵赖
坦白	2, 2	0, 3
抵赖	3, 0	1, 1

- 许多重要的博弈都是两人零和的：如象棋、围棋等
- 因为玩家2的收益 = - 玩家1收益，收益矩阵可以简化

玩家1 \ 玩家2	石头	剪刀	布
石头	0	1	-1
剪刀	-1	0	1
布	1	-1	0

极小极大值定理

- 假设玩家1采用混合策略 $\sigma_1 = (\sigma_{11}, \dots, \sigma_{12}, \dots, \sigma_{1k})$
- 玩家2采用混合策略 $\sigma_2 = (\sigma_{21}, \dots, \sigma_{22}, \dots, \sigma_{2k})$
- 回忆： σ_1 和 σ_2 每一个元素代表选取某一纯策略的概率
- 假设收益矩阵用 A 表示
- 那么玩家1的期望收益为： $\sigma_1^T A \sigma_2$
- 玩家2的期望收益为： $-\sigma_1^T A \sigma_2$

σ_{11}	σ_{12}	A^{11}	A^{12}	σ_{21}
		A^{21}	A^{22}	σ_{22}

$$\sigma_{11} \times A^{11} \times \sigma_{21} + \dots + \sigma_{12} \times A^{22} \times \sigma_{22}$$

极小极大值定理

- John von Neumann 1928年就已经在思考两人零和博弈中各方该如何决策的问题
 - 早于纳什均衡概念的出现, Nash, 1950
- 极大极小策略 (Maximin Strategy) :
 - 站在P1的角度思考最差的情况
 - 如果我选择一个策略 σ_1
 - P2则会选择一个 σ_2 来极小化我的收益
 - 我的目标是选择一个 σ_1 来极大化我最差情况下的收益
 - $V_1^* = \max_{\sigma_1} \min_{\sigma_2} \sigma_1^T A \sigma_2$
 - V_1^* 称为P1的极大极小值
 - 得到的解 σ_1^* 称为P1的极大极小策略
 - 采用 σ_1^* 策略, P1至少可以获得 V_1^* 收益

极小极大值定理

- 极大极小策略与极小极大策略

玩家P1

$$V_1^* = \max_{\sigma_1} \min_{\sigma_2} \sigma_1^T A \sigma_2$$



σ_1^*

玩家P2

$$V_2^* = \min_{\sigma_2} \max_{\sigma_1} \sigma_1^T A \sigma_2$$



σ_2^*

- V_1^* 和 V_2^* 、 σ_1^* 和 σ_2^* 的关系是什么？

极小极大值定理 (Minimax Theorem) Jon von Neumann, 1928

对于两人零和博弈来说：

- $V_1^* = V_2^* = V^*$ ，称为博弈的极大极小值 (Minimax Value)

- σ_1^* 和 σ_2^* 其实构成了博弈的纳什均衡解，互为最优反应
- 该定理被视为博弈论的起点

极小极大值定理(伪)证明

- 严格的证明一般采用凸集分离定理、单纯形理论、Brouwer不动点定理等
- 这里我们假设有了纳什均衡的概念
- 证明 $V_1^* = V_2^* = V^*$ ，只需证明 $V_1^* \leq V_2^*$ 且 $V_1^* \geq V_2^*$
- 证明 $V_1^* \leq V_2^*$ 只需证明：

$$\begin{aligned}\max_x \min_y f(x, y) &\leq \min_y \max_x f(x, y) \\ \min_y f(x, y) &\leq f(x, y), f(x, y) \leq \max_x f(x, y)\end{aligned}$$



