第四讲：算法博弈论

兴军亮

Junliang.Xing@ia.ac.cn

2018年4月20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

- 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。
- 然而，在过去十五年里，由于互联网的飞速发展，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，比如如何找到或者设计包含多个存在相互交互自主个体的系统的均衡点等，于是出现了“算法博弈论”这个新的交叉研究方向。

“The Internet is an equilibrium, we just have to identify the game.”

--Scoott Shenker

什么是算法博弈论？

- 算法博弈论：理论计算机科学和博弈的交叉点，研究博弈论中的计算问题，即如何“求解”游戏。
- 重点研究的问题
  - 一个游戏的所有参与者如何迅速收敛到游戏的均衡点？如果参与者找不到，计算机能不能做到？
  - 实际达到的均衡点和理想的均衡点之间存在多大的差异？
  - 达到均衡点的算法复杂性，能否有多项式复杂度的算法来得到均衡点？

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

-- Kamal Jain

什么是算法博弈论？

- 算法博弈论相对于经济学中讲授的博弈的区别：
  - 应用领域不一样：算法博弈论主要应用于和互联网相关网络模型优化以及和拍卖相关的机制设计等；
  - 研究方法不一样：算法博弈论通常使用具体的优化问题来对实际应用问题进行建模，并且求取最优结果、不可能的结果、可行的近似算法的上界和下界等；
  - 计算复杂度要求不一样：算法博弈论通常要求博弈系统的设计者和参与者行为满足一定的计算复杂度要求（比如满足多项式时间复杂度）。

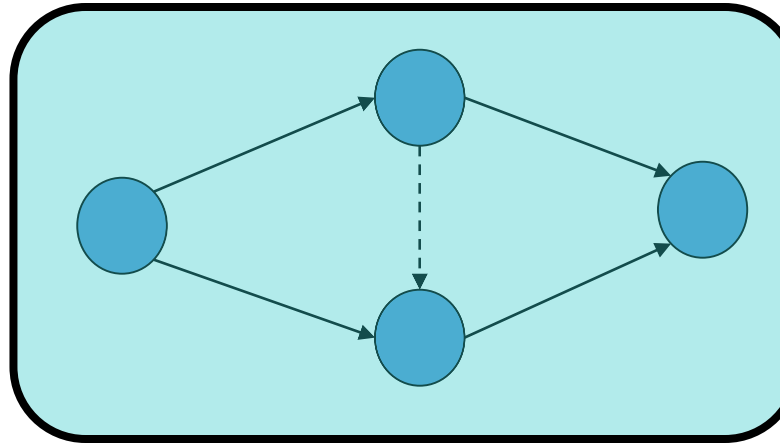
学习算法博弈论的原因

- 研究人工智能游戏：对“理性智能体”和他们之间的交互过程进行建模；
- 研究分析反应系统：计算机辅助验证、 博弈驱动的模型检查等
- 研究计算机算法：一些博弈论算法均具有非常有意思算法分析特点（比如是NP问题还是NPC问题）
- 计算机科学逻辑：计算机科学的逻辑体现出了博弈的特性，包括逻辑模型逻辑和时序逻辑，自我博弈验证；
- 游戏的计算复杂度问题：包括回合制游戏、布尔电路等
- 计算机网络和电子商务：如何为Internet这个巨大的博弈设定规则已达到全社会最优？

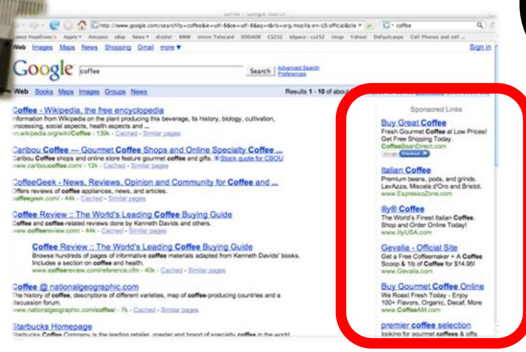
算法博弈论的应用领域

市场定价

网络路由



在线广告



种群进化

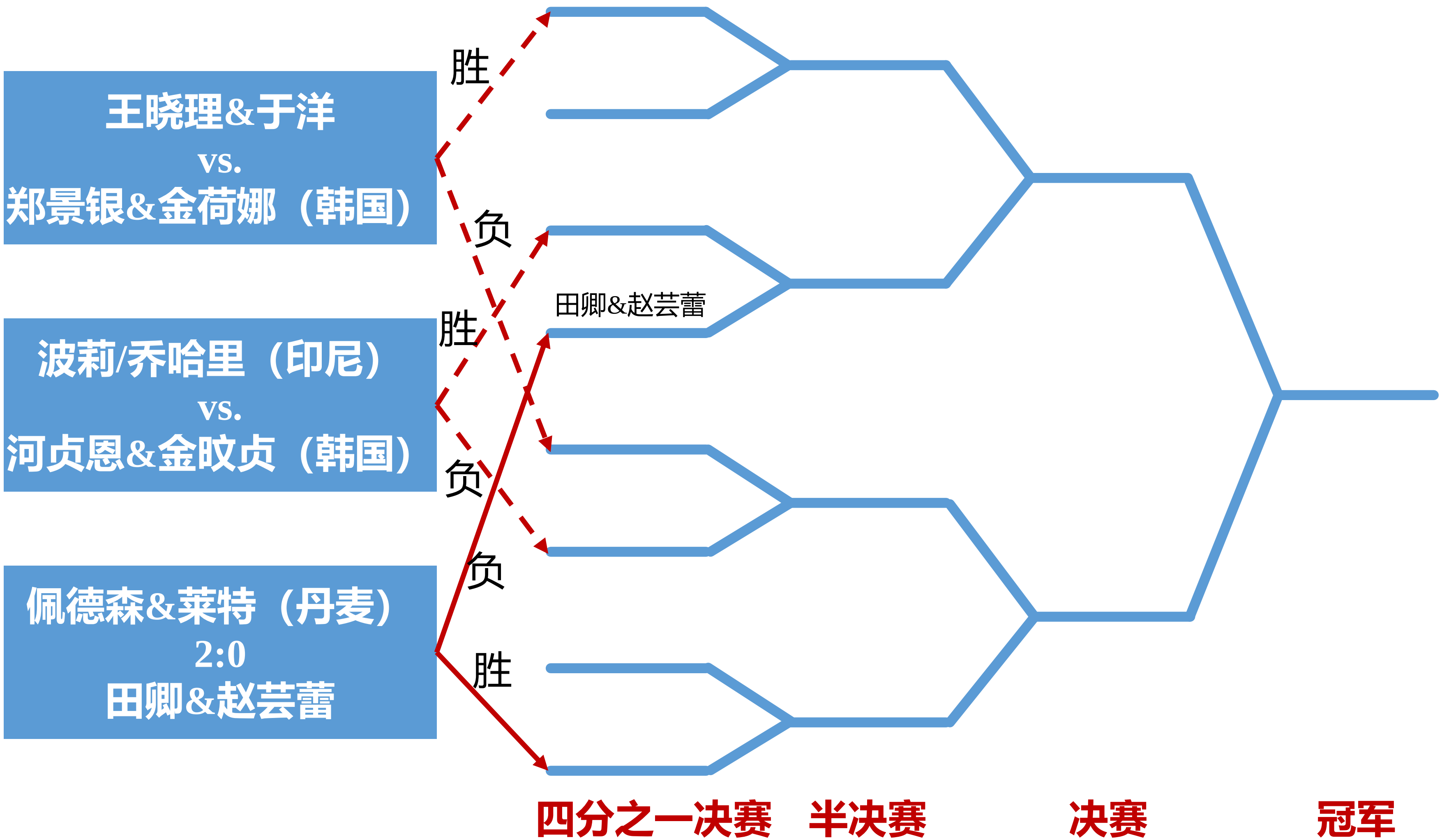
选举

主要研究内容

- 算法化机制设计（Algorithmic Mechanism Design）
  - 单参数机制设计（Single-Parameter Mechanism Design）
  - 多参数机制设计（Multi-Parameter Mechanism Design）
- 均衡的低效率性（Inefficiency of Equilibria）
  - 无序性代价（The Price of Anarchy）
  - 稳定度代价（The Price of Stability）
- 均衡计算复杂度（Complexity of Finding Equilibria）
  - 最佳反应动力（Best-Response Dynamics）
  - 没有遗憾动力（No-Regret Dynamics）等
- 其他研究内容
  - 图形化游戏（Graphical Games）、密码学（Cryptography），计算进化博弈（Computational Evolutionary Game Theory）等

