

CoinFlips 解题报告

绍兴市第一中学 王文涛

1 试题来源

TopCoder SRM 681 Div.1 Level 3 CoinFlips

2 试题大意

有 n 个硬币从左到右依次排成一排。第 i 个硬币的价值为 $vals_i$ 。现在要进行 n 轮游戏。每轮游戏开始时会把每个硬币随机抛一下，正面朝上的概率为 $prob$ 。然后从左往右选中第一个正面朝上的硬币。如果没有正面朝上的硬币，就选中左边第一个。然后，如果选中的这个硬币左右都有硬币，就计算左边硬币的 $vals$ 与右边硬币的 $vals$ 的乘积，作为本次操作得分累加到总得分里。最后我们把选中的硬币移除，剩下的硬币相对位置不变。求期望总得分。

数据范围： $n \leq 300$, $1 \leq vals_i \leq 1000$

3 算法介绍

如果直接记录剩余哪些硬币，状态数会很大很复杂。所以要考虑如何“化整体为局部”，把大的整体的问题分解为小的局部的问题，用较简单的状态表示。由期望的线性性，我们可以分别计算每两个硬币之间产生贡献的期望，再累加起来。由于对于两个硬币的权值固定，只要求两个硬币产生贡献的概率。这样，我们就把问题转化为每两个硬币之间的问题，大大简化了状态。

考虑DP。定义 $f[i][j][k]$ 表示剩余 i 个硬币、第 j 个和第 k 个硬币产生贡献的概率。那么很显然，我们进行下一轮操作时选中的硬币可能在三段区间内： $[1, j-1]$, $[j+1, k-1]$, $[k+1, i]$ ，每段区间的概率也可以用预处理 $prob$ 的0到 n 次方来 $O(1)$ 计算，分别转移到 $f[i-1][j-1][k-1]$, $f[i-1][j][k-1]$, $f[i-1][j][k]$ 。最后如果 $j-i=1$ ，就把 $f[i][j][k]$ 设定为1。

时间复杂度: $O(n^3)$