$$\min_y f(x, y) \leq \max_x f(x, y)$$

左边对任意 x ，右边对任意 y 不等式都成立，证毕

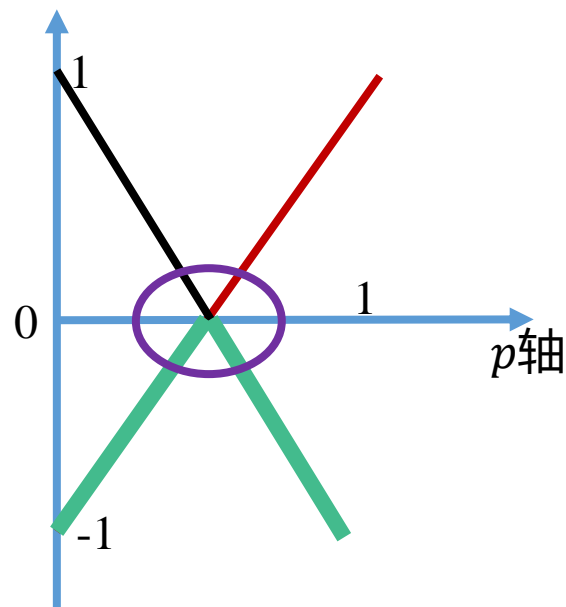
极小极大值定理(伪)证明

- 证明 $V_1^* \geq V_2^*$
 - 假设 $(\tilde{\sigma}_1, \tilde{\sigma}_2)$ 是纳什均衡：
 - P1的期望收益 $\tilde{v} = \tilde{\sigma}_1^T A \tilde{\sigma}_2$
 - $\tilde{\sigma}_1$ 是P1的最优反应 $\tilde{v} = \max_{\sigma_1} \sigma_1^T A \tilde{\sigma}_2$
 - 同理, $\tilde{v} = \min_{\sigma_2} \tilde{\sigma}_1^T A \sigma_2$
- $$\begin{aligned} V_2^* &= \min_{\sigma_2} \max_{\sigma_1} \sigma_1^T A \sigma_2 \leq \max_{\sigma_1} \sigma_1^T A \tilde{\sigma}_2 = \tilde{v} \\ &= \min_{\sigma_2} \tilde{\sigma}_1^T A \sigma_2 \leq \max_{\sigma_1} \min_{\sigma_2} \sigma_1^T A \sigma_2 = V_1^* \end{aligned}$$
- 下面我们通过一个简单例子讲解极大极小策略求法