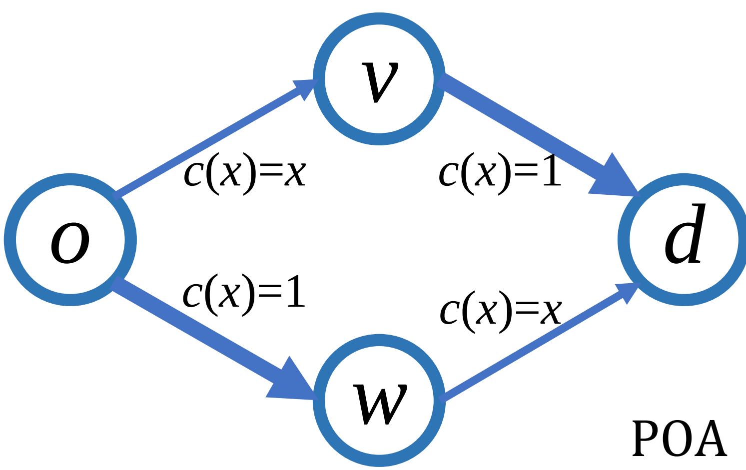
算法化机制设计示例分析

- 2012年奥运会羽毛球女双比赛

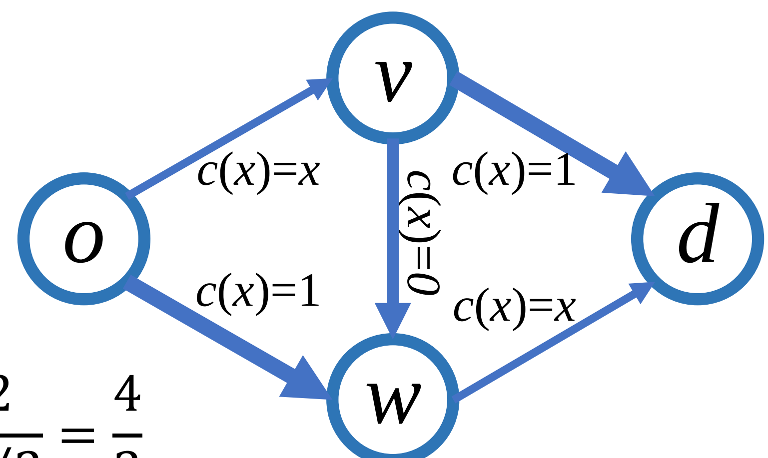


均衡的低效率性示例分析

- 布雷斯悖论（Braess’s paradox）
  - 如下图中所示的道路网
    - 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，一条又长又宽路，一条又窄又短路；
    - 长宽路需要1个小时通过，不管多少车辆通过；
    - 短窄路需要的时间和通过的车辆占总体的比例相同；
  - 试求通过左图的道路所需的平均时间是多少？
  - 如在左图道路网中增修一条超级便道之后呢？

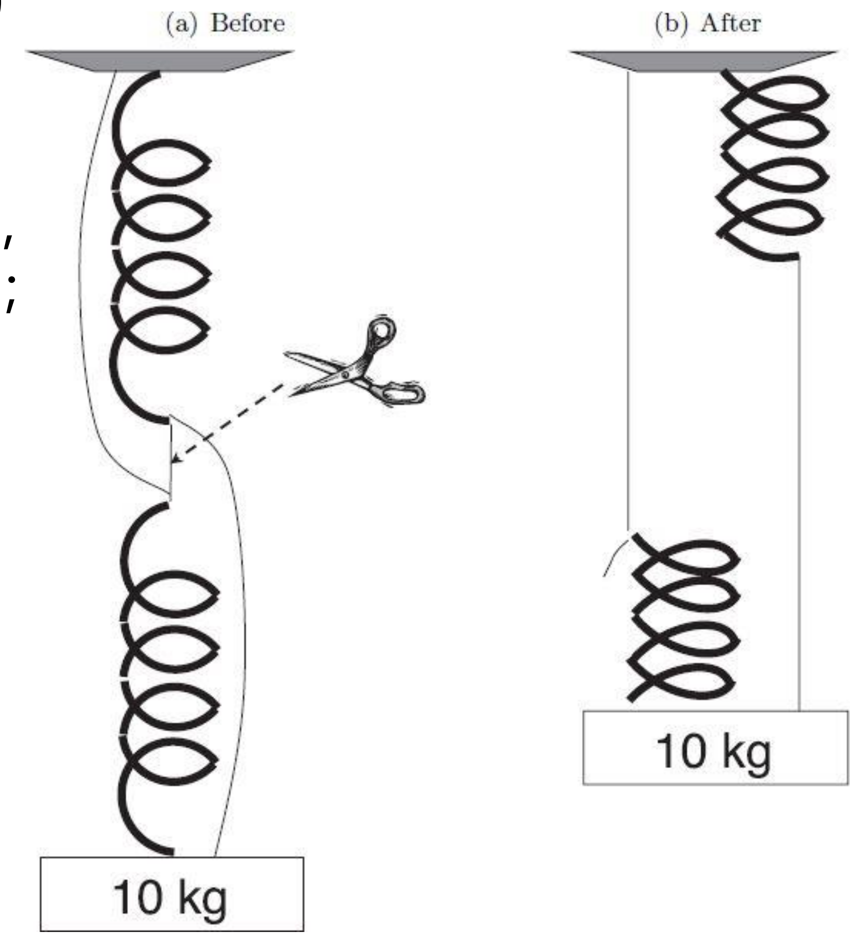


$$\text{POA} = \frac{2}{3/2} = \frac{4}{3}$$



均衡的低效率性示例分析

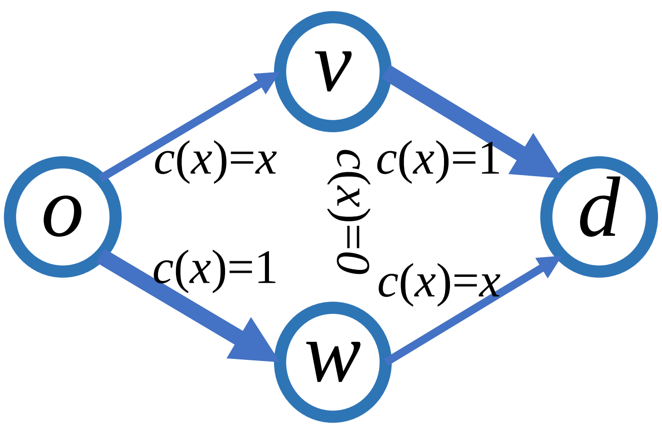
- 布雷斯悖论（Braess's paradox）
  - 绳子和弹簧（Strings and Springs）
    - 通过选择合适长度的弹簧和绳子，可以达到右图(a)的平衡状态；
    - 减掉右图(a)中两根弹簧之间的短绳，弹簧可以迅速切换到右图(b)的状态；
    - 两种连接状态下，弹簧和绳子的承重分别是多少呢？



思考：这个例子其实和上一页PPT中的道路网是等价的。

均衡计算复杂度示例分析

- 聪明的博弈参与者能否学到均衡？
  - 布雷斯悖论：计算机简单搜索既可以得到均衡
  - 石头剪刀布：零和游戏通过线性规划或迭代学习算法可以求解
  - 其他更一般的博弈呢?



均衡计算复杂度示例分析

- 即使聪明的博弈参与者能否学到均衡，计算机能否能否找到更为一般性博弈的均衡？
  - 两人非零和游戏：没有计算复杂度上有效的方法保证得到纳什均衡，虽然这个问题并不属于NP完全问题。
  - 因此，两人博弈问题本身就是一个非常好的呈现出“中间”计算难度的自然问题。
  - 对于存在多个纳什均衡的博弈问题，博弈分析的预测功能愈发显得乏力。
- 因此，需要研究在计算上可行的纳什均衡的算法解或者算法近似解。

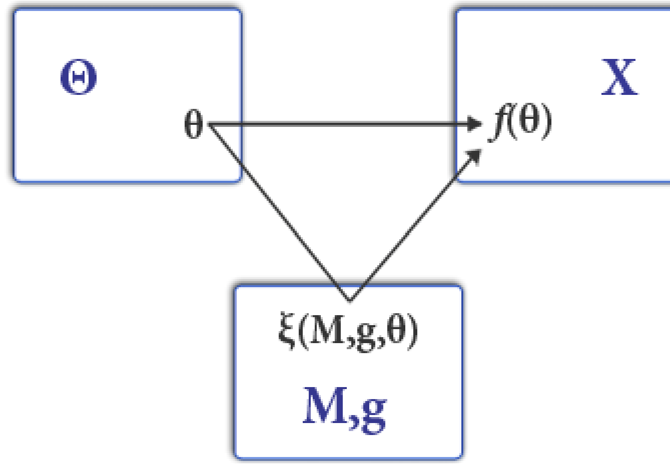
产生和发展历史

- 算法机制设计：最初由希伯来耶路撒冷大学的Noam Nisan和Amir Ronen在他们1999年发表的一篇论文提出。
- 这个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、离散数学和理论计算机中的复杂度分析和算法设计等概念和理论。
- 与经济学中的机制设计的差别
  - 使用理论计算机科学的分析工具，比如最差情况分析和近似度；
  - 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，经典的经济学机制设计模型如VCG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.

