第七讲：算法博弈论简介

李凯
kai.li@ia.ac.cn

2020年11月05日

中国科学院大学
University of Chinese of Academy of Sciences
自动化研究所
CASIA
Institute of Automation

本讲提纲

- 算法博弈论概述
- 算法化机制设计
- 均衡的低效率性
- 均衡计算复杂度

本讲提纲

- 算法博弈论概述
- 算法化机制设计
- 均衡的低效率性
- 均衡计算复杂度

什么是算法博弈论？

- 自从1944年冯诺依曼发表《博弈论与经济行为》之后，博弈论在经济学领域得到了最为广泛的应用
- 我们前面讲的大部分内容基本都是在冯诺依曼和约翰纳什建立的体系上发展起来的，并且主要由经济学家们完成了整个传统博弈论大厦的构建
- 然而，在过去的近二十年里，由于互联网和计算机科学的飞速发展，极大地改变了人类和计算机的关系
- 博弈论和经济学的发展遇到了一些具有共同特点的难题，比如在自由经济市场、互联网等复杂博弈场景中如何找到或者设计系统的均衡点，于是出现了“算法博弈论”（Algorithmic Game Theory）这个新的交叉研究方向

什么是算法博弈论？

- 算法博弈论：计算机科学和博弈论的交叉点，研究博弈论中的计算问题，即如何“求解”现实世界中的博弈
- 这里的求解不光要分析出博弈的均衡，还要设计算法，优化计算得到实际的均衡解
- 重点研究的问题：
  - 如何更好的设计博弈的规则?
  - 博弈的参与者理论上是否能快速收敛到均衡点?
  - 实际达到的均衡点和最优结果之间存在多大的差异?
  - 能否有多项式复杂度的算法来计算均衡点?

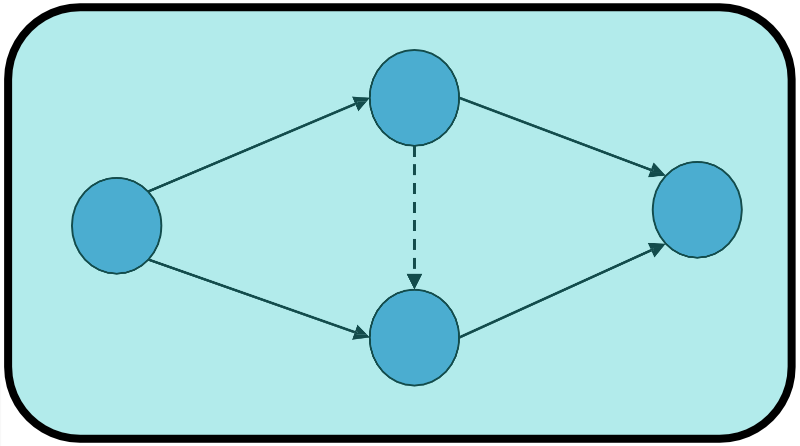
“If your laptop can not find the equilibrium, neither can the market.”

-- 微软研究院Kamal Jain

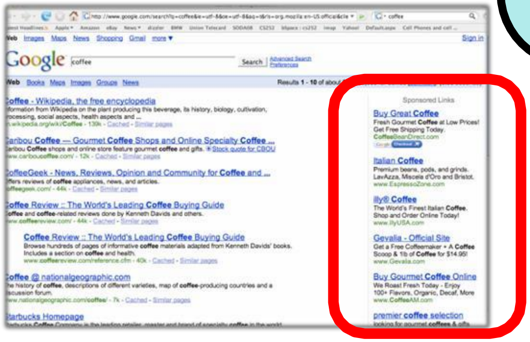
算法博弈论的应用领域

市场定价

网络路由



在线广告



种群进化

选举

主要研究内容

- 算法化机制设计（Algorithmic Mechanism Design）
  - 通俗来讲就是设计博弈的规则，使得各个参与者在追求个人利益的同时能够达到设计者所设定的目标
- 均衡的低效率性（Inefficiency of Equilibria）
  - 研究复杂博弈中均衡解和全局最优解的关系，**可以理解成理想化计划经济和自由市场经济的对比**，如果有一件事情，需要多人合作，在不考虑个人得失的情况下，假设最好能产出A；但如果让这些人自己玩，最后收敛到一个均衡，产出了B，A与B之间的大小关系反映了均衡与最优之间的差距
- 均衡计算复杂度（Complexity of Finding Equilibria）
  - 纳什只证明了均衡的存在性，但没有说如何能找到；研究如何通过算法求解博弈的均衡，如果无法在多项式时间内求解，证明此类问题是难的

算法化机制设计案例分析

• 2012年奥运会羽毛球女双比赛

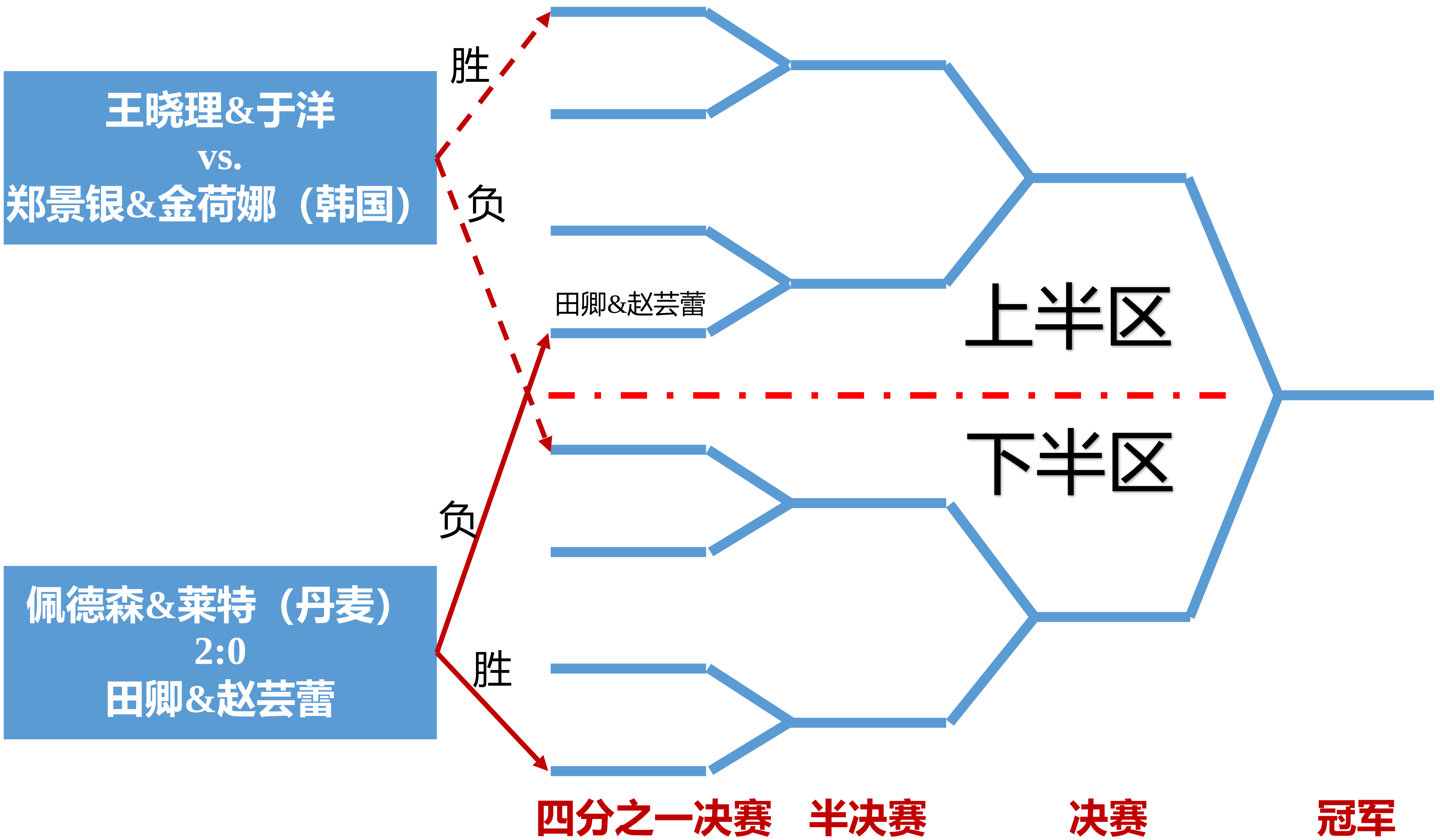
羽毛球女双比赛最初八强

- 郑景银 / 金荷娜 （韩国）
- 王晓理 / 于洋 （中国）
- 程文欣 / 简毓瑾 （中华台北）
- 藤井瑞希 / 垣岩令佳 （日本）
- 河贞恩 / 金旼贞 （韩国）
- 格雷西娅·波利 / 梅利亚娜·乔哈里 （印尼）
- 吕特·尤尔 / 克里斯汀娜·彼德森 （丹麦）
- 田卿 / 赵芸蕾 （中国）

参赛选手	国籍	排名
王晓理 / 于洋	中国	1
田卿 / 赵芸蕾	中国	2
河贞恩 / 金旼贞	韩国	3
藤井瑞希 / 垣岩令佳	日本	4
卡米拉·吕特·尤尔 / 克里斯汀娜·彼德森	丹麦	5
前田美顺 / 末纲聪子	日本	6
郑景银 / 金荷娜	韩国	8
程文欣 / 简毓瑾	中华台北	10
格雷西娅·波利 / 梅利亚娜·乔哈里	印尼	12
欣塔·穆利亚·萨里 / 姚蕾	新加坡	13
潘乐恩 / 谢影雪	中国香港	15
瓦拉·古塔 / 阿什维尼·蓬纳帕	印度	16
瓦莱里亚·索罗金娜 / 尼娜·维斯洛娃	俄罗斯	18
亚历山德拉·布鲁斯 / 李文珊	加拿大	28
周玉莲 / 雷努加·韦兰	澳大利亚	35
米歇尔·克莱尔·爱德华兹 / 安娜里·维尔容	南非	44