极大极小策略：例子

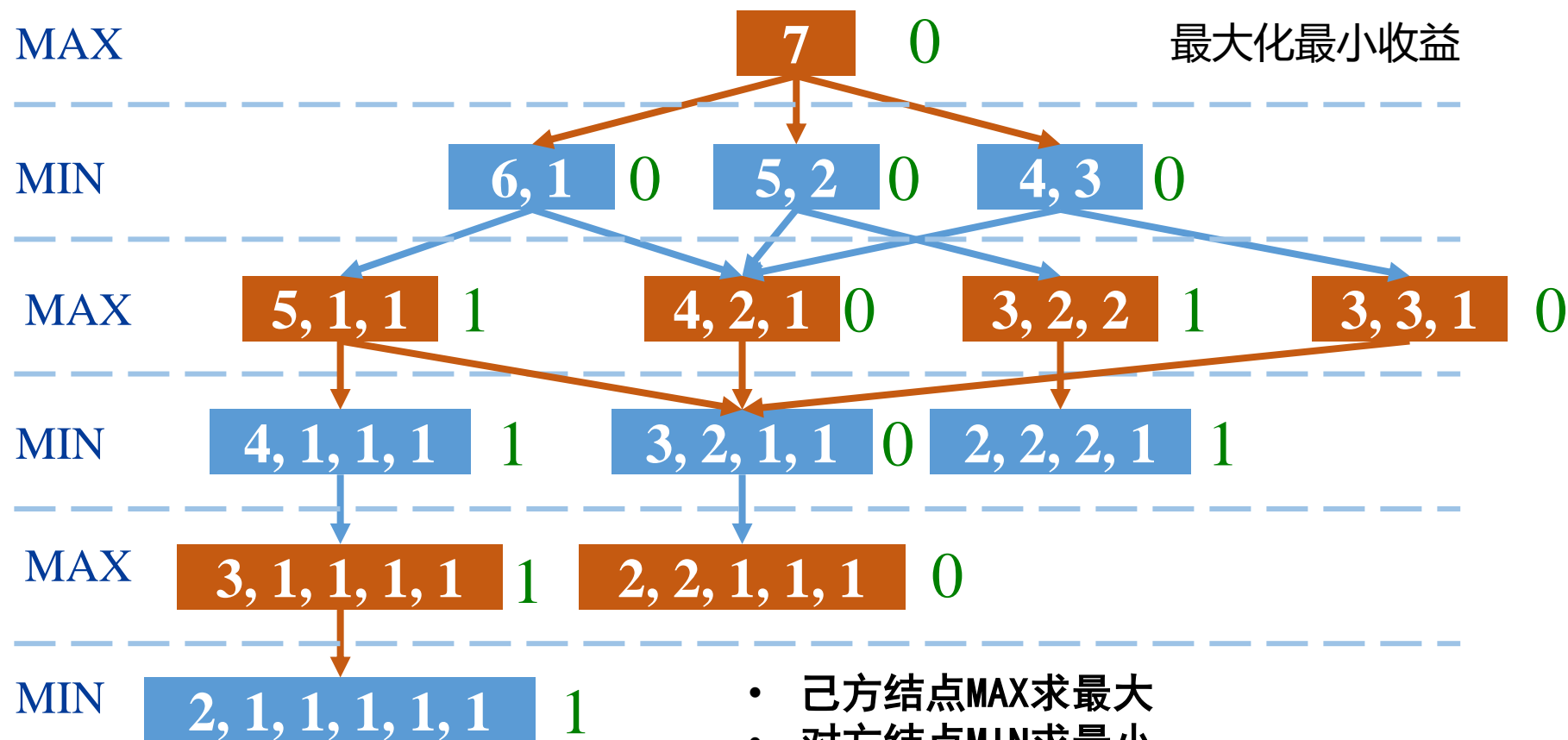
- $V_1^* = \max_{\sigma_1} \min_{\sigma_2} \sigma_1^T A \sigma_2$
- 假设 $\sigma_1 = (p, 1 - p)$
- 求 $\min_{\sigma_2} \sigma_1^T A \sigma_2$
 - 玩家2选h, 玩家1的平均收益 $1 \times p + (-1) \times (1 - p) = 2p - 1$
 - 玩家2选t, 玩家1的平均收益 $(-1) \times p + 1 \times (1 - p) = 1 - 2p$
 - 图像如右图绿色折线所示
- 求得 $p = 0.5$
- 极大极小策略 $\sigma_1 = (0.5, 0.5)$
- 同理, 极小极大策略 $\sigma_2 = (0.5, 0.5)$
- σ_1 和 σ_2 组成博弈的纳什均衡

