第3章 减治法

《人工智能算法》

清华大学出版社

2022年7月

提纲

- 减治法策略
- 拓扑排序
- 总结

减治法策略

减治法

大规模核酸检测：在席卷全球的新冠病毒感染检测和疫情防控中，由于人口众多，如果对每个人的核酸检测样本逐一检测以确诊感染病毒，实施难度较大。

多人混检降低大规模筛查成本：针对40人以内的待检测人员群体，可将被检测人员根据人数均分为两组进行核酸混样检测，将每组检测人员的全部样本放到一起统一检测。随着核酸检测成本的逐渐降低，混检数量从40减少到10和5等。

逐步缩小筛查范围：如果是阴性，则表明该组被检人员均不是病毒感染患者；如果是阳性，则表明该组被检人员中存在病毒感染患者。然后，对该组人员做进一步分组，继续进行核酸混样检测，循环往复，直至找出患者。

减治法策略

- **减治法（Decrease-and-Conquer）**

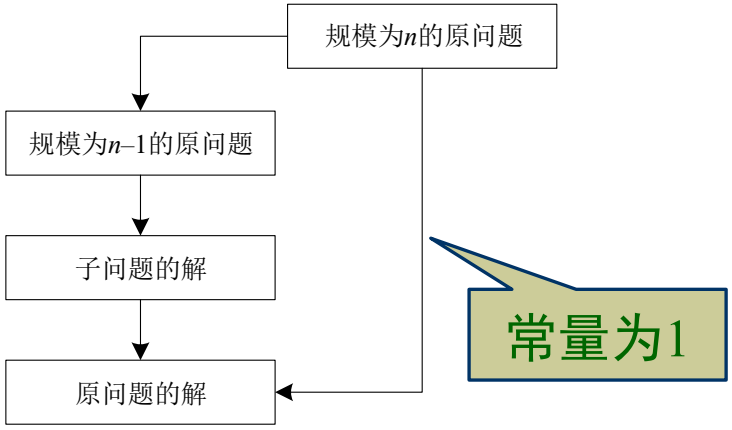
利用给定规模与较小规模问题解之间的关系，从顶至下（递归）或从底至上（非递归）求解问题的一种方法

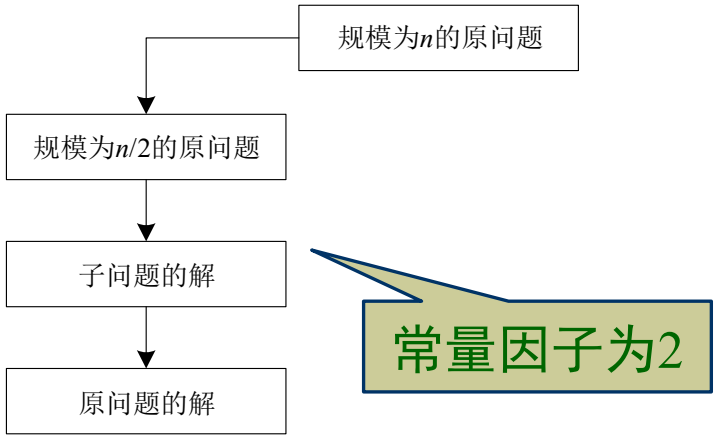
- **3种类型**

（1）减常量法：常量通常为1即减1法，也有减2的（如奇偶数分别处理）

（2）减常因子法：常因子通常为2（减半技术）

（3）减可变规模法：规模减小的模式不同





提纲

- 减治法策略
- 拓扑排序
- 总结

拓扑排序 (1)

- **问题描述**

- 假设我们要安排一系列任务，如任务分工、教学计划中的各门课程的安排顺序（先修课），项目中各子课题的研究顺序，建筑项目等
- 每个任务只有其当先决条件具备时，才能着手安排这个任务去完成
- 找到在满足先决条件情况下，各个任务如何安排的一个**线性序列（先决条件不矛盾）**

