第十三讲

算法博弈论初步

Elements of Algorithmic Game Theory

算法博弈论

均衡解的计算

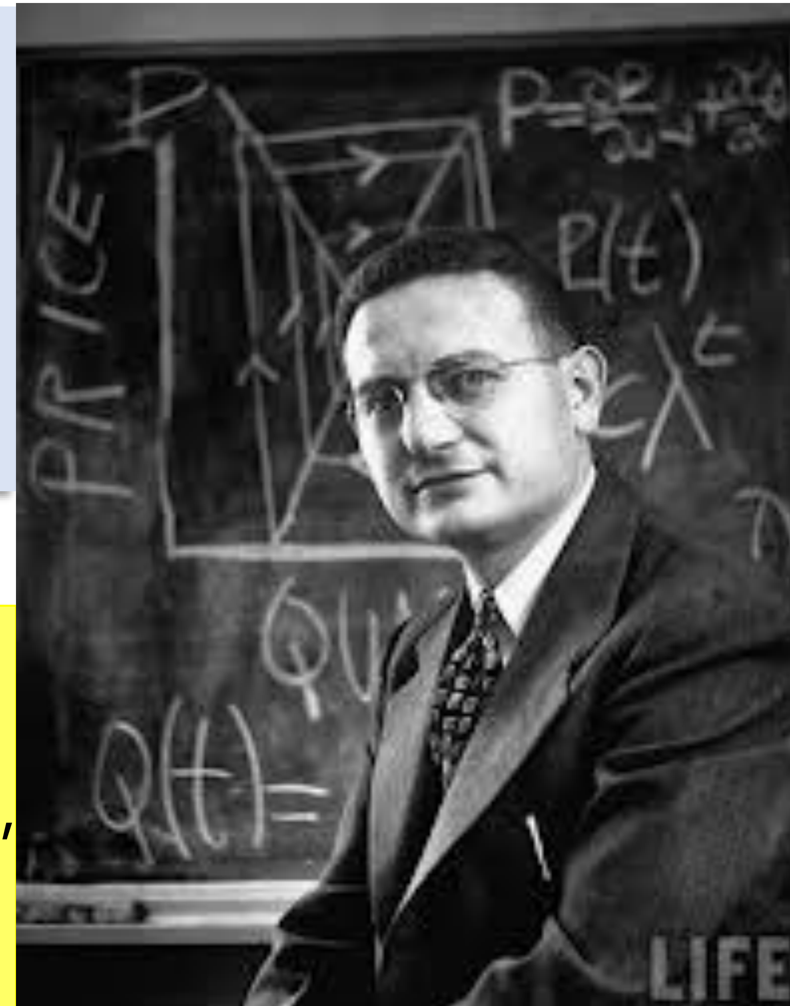
对均衡效率损失的定量分析

算法机制设计

引子

"To be literate in the modern age, you need to have a general understanding of game theory."

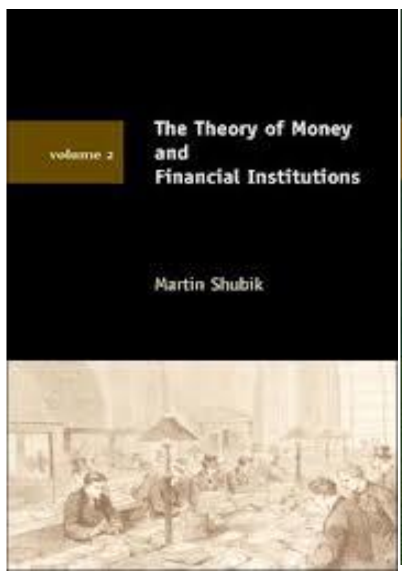
--Nobel Laureate Paul Samuelson (1991)