	h	t
h	1,-1	-1,1
t	-1,1	1,-1



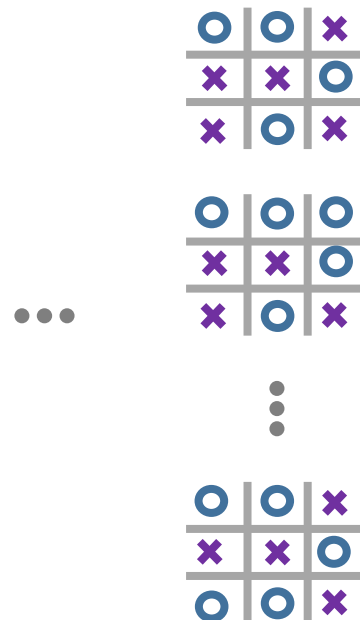
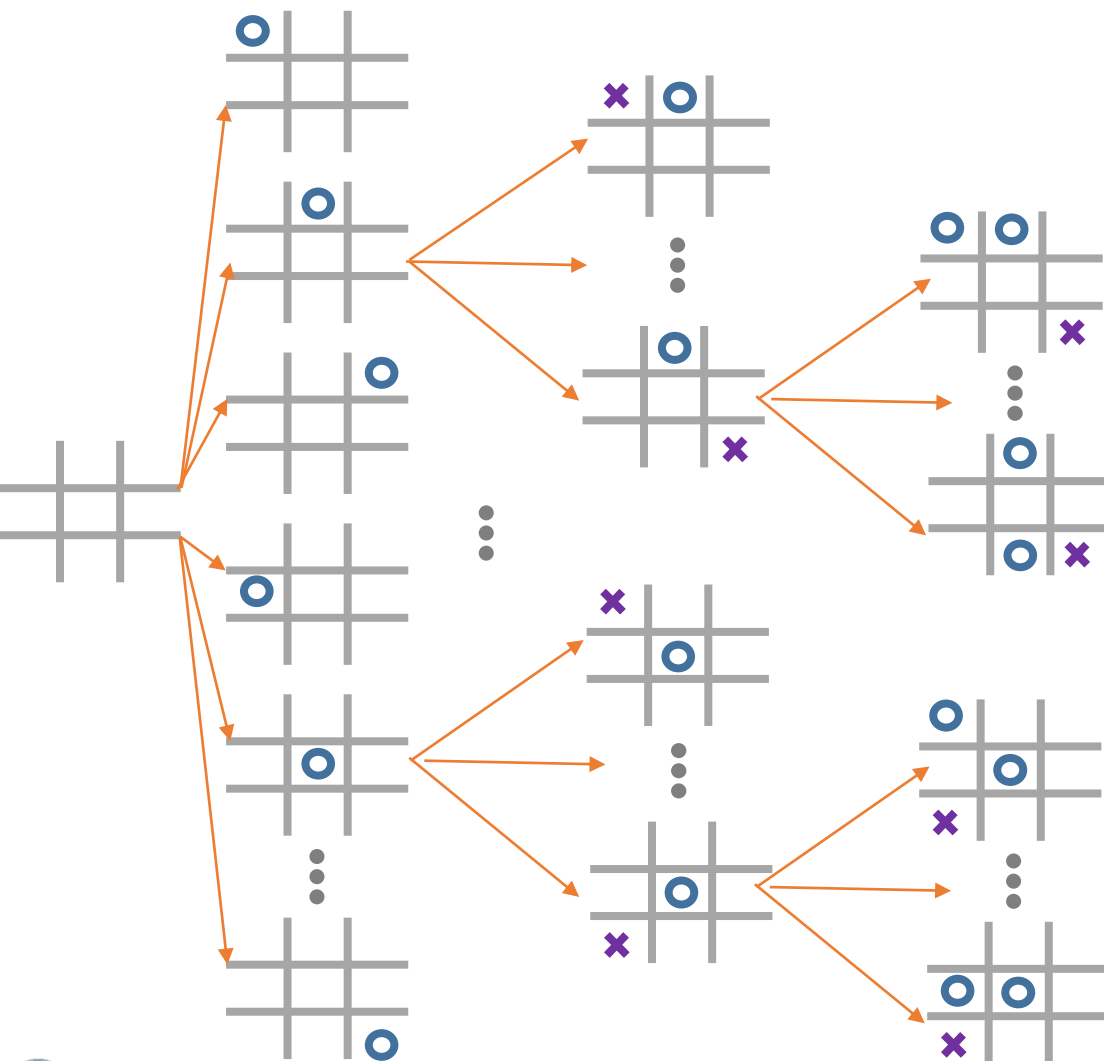
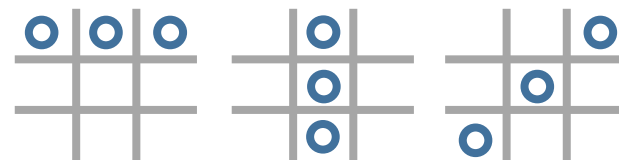
应用：极大极小搜索