例如：

5门必修课的集合$\{C1, C2, C3, C4, C5\}$，学生必须修完这些课程。先决条件：

1）$C1$和$C2$没有先决条件

2）修完$C1$和$C2$才能修$C3$

3）修完$C3$才能修$C4$

4）修完$C3$和$C4$才能修$C5$

问题：

- 学生按什么顺序学习这些课程？
- 解不唯一？

拓扑排序 (2)

- **建模——图**

顶点——任务，边——某个任务的先决条件

(1) 有向图：任务之间有先后关系（有向边）

(2) 无环图：若为有环图，回路中就存在相互矛盾的条件，问题无解

拓扑排序有解的图，必然是**有向无环图**

- **基于深度优先遍历的拓扑排序**

- 执行一次深度优先查找，记住顶点变成死端（出栈）的顺序
- 拓扑排序的一个解：将出栈次序反过来

- **基于减一技术的拓扑排序**

- 在余下的有向图中找一个源（没有入边的顶点），删除该源及出边
- 源不存在，算法停止
- **拓扑排序的一个解：顶点被删除的次序**

拓扑排序 (3)

算法

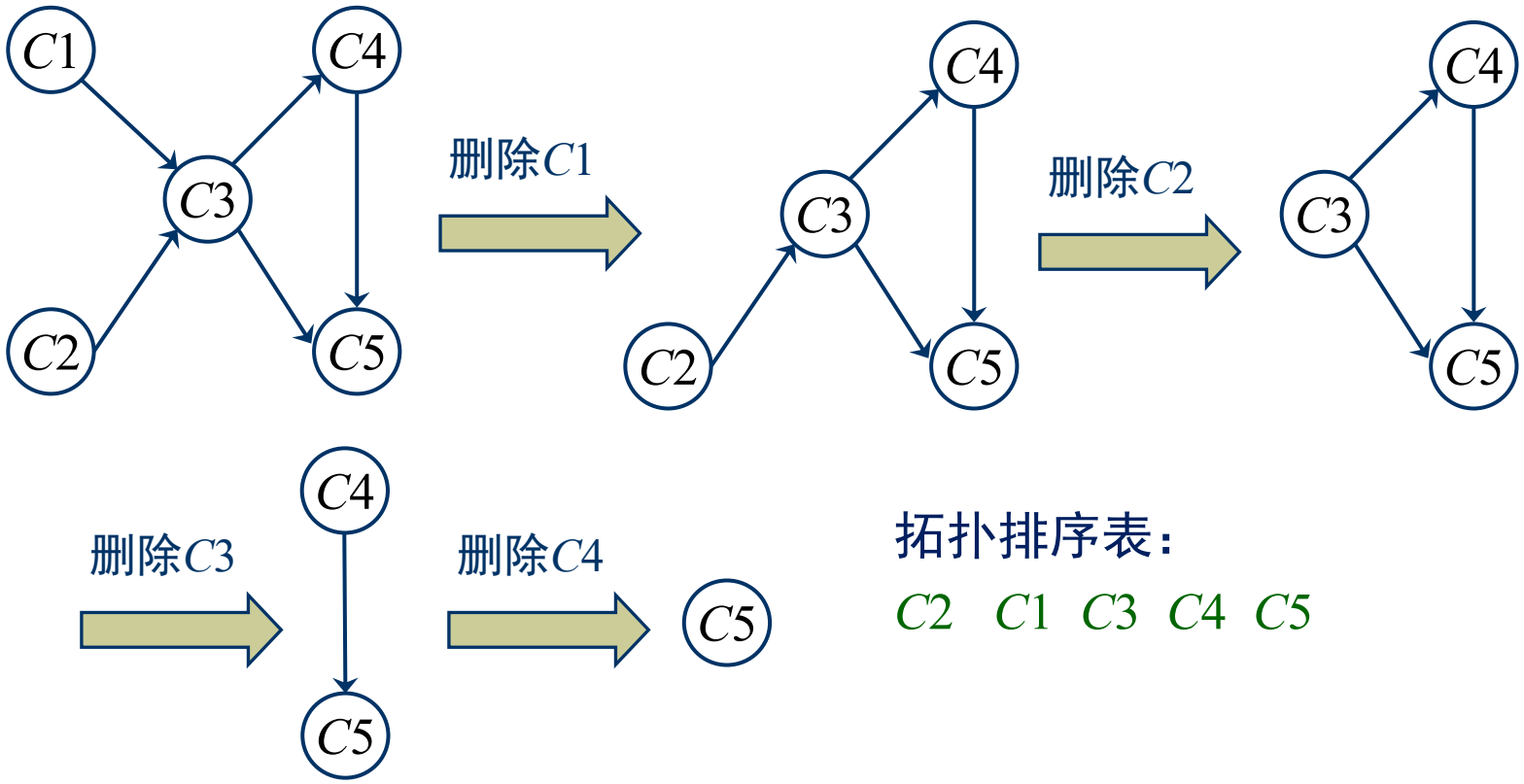
输入：G：给定n个顶点、m条边的有向无环图
输出：$List$或False：输出存储拓扑序列顶点的列表，或不存在拓扑序列