机制设计相关概念介绍

- 机制设计是在策略环境中，针对理性的参与者，使用工程性的方法来设计经济学机制或者奖励措施，从而达到特定的目标。
- 作用：通过设计的“自然”解决博弈中特定参与人的个体决策与整体目标之间的分歧。
- 机制设计相关要素和过程示意图
  - 类型空间：**Θ**
  - 结果空间：**X**
  - 社会选择方程：$f(\theta)$
  - 机制设计过程：参与者在博弈环境 g 中发布消息M，设计的目标是通过实现某个特定的社会选择方程$f(\theta)$来达到博弈的均衡$\xi(M, g, \theta)$。



机制设计相关概念介绍

- 机制设计相对于传统博弈论的不同之处：
  - 机制设计开始于传统博弈过程的结束，然后反过来回到博弈之中去，因此也称作“逆向博弈论”；
  - 重点是博弈分析，呈现是机制设计，目标是期望结果。
- 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 laid the foundations of mechanism design theory”.
  - 获奖原因：为机制设计理论建立了基础。

机制设计相关概念介绍

- 机制：将个体的信息类型（θ）映射为相关结果的一个过程函数，用$y(\theta)$表示
- 机制设计博弈过程：
  - 特定参与者向机制$y(\cdot)$提交自己的结果$y(\theta)$
  - 参与者分别报告自己的信息类型$\hat{\theta}$（有可能撒谎）
  - 机制进行执行（其他参与者收到结果$y(\hat{\theta})$）
- 为了理解谁得到了什么，通常将结果y分成物品分配和奖励转让：$y(\theta)=\{x(\theta),t(\theta)\},x\in\mathbf{X},t\in\mathbf{T}$。
- 设计者确定完全信息下的事件发生标准：社会选择函数$f(\theta):\Theta\rightarrow\mathbf{X}$
- 机制将报告的类型映射到结果：$y(\hat{\theta}):\Theta\rightarrow\mathbf{X}$

机制设计应用：拍卖理论

- 拍卖理论：研究拍卖市场属性以及人们在其中行为的经济学分支。通常关注拍卖设计的效率、最优和均衡拍卖策略、拍卖收益比较等。
- 拍卖的种类
  - 英式拍卖（English Auction）
  - 荷兰式拍卖（Dutch Auction）
  - 密封第一价格拍卖（FPSB）
  - 密封第二价格拍卖（Vickrey Auction）

	密封出价	迭代出价
第一价格	FPSB	Dutch
第二价格	Vickrey	English

- 其他类型的拍卖：单物品拍卖、多物品拍卖、全支付拍卖、有限时长拍卖（烧蜡烛拍卖）、投标缴费拍卖、收购拍卖、组合式拍卖、扩展第一/第二价格拍卖、日式拍卖、唯一价格拍卖、大宗商品拍卖（频谱拍卖）等等。

机制设计应用：拍卖理论

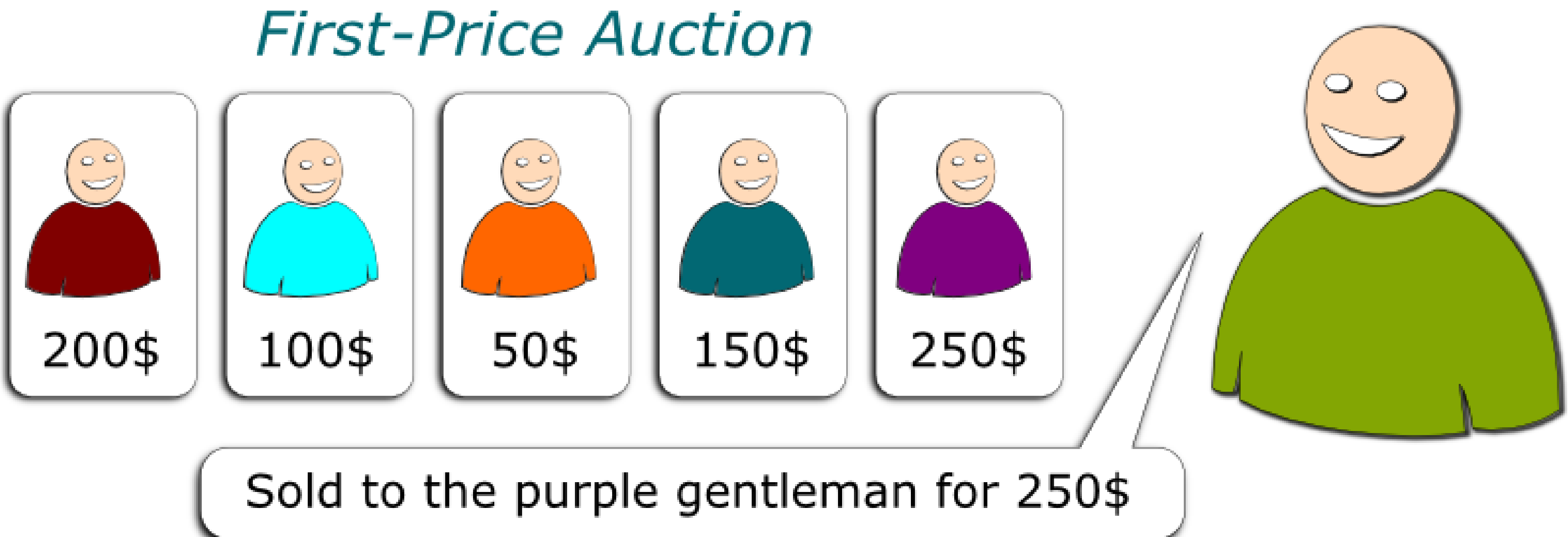
- 单物品拍卖（Single-Item Auctions）
  - 给定一件物品和n个潜在购买者，每个潜在购买者对于该物品都有一个自己的价值评估v_i，$v_i \geq 0$，该价值评估对于其他潜在购买者和拍卖者都是不可知的。
  - 拍卖效益函数：如果购买者i成功以价格p拍得该物品，则他的效益为$v_i - p$；如果未能拍得该物品，则效益为0。

$$f(x) = \begin{cases} v_i - p, & \text{拍到} \\ 0, & \text{未拍到} \end{cases}$$

- 密封出价拍卖
  - 每一个出价者将出价值b_i悄悄地告诉卖家
  - 卖家决定谁成功拍得该物品（如果存在的话）
  - 卖家决定拍得该物品的价格（怎么决定？）

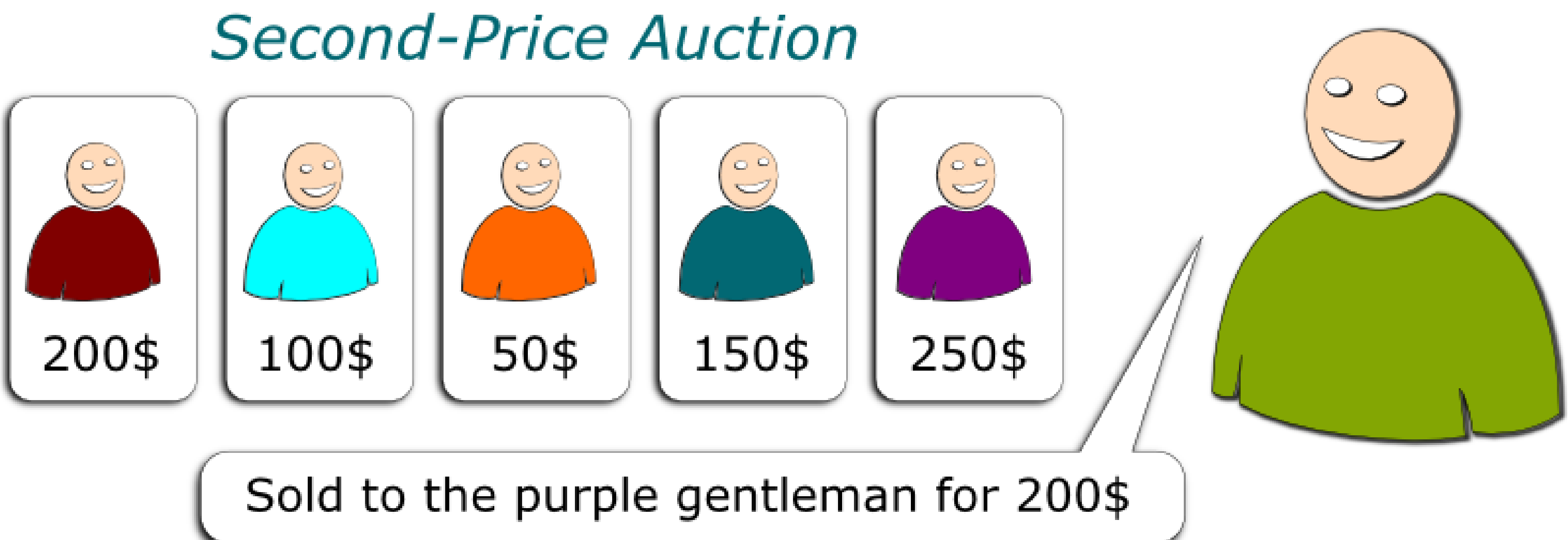
机制设计应用：拍卖理论

- 第一价格拍卖（First-Price Auctions）：赢得拍卖的购买者以自己的出价购买拍到的物品。
  - 是实际中非常常见的一种拍卖定价方式
  - 拍卖者貌似卖出了最高的价格，但是他能预测会发生什么吗?
  - 竞拍者虽然拍出了价格要购买，但他是否心甘情愿出最高价?