- 分硬币游戏：有7枚硬币，只能将已分好的一堆硬币分成个数不等的两堆，当每堆只有1枚或2枚硬币则不能再分，红蓝双方轮流进行，直到谁不能再分，则为输



极大极小搜索：井字棋AI

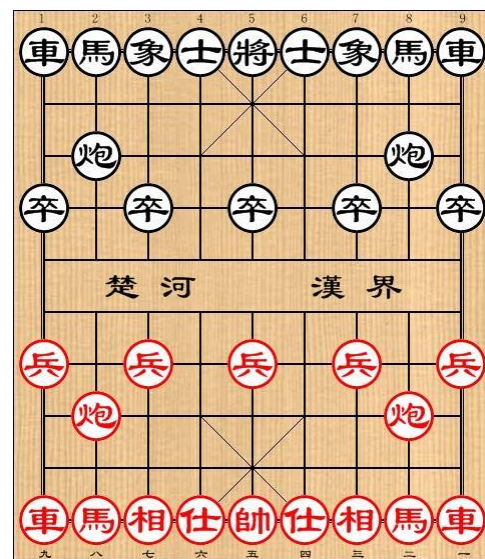
- 井字棋游戏：三子最先成线者胜



极大极小搜索：中国象棋和五子棋

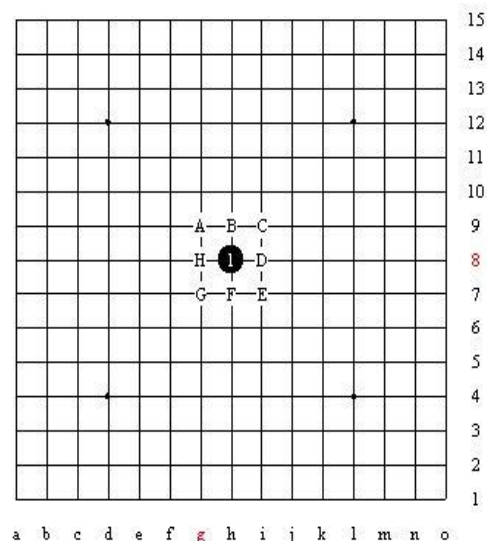
• 中国象棋AI搜索复杂度

- 棋盘大小 9×10 ，状态空间复杂度 10^{48}
- 一盘棋平均走50步，每一步大概有20多种走法，决策空间复杂度约为 10^{65}



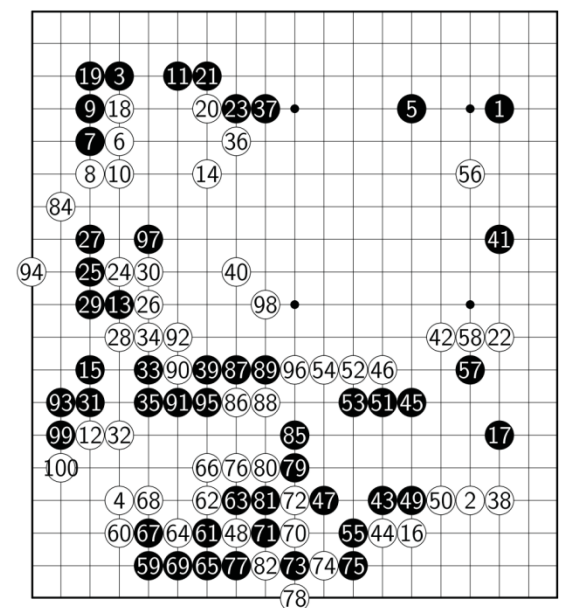
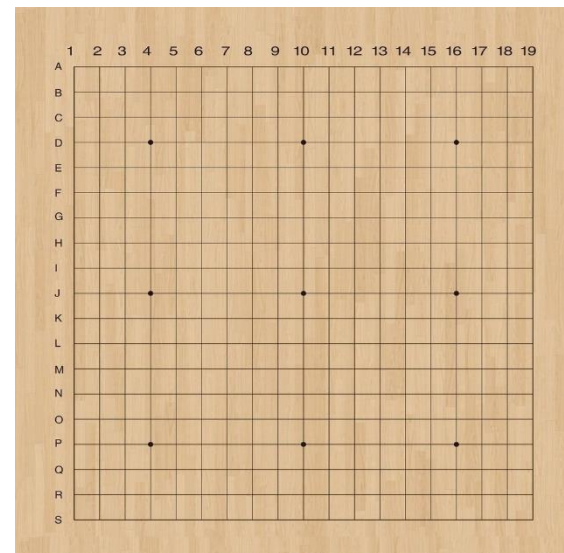
• 五子棋AI的搜索复杂度