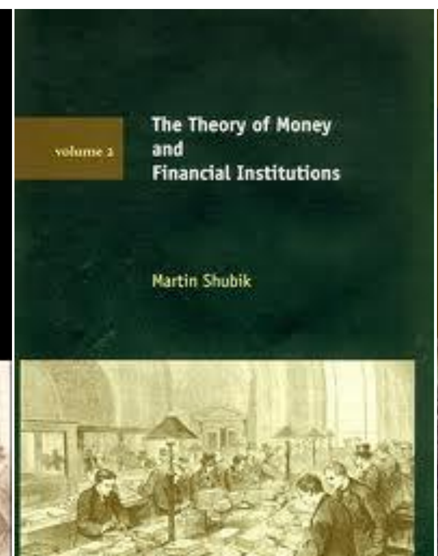


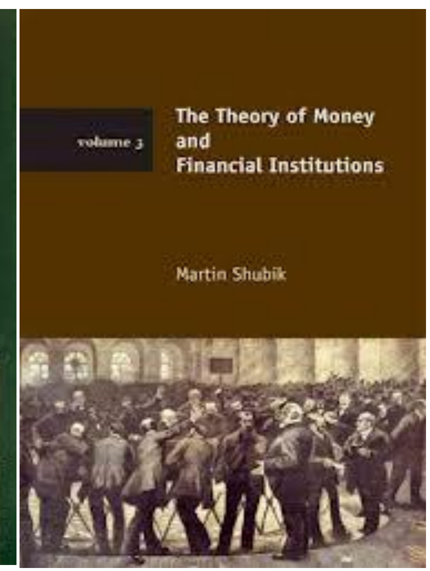
经济学家、1991年诺贝尔经济学奖得主保罗• 萨默尔森说："如果你想要在现代社会做一个有文化的人，那么你就要对博弈论有一个大致的了解。"

“理性” - 百元大钞竞拍

耶鲁大学经济学家舒比克(**M. Shubik**) 设计了如下游戏（dollar auction)：准备一张百元大钞，告诉他们你要将这张百元大钞拍卖给出价最高的那位朋友，大家互相竞价，以**5**元为增量单位，到没有人再加价为止。出价最高的人，只要付给你他所开的价码，即可获得这张百元大钞，但出价第二高的人，虽无法获得百元大钞，仍需将他所开的价码付给你。







4

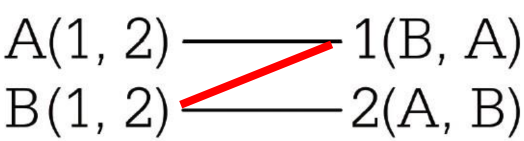
稳定匹配

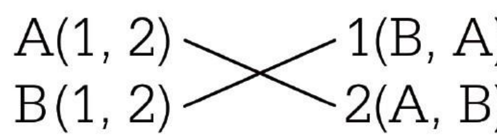
假如你是一位红娘。现有若干个单身男子登门求助，还有同样多的单身女子也前来征婚。如果你已经知道这些女孩儿在每个男孩儿心目中的排名，以及男孩儿们在每个女孩儿心中的排名，你应该怎样为他们牵线匹配呢？

最理想的匹配方案当然是，每个人的另一半正好都是自己的“第一选择”。但绝大多数情况下都不可能实现。当这种最为理想的匹配方案无法实现时，怎样的匹配方案才能令人满意呢？

稳定匹配

理想的匹配固然重要，但是和谐才是关键。给定一个匹配(M,F)，若存在俩男女(M_i,F_j)，他们各自的匹配对象均不如对方: $F_j \succ F_i$，$M_i \succ M_j$, 则称(M_i,F_j)为一“拆分对”，没有拆分对的匹配称为**稳定匹配**。