算法化机制设计案例分析

- 2012年奥运会羽毛球女双比赛



算法化机制设计案例分析

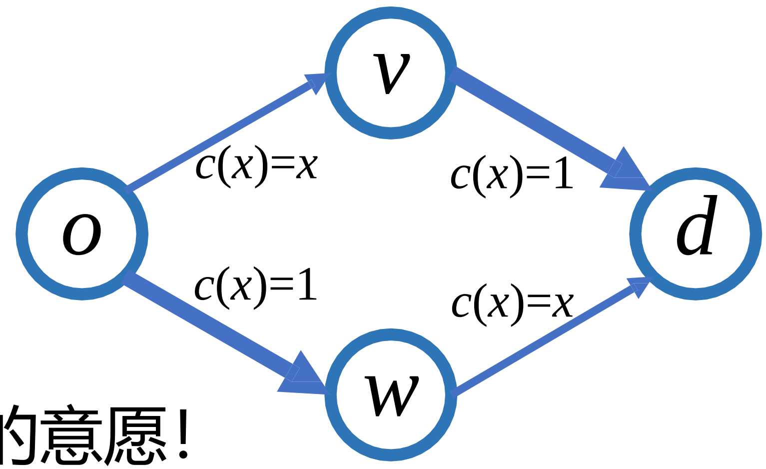
- 消极比赛视频

算法化机制设计案例分析

- 这个案例中比赛设计者和运动员的动机不完全一致！
  - 运动员想要拿到更好的名次
  - 比赛设计者希望运动员打出体育精神，比赛越精彩越好
- 给我们的启示：博弈的规则很重要
- 失败的规则会导致我们不想见到甚至不可控的结果
- 博弈设计者应该考虑到博弈各方的可能行为，而不是让参与者做出违背自己利益的行为
- 或许这里的运动员缺乏体育精神，但是责任更大的其实是比赛组织者，毕竟运动员也是极大化他们的收益，这本身并没有错
- 机制设计就是研究博弈规则制定的理论

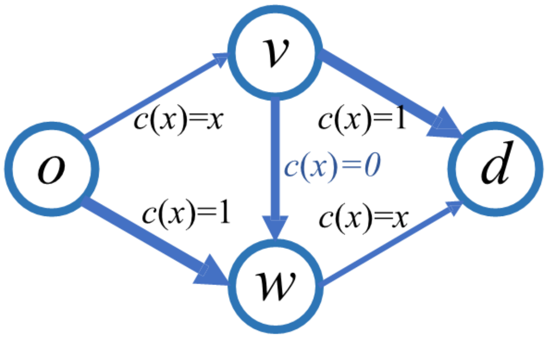
均衡的低效率性案例分析

- 布雷斯悖论（Braess’s Paradox）
  - 有一个起点 o 和终点 d，有固定数量的司机从起点到终点
  - 从起点到终点有两条路线，每条路线都分别包含一条长宽路，一条窄短路
  - 长宽路需要1个小时通过，不管多少车辆通过
  - 短窄路需要的时间=通过的车辆占总体的比例
- 通过道路所需的平均时间是多少？
  - 由于两条路线其实是一样的
  - 两条路线会平分所有的车流量
  - 因此，平均时间是1+0.5=1.5小时
  - 这是达到均衡的平均用时
  - 为什么是均衡？没有人有改变路线的意愿！



均衡的低效率性案例分析

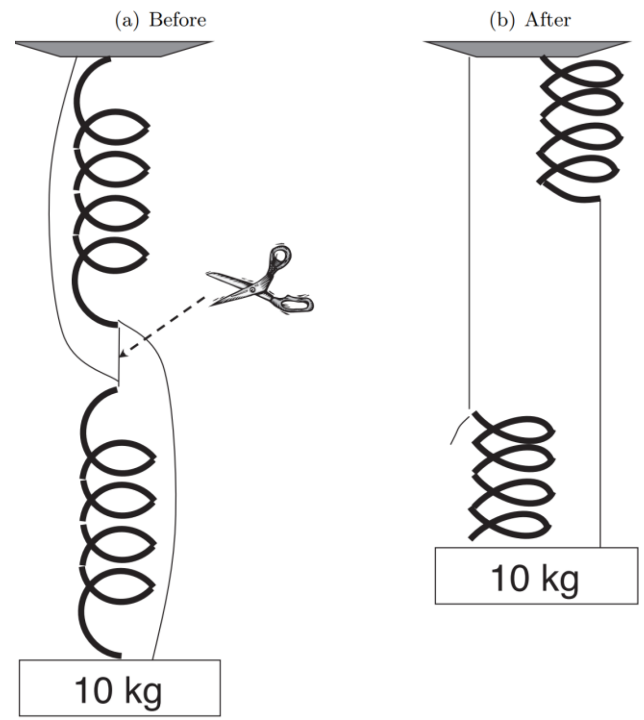
- 布雷斯悖论（Braess’s Paradox）
  - 如果我们在道路网中增修一条超级便道之后呢？可以瞬间传送
  - 如果你是司机，你会做什么样的选择？
  - $o \to v \to w \to d$不会比原来两条路线差
  - 因此，司机都会采用$o \to v \to w \to d$这条路线，达到均衡
  - 均衡时平均的通勤时间变为1+1=2!
  - 而最优的平均通勤时间是1.5
  - 好心办了坏事!



- 无秩序代价（Price of Anarchy，POA）
  - 均衡/最优
  - 这里，$\mathrm{POA} = \frac{2}{3/2} = \frac{4}{3}$
  - POA理想条件下为1，有些场景均衡可以接近最优

均衡的低效率性示例分析

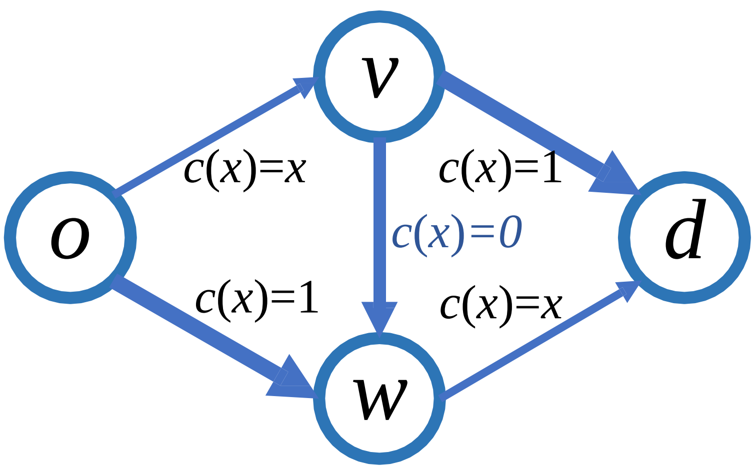
- 绳子和弹簧（String and Springs）
  - 通过选择合适长度的弹簧和绳子，可以达到右图(a)的平衡状态
  - 剪掉右图(a)中两根弹簧之间的短绳，弹簧可以迅速切换到右图(b)的状态
  - 物体为什么会上升?
  - 左图两根弹簧都承重10Kg，右图则分别承重5Kg



- 这个例子其实和上页中的道路网是等价的
- 剪掉短绳等价于撤掉瞬时传送装置
- 分析不同场景下，均衡和最优的关系非常重要

均衡计算复杂度案例分析

- 计算机能否快速求解均衡？
  - 布雷斯悖论：计算机简单搜索就可得到均衡
  - 石头剪刀布：两人零和博弈通过线性规划或迭代学习可以求解
  - 其他更一般的博弈呢?



均衡计算复杂度案例分析

- 计算机能否找到更为一般性博弈的均衡？
- 对两人非零和博弈来说，求解纳什均衡已经是PPAD完全问题了，目前不存在多项式的解法
- 多人博弈的纳什均衡求解更为困难
- 由于纳什均衡的求解复杂性，纳什均衡的意义也越来越被质疑
- 连计算机都求解不了纳什均衡，博弈参与者应该也很难达到纳什均衡
- 因此越来越多的研究转向可高效求解的均衡概念，比如相关均衡（Correlated Equilibria）、粗相关均衡（Coarse Correlated Equilibria），或者近似求解纳什均衡

算法博弈论概述小结

- 算法化机制设计
  - 设计博弈的规则，使得各个参与者在追求个人利益的同时能够达到设计者所设定的目标
  - 奥运会羽毛球比赛的丑闻就是由于机制设计缺陷所导致的
- 均衡的低效率性
  - 探究博弈中均衡解和全局最优解的关系
  - 无秩序代价POA：均衡/全局最优，越接近1越好
- 均衡计算复杂度
  - 除了小部分特殊类型的博弈，求解纳什均衡是困难的
  - 纳什均衡的意义
  - 可高效计算的均衡概念
  - 近似求解

本讲提纲

- 算法博弈论概述
- 算法化机制设计
- 均衡的低效率性
- 均衡计算复杂度

产生和发展历史

- 算法化机制设计：最初由希伯来耶路撒冷大学的研究者在他们1999年发表的一篇论文中提出
- 该研究方向结合了经济学中的效用最大化和机制设计、博弈论中的理性假设和纳什均衡、理论计算机科学中的复杂度分析和算法设计等概念和理论
- 与经济学中机制设计的差别
  - 将计算复杂度作为一个核心限制，比如必须多项式时间内可实现，一些经典的经济学机制设计模型就不再适用
- 应用领域
  - 政治选举、市场活动、物品拍卖、政府政策等

Nisan, Noam; Ronen, Amir (1999), "Algorithmic mechanism design", Proceedings of the thirty-first annual ACM symposium on Theory of computing: 129–140.