- 棋盘大小 15×15
- 状态空间复杂度 $3^{225} \approx 10^{107}$
- 每一步有几十种走法，决策空间复杂度约为 10^{105}



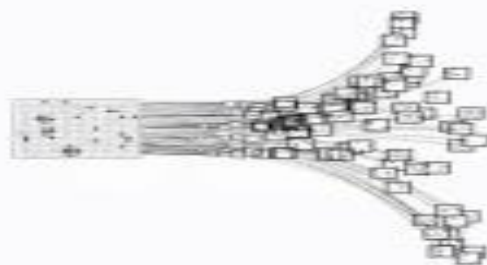
极大极小搜索应用示例：围棋AI

- 围棋AI的搜索复杂度
 - 棋盘大小 19×19 ，每一处三个状态
 - 状态空间大小： $3^{361} \approx 10^{172}$
 - 下完一局围棋约需要150步，每一步约有250走法，决策空间大小： $250^{150} \approx 10^{360}$
- 假设1微秒走一步，遍历象棋、五子棋、围棋博弈分别需要 10^{34} 、 10^{93} 和 10^{158} 年
- 结论：无论是象棋、五子棋和围棋，都无法穷举搜索！



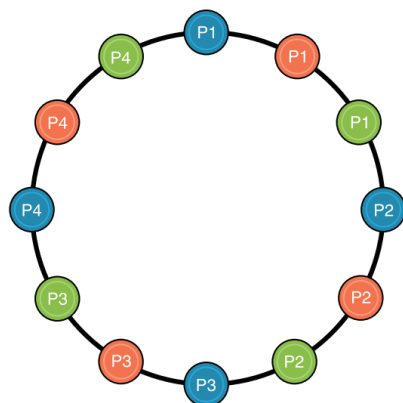
极大极小搜索应用示例：围棋AI

- 国际象棋AI和围棋AI中的博弈树搜索复杂度对比

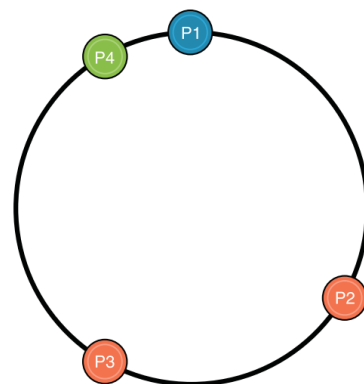


纳什均衡的选择问题

- 两人零和博弈的纳什均衡不存在选择问题，任意两个纳什均衡互相组合同样是纳什均衡
 - $(\sigma_1, \sigma_2), (\sigma'_1, \sigma'_2)$ 都是纳什均衡 $\rightarrow (\sigma_1, \sigma'_2), (\sigma'_1, \sigma_2)$ 都是纳什均衡
- 多人博弈中的纳什均衡存在选择问题，不同纳什均衡的组合通常不再是纳什均衡
- 因此，纳什均衡在多人博弈中并没有特别好的理论和性能保证



