中国矿业大学

第四章 贪心算法

§4.2 贪心算法的基本要素

韩丽霞

计算机科学与技术学院
School Of Computer Science And Technology, CUMT

学习要点

01 贪心选择性质

02 贪心算法与动态规划算法的差异

03 背包问题和0-1背包问题

基本要素

(1) 贪心选择性质

(2) 最优子结构性质

以活动安排问题为例，讲解贪心选择性质。

贪心选择性质

贪心选择性质：指所求问题的整体最优解可以通过一系列局部最优的选择，即贪心选择来达到。

步骤如下：

1)假设问题有一个整体最优解，并证明可修改这个最优解，使其以贪心选择开始。

2)运用数学归纳法证明：

每一步贪心选择→问题的整体最优解。

贪心算法解决活动安排问题

活动安排问题：n个活动的集合$E=\{1,2,\cdots,n\}$，

选出最大的相容活动子集合。

贪心策略：最早结束的活动，优先安排。

将n个活动按结束时间非减序排列

课后作业

思考如下具有11个活动安排的问题？

活动	1	2	3	4	5	6	7	8	9	10	11
s_i	0	4	4	5	3	1	8	6	8	12	2
f_i	3	6	5	6	8	4	11	10	12	14	13

解：待安排的11个活动按结束时间的非减序排序。

活动	1	6	3	2	4	5	8	7	9	11	10
s_i	0	1	4	4	5	3	6	8	8	2	12
f_i	3	4	5	6	6	8	10	11	12	13	14

最大相容子集{1,3,4,8,10}

{6,3,4,8,10}　{1,3,4,8,10}

{6,3,4,7,10}　{1,3,4,7,10}

贪心选择证明

证明：1)证明活动安排问题有一个最优解以贪心选择开始。

设A是所给活动安排问题的一个最优解，且A中活动按f_i非减序排列，A中的第一个活动是活动k。

若$k=1$，则A是一个以贪心选择开始的最优解。 $\{1,3,4,8,10\}$

否则，构造 $B = A - \{k\} \cup \{1\}$

A={6,3,4,8,10}　B={1,3,4,8,10}

A={6,3,4,7,10}　B={1,3,4,7,10}

贪心选择证明

下面证明B也是该活动安排问题的最优解。

因为$f_1 \le f_k$， 故B中的活动是相容的。

$|A| = |B|$，因此B也是该活动安排问题的最优解

由此可见，总存在以贪心选择开始的最优活动安排方案。

贪心选择-最优解

2)每一步贪心选择→问题的整体最优解。

若A是原问题的最优解，则$A' = A - \{1\}$是活动安排问题$E' = \{i \in E: s_i \geq f_1\}$的最优解。

反证法：若A'不是E'的最优解，B'是其最优解。

$|A'| \leq |B'|$ 则$A'' = B' \cup \{1\}$是活动安排问题E的最优解。

$$|A''| > |A|$$

矛盾

对贪心选择次数用数学归纳法即知，贪心算法产生原问题的最优解。

贪心算法-动态规划

共同点：均具有最优子结构性质。

不同点：贪心算法具有贪心选择性质，仅在当前状态下做出最好选择，不依赖于子问题的解；动态规划法：每步所作的选择依赖于相关子问题的解。

背包问题

背包问题：给定n种物品和一个背包，物品i的重量是w_i，其价值为v_i，背包的容量为c。问如何选择装入背包中的物品，使得装入背包中物品的总价值最大？

$$\begin{cases} max \quad \sum_{i=1}^{n} v_i x_i \\ s.t. \quad \sum_{i=1}^{n} w_i x_i \le c \\ \qquad \underline{0 \le x_i \le 1} \quad i = 1,2,\cdots,n \end{cases}$$

贪心思路

价值最大 → 性价比高优先

贪心策略：单位重量价值最高的物品，优先装包。

将n个物品按单位重量价值从大到小的顺序排序

贪心算法步骤

1) 计算每种物品的单位重量价值$\frac{v_i}{w_i}$；

2) 按照物品单位重量价值由大到小排序；

$O(nlogn)$

3) 依次选择单价高的物品优先装包，直至装满。

背包问题

例1 背包问题 $n=3, c=20, W=(18,15,10), V=(25,24,15)$

解：$\dfrac{v_1}{w_1}=1.39 \qquad \dfrac{v_2}{w_2}=1.6 \qquad \dfrac{v_3}{w_3}=1.5$

在背包中放入物品2，重量为15，价值为24；

放入一半物品3，背包重量为 20，价值为31.5。

背包问题的最优解为(0,1,0.5)，价值为31.5。

0-1背包问题

$$
\begin{cases}
\max & \displaystyle\sum_{i=1}^{n} v_i x_i \\
s.t. & \displaystyle\sum_{i=1}^{n} w_i x_i \leq c \\
 & \underline{x_i \in \{0,1\}} \quad i = 1,2,\cdots,n
\end{cases}
$$

贪心算法能否求解0-1背包问题？

小结

01 贪心选择性质

02 贪心算法与动态规划的对照

03 背包问题与0-1背包问题