产生和发展历史

- 2007年诺贝尔经济学奖：赫维茨（Hurwicz），马斯金（Maskin）和迈尔森（Myerson）
  - Prize motivation: “for having lid the foundations of mechanism design theory.”
  - 获奖原因：为机制设计理论建立了基础

- 接下来我们从机制设计最重要的应用之一：拍卖理论，对算法化机制设计进行介绍

单物品拍卖

- 单物品拍卖（Single-Item Auctions）
- 给定一件物品和n个潜在购买者
- 每个购买者i对于物品都有一个自己的价值评估v_i，$v_i \geq 0$，v_i代表i愿意出的最高价格，v_i对于其他购买者和拍卖者都是不可知的，购买者都想出最少的钱获得商品
- 购买者效益模型（Bidder Utility Model）：如果购买者i成功以价格p拍得该物品，则他的效益为$v_i - p$; 如果未能拍得该物品，则效益为0。

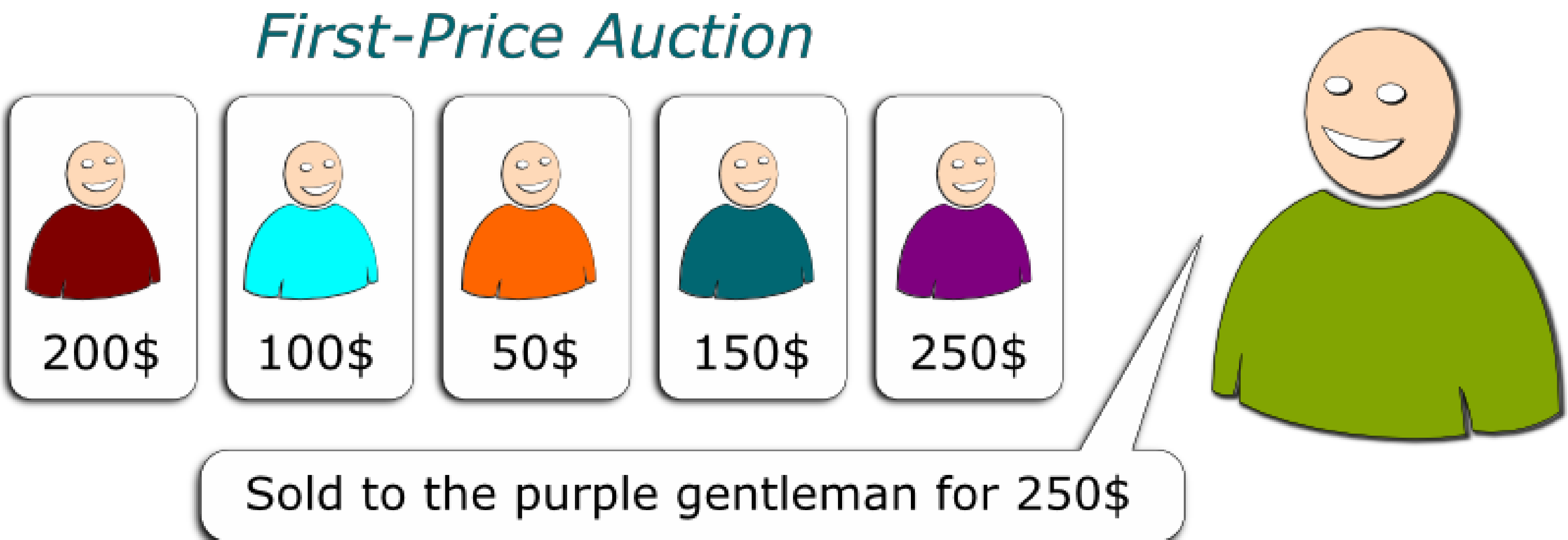
$$\begin{cases} v_i - p, & \text{拍到} \\ 0, & \text{未拍到} \end{cases}$$

密封出价拍卖

- 密封出价拍卖（Sealed-Bid Auctions）
- 每一个购买者将自己的出价b_i悄悄地告诉卖家
  - 比如通过一个密闭的信封
- 卖家决定谁成功拍得该物品（如果存在的话）
  - 最自然的是出价最高者获得商品，这也是本课程默认的选择
- 卖家决定拍得该物品的售价（Selling Price）
  - 指定售价很重要，可以影响购买者的行为
  - 举一个极端的例子，卖家是大公无私的，不收取获得商品者任何钱，那么结果会怎样?
  - 拍卖会变成→谁能写出最大的数字!

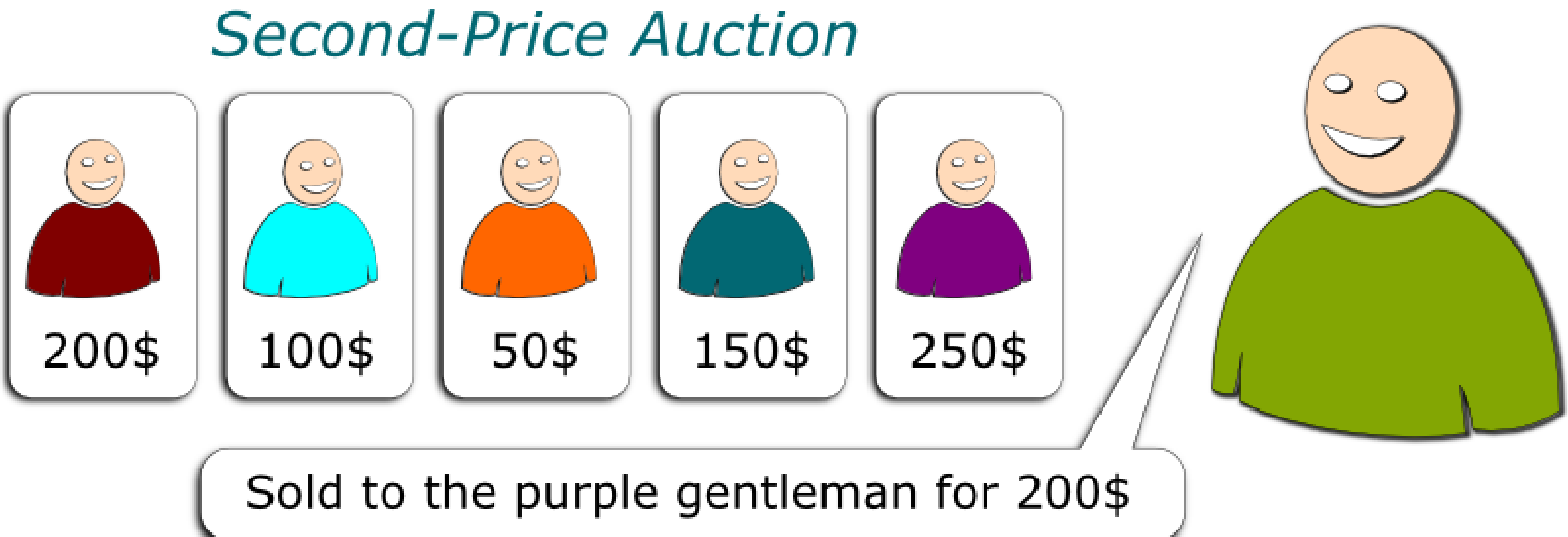
第一价格拍卖

- 第一价格拍卖 (First-Price Auctions)： 出价最高的购买者以自己的出价购买拍到的物品
  - 是实际中常见的一种拍卖定价方式，但是理论上其实很难分析
  - 对购买者来说，很难确定如何出价
  - 对卖家来说，很难预测会发生什么



第二价格拍卖

- 第二价格拍卖（Second-Price Auctions）：出价最高的购买者以“所有出价第二高的价格”购得物品
  - 英式拍卖、eBay、常见拍卖公司拍卖等
  - 第二价格拍卖也称为Vickrey拍卖，具有很多优良的性质，是非常重要的拍卖类型



第二价格拍卖

- 第二价格拍卖中的动机（Incentives）

在一个第二价格拍卖中，每一个参与者i都有占优策略：设定他的出价b_i为自己对于拍卖品的私有评估值v_i

证明：对于任意一个竞拍者i及其出价b_i和评估值v_i，以及其他竞拍者出价b_{-i}，我们需要证明在$b_i = v_i$竞拍者i的收益最大化，令$B = \max_{j \neq i} b_j$,对于i来说，只有两种可能：

- 当$b_i < B$, 那么i竞拍失败，收益为0;
- 当$b_i \geq B$, 那么i竞拍成功，收益为$v_i - B$;

当$v_i < B$时，i最大的收益是$\max(0, v_i - B) = 0$
当$v_i \geq B$时，i最大的收益是$\max(0, v_i - B) = v_i - B$
对于任意一种情况，竞拍者i都可以通过真实地报告自己的的评估值来达到，因此$b_i = v_i$是i的占优策略

第二价格拍卖

- 第二价格拍卖中每一个参与者i都有占优策略：设定他的出价b_i为自己对于拍卖品的私有评估值v_i
- 这使得参与者很容易做决策
- 当参与者i报价时，他不用去分析其他参与人的信息
- 不用去管报价的总人数有多少
- 不用去管其他人的私有评估值是多少
- 不用去管他们是否按照自己的私有评估值报价
- 这与第一价格拍卖完全不同！
- 第一价格拍卖中如果按照自己的评估值报价，那么收益总是0，需要报价略低一些，低多少依赖于其他人的报价

第二价格拍卖

- 第二价格拍卖中的非负效益（Nonnegative Utility）

在一个第二价格拍卖中，每一个真实报告自己对于商品评估值的参与者的收益必定为非负值

证明：对于任意一个竞拍者i及其出价b_i和评估值v_i

当竞拍失败时，收益为0

当竞拍成功时，收益为$v_i - p, p$是第二高的出价值

由于i赢得竞拍（i是出价最高者，$b_i > p$）并且他的出价$b_i = v_i$，那么$v_i - p > 0$

因此，对于任意一种情况，竞拍者i的收益值均为非负值

拍卖的DSIC和社会收益最大化特性

DSIC特性：Dominant-Strategy Incentive Compatible

在一个拍卖中，如果对于每一个竞拍者，真实报价总是占优策略，并且真实报价总是得到非负的收益，那么我们就称该拍卖具备DSIC特性

- 拍卖的社会收益：定义$\sum_{i=1}^{n} v_i x_i$为一个单物品拍卖结果的社会收益，其中当i拍卖成功$x_i = 1$，反之$x_i = 0$，对于单物品拍卖，只有一件商品，$\sum_{i=1}^{n} x_i \leq 1$