机制设计应用：拍卖理论

- 第二价格拍卖（Second-Price Auctions）：赢得拍卖的购买者以次高价购买拍到的物品。
  - 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，是非常重要的一类拍卖类型。



机制设计应用：拍卖理论

- 命题：关于第二价格拍卖中的奖励措施

第二价格拍卖的奖励措施

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，以及其他竞拍者的出价b_{-i}，我们需要证明在$b_i = v_i$竞拍者i的收益最大化，令$B = \max_{j \neq i} b_j$，则第二价格竞拍的收益函数为$\max(0, v_i - B)$。

当$b_i < B$，那么竞拍者i竞拍失败，收益为$\max(0, v_i - B)$ =0；

当$b_i \geq B$，那么竞拍者i竞拍成功，收益为$\max(0, v_i - B) = b_i - B$；

对于任意一种情况，竞拍者i的最优策略都是真实地报告自己的评估值。

机制设计应用：拍卖理论

- 命题：关于第二价格拍卖中的非负效益

第二价格拍卖的任一竞拍者收益为非负值

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竞拍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$v_i - p$，由于赢得竞拍并且他真实报告自自己对于物品的私有评估值，那么$v_i - p \geq 0$；

因此，对于任意一种情况，竞拍者i的收益值均为非负值。

机制设计应用：拍卖理论

- 拍卖的DSIC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$x_i=0$。

社会收益最大化特性：welfare maximizing

在一个拍卖中，如果对于每一个竞拍者真实报价后，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。

机制设计应用：拍卖理论

- 完美拍卖的定义

定义：完美拍卖（Ideal Auction）

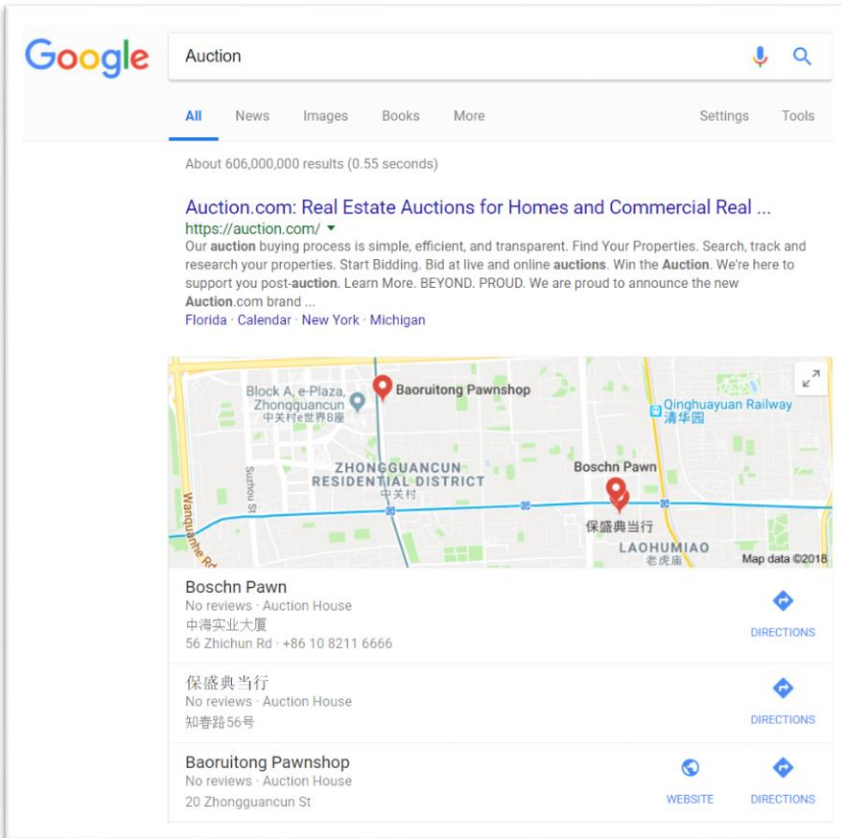
满足下面三个条件的拍卖称为完美拍卖：

1. 强激励保证：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：它可以在输入大小（表示$v_1, \ldots, v_n$所需要的数量）的多项式（最好是线性）时间复杂度内被计算实现；

- 定理：第二价格拍卖是完美拍卖。
  - 证明：根据之前的命题结论即可证明条件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。

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：搜索引擎竞价排名

- 搜索引擎竞价排名基本模型：多物品拍卖模型
  - 每个特定关键词的搜索页面包含k个位置可用于出售
  - 每个位置的出售价格不一样，使用点击率（CTR）评估其价格
  - 有多个广告商来竞标这些位置
  - 问题：如何设计拍卖规则?
- 设计要求
  - 要求1：满足DSIC特性
  - 要求2：满足社会收益最大化特性
  - 要求3：满足计算复杂度尽可能高效性

案例研究：搜索引擎竞价拍卖

- 搜索引擎竞价拍卖机制设计
  - 选择谁赢得拍卖物品
  - 选择由谁来支付什么
- 两阶段设计方法
  - 步骤一：假设所有竞拍者都真实报价，我们如何安排竞拍者到不同的位置从而使得要求2和要求3满足?
  - 步骤二：假设已经有了步骤一的答案，我们如何设定出售价格从而使得要求要求1得到满足?
- 答案：
  - 步骤一：贪心算法既可以满足要求，即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。
  - 步骤二：每一个广告商按照下一名的广告商的出价进行支付?

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价值v_i，表示它对于某个物品的估价，另外，有一个可行集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。

$$\mathcal{E} = \{\{v_i\}_{i=1}^{n}, (x_1, \ldots, x_n)，\ x_i \in X\}$$

- 实例对应
  - 单物品拍卖：$X = \{0,1\}$，$\sum_i^n x_i \leq 1$
  - K-物品拍卖：$X = \{0,1\}$，$\sum_i^n x_i \leq k$
  - 搜索引擎竞价拍卖：X是n个位置的指派向量，每个位置最多安排1个广告商，每个广告商最多安排一个位置；如果第i个广告商安排了第j个位置，x_i就等于位置j的点击率α_j
- 拍卖：一个用于交换物品和钱的机制。

麦尔森定理（Myerson's Lemma）

- 分配和支付规则：对应密封拍卖中的“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 $\mathbf{b} = (b_1, \dots, b_n)$ ，组成竞价向量；
  - 分配规则：选择一个合适的分配$x(b) \in X \subseteq \mathbb{R}^n$作为竞价的函数；
  - 支付规则：选择一个支付$p(b) \in \mathbb{R}^n$作为竞价的函数；
- 上述这个过程也成为直接显式机制（direct-revelation mechanisms）
- 收益函数：$u_i(b) = v_i \cdot x_i(b) - p_i(b)$
- 对于支付规则的限制：$p_i(b) \in [0, b_i \cdot x_i(b)]$
  - 大于0要求拍卖者不能赔钱
  - 小于$b_i \cdot x_i(b)$要求真实报价的竞拍者收益为正

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，当一个分配规则**x**有一个对应的支付规则p使得直接显式机制(**x**,**p**)具有DSIC特性，那么就称这个分配规则**x**是实现的。

DSIC机制要求分配规则必须可实现。

定义：分配规则的单调性（Monotone Allocation Rule）

在单参数环境下，对于任意一个个体 i 和其他个体的出价$\mathbf{b}_{-i}$，如果分配$x_i(z, \mathbf{b}_{-i})$相对于个体i的出价z是非递减的，那么就称这个分配规则**x**具有单调性。

出价越高，得到越多。

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直接显式机制(**x**, **p**)具有DSIC特性，并且只要b_i=0时即满足p_i(**b**)；

c. 在b中的支付规则可以通过具体的计算公式给出。

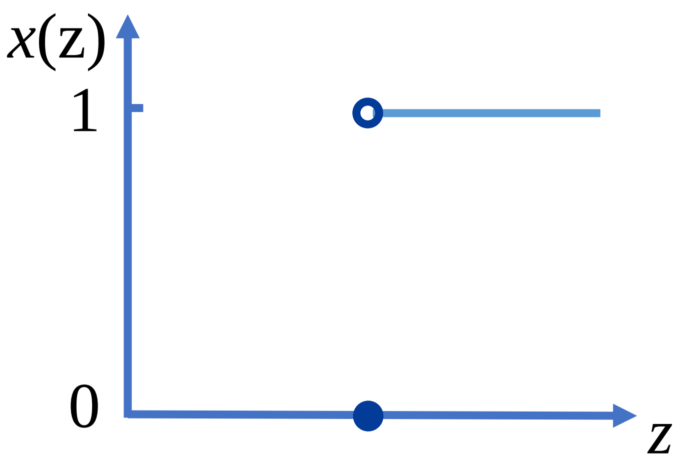
- 定理分析
  - 内容a表明可实现性和单调性对应了同一类分配规则；
  - 内容b说明当分配规则可实现，就可以确保具有DSIC特性；
  - 内容c说明有直接的计算公式来实现分配规则，因而很高效。

麦尔森定理（Myerson's Lemma）

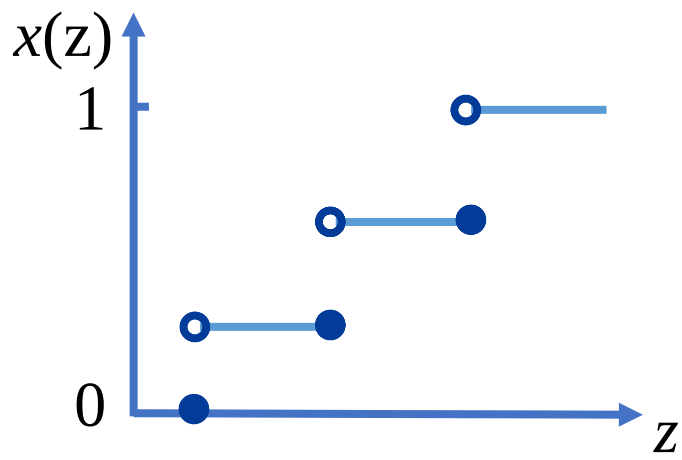
- 麦尔森定理的证明

证明：使用DSIC特性的要求限制支付规则到一个具体的候选。详细见参考书籍《Twenty Lectures on Algorithmic Game Theory》的第3.4章。需要记住证明中得到的一个公式：麦尔森支付公式

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



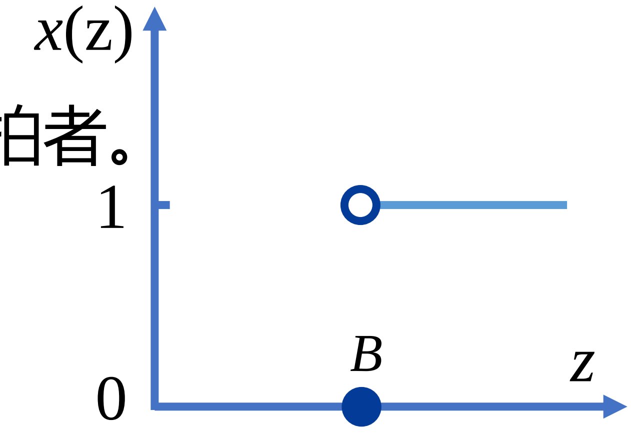
0-1单调曲线



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。
  - 如何设计支付规则?



