第十三, 十四讲 算法博弈

主讲教师: XXX
讲义整理: XXX

2021 年 1 月 9 日

1 稳定匹配问题

1.1 问题描述

已知男女集合 $B = b_1, b_2, ...b_n$ $G = g_1, g_2, ...g_n$, 并且知道每个男性在女性心中的地位, 以及女性在男性心中的地位，通过一个 n 元组表示，如 $b_1 = g_2, g_3, g_1, ...g_i$ 代表在男性 1 中 g_2 地位最高，g_i 地位最低。

我们在男女构成的二部图中建立一个完全匹配 M 来达到理想状态，理想状态定义如下：

拆分对: 对于一个男女匹配 (b_i, g_j)，若 $\exists g_m, b_n$, 在 b_i 中 $g_m > g_j$(前者比后者地位高), 在 g_j 中 $b_n > b_i$, 则称匹配 (b_i, g_j) 为**拆分对**。

稳定匹配: 任对完全匹配 M 中一个匹配 (b_i, g_j) 均不是拆分对则称 M 为稳定匹配。

1.2 Gale-Shapley 算法

通过每次见到拆分对就进行改变的 locol change 算法，不一定能构造稳定匹配，因为可能构成一个循环。

Gale-Shapley 算法
1. 每一个男性 b_i 同时向其最优女性（心中地位最高）告白；
2. 每一个女性，若存在向其告白的男性，则选择自己心中最优的一位，**暂时**构成匹配；若不存在向其告白的男生，则等待
3. 够称匹配的男性无需再进行告白，其余男性向其最优的，**尚未被拒绝的女性**告白。
4. 女性继续选择最优的构成匹配，如果有超过当前匹配的男性的地位的男性向其告白，则拆分原来的匹配，构成新的临时匹配。
5. 重复 3-4 步，直到所有人构成匹配。

1.3 Gale-Shapley 算法的有趣结论

1. 如果把算法的主动一方与被动一方置换，结果可能不同；且对于主动追求的一方，其匹配的平均满意度更高（定义满意度为：当前对象的地位相比最优对象的地位的差值）

2. 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。3. Gale-Shapley 算法只能处理如二部图的匹配问题，对一般图无法构成稳定匹配。(考虑同性别的组宿舍问题)

2 算法博弈重要概念

纳什均衡：动态博弈的稳定的最终结果，在纳什均衡下，任何一个博弈参与者的任何调整策略的方式都无法使自己的收益更优；

社会效用：定义博弈的结果集为 O，第 i 个博弈参与者的效用函数为 U_i，则社会效用函数 $C(O) = f[U_1(0), U_2(0), U_3(0)......U_n(0)]$。

Price of Anarchy-PoA: 对于一个博弈系统 I，其中社会效用最**低**的纳什均衡与最优社会效用的比值计为 $PoA(I)$, 则对于任意实例 I 中最低 $PoA(I)$ 记为 PoA。

Price of Stability-PoS: 对于一个博弈系统 I，其中社会效用最**高**的纳什均衡与最优社会效用的比值计为 $PoS(I)$, 则对于任意实例 I 中最低的 $PoA(I)$ 记为 PoS。

3 机制设计

3.1 cake cutting 问题

设计一种机制，将一个蛋糕分成 n 份，使每个人对其分配到的那个蛋糕“满意”。
” 满意” 的具体含义由不同要求决定。

需要注意的是，每一个参与者对于蛋糕的衡量标准是不一样的，例如一个 3-分割在 A 看来是 $(1/3, 1/3, 1/3)$，但是在 B 看来是 $(1/4, 1/4, 1/2)$。但是无论对于哪个衡量标准，都符合基本运算规则（例如其和必然为 1）