社会收益最大化特性：Welfare Maximizing

在一个拍卖中，如果竞拍者都真实报价，竞拍的结果能够最大化社会收益，那么我们就称该拍卖具备社会收益最大化特性

第二价格拍卖是完美拍卖

- 完美拍卖的定义

定义：完美拍卖（Ideal Auction）

满足下面三个条件的拍卖称为完美拍卖：

1. 强动机保证：它是一个满足DSIC特性的拍卖
2. 强效果保证：它是社会福利最大化的拍卖
3. 计算高效性：可在多项式（最好是线性）时间复杂度内计算实现

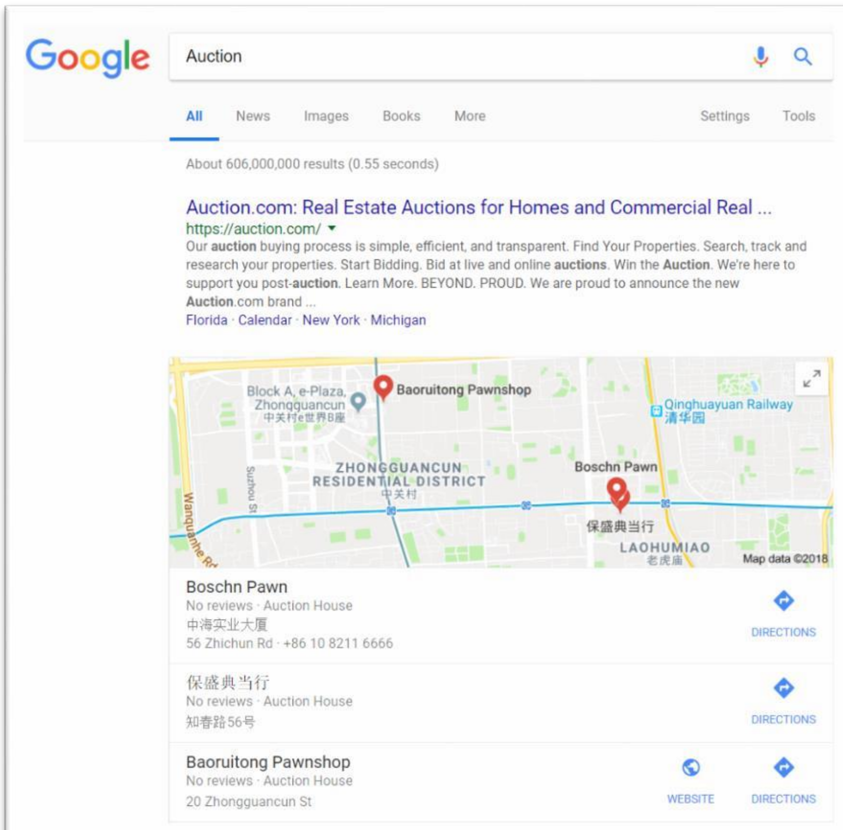
- 定理：第二价格拍卖是完美拍卖
  - 证明：根据之前的结论即可证明条件1和2，求解第二价格拍卖所需的时间复杂度为线性，因此定理得证

第二价格拍卖是完美拍卖

- 第二价格拍卖这一完美拍卖的三个特性都非常重要
- DSIC：
  - 从竞拍者的角度看，DSIC特性使得他们很容易报价
  - 从卖家的角度看，DSIC特性使得竞拍的结果很容易分析
- 社会福利最大化：
  - 虽然各竞拍者的评估值是私有的，但是通过第二价格拍卖，可以将物品给予“认为该商品价值最大的人”，也就是评估值最大的人
- 计算高效性：
  - 拍卖想要有实用价值，相应的算法计算层面必须足够快
  - 第二价格拍卖时间复杂度是线性的

案例研究：搜索引擎竞价拍卖

- 搜索引擎竞价拍卖（ Sponsored Search Auctions ）
  - 谷歌的广告搜索收益占据了整个公司87%的收入
  - 百度的广告搜索收益占据了整个公司91%的收入



案例研究：搜索引擎竞价拍卖

- 搜索引擎竞价排名基本模型：多物品拍卖模型
  - 物品：每个特定关键词的搜索页面包含的k个广告位
  - 竞拍者：比如相机关键词的竞拍者可能是索尼、尼康等厂商
  - 每个物品的价值不一样，使用点击率（CTR）评估其价值
    - 第个i广告位的CTR用α_i表示，$\alpha_1 \geq \alpha_2 \geq \cdots \geq \alpha_k$
  - 每个广告商i有一个私有价值v_i
    - v_i可以认为是他的广告被点击给他带来的收益
    - 因此，第j个广告位给他带来的期望收益为$v_i \alpha_j$
- 问题：如何设计拍卖规则？是否有完美的搜索引擎竞价拍卖呢？
  - 社会福利最大化：$\sum_{i=1}^{n} v_i x_i$，x_i代表厂商i竞得的广告位对应的CTR值，如果没有竞得任何广告位则为0
  - 一个厂商只能得一个广告位，一个广告位只能给一个厂商

案例研究：搜索引擎竞价拍卖

- 搜索引擎竞价拍卖机制设计
  - 选择哪个厂商赢得哪个广告位
  - 广告位售价多少?
- 两阶段设计方法
  - 步骤一：假设所有竞拍者都真实报价，我们如何为广告商分配广告位从而使得要求2和要求3满足?
  - 步骤二：假设已经有了步骤一的答案，我们如何设定广告位售价从而使得要求1得到满足?
- 如果我们能完成这两步，就会得到一个完美拍卖
  - 步骤2保证了DSIC特性，因此竞拍者会真实报价
  - 这正好满足了步骤1的条件，而步骤1可以保证社会福利最大化和计算高效性，因此满足完美拍卖的三个要求

案例研究：搜索引擎竞价拍卖

- 步骤一：假设所有竞拍者都真实报价，如何为广告商分配广告位从而使得社会福利最大化并且可以高效计算？
  - 直接采用简单的贪心算法就可以满足
  - 为出价第i高的广告商分配第i好的广告位
  - 也就是将广告商的出价按照由高到低排序，分别放到按照点击率由高到低排序的位置中
  - 只需一个排序即可，因此计算效率非常高
- 步骤二：如何设定广告位的售价从而使得DSIC特性得到满足？
  - 是否有类似于第二价格拍卖那样，找到一种售价的制定规则，使得所有广告商的占优策略是真实报价且收益非负？
  - 麦尔森定理（Myerson’s Lemma）

麦尔森定理（Myerson's Lemma）

- 单参数环境：
  - n个智能体，每个智能体i有一个私有的非负评估值v_i，表示它对于得到的单位物品的估价
  - 可行集X，它的每一个元素是一个非负的n维向量$(x_1, \ldots, x_n)$，其中x_i表示分给智能体i的物品数量
- 实例对应
  - 单物品拍卖：X是由01向量（至多有一个1）组成的集合，$\sum_{i=1}^{n} x_i \leq 1$
  - k-物品拍卖：k个相同的物品，X是由01向量组成的集合，$\sum_{i=1}^{n} x_i \leq k$
  - 搜索引擎竞价拍卖：如果第i个广告商安排了第j个位置，x_i等于位置j的点击率α_j
- 拍卖：一个用于交换物品和钱的特殊机制

麦尔森定理（Myerson's Lemma）

- 分配和支付规则：对应密封拍卖中“谁赢得拍卖”和“谁支付多少”的问题
  - 从所有竞拍者收集他们的竞价b= $(b_1, \dots, b_n)$组成竞价向量
  - 分配规则：按照竞价向量b，从可行集X选择一个合适的分配方式$x(\boldsymbol{b}) \in X \subseteq \mathbb{R}^n$
  - 支付规则：按照竞价向量b，选择一个支付方式$p(\boldsymbol{b}) \in \mathbb{R}^n$
- 竞拍者i的收益： $u_i(\boldsymbol{b}) = v_i \cdot x_i(\boldsymbol{b}) - p_i(\boldsymbol{b})$
- 对于支付规则的限制： $p_i(b) \in [0, b_i \cdot x_i(b)]$
  - 大于等于0要求拍卖者不能赔钱
  - 小于等于$b_i \cdot x_i(b)$ 要求真实报价的竞拍者收益不为负

麦尔森定理（Myerson's Lemma）

- 麦尔森定理相关的一些定义

定义：分配规则的可实现性（Implementable Allocation Rule）

在单参数环境下，当分配规则x有一个对应的支付规则p使得(x, p) 具有DSIC特性，那么就称这个分配规则x是可实现的

- 如果我们想要设计一个满足DSIC特性的拍卖，那么设计的分配规则必须是可实现的
  - 回到我们的竞价拍卖问题，为出价第i高的广告商分配第i好的广告位这一分配规则（已经满足社会福利最大化和可高效计算）是否可实现呢？如果可以，我们就找到了一个完美的竞价拍卖！
  - 回忆单物品拍卖，将物品给出价最高者这一分配规则可实现吗？
  - 可以，对应的支付规则是：所有出价第二高的价格购得物品！

麦尔森定理（Myerson's Lemma）

- 麦尔森定理相关的一些定义

定义：分配规则的单调性（Monotone Allocation Rule）

在单参数环境下，对于每一个竞拍者i和其他竞拍者的出价$\boldsymbol{b}_{-i}$，如果$x_i(z, \boldsymbol{b}_{-1})$相对于$i$的出价$z$是非递减的，称该分配规则x是单调的

- 出价越高，得到的物品数不会减少
- 对于单物品拍卖来说，“将物品给出价最高者”这一分配规则是单调的，将物品给出价第二高的呢？非单调
- 回到竞价拍卖问题，“为出价第i高的广告商分配第i好的广告位”这一分配规则是单调的
  - 增加出价只会使得分配的广告位变好

麦尔森定理（Myerson's Lemma）

- 麦尔森定理的内容

麦尔森定理（Myerson's Lemma）

固定一个单参数环境：

a. 一个分配规则x是可实现的当且仅当它是单调的
b. 如果一个分配规则x是单调的，那么存在一个唯一的支付规则使得 (x, p)具有DSIC特性，$b_i = 0$时$p_i(b)$=0
c. 上述支付规则可以通过具体的计算公式给出

- 定理分析
  - 内容a表明分配规则的可实现性和单调性是等价的，单调性更有实用价值，因为比较容易验证
  - 内容b说明当分配规则可实现时，有唯一的支付规则可以确保DSIC特性
  - 内容c说明有公式来直接计算支付规则，非常方便