彼此之间距离越远越好
存在无穷多个纳什均衡，每一个都是均匀分布在环上



但是不同均衡的组合一般不再是纳什均衡了

重复博弈中的均衡

- 回忆：重复博弈，同样结构的博弈重复多次或无限次
 - 每次博弈称为阶段博弈 (stage games)
 - 多轮囚徒困境 (Iterated Prisoners' Dilemma)
- 重复博弈的子博弈精炼纳什均衡：
 - 最后一轮，所有玩家肯定会采取纳什均衡策略
 - 前面轮次的事情已经发生了，我们控制不了
 - 理性玩家会在最后一轮采取最优策略，也就是纳什均衡
 - 每轮都采取纳什均衡组成重复博弈的一个子博弈精炼纳什均衡
 - 数量非常庞大，如果每轮都有3个纳什均衡，共10轮→ 3^{10} 个！
 - 将各个轮次看作独立的，不是特别有用
 - 可能有其他形式的子博弈精炼纳什均衡，比如玩家可以通过对手的历史动作来决定自己的行为
 - 可以实现玩家之间的合作，比如实现囚徒困境的合作！

有限轮囚徒困境中的均衡

- 一次囚徒困境博弈，双方都抵赖
- 能否通过多轮博弈实现合作？坦白坦白
- 如果可以需要多少轮博弈才行呢？
- 从两轮开始：
 - 逆向递归法，第二轮会怎么做？互相抵赖
 - 第一轮会怎么做？这里的动作无法影响第二轮，因为第二轮总会互相抵赖，因此理性玩家会极大化自己第一轮的收益，同样会采取纳什均衡策略，也就是互相抵赖
 - 两轮中都互相抵赖（是一个子博弈精炼纳什均衡），无法合作
- 三轮、四轮、N轮同样的逻辑，都不能实现合作
- 有限轮囚徒困境，每轮都抵赖是唯一的子博弈精炼纳什均衡！

囚徒A \ 囚徒B	坦白	抵赖
	坦白	抵赖
坦白	3,3	1,4
抵赖	4,1	2,2

无限轮囚徒困境中的均衡

- 每轮都抵赖同样是子博弈精炼纳什均衡，但不唯一
- 冷酷策略（Grim Trigger Strategies）：
 - 玩家在开始时选择合作，在接下来的博弈中，如果对方合作则继续合作，而如果对方一旦背叛，则永远选择背叛，永不合作
 - 对任何玩家的一次性不合作将触发永远的不合作，在冷酷策略下，玩家没有改正错误的机会，所以才被称为冷酷
 - 冷酷的结果是双方都没有背叛对方的积极性，可以实现合作！
- 采取冷酷策略是无限轮囚徒困境的子博弈精炼纳什均衡
 - 如果触发背叛进入不合作阶段，双方都抵赖→子博弈精炼纳什均衡
 - 合作阶段，玩家是否有背叛的动机呢？
 - $3 + 3\sigma + 3\sigma^2 + \dots \geq (?)4 + 2\sigma + 2\sigma^2 + \dots \rightarrow \sigma \geq 0.5$
 - 当玩家比较在意未来收益的时候，他们没有动机选择背叛！

囚徒A \ 囚徒B	坦白	抵赖
坦白	3,3	1,4
抵赖	4,1	2,2

随机博弈中的均衡

- 回忆：随机博弈五元组 (Q, N, A, P, R)
 - 状态、玩家、动作、转移概率、奖励函数
 - 多智能体强化学习的基础，是目前研究的热点
 - Minimax-Q学习
 - Nash-Q学习
 - ...
 - 谷歌搜索multiagent learning survey
 - Multi-agent reinforcement learning: An overview
 - A survey and critique of multiagent deep reinforcement learning
 - Multi-Agent Reinforcement Learning: A Selective Overview of Theories and Algorithms
 - If multi-agent learning is the answer, what is the question?