3.1.1 公平分割

公平：每个参与者获取到的蛋糕，在其衡量标准下为 1/n 个。

二人的 cake cutting 问题非常简单，只需要一人切一人分即可，不作赘述。

三人及以上的 cake cutting 问题不能完全套用二人的模式（一人分，轮流选，分的人最后选），因为否则只有分蛋糕以及第一个选取的人会满意，中间的人则不会。

考虑 $n > 2$ 的情况：
迭代算法：
迭代的起点为 2，采用 2 人 cake cutting 方式使二人满意。

迭代过程：
设 i 个人已经满意，对于第 i+1 个人，前 i 个人将其蛋糕分为 i+1 份，第 i+1 个人选取其认为最大的一块。

迭代直至 i=n 为止。

3.1.2 无嫉妒分割

无嫉妒：如果每个参与者服从了既定规则后，都能保证自己得到的蛋糕不少于其他任何一个参与者得到的蛋糕（从自己的衡量标准来看）。

n=2 使依然采用公平分割的方式；

n=3 时的算法如下：
设三人为 A，B，C；
第一阶段：
1.A 按照自己衡量标准等分蛋糕为 3 份。
2.B 按照自己衡量标准，对最大的两个蛋糕块，若一样大则不处理；若不一样大则裁剪最大的蛋糕，将裁剪的部分称为 S，剩下的部分称为 T。S 暂时移除。
3. 按照 C，B，A 的顺序来选取除 S 以外的蛋糕，若 C 没有选择 T，则 B 必须选择 T

第一阶段必然是无嫉妒的：
1.A 等分蛋糕，且不会选到被切除的块 T，故不会嫉妒；
2.C 最先选取蛋糕，会选择最优解，不会嫉妒；
3.B 确保最大的两块蛋糕对自己的价值是相同大的，且第二个选择，则不会嫉妒；

第二阶段：
1. B 或 C 中没有在第一阶段选择 T 的等分蛋糕为三块；
2. 切蛋糕的最后选择，A 第二个选择，剩下一人（即第一阶段选择 T 的）第一个选择；

第二阶段必然是无嫉妒的：
1. 同理，切分蛋糕的不会产生嫉妒；2. 第一个选择蛋糕的人不会产生嫉妒；3.A 不会嫉妒第一阶段选择 T 的人（因为在第一阶段的 A 视角下，选择 T 的人即使获取整个 S 块也不会使 A 嫉妒）。A 不会嫉妒第三个选择的人。故 A 不会产生嫉妒；

两阶段都是无嫉妒的，则整个过程是无嫉妒的。

3.2 设施选址博弈

3.2.1 问题描述

输入：
公有信息：网络上的 n 个局中人；
私有信息：每个局中人的私有位置 x_i,$x = (x_1, x_2, ... x_n)$
(私有信息表示，机制并不知道这些信息的真实数据，只能由局中人告知，而告知的信息不一定是真的，可能存在欺诈行为)

输出：
设施的位置 y

机制：
$f(x) = y$

效用函数：
局中人费用函数：$cost(y, x_i) = |y - x_i|$

社会目标函数：（即要达到的目标）
距离之和最小:$sc(y, x) = min\Sigma_i cost(y, x_i)$
最大距离最小:$mc(y, x) = min max_i cost(y, x_i)$

3.2.2 算法博弈概念

近似比 ρ：

$sc(f(x), x) \leq \rho * sc(opt(x), x)$
$mc(f(x), x) \leq \rho * mc(opt(x), x)$

防策略操纵性 Strategy proofness：
直观理解,防策略操纵性即对于一个机制，每个局中人谎报其私有信息不能获取更高的个

人效益。

形式化表述如下：
$\forall$ 策略组合 $x \in R^n$,$\forall i \in N$ $\forall x_i' \in R$，均有

$$cost(f(x), x_i) \leq cost(f(x_i', x_{-i}), x_i)$$

其中 $x_{-i} = (x_1, ..., x_{i-1}, xi+1, ..., x_n)$。

3.2.3 网络为线，以 $sc(f(x), x)$ 为目标

$y* = med(x)$
med 表示中位数

易证改算法为最优，且是防策略操纵的。
任意一个位于 y 右侧的局中人，可以选择向左侧谎报或向右侧谎报；
向左侧谎报不会影响 y 的值；向右侧谎报将导致 y 右移，使自己的费用函数增加。
y 左侧同理，故不存在有人谎报的情况

3.2.4 网络为线，以 $mc(f(x), x)$ 为目标

$l(x) = min\{x_1, x_2......x_n\}$
$r(x) = max\{x_1, x_2......x_n\}$
$y* = cen(x) = (l(x) + r(x))/2$