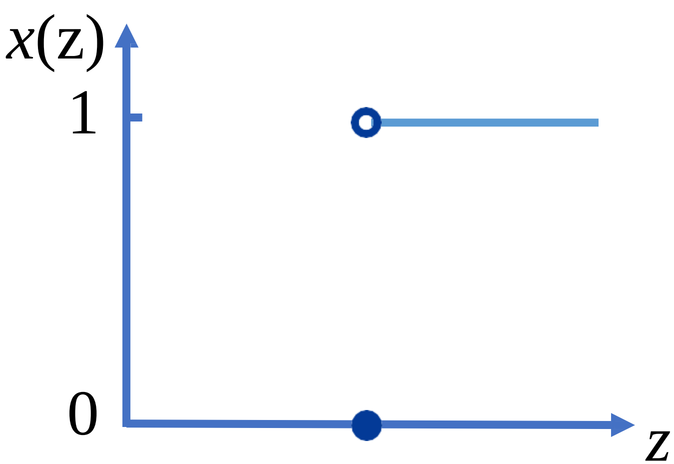
麦尔森定理（Myerson's Lemma）

- 麦尔森定理的证明

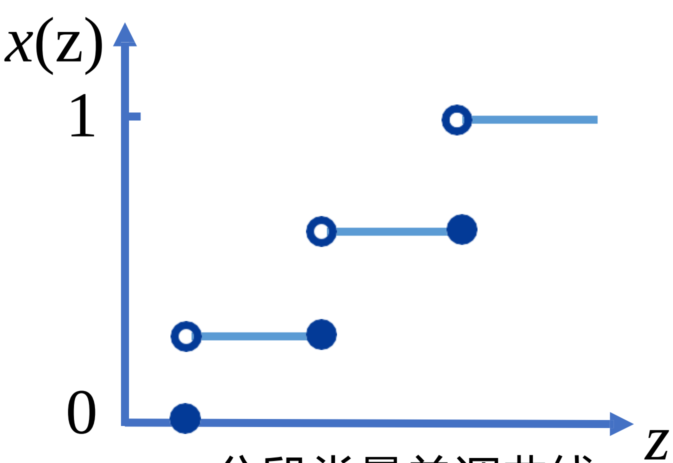
证明：见参考书籍《Twenty Lectures on Algorithmic Game Theory》的第3.4节。
需要记住证明中得到的一个公式：**麦尔森支付公式**

$$p_i(b_i, \boldsymbol{b}_{-i}) = \sum_{j=1}^{l} z_j \cdot \left[\text{jump in } x_i(\cdot, \boldsymbol{b}_{-i}) \text{ at } z_j\right]$$

其中$z_1, z_2, \ldots, z_l$为分配函数$x_i(\cdot, \boldsymbol{b}_{-i})$在区间$[0, b_i]$中的断点



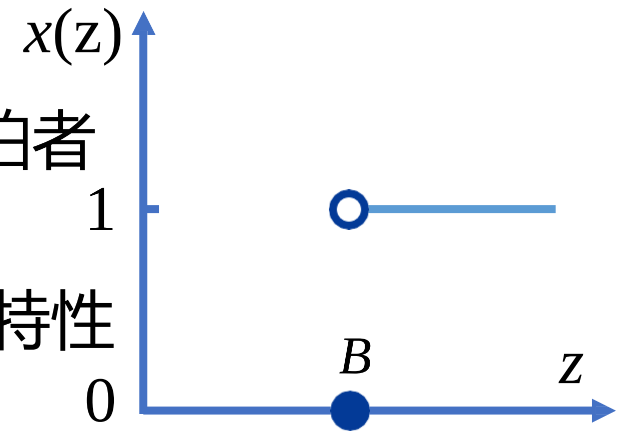
0-1单调曲线



分段常量单调曲线

麦尔森定理（Myerson's Lemma）

- 麦尔森定理的应用：单物品拍卖
  - 分配规则是将物品分配给最高出价的竞拍者
  - 分配规则满足单调性，因此可实现
  - 存在唯一的支付规则使得拍卖具有DSIC特性
  - 如何设计支付规则？



解答：使用麦尔森支付公式

$$p_i(b_i, \boldsymbol{b}_{-i}) = \sum_{j=1}^{l} z_j \cdot [\text{jump in } x_i(\cdot, \boldsymbol{b}_{-i}) \text{ at } z_j]$$

$$B = max_{j \neq i} b_j$$

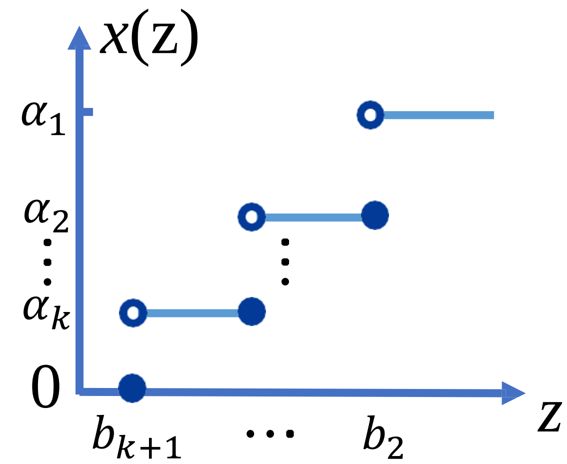
对于单物品最高出价获得拍卖品的原则，只有一个跳跃点：在B处，跳跃的高度为1，因此，当参与者i成功竞拍 $p_i(b_i, \boldsymbol{b}_{-i}) = B$，否则$p_i(b_i, \boldsymbol{b}_{-i}) = 0$

单物品拍卖所用的第二价格支付规则合理

因此，第二价格支付是麦尔森定理的一个特殊形式

麦尔森定理（Myerson's Lemma）

- 麦尔森定理的应用：搜索引擎竞价拍卖
  - 给定k个按照点击率由高到低排序($\alpha_1 \geq \alpha_2 \geq \cdots \geq \alpha_k$)的位置，分配规则：为出价第$i$高的广告商分配第$i$好的广告位
  - 分配规则满足单调性，因此可实现
  - 存在唯一的支付规则使得拍卖具有DSIC特性
  - 如何设计支付规则?



解答：给定一个竞价向量**b**，从高到低将出价重排：$b_1 \geq b_2 \geq \cdots \geq b_n$。以第一个竞价者为例，固定其他出价，将第一个竞价者的出价由0逐渐增加到b_1，当z成为第j高的出价时，就会有一个在$\alpha_j - \alpha_{j+1}$的跳跃

根据麦尔森支付公式，可以得到:

$$p_i(\mathbf{b}) = \sum_{j=i}^{k} b_{j+1} \cdot (\alpha_j - \alpha_{j+1})，其中\alpha_{k+1} = 0$$

最右边的跃点在$z = b_2$，一旦$z > b_2, x_i(z, \boldsymbol{b}_{-i}) = \alpha_1$，跳跃大小$\alpha_1 - \alpha_2$

最左边的跃点在$z = b_{k+1}$，一旦$z > b_{k+1}, x_i(z, \boldsymbol{b}_{-i}) = \alpha_k$，跳跃大小$\alpha_k - \alpha_{k+1} = \alpha_k$

算法化机制设计小结

- 拍卖的DSIC特性：对于每一个竞拍者，真实报价总是占优策略，并且真实报价总是得到非负的收益
- 社会收益最大化：如果竞拍者都真实报价，竞拍的结果能够最大化社会收益$\sum_{i=1}^{n} v_i x_i$
- 完美拍卖：满足DSIC、社会收益最大化、可以高效计算
- 第二价格拍卖是完美拍卖
- 设计完美拍卖两步走：
  - 1. **假设真实报价，设计分配规则满足：社会收益最大化并可高效计算**
  - 2. **设计支付规则使得DSIC特性成立，也就是真实报价是占优策略**
  - 分配规则可实现：存在一个支付规则，使得DSIC满足，因此目标变成寻找可实现的分配规则
  - 分配规则可实现与单调是等价的，因此寻找单调的分配规则即可
  - 有了单调的分配规则，麦尔森支付公式可以直接找到对应的支付规则

本讲提纲

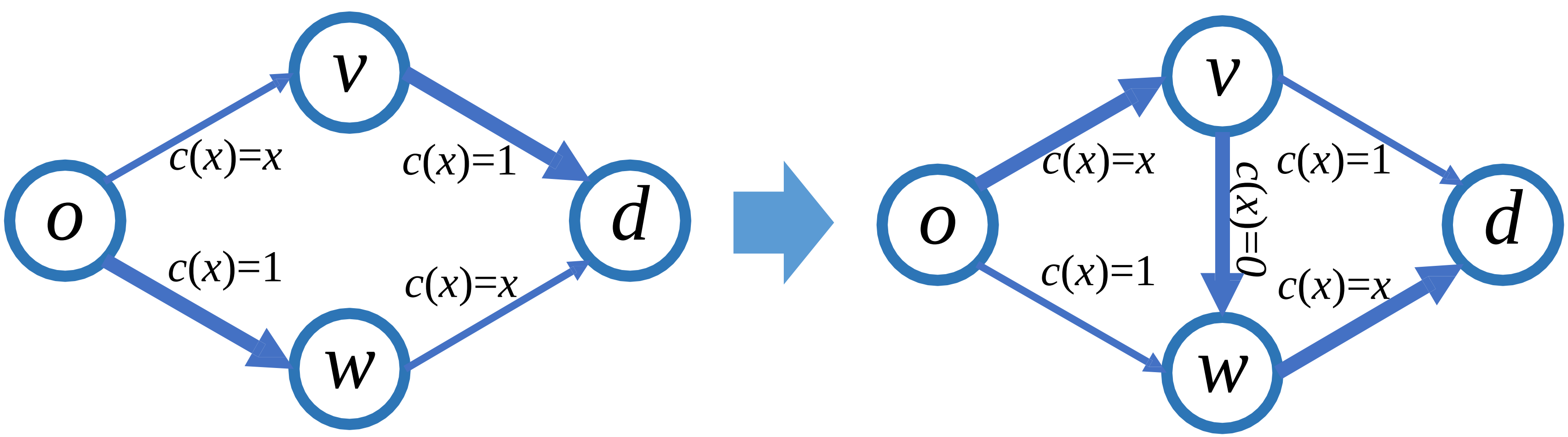
- 算法博弈论概述
- 算法化机制设计
- 均衡的低效率性
- 均衡计算复杂度

自私路由（Selfish Routing）

- **自私路由**：最初由计算机科学家蒂姆·拉夫加登（Tim Roughgarden）提出，每个人在道路网络中的移动方式在自己看来是最佳的（“用户优先”），不过大家的整体行为却不一定是最优的（未达到“系统优先”）
- 每个人都是“自私的出行人”，纳什均衡有可能会陷入局部最优陷阱，没有达到系统最优，这一现象称为均衡的低效率性
- 但是有时候纳什均衡和系统最优很接近，系统满足什么样的条件才能使得纳什均衡接近最优呢？

