目录

本节概要

1 综述

2 理论构建
- 限制论域
- 基本概念
- 排课函数
- 排课偏好
- 生成个体 i 对所有时间段的排序 Q_i
- 处理争端
- 输出结果
- 排课函数 H 的性质
- 扩展到更一般的情况

3 编程实现

本节概要

1 综述

2 理论构建
- 限制论域
- 基本概念
- 排课函数
- 排课偏好
- 生成个体 i 对所有时间段的排序 Q_i
- 处理争端
- 输出结果
- 排课函数 H 的性质
- 扩展到更一般的情况

3 编程实现

限制论域

1. 假定每个老师需要上相同时间长度的课。
2. 限制输入，使偏好组合（理论中表示为排课组合 K）满足单峰性。
3. 假定不存在多个老师上同一门课的情况。
4. 假定一个时间段内只有一个老师上课。
5. 不考虑课程的搭配问题。
   - 工作人员难以按课程名把课程分为文科、理科、工科等。(如哲学逻辑)
   - 同一类别的课程可能有不同的难度，有些很轻松，有些很硬核. 而工作人员很难了解所有课程的难度。
6. 假定没有金钱交易。

基本概念

个体集 $N=\{0,1,2,\ldots,n-1\}$. 设有 n 位老师.

日期集 $D=\{d_0,d_1,d_2,\ldots,d_{m-1}\}$. 设一周有 m 天要上课.

课序集 $P=\{p_0,p_1,p_2,\ldots,p_{t-1}\}$. 设一天有 t 节课。如果没有特殊说明，“时间段”指的是一周中的某个上课时间段.

课时 $r\in N^*$ 是每个老师一周里要上的课时。

排课函数

排课函数 $H : \overline{K} \to \overline{R}$

- $K = (K_0, K_1, K_2, \ldots, K_{n-1})$ 是一个排课组合，其中 K_i 指的是序号为 i 的老师（个体 i）给出的排课偏好.
- R 是排出来的课程表.
- $\overline{K}$ 是所有排课组合的集合.
- $\overline{R}$ 是所有可能的课程表的集合.

排课偏好

个体 i 的排课偏好 $K_i = (D_i, P_i, A_i, a_i, b_i, c_i)$

- 精确偏好
  - a_i 是个体 i 的第一志愿.
  - b_i 是个体 i 的第二志愿.
  - c_i 是个体 i 的第三志愿.
- 模糊偏好
  - D_i 是个体 i 对日期的排序.
  - P_i 是个体 i 对课序的排序.
  - $A_i \in \{0, 1\}$ 是个体 i 对 D 与 P 的重视程度的比较.
    0 表示 $D > P$, 1 表示 $P > D$.
- Q_i 是个体 i 对所有时间段的排序.

生成个体 i 对所有时间段的排序 Q_i

操作步骤

已知 $D_i = (d^i_0, d^i_1, d^i_2, \ldots, d^i_{n-1})$，$P_i = (p^i_0, p^i_1, p^i_2, \ldots, p^i_{n-1})$，$A_i \in \{0, 1\}$.

- $A_i = 0$，即 $D > P$. 则优先排 D_i，生成元组

$$((d^i_0, p^i_0), (d^i_0, p^i_1), \ldots, (d^i_1, p^i_0), (d^i_1, p^i_1), \ldots, (d^i_2, p^i_0), (d^i_2, p^i_1), \ldots)$$

- $A_i = 1$，即 $P > D$. 则优先排 P_i，生成元组

$$((d^i_0, p^i_0), (d^i_1, p^i_0), \ldots, (d^i_0, p^i_1), (d^i_1, p^i_1), \ldots, (d^i_0, p^i_2), (d^i_1, p^i_2), \ldots)$$

在生成的元组中把 a_i, b_i, c_i 挑出来放到前三位，得到 Q_i.

处理争端

基本思想

用计算复杂性来防止策略操纵。实现这一目标的方法:

- **比较落选次数**.
- **Collection.shuffle(List<?> list) 方法**. （计算的时候会打乱顺序，输出结果时仍然按照原来的顺序）
- **对处理争端时系统时间的毫秒数取模**.

操作过程

在第 i 轮（从 0 开始计数）考虑未分配完课时的老师们的第 $i+1$ 志愿. 一轮可以分配多个时间段. 对于每一个被给出的时间段:

该时间段已被分配

每位老师的落选次数加一，进入第 $i+1$ 轮.

该时间段未被分配

- 如果只有一位老师想要这个时间段或者某位老师的落选次数大于其他老师，则把这个时间段分配给这位老师. 剩余老师(可以没有）的落选次数加一.
- 如果想要这个时间段的的老师中有多位老师拥有最大落选次数，则
  1. 用 Collection.shuffle(List<?> list) 方法打乱老师排序. 给这些拥有最大落选次数的老师们从 0 开始标号.
  2. 对计算机处理这个争端时的系统时间的毫秒数取老师数量的模，拥有与结果相对应的序号的老师获得这一个时间段.
  3. 其他老师的落选次数加一.

输出结果

课程表 R

在所有老师都分配完课时后，

1. 将各个时间段对应的老师的名字填入课程表.
2. 输出课程表 R.

排课函数 H 的性质

- **非独裁的**.
- **满足中立性**. 两位老师交换 K_i 和工作人员输入他们信息的时间，在那一次随机排课中两人的上课时间段交换。
- **用计算复杂性作为操纵的障碍**. 操纵者需要知道 $shuffle()$ 方法打乱老师排序的结果、处理争端时系统时间的毫秒数、随机过程对其他老师的影响.
- **不是帕累托最优的**. 这是使用模糊偏好的结果，不过这是合理的，因为老师不可能给出关于所有时间段的精确排序，更不可能给出关于所有时间段的集合的精确排序. 此外还能减少老师输入的数据量.

排课函数 H 的性质

- **要输入的数据过多**. 不过与输入对所有时间段的排序相比减少了输入的数据.
- 精确偏好与模糊偏好的**数量分配很难设置**.
- **一致同意性**在这个场合下**无意义**. 每个老师只能对自己的课程安排发表看法，并且评价一个课程表的角度是多样的，几乎不可能有一个所有人一致同意的课程表.
- **匿名性、单调性**在这个场合下**无意义**.

扩展到更一般的情况

这个排课函数可以被用于给一个集合中所有主体分配满足以下性质的元素 x:

1. 该元素有两个参数，记为 a 和 b，其中 $a \in A, b \in B$.
2. 该元素可以等于 $A \times B$ 中的任何元素. 即各种情况都可能出现. 类似于每天有相同数量的课时。
3. 主体对元素 x 的偏好由参数 a 和 b 共同决定.

本节概要

天罡三十六星

在遥远的的 M108 星云有一颗行星，它的公转周期较长，一周有 9 天；自转周期特别长，白天长得足够上 16 节课. 行星上的“施耐庵大学”要给 36 个“梁山”老师排课表：宋江、卢俊义、吴用、公孙胜……每个老师要上 4 节课.

谢谢观看