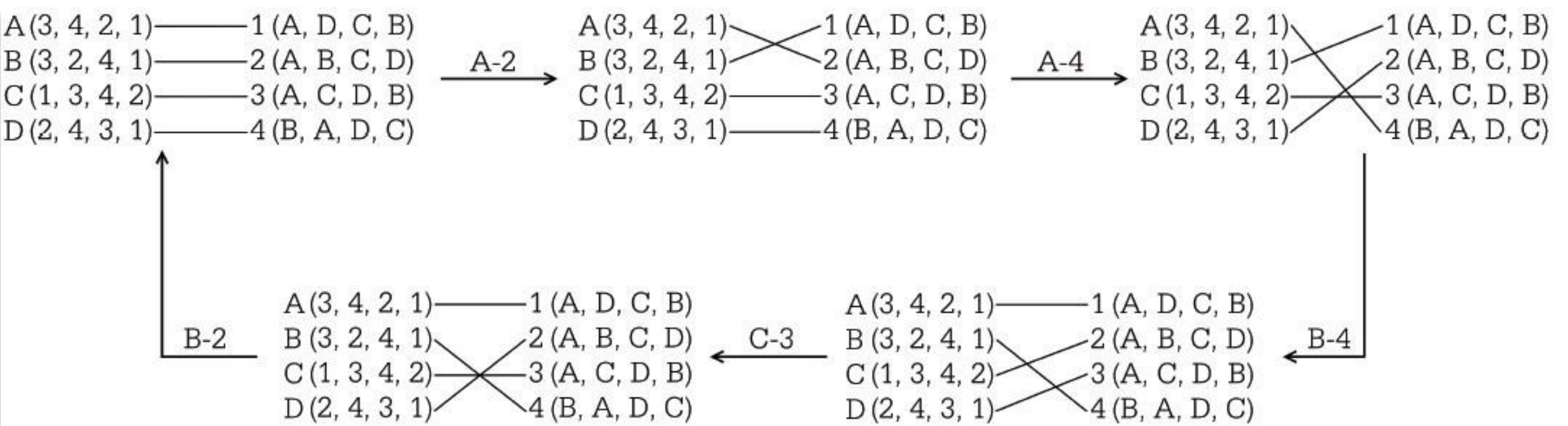


稳定的匹配总存在吗？如何找到一个稳定的匹配呢？

Local-Change

构造稳定婚姻匹配的一个非常自然的想法是“逐步改进”策略：

- 若两个人互相都觉得对方比自己当前的伴侣更好，就让这两个人成为一对，被甩的那两个人组成一对；
- 重复逐步改进，直到消除所有的拆分对。



➢ Gale-Shapley算法

1962年，美国数学家 **David Gale** 和 **Lloyd Shapley** 发明了一种寻找稳定匹配的策略。

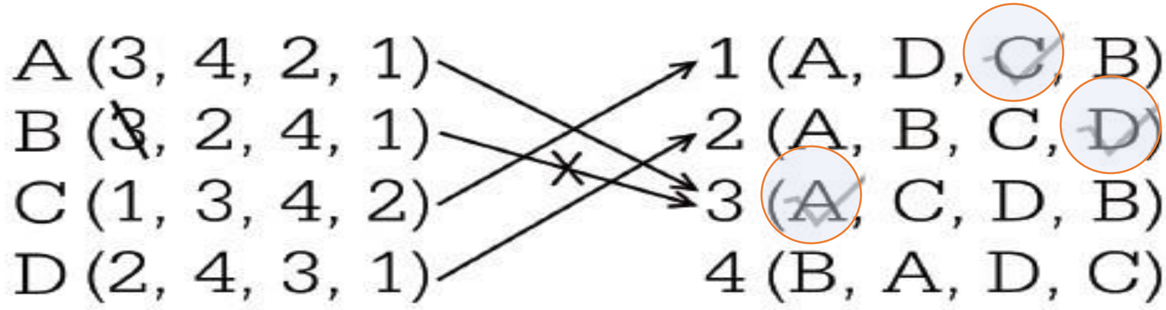
Gale-Shapley 策略/算法：男孩儿将一轮一轮地去追求他最中意的（且没有拒绝过他的）女孩儿，女孩儿可以选择接受或者拒绝：