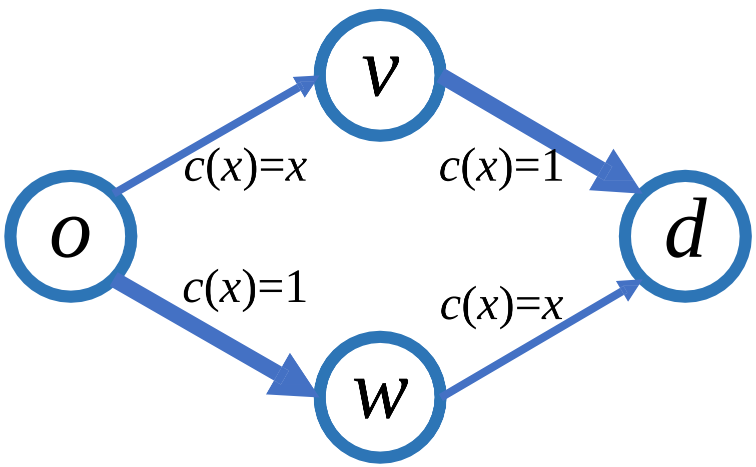
布雷斯悖论详细讲解

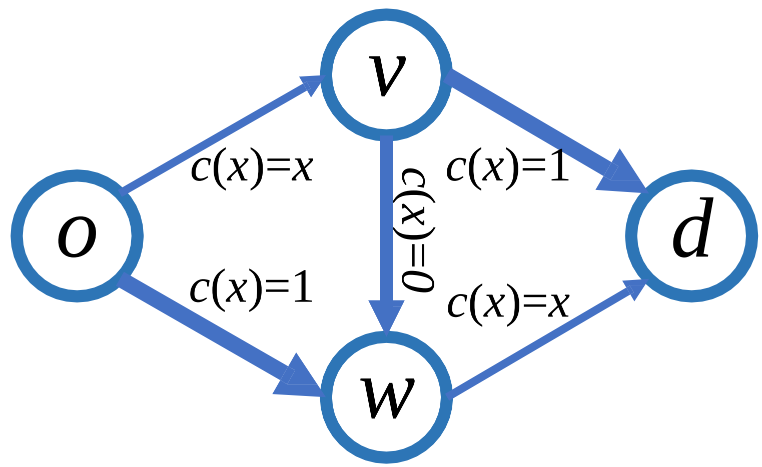
- **布雷斯悖论**（名字来自德国数学家迪特里希·布雷斯）指在一个交通网络上增加一条路段反而使网络上的旅行时间增加；这一附加路段不但没有减少通勤时间，反而降低了整个交通网络的服务水准，这种出力不讨好且与人们直观感受相背的交通网络现象主要源于“纳什均衡点并不一定是社会最优”这一现象



布雷斯悖论详细讲解

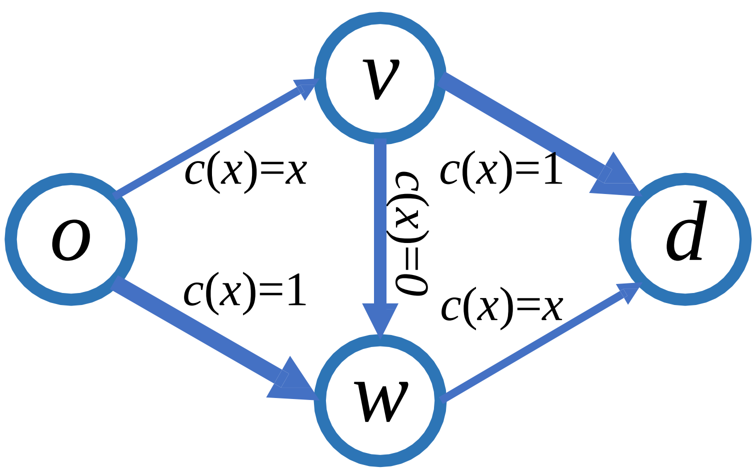
- 符号含义：o为起点，d为终点，x代表路径上的流量比例($x \in [0,1]$)， c(·)为路径的代价函数，即通过时间
- 在原始网络（左图）中，均衡条件下，50%司机选择o-v-d，剩下50%司机选择o-w-d，最终期望代价为1.5
- 在扩展网络（右图）中，加入一条通过时间近于0的v-w，此时所有司机都会选择 o-v-w-d，最终期望代价为2.0





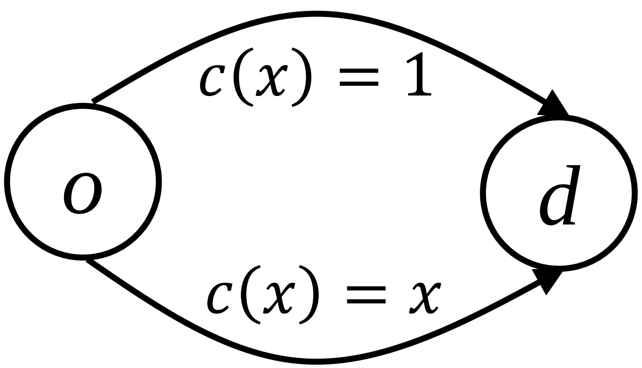
无秩序代价（Price of Anarchy，PoA）

- 无秩序代价：博弈论中用于评测在一个系统中，因为其参与单位的利己行为（或自私行为）而导致的效率下降程度，PoA值越大，说明均衡的效率越低
- $\mathrm{PoA} = \dfrac{\text{均衡策略的代价}}{\text{最优策略的代价}}$
- $\mathrm{PoA} \geq 1$
- 以布雷斯悖论中的扩展网络为例，$\mathrm{POA} = \dfrac{2}{3/2} = \dfrac{4}{3}$



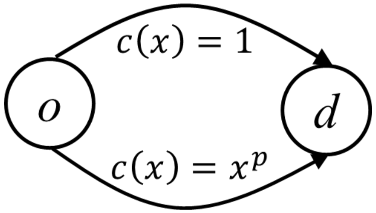
Pigou网络

- Pigou网络：
  - 网络包含两个顶点，出发点o和目标点d
  - 从o到d有两条路径，上路径与下路径
  - x：使用某路径的交通量占所有交通量的比例
  - 在上路径中，代价函数为常数$c(x) = 1$
  - 在下路径中，代价函数为 $c(x) = x$
  - 纳什均衡策略：都选下路径，平均通勤时间1小时
  - 最优策略：上下路各一半，平均通勤时间3/4小时
  - $\text{POA} = \frac{1}{3/4} = \frac{4}{3}$



非线性Pigou网络

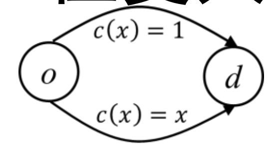
- 如果在下路径中代价函数变为 $c(x) = x^p$，p可以很大
- 选择下路径仍然是占优策略，因此，均衡下的平均通勤时间仍然是1小时



- 但是最优情况会变得明显不同
- 假设交通还是上下路各一半，当$p \to \infty$时，下路径几乎不费时间，因此平均通勤0.5小时
- 当$1-\varepsilon$的流量用下路径，$\varepsilon \to \infty, p \to \infty$时，下路径还是几乎不费时间，上路径几乎没车，因此平均通勤时间接近0
- 因此，非线性Pigou网络的POA值在$p \to \infty$时，会变得无穷大

自私路由网络的POA界

- 线性Pigou网POA接近1，非线性的POA则可以无限大
- 是否是非线性导致了自私路由网络的POA值变大？
- 定义C为一组类型的代价函数



  - 比如$C = \{ax + b : a, b \geq 0\}$代表仿射类型的代价函数

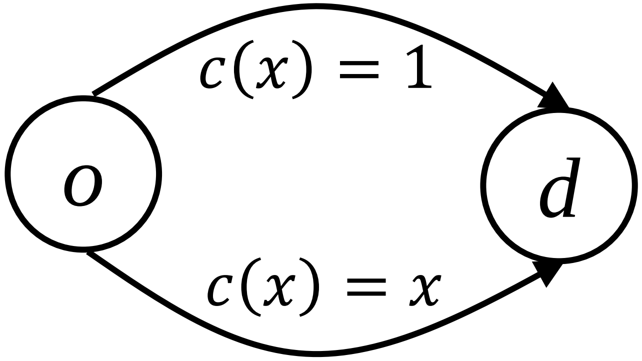
自私路由网络的POA界（Tight POA Bounds for Selfish Routing）

指定代价函数类型C，任意代价函数$c \in C$的自私路由网络，其POA最大值是通过类Pigou网络（Pigou-like network）得到

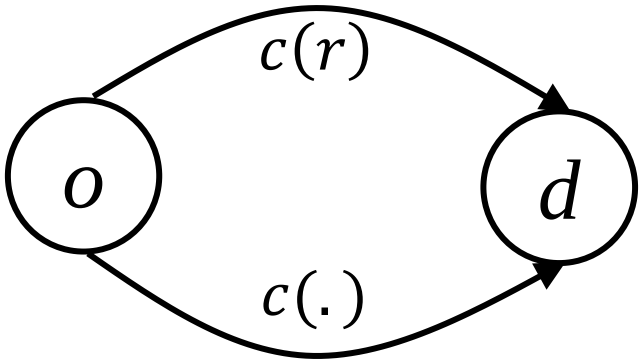
- POA最差情况的网络结构总是最简单的：类Pigou网络！
- 均衡的低效性和网络结构的复杂度无关，而和代价函数有关！不用遍历所有的网络结构，只需要遍历类Pigou网络就可以找到最差情况的POA！