该算法最优，但是不是防策略操纵的。任何一个人可以向左或向右谎报位置，使 y 离他们更近。

以下给出近似比为 2 的防策略操纵机制 $y* = cen(x) = l(x)$

Theorem 1 *该问题的防策略操纵机制的近似比至少为 2*

Proof 3.1 *对于防策略操纵机制 M，对于实例 I={0,1}, 设输出为 y，*$0 < y < 1$

再考虑实例 I={0,y}，此时的输出必须为 y，否则在 y 上的局中人会谎报位置为 1，迫使输出为 y。

而此时最优解显然为 y/2，故近似比至少为 2。

3.3 厌恶型设施选址博弈

3.3.1 问题描述

问题与上述设施选址博弈基本相同，唯一区别在于社会目标函数：
使所有局中人到距离之和最小:$sc(y,x) = min\Sigma_i cost(y,x_i)$
同时，所有人都位于 [0,1] 之间，且设置也必须在 [0,1] 之间

设计机制如下

$$f(x) = \begin{cases} 0, & \sum_{i=1}^{n} x_i \geq \sum_{i=1}^{n} (1-x_i) \\ 1, & \text{otherwise} \end{cases}$$

上述机制不是防策略操纵性的，因为每一个在左侧或在右侧的人，可以谎报位置在在 0 或 1 处，使设施尽可能离自己远。

不过注意到，在上述情况下，那一侧的人多，最终设置的位置就位于另一次，故修改机制如下:
n_1 表示位于 [0,1/2] 的人的数量，n_2 表示位于 (1/2,1] 的人的数量。

$$f(x) = \begin{cases} 0, & n_1 > n_2 \\ 1, & \text{otherwise} \end{cases}$$

不难证明这一机制是防策略操纵性的，且近似比为 3

最坏情况:
有 $1/2+\sigma$ 的人位于 0，$1/2-\sigma$ 的人位于 1/2。

3.4 装箱博弈

3.4.1 模型描述

记每个箱子费用为 1，箱子内的物体要以某种方式分配箱子的费用，且每个箱子趋向于使自己分配的费用最少。

稳定装箱：在某种分配算法结束后，没有箱子会改变自己的位置，即分配算法的结果是一种纳什均衡。

问题：提出一种机制，使得最后消耗的箱子（社会效益）最少。通过 PoA 来衡量。

3.4.2 公平机制

按比例分配：箱子内的物体按物体的体积比分配箱子费用。
PoA 约为 1.64;
同等分配：箱子内的物体均摊箱子费用。
PoA 接近 1.7;

问题：是否能提出一种算法，在公平机制下能实现稳定分配：
（算法课中熟悉的 NF，FF，WF，BF 均不是稳定分配。）
例如：对于同等分配机制，将物体从小到大排序，然后依次分配，是一种稳定匹配。

3.4.3 差异机制

首先考虑如下机制 **Large Pay Public**：
每个物体支付自己所占空间的费用，而空余部分的费用由大物体支付。

这一机制的 PoA 为 1.5，考虑如下分配情况：

A 箱中装有一个 $1/2+\lambda$ 的物体，B 箱中装有一个 $1/2+\lambda$ 的物体，C 箱中装满大小为 λ 的物体，将这套组合重复 k 次。

上述结果由 Large Pay Public 产生，而最优的方式为：
A 箱中装有一个 $1/2+\lambda$ 的物体，以及数个 λ 直至填满；B 箱中装有一个 $1/2+\lambda$ 的物体，以及数个 λ 直至填满，将这套组合重复 k 次。最后一个箱子内有数个大小为 λ 物体。

二者之比为 1.5。

为了改进上述机制，关键在于如何促使小物体进入大物体所在的箱子：
(大物体，通常指体积超过 1/2 的物体)
故引入折扣机制，小物体装入大物体所在的更满的箱子内中会得到折扣，越满折扣越大;

但是折扣不能是任意的，否则对于大箱子，当箱子过满后大箱子可能会移动到其他箱子，以避免分担小物体的巨大折扣，故折扣函数同样需要使得箱子越满，大物体分摊的费用越低。

综合考虑上述两个要求，给出如下**折扣函数**：
$f(x)=1-(2/3)x$
其中 x 代表箱子内物体大小，而函数值则是小物体的折扣值 (折扣值乘小物体大小即小物体支付费用)。

无大物体的箱子依然使用按比例分配原则。

可以证明，这一机制的 PoA 不超过 22/15。