- 有人向她表白：从所有追求者中选择自己最中意的那一位，答应和他暂时做情侣，并拒绝所有其他追求者
- 没有人向她表白：女孩儿只需要继续等待

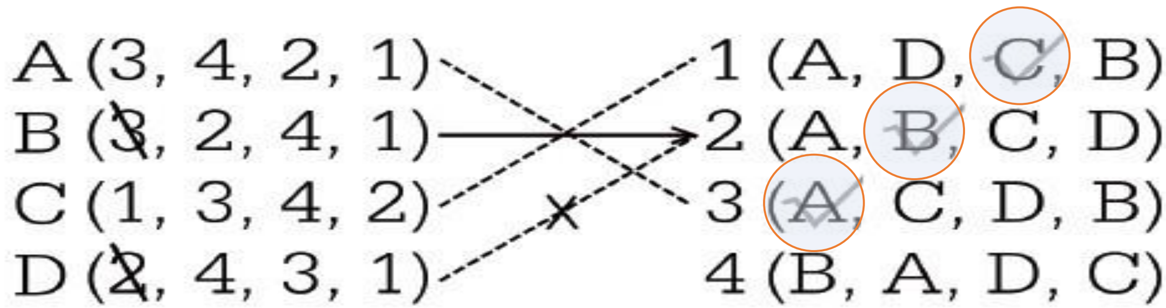
8

Gale-Shapley算法

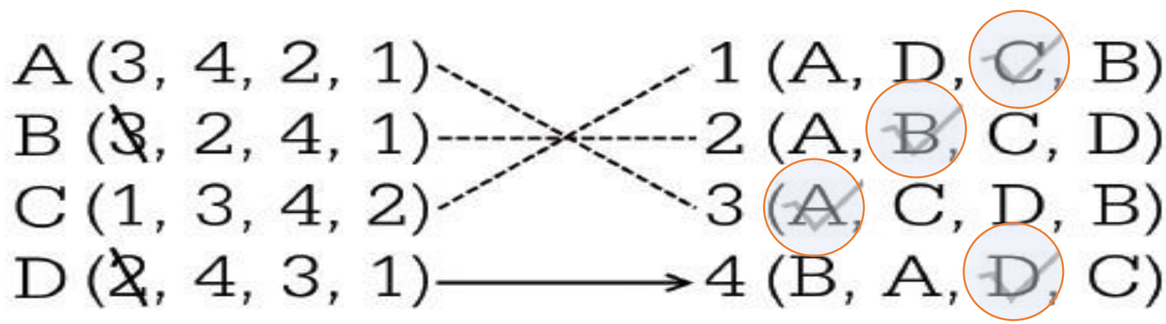
第一轮



第二轮



第三轮



9

Gale-Shapley算法

定理 Gale-Shapley 算法给出的匹配是稳定的。

练习 针对前面所考虑的实例，让女孩儿开始追男孩儿，男孩儿拒绝女孩儿，应用**Gale-Shapley 算法**求出一个稳定匹配。

这种男追女-女拒男的方案对男性更有利。

倘若有某位女性知道所有其他人的偏好情况，经过精心计算，她有可能发现，故意拒绝本不该拒绝的追求者（暂时答应一个较差的男性做情侣），或许有机会等来更好的男性。因而，在实际生活中应用**Gale-Shapley 算法**，不得不考虑博弈中的**欺诈**行为。

Gale-Shapley算法

Gale-Shapley 算法有一些局限。例如，它无法处理一般图的稳定匹配问题：假设每个宿舍可住两个人，且 $2n$ 个学生中每一个学生对其余 $2n - 1$个学生的偏好评价，如何寻找一个宿舍的稳定分配呢？此时，**Gale-Shapley 算法**就失效了。而事实上，宿舍的稳定分配问题中有可能不存在稳定的分配。

4人偏好关系

A: B > C > D

B: C > A > D

C: A > B > D

3个可能的分配

A + B; C + D

A + C; B + D

A + D; B + C

11

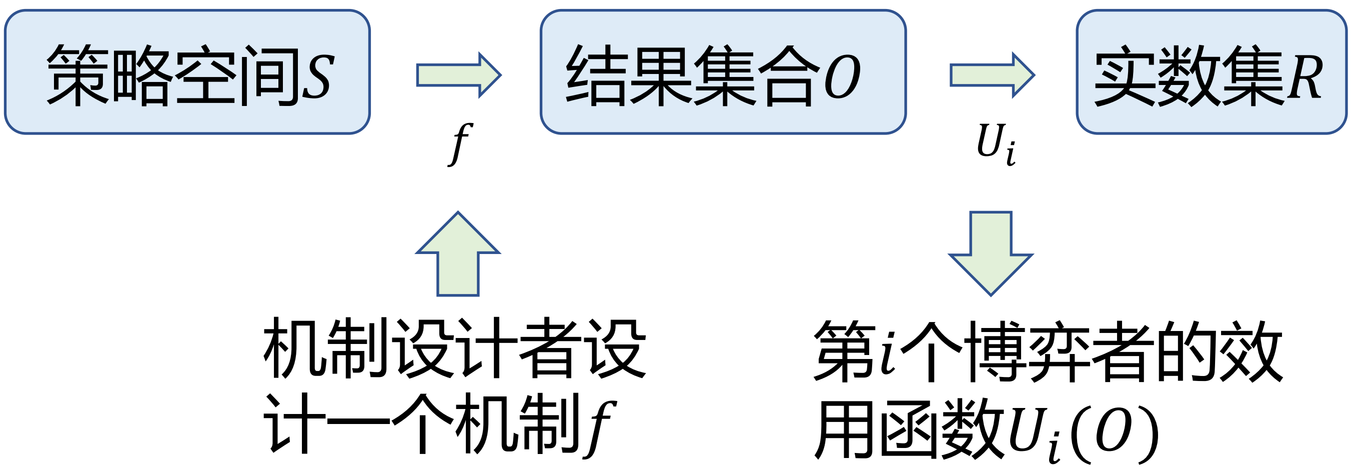
博弈分类

- 非完全信息（incomplete）： type不为人知
- 完全信息（complete）： type为人所知

- 动态博弈：下一个要作决策的博弈者可以根据前一个博弈者的决策来决定自己的决策
- 静态博弈：多个博弈参与者在不知道他人的决策的前提下同时给出自己的决策

博弈

- n个参与者
- 每个参与者i都有一个策略集S_i
- $S = S_1 \times S_2 \times \cdots \times S_n$成为一个策略空间
- 第i个参与者所作决策的效用函数U_i