自私路由网络的POA界

- 类Pigou网络的正式定义
  - 与前面的Pigou网络类似
  - 网络包含两个顶点，出发点o和目标点d
  - 从o到d有两条路径，上路径与下路径
  - 非负的总交通量r
  - 在上路径中，代价函数为常数$c(r)$
  - 在下路径中，代价函数为 $c(.)$，比如可以是线性、二次等
  - 代价函数非负，连续，非减，代表单位流量的通勤时间



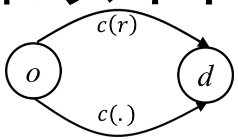
Pigou网络



类Pigou网络

自私路由网络的POA界

- 类Pigou网络，选择下路径仍然是占优策略，因为下路径最大的代价等于$c(r)$：当所有流量都选择下路径时
- 因此，均衡时的总交通时间为$r \cdot c(r)$：总流量乘以单位流量的通勤时间



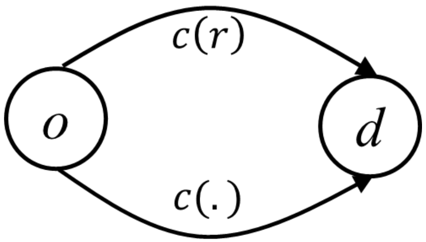
- 最小总交通时间：$\min_{0 \le x \le r} \{x \cdot c(x) + (r - x) \cdot c(r)\}$，其中$x$是走下路径的交通量，条件可以从$0 \le x \le r$变为$0 \le x$因为代价函数非减，最小值总是在$0 \le x \le r$中达到
- 类Pigou网的POA：

$$\max_{0 \le x} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r - x) \cdot c(r)} \right\}$$

自私路由网络的POA界

- Pigou界（Pigou bound）：对于任意一个包含非负、连续、非减代价函数的集合C，定义Pigou界$\alpha(C)$
  - $\alpha(C)$为类Pigou网络 PoA的最大值，其中这些类Pigou网络的下路径的代价函数来源于C，形式上可以表示为

$$\max_{c\in C}\max_{r\ge 0}\max_{0\le x}\left\{\frac{r\cdot c(r)}{x\cdot c(x)+(r-x)\cdot c(r)}\right\}$$



  - 分别从代价函数、总交通量、选择下路径的交通量这三个变量里搜索类Pigou网络的PoA的最大值

代价函数类型	代表公式	最差情况的PoA值（Pigou界）
线性函数	$x+1$	$4/3\approx 1.3$
二次函数	x^2+x+1	$\frac{3\sqrt{3}}{3\sqrt{3}-2}\approx 1.6$
三次函数	x^3+x^2+x+1	$\frac{4\sqrt[3]{4}}{4\sqrt[3]{4}-3}\approx 1.9$
p次函数	$x^p+x^{p-1}+\cdots+x+1$	$\frac{(p+1)\sqrt[p]{p+1}}{(p+1)\sqrt[p]{p+1}-p}\approx\frac{p}{\ln p}$

自私路由网络的POA界

- 自私路由网络POA界再回顾：
- 对于任意形式的自私路由网络，假设它每一条边上的代价函数都属于集合C，那么该网络的最大PoA值（最坏情况下的PoA值）可以在对应的类Pigou网络中达到
- 也就是说对于任意的自私路由网络，其每一条边上的代价函数都属于集合C，则该网络的PoA$\leq \alpha(C)$
- 重点：对任意的自私路由网络都成立！
- 自私路由网络的PoA值和结构无关，而和代价函数有关
- 计算复杂网络的最大PoA值问题可以转化为计算相对应的类Pigou网络的最大PoA值，也就是求Pigou界$\alpha(C)$

均衡的低效率性小结

- 布雷斯悖论
- 无秩序代价：均衡/最优，大于等于1，值越大表明均衡越低效
- 线性Pigou网络
- 非线性Pigou网络
- 自私路由网络的PoA值和结构无关，而和代价函数有关
- 类Pigou网络
- Pigou界$\alpha(C)$
- 计算复杂网络的最大PoA值问题可以转化为计算相对应的类Pigou网络的最大PoA值，也就是求Pigou界

本讲提纲

- 算法博弈论概述
- 算法化机制设计
- 均衡的低效率性
- 均衡计算复杂度

研究均衡计算复杂度的原因

- 经济学中的博弈论研究中存在的尚未回答的问题

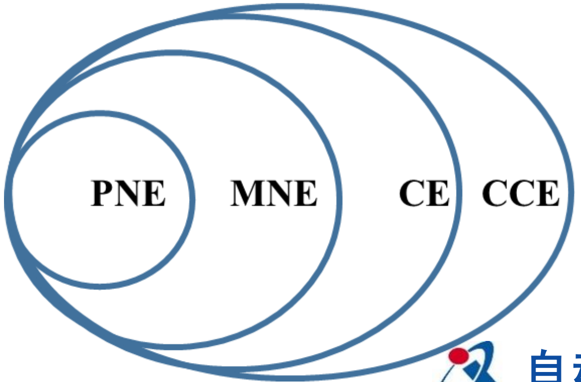
对于一个存在均衡的博弈，所有参与人都足够聪明，他们能找到均衡吗？

如果能够找到均衡，用来找均衡的算法是什么呢？

如果有一个能找均衡的算法，算法会在多长时间找到均衡呢？

不同类型的均衡

- 不同类型均衡的存在性与可高效计算性
- PNE：Pure Nash Equilibria纯策略纳什均衡
  - 不一定存在，除了一些特殊的博弈之外都难以计算
- MNE：Mixed Nash Equilibria混合策略纳什均衡
  - 一定存在，除了两人零和博弈之外都难以计算
- CE：Correlated Equilibria相关均衡
  - 一定存在，容易计算
- CCE：Coarse Correlated Equilibria粗相关均衡
  - 一定存在，更容易计算
- 后面课程会详细讲述CE、CCE



纯纳什均衡求解算法

- 代价最小化博弈（Cost-Minimization Games）
  - 包含k个智能体
  - 每个智能体i有一个纯策略集合S_i，$s_i \in S_i$
  - 每个智能体都有一个非负代价函数$C_i(\mathbf{s})$，$\mathbf{s} = (s_1, \dots, s_k)$
- 自私路由网络是一个CMG，$C_i(\mathbf{s})$代表给定其他智能体路径选择的基础上，智能体i在它所选路径上的总耗时

纯策略纳什均衡（Pure Nash Equilibrium，PNE）

一个策略组合$\mathbf{s} = (s_1, \dots, s_k)$是PNE，如果对于任意智能体$i$来说，$C_i(\mathbf{s}) \le C_i(s_i', \mathbf{s}_{-i})$

- 不一定存在，什么时候一定存在后面课程会讲述
  - 势博弈

纯纳什均衡求解算法

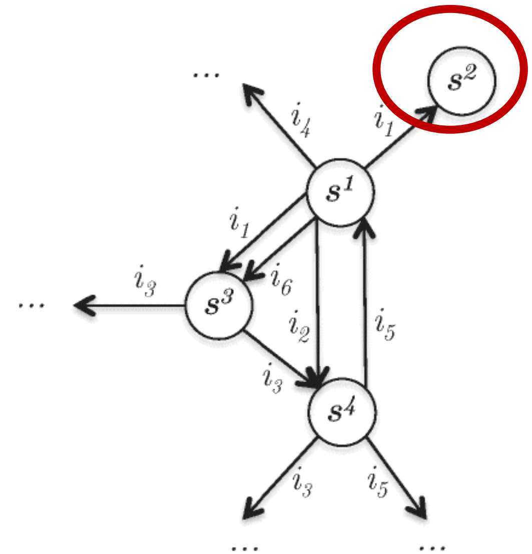
- 动态最优反应算法（Best-Response Dynamics）

```
While 当前策略组合s不是PNE
    选择任意一个参与者i
    针对i选择一个单方面的有利偏离s'_i，使得
        C_i(s'_i, s_{-i}) < C_i(s)
    更新s = {s'_i, s_{-i}}
End While
```

  - 第二行中i可以任意选择，也可以按照预定次序选择，比如$i = 1, \dots, k$
  - 上述算法更准确地应该称为Better-Response Dynamics
  - Best-Response Dynamics，选择s_i'使之满足$s_i' = \operatorname{argmin}_{s_i'} c_i(s_i', \boldsymbol{s}_{-i})$
  - 该算法可以用任意策略组合作为初始化

纯纳什均衡求解算法

- BRD可以看作是有向图上的行走过程
  - 顶点代表一个策略组合
  - 顶点的每条出边代表一个单方面的有利偏离
- 当一个顶点只有进入的边，没有出去的边，则该顶点为当前博弈问题的PNE，每个玩家都没有有利的偏离！
- 如果博弈没有PNE，则会出现循环（Cycles）
- 如果有PNE，收敛速度有多快？
- 是否能多项式时间复杂度收敛？



计算复杂性理论简介

- Computational Complexity Theory
  - 理论计算机科学和数学的一个分支
  - 对于计算机而言，任何一个问题的求解都需要资源，计算复杂性理论定量分析计算解决问题所需的资源：时间和空间
  - 将计算问题按照所需时间资源的不同予以分类，就得到了常见的P、NP、NP完全、NP难这样的概念
- 时间复杂度的概念
  - 并不是表示一个程序解决问题需要花多少时间，而是当问题规模扩大后，程序需要的时间增长得有多快
  - $O(1)$：常数级复杂度，不管数据有多大，花的时间总是那么多
  - $O(n)$：数据规模变得有多大，花的时间也跟着变得有多长
  - $O(n^2)$：数据扩大2倍，时间变慢4倍
  - $O(a^n)$：指数级复杂度