解答：使用麦尔森支付公式

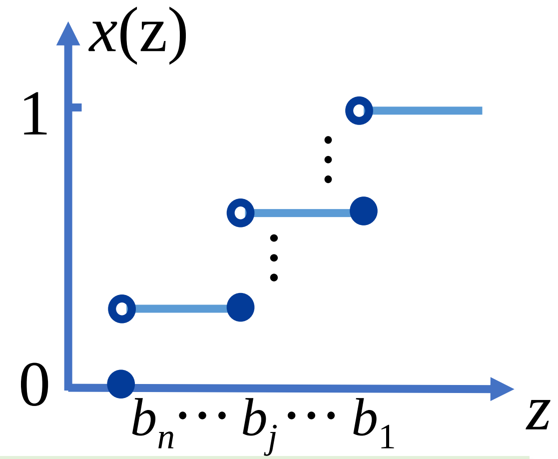
$$p_i(b_i, \mathbf{b}_{-i}) = \sum_{j=1}^{l} z_j \cdot [jump\ \text{in}\ x_i(\cdot, \mathbf{b}_{-i})\ at\ z_j]$$

对于单物品最高出价获得拍卖品的原则，只有一个跳跃点：在出价B处跳跃为1的点。因此，当参与者i成功竞拍 $p_i(b_i, \mathbf{b}_{-i}) = B$，否则$p_i(b_i, \mathbf{b}_{-i})$ =0。单物品分配规则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$x$是将第$i$出价的广告商分配到 第$i$个位置。
  - 如何设计支付规则?



解答：给定一个支付向量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称为第j高的出价时，就会有一个在$a_j - a_{j+1}$的跳跃，根据麦尔森支付公式，可以得到：

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

考虑到广告商只关心点击率，支付规则可以使用点击率进行归一化：

$$p_i(\mathbf{b}) = \sum_{j=1}^{k} b_{j+1} \cdot \frac{(\alpha_j - \alpha_{j+1})}{a_i}$$

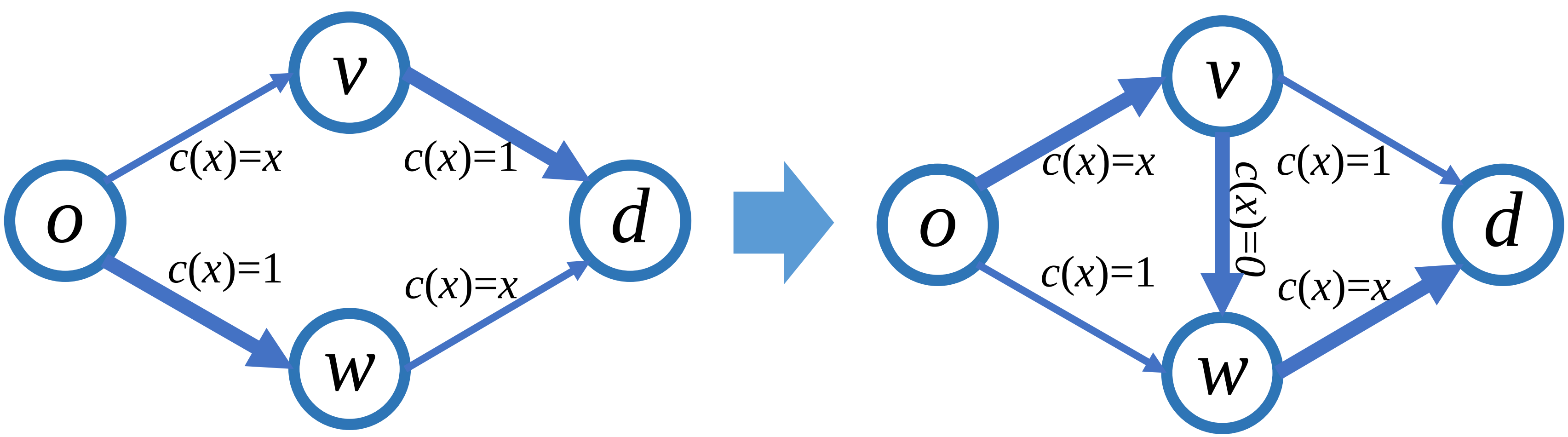
这样，当拍得的广告被点击，竞拍者支付一个由低出价组成的凸包。

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对道路网络产生的影响却可能是最差的（未达到“系统优先”）。
- **约翰·纳什**（John Nash）：“在一个国家中，没有任何一个实验性游戏的参与者可以仅凭自己的努力来改善状况。具有讽刺意味的是，如果大家所做的事情对自己最有利，他们的做法就不会对每个人都有益。”
- 每个人都是“自私的出行人”，纳什均衡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。

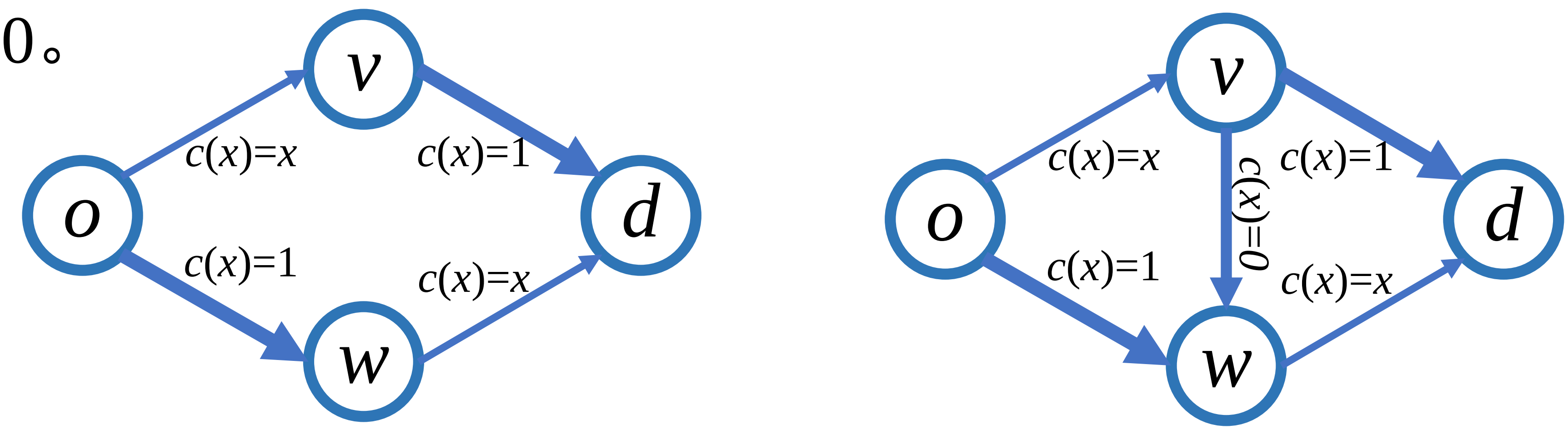
布雷斯悖论详细讲解

- **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交通延滞，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化”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（$x \in [0,1]$）。$c(\cdot)$为路径的代价函数，即通过时间。
- 在原始网络（左图）中，o-v-d 和 o-w-d路径代价函数完全相同，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，此时最终期望代价为 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。
- 考虑一个博弈系统 $G = (N, S, u)$，N为参与人员，S为所有策略组合，u为功效函数。某一个策略s对整个系统的收益函数为：$\mathrm{Welf}(s) = \sum_{i \in N} u_i(s)$. 在博弈过程中，我们期望收益函数越大越好。同理可以定义代价函数$\mathrm{Cost}(s)$，代价函数越小越好。

无秩序代价（Price of Anarchy，PoA）

- 从收益函数角度定义无秩序代价：

$$\mathrm{PoA}=\frac{\max_{s\in S}\mathrm{Welf}(s)}{\min_{s\in E}\mathrm{Welf}(s)}$$