社会效用：

结果集合O $\xrightarrow{U_i}$ 实数集R

第i个博弈者的效用函数U_i

社会效用函数$C(O) = f[U_1(O), U_2(O), \ldots, U_n(O)]$

纳什均衡：

任何一个博弈者均无法通过单方面改变策略使自己的效用函数值增大

纳什均衡：

纳什均衡是全部参与者所选策略的一个组合，在这个策略组合中，每个人的策略都是针对其他人策略的最佳反应

占优策略：

占优策略就是无论对方采取何种策略，其都是最佳反应的策略

占优策略均衡：

当一个博弈中的每位参与者都选择了各自的占优策略时，相应的博弈结果就是占优策略均衡

囚徒困境

		B	
		认罪	不认罪
A	认罪	10年，10年	0，20年
	不认罪	20年，0	1年，1年

“认罪”对A和B来说都是占优策略,（认罪，认罪）是一个占优策略均衡,（认罪，认罪）是一个纳什均衡

采用这种均衡策略的社会收益没有采用非均衡策略的收益好，故类似这种博弈属于社会两难。

囚徒困境

故事发生在**BBC**的一个“金球”(**Golden Balls**) 游戏节目里。节目开始有四名选手参加，经过许多轮对人性的考验和互相之间的角力后，到最后只剩下两名选手，**Ibrahim** 和 **Nick**，和一大笔的奖金。此时，主持人会给每人两个球，一个写着 **Split** (平分)，另一个写着 **Steal** (偷走)，他们需要从中选择一个球。

囚徒困境

根据两人的选择，会出现三种情况：

1. 如果两个人都选择了**Split**，那么皆大欢喜，
   两个好人可以平分之前累积的奖金，这是最理想的情况；
2. 如果其中一个人选择 **Split**, 另一个人选择 **Steal**，
   那么选择了 **Steal** 的坏人可以拿走全部的奖金，
   而选 **Split** 的好人则一分也拿不到；
3. 如果两个人都选择了**Steal**，那么两个坏人都一分钱也拿不到。

囚徒困境

金球博弈		Nick	
		平分	偷走
Ibrahim	平分	(1/2, 1/2)	(0, 1)
	偷走	(1, 0)	(0, 0)

博弈论-囚徒两难博弈（续）

在 Nick 一定选择 Steal 的情况下，Ibrahim 面临的情况是：选择 Steal，两个人都拿不到钱；选择 Split，那么 Nick 拿到所有的钱，然后节目结束后两人平分，但 Nick 是不是会遵守承诺可不好说。

但是在 Nick 如此强硬的情况下，Ibrahim 没有别的选择，他只能选择 Split。

而 Nick 呢？对着镜头，他展示了自己的选择：
也是 Split，而不是像他之前表示一定会选的 Steal。

就这样，两个人皆大欢喜，平分了奖金。

从优化到博弈：如何衡量均衡?

给定一个博弈系统的例子I，其最差的纳什均衡解NE(I) 与社会最优解OPT(I) 的比值记着PoA(I), 其最好的纳什均衡与社会最优解的比值记成PoS (I)

Price of Anarchy (PoA): 所有PoA (I)中最差的值

Price of Stability (PoS): 所有PoS (I)中最差的值

PoA 和 PoS的分析

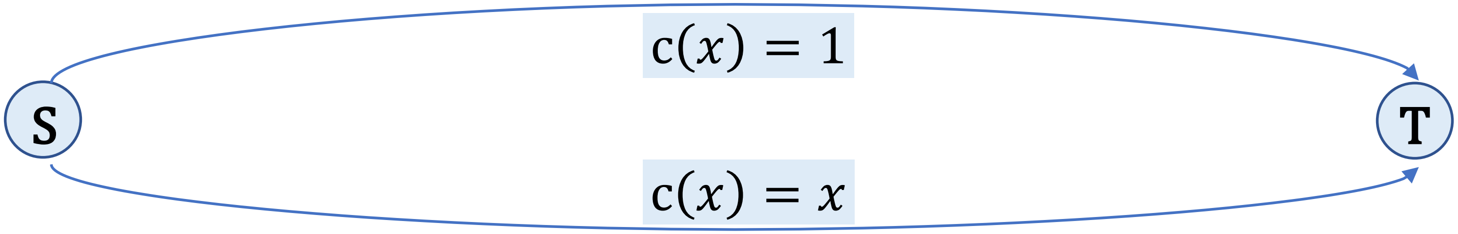
- Price of Anarchy (PoA)