博弈的解概念小结

- 帕累托最优
- 纳什均衡
 - 最优反应、静态、动态、完全信息、不完全信息
- 极小极大值定理
 - 极大极小策略、极大极小值
- 纳什均衡的选择问题
- 重复博弈中的均衡
 - 有限轮囚徒困境、无限轮囚徒困境、冷酷策略
- 随机博弈中的均衡
 - Minimax-Q学习
 - Nash-Q学习

本讲提纲



1 博弈表示方法

2 常见博弈类型

3 博弈的解概念

4 课程设计任务

课程设计可选任务一

- **任务名称：**计算博弈前沿问题调研 **撰写语言：**中文
 - 具体内容：阅读计算博弈领域相关论文或综述，了解与人工智能专业相关的前沿研究领域和方向，选择自己感兴趣的前沿研究问题或者与自己专业相关的研究方向，撰写关于智能博弈某类技术的法扎历史、研究现状和发展趋势的调研报告
 - 建议方向
 - 自博弈学习技术、不完美信息博弈学习技术、联盟博弈学习技术等
 - 多智能系统研究、多智能体学习技术、多智能体博弈技术、多智能体深度强化学习技术等
 - 不完美信息博弈性能评估、神经演化计算技术、遗传算法进化目标、多智能体系统学习目标评估等
 - 评分标准
 - 对所选方向调研内容是否全面、是否深入、是否有新的思考
 - 调研报告语言和结构规范性（调研报告模板会提供，篇幅要求加上参考文献最低不少于10页）

课程设计可选任务二

• 任务名称：智能博弈棋牌类AI设计

- 具体内容：以极大极小搜索为起步，通过调研Alpha-Beta搜索、蒙特卡洛树搜索、基于模型的树搜索（如MuZero）、深度反事实值最小化（DeepCFR）等相关技术，实现一个两人或多人零和博弈的AI，包括但不限于：五子棋、围棋、中国/国际象棋、麻将、德州扑克、桥牌、星际争霸、王者荣耀等
- 参考程序：建议采用C++或Python语言，可参考现有已发表相关论文或者GitHub上开源的相关程序，但需在实验报告中说明
 - 已经发表的论文：AlphaGo、AlphaGo Zero、AlphaGo Zero; DeepStack、Libratus、Pluribus; SuphX; AlphaStar、OpenAI Five等
 - GitHub开源程序有很多，比如：<https://github.com/Aleum/AlphaGo>等
- 评分准则：1) 综合考虑所设计AI的效果和效率以及算法创新性；2) 课程设计报告撰写的规范性和质量以及表述的清晰性

课程设计可选任务三

• 任务名称：计算博弈开放问题求解

- 具体内容：结合自己的兴趣爱好和研究方向，在计算博弈前沿研究方向中选择某一具体问题，针对该问题研究现状进行分析，总结现有方法存在的普遍问题，给出解决思路和方法，如有需要，可以对方法进行分析证明或者设计算法进行实验验证
- 候选问题
 - 选择特定博弈问题，分析其均衡解的存在性、可计算性等
 - 选择特定博弈问题，研究其均衡解和最优解之间的关系
 - 研究智能博弈问题中的非理性因素对博弈结果的影响
 - 研究不完美信息博弈算法的性能评估问题或学习求解框架
- 撰写语言：英文或者中文
- 评分标准：1) 所选择问题的代表性、解决思路的合理性和完整性等；2) 方案报告撰写的质量（方案报告模板会提供，篇幅要求加上参考文献最低不少于6页）

任务完成方式和提交要求

- 完成方式：个人独立完成或组队（不多于三人）完成，组队完成的任務需单独论述每个队员在其中发挥的作用
- 提交时间：课程倒数第三次课之前（预计在12月3日），在课程倒数第二次课上会选择优秀的调研报告、AI程序及方案报告进行现场讲解展示，其中具有原创性的AI程序或方案报告会邀请相关学生参与人工智能顶级国际会议IJCAI2021或者ICML2021论文的投稿
- 提交方法：在课程网站上提交，同时提交Word版实验报告和代码到助教邮箱（kangyongxin2015@ia.ac.cn）
- 邮件发送规范（组队学号为所有队员学号加号串联）
 - 邮件主题：计算博弈课程设计_学号_姓名
 - 附件名称：计算博弈课程设计_学号_姓名.zip

感谢聆听！

李凯

kai.li@ia.ac.cn

2020年9月24日



中国科学院大学
University of Chinese Academy of Sciences



自动化研究所
Institute of Automation