其中E为纳什均衡集合中的策略组合，S为所有的策略组合。

根据定义可以得到：$\mathrm{PoA}\geq 1$

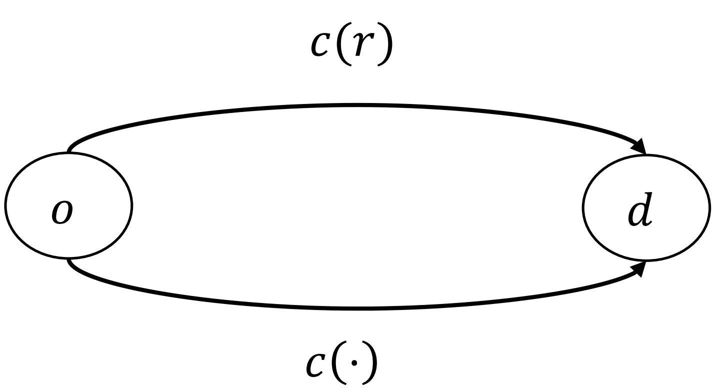
- 从代价函数角度定义无秩序代价：

$$\mathrm{PoA}=\frac{\max_{s\in E}\mathrm{Cost}(s)}{\min_{s\in S}\mathrm{Cost}(s)}$$

- 以布雷斯悖论中的扩展网络为例，$\mathrm{PoA}\geq 4/3.$

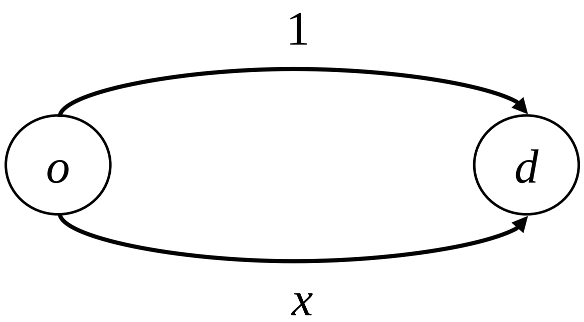
布雷斯悖论分析：Pigou网络

- 布雷斯悖论中的网络可简化为更一般的情况：
  - 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，“上边一条”和“下边一条”；
  - 网络中有非负的交通流量 r；
  - 在“上边一条”中，代价函数为 常数$c(r)$，与当前路径上的交通比例无关；
  - 在“下边一条”中，代价函数为 $c(\cdot)$，函数变量为当前路径上的交通流量（小于等于 r）。其中$c(\cdot)$非负、连续、非减；
  - “下边一条”是每个参与者的纳什均衡策略；



布雷斯悖论分析：Pigou网络

- Pigou网络举例：线性代价函数



- 对每一个单位来说，“下边一条”为纳什均衡策略。因此在纳什均衡策略集合中，系统的整体代价期望为1；
- 对整个系统来说，代价期望为$x^2 - x + 1$，当$x = {}^1/_2$, 代价期望达到最小值，为 ${}^3/_4$；
- 因此在该Pigou网络中，PoA为 ${}^4/_3$；

布雷斯悖论分析：Pigou网络

- Pigou 网络举例：非线性代价函数

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

- 随着代价函数阶 p 的增大，PoA增大。$p\to+\infty$ 时，PoA $\to+\infty$.

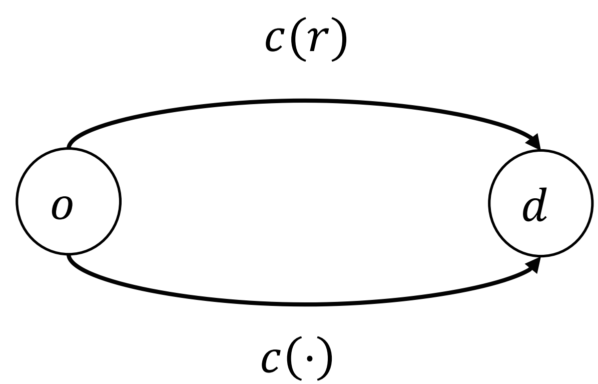
布雷斯悖论分析：Pigou网络

- 考虑通用的 Pigou 网络，其系统的最小代价为 $\inf_{0\le x\le r}\{x\cdot c(x)+(r-x)\cdot c(r)\}$, 则对应的 PoA为：

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

- Pigou界：对于任意一个包含非负、连续、非减的函数集合C，定义Pigou 界 $\alpha(C)$为 Pigou 网络 PoA的最大值，其中这些 Pigou 网络“下边一条”的代价函数来源于C。形式上可以表示为

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



布雷斯悖论分析：Pigou网络

- 自私路由网络中的Pigou界定理

定理1：最大PoA值可达性

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函数集合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Pigou网络中达到，其中对应Pigou网络的“下一条边”的代价函数属于函数集合C。

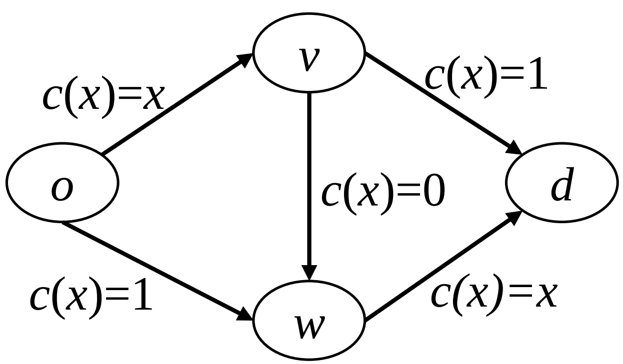
定理2：PoA值的Pigou界定理

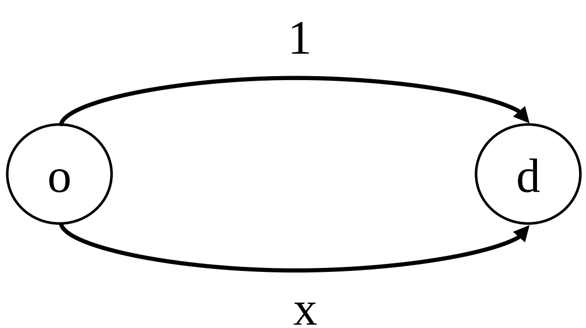
对于任意的自私路由网络，其每一条边上的代价函数都属于集合C，则该网络的PoA $\leq \alpha(C)$

- 关于定理的一些解释
  - 自私路由网络的PoA值和网络结构复杂度无关，而和代价函数有关。
  - 计算复杂网络的PoA值问题可以转化为计算相对应的Pigou网络的PoA 值

布雷斯悖论分析：Pigou网络

- 应用举例：求解布雷斯悖论中扩展网络的PoA值
  - 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。在均衡策略o-v-w-d中，最终期望代价为 2.0。根据PoA的定义，PoA $\geq 4/3$
  - 将扩展网络转化为右边的 Pigou 网络，该Pigou 网络的PoA值为4/3。根据Pigou界定理，扩展网络的$PoA \leq 4/3$。
  - 综上所述，布雷斯悖论中扩展网络的PoA值等于4/3





对自私路由网络的改进策略

- 改进策略：网络预留
  - 自私路由网络适用于实际应用中许多不同类型的网络，包括交通、通信和电子网络等。
  - 在通信网络中，一个很大的优势是，向网络添加额外的容量通常相对便宜和方便。正因为如此，一个通用的通信网络管理策略是安装比需要更多的容量，这意味着网络的容量通常不会被充分利用。
  - 这种网络预留的动机一方面是为了预测未来需求的增长。另外一方面，过度供应也和网络性能有关，因为根据经验，在有额外容量的情况下，网络往往会遭受更少的数据包丢失和延迟。
  - 类似的还有排队等待限流。在上下班高峰期的地铁站中，通常用排队等待来限制地铁站中的人流。

对自私路由网络的改进策略

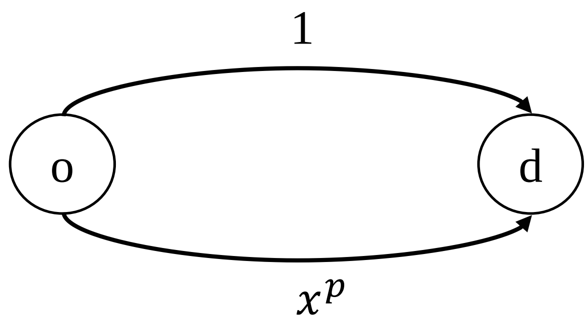
- 改进策略：网络预留
  - 引入预留参数$\beta \in (0，1)$，则β预留网络定义为: 当网络达到均衡时，每一条边e上的流量都满足$f_e \le (1-\beta)u_e$，也就是说在均衡时刻，每条边上的最大利用率不超过$(1-\beta)$。
  - 利用Pigou界定理可得β预留网络中最坏情况下的 PoA值：

$$\frac{1}{2}\left(1+\sqrt{\frac{1}{\beta}}\right)$$

  - 当$\beta \to 0$时，$\mathrm{PoA} \to +\infty$,队列发生完全阻塞情况。
  - 当$\beta \to 1$时，$\mathrm{PoA} \to 1$，不存在自私路由带来的低效率，但流量利用率为0%，此时网络没有通信功能。
  - 因此实际使用中采用一个折中的β， 比如β=0.1，此时网络流量利用率为 90%，PoA值最多为 2.1

对自私路由网络的改进策略

- 改进策略：技术升级
  - 考虑 Pigou 网络



- ➢ 当流量为1时，该网络均衡策略的代价为 $1^p = 1$
- ➢ 当流量为2时，该网络最优策略的代价为 $(1+\varepsilon)+(1-\varepsilon)^{p+1}$
- ➢ 流量为1时网络均衡策略的代价小于流量为2时的网络最优策略代价