初始化队列Q，用于存储入度为0的顶点
For i =0 To $n-1$ Do //遍历图G中的顶点
  If $Indegree[i]$=0 Then //将入度为0的顶点加入队列
    Enqueue(i, Q)
  End If
End For
$List \leftarrow \emptyset$ //初始化用于存储输出顶点的列表
While $Q \neq \emptyset$ Do
  Dequeue(e, Q) //按入队顺序输出队列Q中的顶点e
  $List \leftarrow List \cup \{e\}$
  For 每一个与e邻接的u Do //对e顶点的每个邻接顶点u的入度减1
    $Indegree[u] \leftarrow Indegree[u]-1$
    If $Indegree[u]$=0 Then //若减一后入度为0，则加入队列
      $Q \leftarrow Q \cup \{u\}$
    End If
  End For
End While
If $|List|=n$ Then
  Return $List$ //按序输出列表中的顶点，作为拓扑排序结果
Else
  Return False //图G不是有向无环图，不存在拓扑排序
End If

时间复杂度：$O(n \times m)$

拓扑排序 (4)

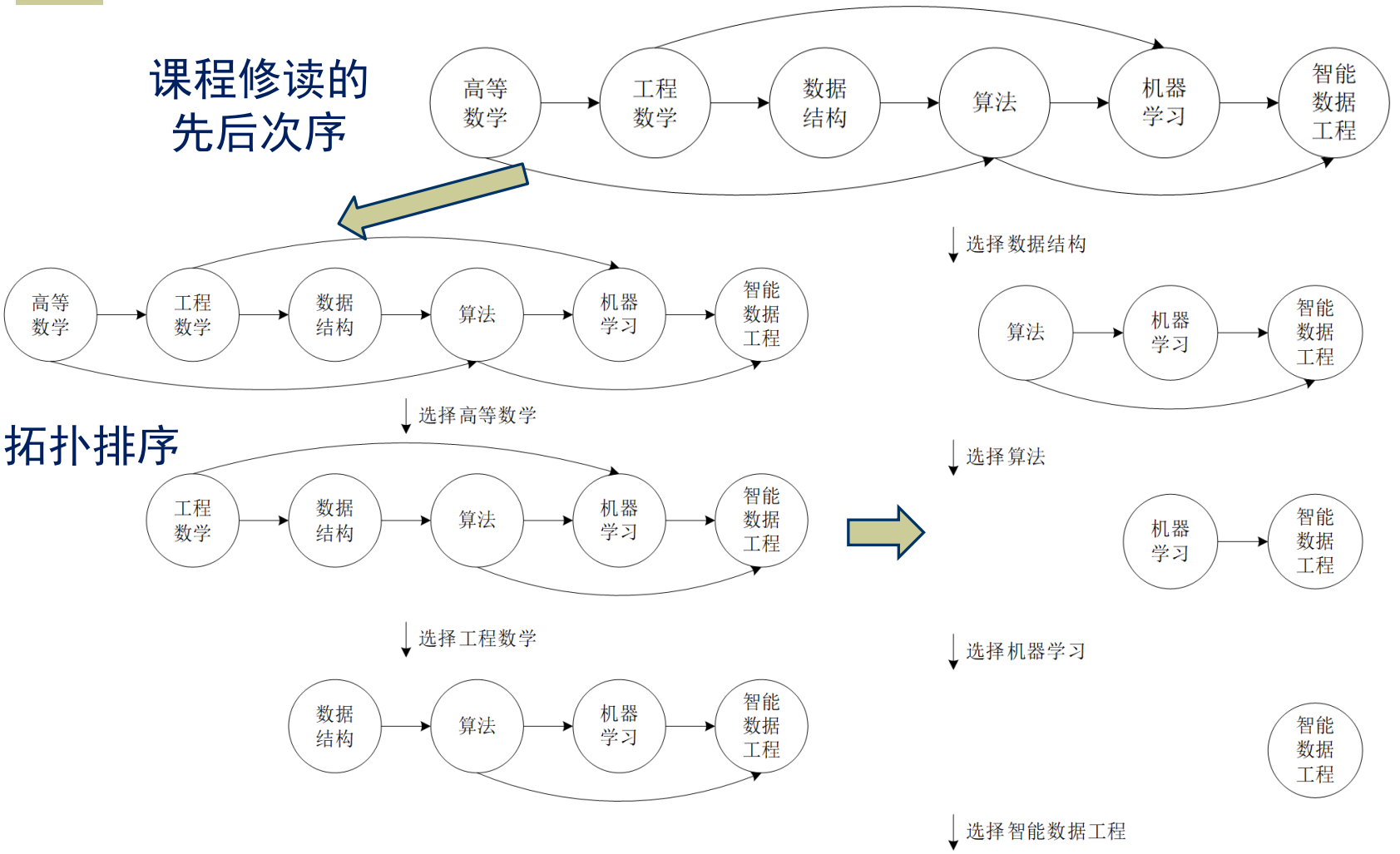
基于减一技术的拓扑排序（源删除算法）



拓扑排序表：

C2 C1 C3 C4 C5

拓扑排序 (5)


课程修读的
先后次序
高等
数学
工程
数学
数据
结构
算法
机器
学习
智能
数据
工程
拓扑排序
选择高等数学
选择工程数学
选择数据结构
选择算法
选择机器学习
选择智能数据工程

拓扑排序 (6)

- **源删除算法的实现——基于数组（图存储：邻接矩阵）**

用一个入度数组保存每个顶点的入度**（无取出规则）**

(1) 找到入度为0的点，将其存入数组中，再将其从图中删除（与它相关的边都删除，相邻的顶点的入度均减1）

(2) 重复步骤(1)执行，直至所有的顶点都被找到为止

时间复杂度为$O(n^2)$（即图的遍历）

- **源删除算法的实现——基于队列（图存储：邻接表）**

(1) 在图中找出所有无先决条件的源顶点，将它们全部入队**（取出规则）**

若无源，算法停止，输出记录的顶点序列，得到解（无未访问顶点）

(2) 队头顶点出队（实现删除操作），且按出队顺序记录顶点，并同时删除从这个顶点出发的所有的边（队列的出入队顺序相同）

(3) 返回步骤(1)执行

时间复杂度为$O(n+e)$（初始源顶点$O(n)$，基于队列找邻接点$O(e)$）

结语

谢谢！