计算复杂性理论简介

- 时间复杂度的概念
  - $O(n!)$：阶乘级复杂度
  - $O(2n^2) = O(n^2)$
  - $O(n^3 + n^2) = O(n^3)$
  - $O(0.01n^3)$效率低于$O(100n^2)$
  - $O(n^{100})$效率低于$O(1.01^n)$
  - $O(1), O(\log(n)), O(n^a)$：多项式复杂度
  - $O(a^n), O(n!)$：非多项式复杂度
  - 后者的复杂度无论如何都远远大于前者
  - 我们选择的算法通常都需要是多项式级的复杂度
  - 非多项式级的复杂度需要的时间太多，其复杂度计算机往往不能承受

计算复杂性理论简介

- 是不是所有的问题都可以找到多项式复杂度的算法呢？
  - Hamilton回路问题：给一个图，能否找到一条经过每个顶点一次且恰好一次（不遗漏也不重复）最后又走回来的路（满足这个条件的路径叫做Hamilton回路）
  - 这个问题现在还没有找到多项式的算法
- P类问题
  - 如果一个问题可以找到一个能在多项式的时间里解决它的算法，那么这个问题就属于P问题
  - 大学《算法与数据结构》课程中的大部分问题都是P类问题
- NP问题
  - NP问题不是非P类问题，NP问题是指可以在多项式的时间里验证一个解的问题
  - 有可能找一个解很困难，但验证一个解很容易

计算复杂性理论简介

- Hamilton回路是一个NP问题
  - 因为验证一条路是否恰好经过了每一个顶点非常容易
  - 但是找到一条这样的路径却很难
  - 通常只有NP问题才可能找到多项式的算法，我们不会指望一个连多项式时间验证一个解都不行的问题存在一个解决它的多项式时间复杂度的算法
- P类问题都是NP问题
  - 能多项式时间地解决一个问题，必然能多项式时间验证一个问题的解
- 是否所有的NP问题都是P类问题？
  - 即是通常所谓的NP问题：证明或推翻P=NP
  - 世界七大数学难题之一

计算复杂性理论简介

- NP问题： P=NP？
  - 这是一个耗费了很多时间和精力也没有解决的终极问题，好比物理学中的大统一和数学中的歌德巴赫猜想等
  - 人们普遍认为，P≠NP
  - 也就是，存在至少一个不可能有多项式复杂度算法的NP问题
  - 在研究NP问题的过程中找出了一类非常特殊的NP问题：NP完全（NPC）问题
  - 正是NPC问题的存在，使人们相信P≠NP
- 约化Reducibility
  - 一个问题A可以约化为问题B的含义是：可以用问题B的解法解决问题A
  - 也就是问题A可以“变成”问题B

计算复杂性理论简介

- 约化Reducibility
  - 求解一个一元一次方程和求解一个一元二次方程
  - 前者可以约化为后者，知道如何解一个一元二次方程那么一定能解出一元一次方程
  - 核心是找到一个规则，按照这个规则把解一元一次方程程序的输入数据变一下，用在解一元二次方程的程序上，两个程序总能得到一样的结果
  - 对应项系数不变，一元二次方程的二次项系数为0，按照这个规则把前一个问题转换成后一个问题，两个问题就等价了
  - 问题A可约化为问题B：B的时间复杂度高于或者等于A的时间复杂度。也就是说，问题A不比问题B难
  - 约化具有传递性：如果问题A可约化为问题B，问题B可约化为问题C，则问题A一定可约化为问题C

计算复杂性理论简介

- 约化Reducibility
  - 较正式定义：如果能找到一个变化法则，对任意一个程序A的输入，都能按这个法则变换成程序B的输入，使两程序的输出相同，那么我们说，问题A可约化为问题B
  - 可约化指的是可多项式地约化(Polynomial-time Reducible)，即变换输入的方法能在多项式的时间里完成，约化的过程只有用多项式的时间完成才有意义
  - 一个问题约化为另一个问题，时间复杂度增加了，问题的应用范围也增大了，通过对某些问题不断约化，能够不断寻找复杂度更高，但应用范围更广的算法来代替复杂度虽然低，但只能用于很小的一类问题的算法
  - 如果不断地约化上去，不断找到能通吃若干小NP问题的一个稍复杂的大NP问题，那么最后是否有可能找到一个时间复杂度最高，能通吃所有NP问题的这样一个超级NP问题呢？

计算复杂性理论简介

- 存在一个NP问题，所有的NP问题都可以约化成它
- 只要解决了这个问题，所有的NP问题都解决了！
- 这种问题的存在难以置信，更加不可思议的是，这种问题不只一个，它有很多个，它是一类问题：NPC问题
- NPC问题：
  - 它是一个NP问题；所有的NP问题都可以约化到它
  - 证明一个问题是NPC问题也很简单：先证明它是一个NP问题，再证明一个已知的NPC问题能约化到它
  - 所有的NP问题都能约化成NPC问题，那么只要任意一个NPC问题找到了一个多项式算法，那么所有的NP问题都能用这个算法解决，NP也就等于P了，正是NPC问题的存在，使人们相信P≠NP，NPC问题目前没有多项式的有效算法，只能用指数级甚至阶乘级复杂度的搜索

计算复杂性理论简介

- NP-Hard问题：
  - 所有的NP问题都可以约化到它，但他本身不一定是NP
  - NP-Hard问题要比NPC问题的范围广
  - 即使NPC问题发现了多项式级的算法，NP-Hard问题有可能仍然无法得到多项式级的算法
- 第一个NPC问题从哪里来的？
  - 逻辑电路问题，其它的NPC问题都是由这个问题约化而来的
  - 逻辑电路问题是NPC类问题的鼻祖
  - 给定一个逻辑电路，问是否存在一种输入使输出为True
  - 有了第一个NPC问题后，一大堆NPC问题就出现了，因为再证明一个新的NPC问题只需要将一个已知的NPC问题约化到它就行了，后来，Hamilton 回路成了NPC问题，旅行商问题也成了NPC问题

纯纳什均衡计算复杂度

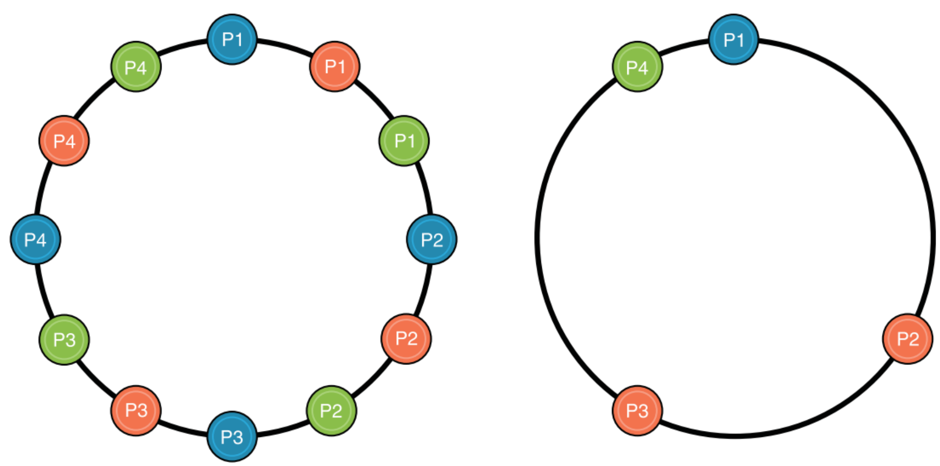
- PLS问题（polynomial local search）：后面课程讲述
- PLS完全问题：
  - 如果所有的PLS问题都可以约化到L问题，且$L \in PLS$，则称L为PLS完全问题，绝大多数专家认为不存在多项式复杂度的算法来求解PLS完全问题
- Congestion Games阻塞博弈是自私路由网络的扩展
  - 边变成了资源集合E，每个玩家策略集合$S_i \subseteq 2^E$（资源子集）
- 计算阻塞博弈中的PNE是一个PLS完全问题
  - 和NP完全问题证明类似，先证明它是一个PLS问题，再证明一个已知的PLS完全问题能约化到它
  - 计算k智能体的阻塞博弈的PNE，需要k的指数级的迭代次数，无论每次迭代如何选择获益的偏离

混合纳什均衡计算复杂度

- 有向图的多项式校验参数问题
- Polynomial Parity Arguments on Directed Graphs，PPAD
- MNE不是判定问题，存在性是由纳什定理直接保证的
- 也不是优化问题，因为没有一个优化的目标
- 从本质上讲，它是一类不动点的计算问题
- 美国加州大学伯克利分校的克里斯托斯教授定义了PPAD计算复杂类来描述这类问题
- The complexity of computing a Nash equilibrium
- 求解一般的两人博弈纳什均衡是PPAD完全问题
- 求解三人/四人博弈纳什均衡是PPAD完全问题

混合纳什均衡计算复杂度

- 纳什均衡遭受的批评不仅是计算上困难的问题，还有选择性问题：
  - 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
  - $(\sigma_1, \sigma_2), (\sigma_1', \sigma_2')$是纳什均衡→ $(\sigma_1, \sigma_2'), (\sigma_1', \sigma_2)$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



α-Rank

SCIENTIFIC REPORTS

OPEN

α-Rank: Multi-Agent Evaluation by Evolution

Shayegan Omidshafiei[1], Christos Papadimitriou[5], Georgios Piliouras[4], Karl Tuyls[1], Mark Rowland[2], Jean-Baptiste Lespiau[1], Wojciech M. Czarnecki[2], Marc Lanctot[3], Julien Perolat[2] & Remi Munos[1]

- Google DeepMind提出来的一种新型均衡概念
- 可以处理多人一般和博弈
- 唯一不存在选择性问题
- 可高效计算

均衡计算复杂度小结

- 均衡的存在性问题
- 不同类型的均衡
  - PNE、MNE、CE、CCE
- 动态最优反应算法求PNE
- 计算复杂性理论
  - P类问题、NP问题、约化、NPC问题、NP-Hard问题
- 计算阻塞博弈中的PNE是一个PLS完全问题
- 求解一般的两人博弈纳什均衡是PPAD完全问题
- 求解三人/四人博弈纳什均衡是PPAD完全问题
- 纳什均衡的选择性问题

感谢聆听！

李凯

kai.li@ia.ac.cn

2020年11月05日