这一现象可以扩展为更一般的定理，即资源扩大边界（Resource Augmentation Bound）定理：对于任意的自私路由网络和流量 r，其均衡策略的代价小于等于当流量为2r时网络最优策略的代价。

对自私路由网络的改进策略

- 对资源扩大边界定理的解释
  - 同一代价函数不同流量的网络进行比较，等价于对不同代价函数相同流量的网络进行比较。前者更容易解释和推理。
  - 直观地，对网络的通信技术进行升级，信息通过速度变得更快，相当于网络流量变小，假定为原来的1/2.
  - 根据资源扩大边界定理，流量变为为原来的1/2时，网络均衡策略代价小于等于技术改进之前网络的最优策略代价。
  - 因此，技术升级可以改进自私路由网络。

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

对于一个存在纳什均衡的博弈，即使所有参与人都足够“聪明”，他们都能找到均衡吗？

即使所有参与人都能够找到纳什均衡，他们用来找到纳什均衡的算法是什么呢？

即使他们都有一个能够找到纳什均衡的算法，算法会在多长时间找到均衡呢？

纯纳什均衡求解算法

- 纯纳什均衡定义回顾
  - 假定G是最小化代价的博弈，$s=\{s_1,\dots,s_k\}$是每一个参与者的策略组合。S_i代表参与者 i可选的策略集合。s_{-i} 代表除了 i之外其他人的策略组合。$C_i(.)$ 为参与者i的代价函数。如果对于每一个参与者 i 和任意一个单方面的偏差s_i', 都满足：$C_i(s)\le C_i(s_i',s_{-i})$. 则称 s 为$G$的纯纳什均衡。
- 势博弈（potential game，也称为潜在博弈）定义：
  - 存在有界实值势函数 Φ，使得对于任意单方面偏差s_i'引起的代价改变都等于势函数的变化：$\Phi(s_i',s_{-i})-\Phi(s)=C_i(s_i',s_{-i})-C_i(s)$
- 纯纳什均衡在很多博弈问题中不存在
- 势博弈中存在至少一个纯纳什均衡

纯纳什均衡求解算法

- 最优反应（Best Response）：在博弈中，假如其他人所采取的行动（策略选择）是已知或者能被预测的，根据这个已知的或可预测的行动而采取的能使自己的代价最小化（收益最大化）的策略，称为最优反应。
- 动态最优反应算法 （Best-Response Dynamics）

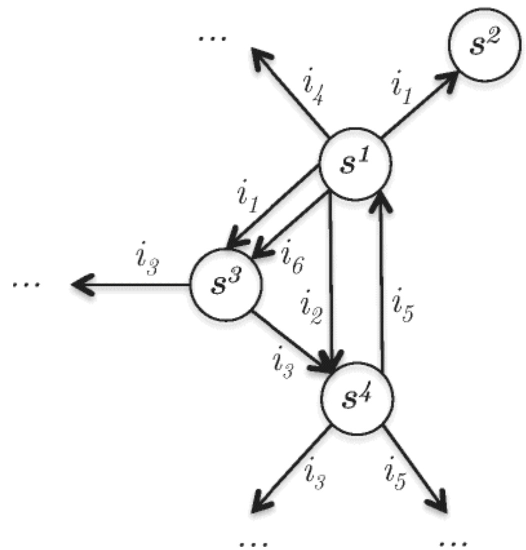
```
While当前策略 s 不是纯纳什均衡
选择任意一个参与者 i
针对 i 选择一个单方面偏差s_i'，使得
        C_i(s_i', s_{-i}) ≤ C_i (s)
更新s = {s_i', s_{-i}}
End While
```

  - 第二行中 i 可以任意选择，也可以改进为按照预定次序选择，比如 $i = 1, \ldots, k$
  - 为了提升算法效率，选择s_i'的时候也可以使之满足$s_i' = \mathrm{argmin}_{s_i'} C_i(s_i', s_{-i})$
  - 该算法可以用任意随机策略作为初始化

纯纳什均衡求解算法

- 动态最优反应算法可以看作是有向图上的行走过程，每个顶点代表一个策略组合，每条有向边的末端策略对应的代价函数要小于前端策略对应的代价函数。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纯纳什均衡策略。比如：

s^2为该博弈问题的纯纳什均衡策略



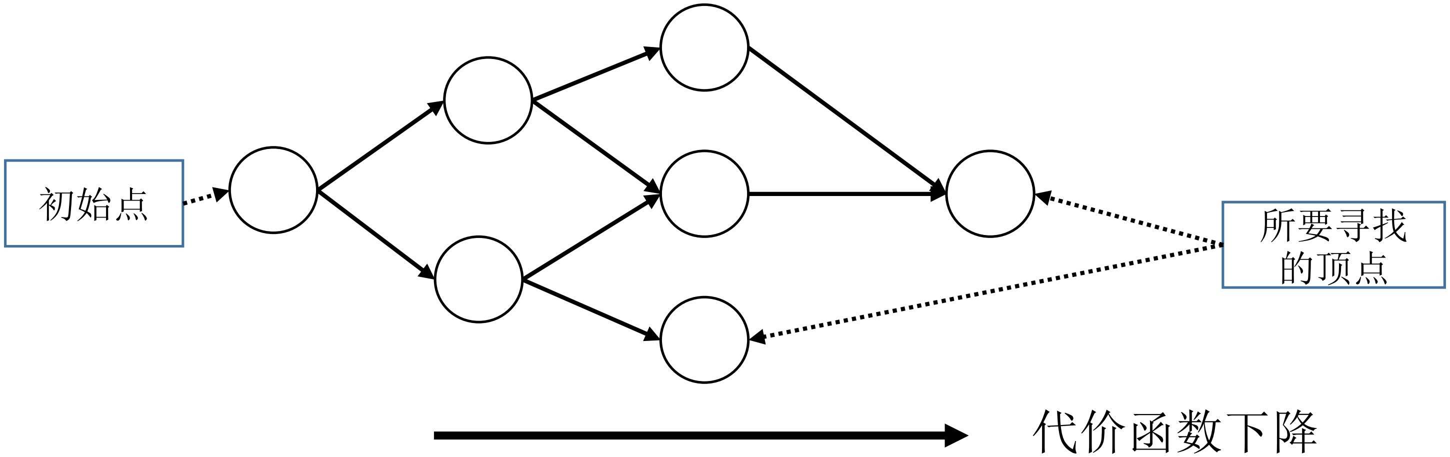
- 在势博弈问题中，从任意初始策略开始，动态最优反应算法都会收敛到一个纯纳什均衡上。

纯纳什均衡求解算法

- **局部搜索问题包含三个子算法：**
  - 第一个多项式算法：输入某个样例，输出可行解
  - 第二个多项式算法：输入该样例和可行解，输出目标函数值
  - 第三个多项式算法：输入该样例和可行解，输出是否局部最优或者给出更好的解
- **在势博弈中，动态最优反应算法是一个局部搜索问题：**
  - 第一个算法：任意选择一个策略组合
  - 第二个算法：计算该策略组合下的代价函数值
  - 第三个算法：判断是否为纯纳什均衡，并更新策略组合
  - 第三个算法需要遍历所有可能，通常遍历次数会根据输入空间呈指数增长

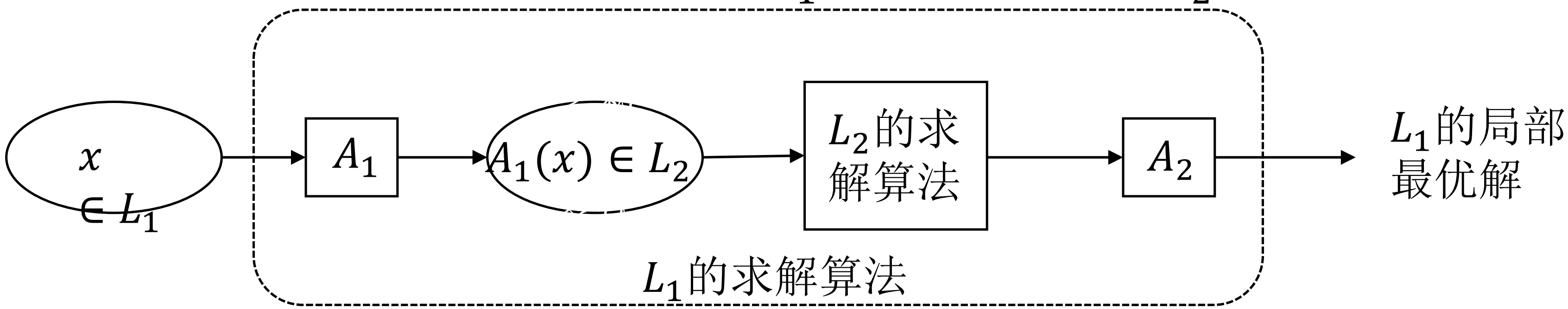
纯纳什均衡求解算法

- 关于局部搜索算法的一些说明
  - 局部搜索的第三个算法通常需要进行多次迭代才能确认。
  - 如果求解全局最优解，则局部搜索问题为NP-难问题
  - 如果求解局部最优解，则局部搜索问题很容易被解决
  - 局部搜索算法可以看作是在有向无圈图上的行走过程。每一个顶点代表可行解，终点代表一个局部最优解。局部搜索问题可以描述为从任意一个点出发寻找终点的问题。