对任何一个实例，证明任何一个纳什均衡对应的目标函数值不超过最优解的某个比值（上界）;构造一个例子证明比值的下界。

- Price of Stability (PoS)

对任何一个实例，证明某一个纳什均衡对应的目标函数值不超过最优解的某个比值（上界）;构造一个例子证明比值的下界。

Pigou's example

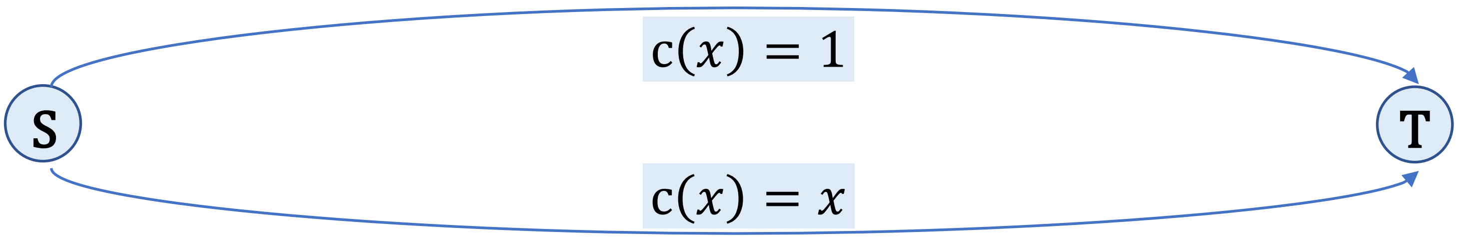


两条不相交的边分别连接着起点S和终点T，函数$c(x)$表示一条边上产生的延迟费用，x表示这条边上的交通量

$c(x) = 1$表示这条路相当长而没有交通堵塞；$c(x) = x$表示这条路的费用随着堵塞时间越长而越高

特别地，下方那条边费用不超过上方那条边的费用当且仅当交通量不超过一个单位

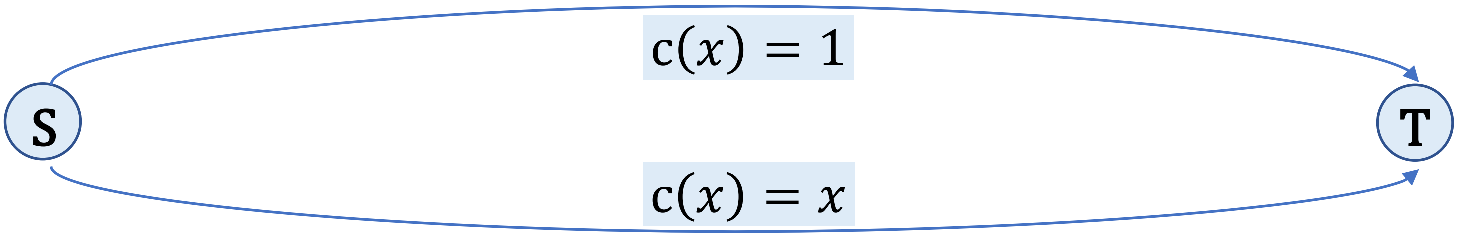
Pigou’s example



假设只有一个单位的交通量，
参与者有很多且相互独立地选取从起点S到终点T的路

假设每个参与者的目标都是最小化费用，
每个人选择下方那条边即是占优策略；在这个纳什均衡中，所有参与者都会产生 1 的费用。

Pigou's example



目标函数是最小化所有参与者费用的平均值。

在以上的均衡中，平均值为1。

设上方交通量为x，总费用为$x \cdot 1 + (1 - x) \cdot (1 - x)$，欲使其达到最小，$x$取0.5。

把交通量均分到两条线路上才是这个例子的“最优结果”（$0.5 \cdot 1 + 0.5 \cdot 0.5 = 0.75$）。

故$PoA = PoS = 1/0.75 = 4/3.$