纯纳什均衡的计算复杂度

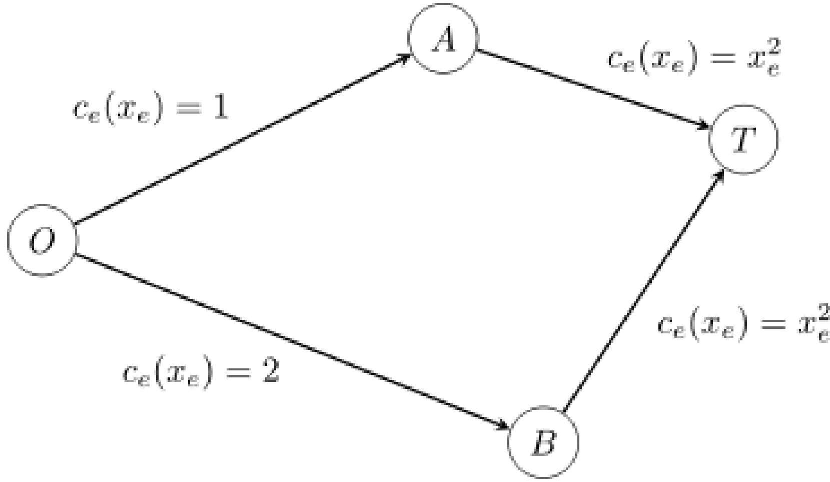
- PLS：局部搜索问题的集合
- PLS 归纳：
  - 定义：设 $L_1 \in PLS, L_2 \in PLS$,如果存在两个多项式复杂度算法 A_1, A_2 满足如下条件，则称为 L_1 问题归纳到 L_2 问题：
    - A_1把每L_1的一个实例 x映射到L_2的一个实例，即$A_1(x) \in L_2$
    - A_2把L_2的每一个局部最优解$A_1(x)$映射到L_1的局部最优解x
  - 解释：可以在多项式时间内将求解L_1问题转化为求解L_2问题



- PLS 完全问题：如果所有的PLS问题都可以归纳到 L 问题，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。

纯纳什均衡的计算复杂度

- 堵塞博弈（Congestion game）
  - 通信网络中的博弈
  - 包含所有堵塞元素的基本集合E，n个参与人，每个参与人i的策略输入策略集合$S_i \subseteq 2^E$。
  - E代表每一条边，每个人的策略为是否选择其中的每一条通道。每条边上都有代价函数，代价函数非负且单调递增。如右图：



- 在堵塞博弈中，计算纯纳什均衡是一个PLS完全问题
- 假设所有的参与人都有同一个出发点和目的点，计算堵塞博弈的纯纳什均衡依然是一个PLS完全问题。

混合纳什均衡及其相关概念

- 混合纳什均衡（MNE）：
  - $\sigma_1, \dots, \sigma_k$是策略集合$S_1, \dots, S_k$上的概率分布，$\sigma = \sigma_1 \times \cdots \times \sigma_k$。对于每一个参与人 $i \in \{1, \dots, k\}$和任何的单方面偏差s_i'，都满足
    $$E_{s\sim\sigma}[C_i(s)] \le E_{s\sim\sigma}[C_i(s_i', s_{-i})]$$
  - 在所有有限博弈中，混合纳什均衡一定存在，但是很难计算。
- 后悔值（Regret）
  - 设已经博弈了T个回合，对于参与人i，执行策略$s_i^1, \dots, s_i^T$的后悔值为：
    $$\frac{1}{T}\left[\sum_{t=1}^{T} c_i^t\left(s_i^t\right) - \min_{s\in S_i} \sum_{t=1}^{T} c_i^t(s)\right].$$
  - 定义中的代价函数(c)与MNE定义中的代价函数(C)不同，在本定义中，代价函数本身包含了对手的决策，因此是回合相关的。MNE中，代价函数只与博弈规则有关。
- 不后悔算法（No-Regret Algorithm）
  - 对于任意小的数$\varepsilon > 0$，存在一个回合范围T_0，当$T > T_0$时，根据算法A选择的决策的后悔值都小于ε，则称 A 为不后悔算法。换言之，在不后悔算法中，$T \to +\infty$时，后悔值趋于0

不后悔算法

- 不后悔算法设计原则
  - 过去的博弈经验应该用于指导未来的选择策略的概率分布
  - 如果某个策略带来了高代价，则需要降低该策略的选择概率
- 不后悔算法举例-权重更新算法

初始化每个策略的权重 $w^1(s) = 1, s \in S$
for $t = 1,2,\ldots,T$ do
利用权重计算每个策略的选择概率 $p^t(s) = {w^t(s)}/{\sum_{\tilde{s}\in S} w^t(\tilde{s})}$
根据代价重新计算每个策略的权重 $w^{t+1}(s) = w^t(s)\big(1-\beta c^t(s)\big)$

代价越大，
权重越低.

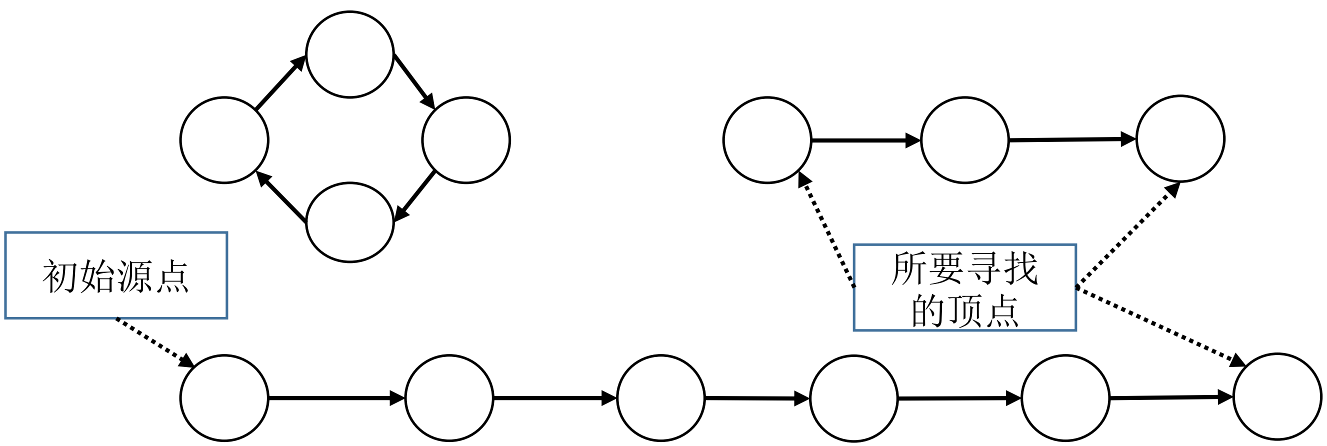
- 动态不后悔算法

初始化每个策略的选择概率
for t = 1,2,..., T do
每一个参与人 i 根据当前各自的概率分布选择决策，看作是纯决策
对每个参与人 i 的决策计算代价值$c_i^t(s_i) = E_{e_{-i}^t \sim \sigma_{-i}^t}\left[C_i\left(s_i, s_{-i}^t\right)\right]$，$\sigma_{-i}^t = \prod_{j\neq i} \sigma_j^t$
选择不后悔算法更新每个参与人 i 的策略选择分布函数

- 在二人零和博弈中，动态不后悔算法收敛于混合纳什均衡

混合纳什均衡算法复杂度

- PPAD（有向图的多项式校验参数）问题定义
  - 假设G（尺度很可能是指数增长）为一个不包含孤立结点的有向图，每一个顶点最多只有一个前驱顶点和后向顶点，即出度和入度最多为1。G 中存在一个多项式时间的函数$f(v)$可以从顶点 v 中得到它的前驱顶点和后向顶点。给定某一个源点 s（入度为0），从 G 中找出除了 s 之外的所有出度为 0 或者入度为 0 的顶点。 如下图所示：



混合纳什均衡算法复杂度

- 双矩阵博弈问题
  - 两个参与人分别有不同的 $n \times m$ 维的收益矩阵 A 和 B，其中 n 和 m分别为两个人可采取的策略数目。
  - 二人零和博弈问题（如猜拳游戏）是双矩阵博弈问题的一种特殊情况，即：

$$A = -B$$

  - 目前没有结论能证明，双矩阵博弈的混合纳什均衡问题可以用多项式复杂度的算法解决。
  - 在双矩阵博弈问题中，混合纳什均衡是一个 PPAD 完全问题。虽然PPAD 和 NP 很接近，但PPAD完全是 NP完全的不充分条件。

本次课程作业

- 作业内容：总结博弈论课程所有讲授内容的重要知识点。
  - 要求：使用一张A4打印纸（可正反面）总结自己归纳的博弈论前四讲的重要知识点，使用Word排版，可以记录文字、图形等，最小字号打印出来应清晰可见。同时在打印纸醒目位置注明自己的姓名、学号和专业。
- 提交时间：2018年5月20日17:00之前
- 提交方法：在课程网站上提交，同时提交电子版Word到助教邮箱（peixi.peng@ia.ac.cn）
- 邮件发送规范
  - 邮件主题：博弈论第四次作业_学号_姓名
  - 附件名称：博弈论第四次作业_学号_姓名.docx

感谢聆听！

兴军亮

Junliang.Xing@ia.ac.cn

2018年4月20日