

# COCOTTT 命题思路

清华大学 陈俊锐

2018 年 2 月 12 日

此文件仅作为上报命题思路之用，切勿外传

## 1 题意简述

给出一棵  $n$  个节点的无根带权树，每个点有一个权值  $a_i$ 。求树上所有大于  $k$  的连通块的第  $k$  大权值的和。模任意大质数。

$n, k, w = \max(a_i) \leq 2500$ ，时间限制 5s，空间限制 512MB。

## 2 算法简介

最直观的想法为将问题转化为求第  $k$  大权值不小于  $x \in [1, w]$  的连通块个数，转化为背包问题（重量为  $[a_i \geq x]$ ），复杂度为  $O(n^2w)$ 。

一种非常暴力的优化是用 FFT 优化这个背包，复杂度为  $O(nw \log^3 n)$ 。这个算法根本不可写，也太慢。

一个好一些的优化是用动态 DP 去优化这个 DP，按从小到大的顺序往树上加入点，使用动态 DP 高效维护背包数组，复杂度为  $O(nw^2 \log^2 n)$ 。如果用 FFT 优化背包卷积可以做到  $O(nw \log^2 n \log w)$ ，但是注意到背包所有的运算只有乘法和加法，因此可以在一开始进行 DFT，中间用 DFT 的结果  $O(n)$  运算，最后再把结果加起来 IDFT 回去，就可以省掉一个大常数  $O(\log n)$ 。并且注意到这里可以用  $O(n^2)$

的时间进行 DFT 和 IDFT，所以把它们换成普通的暴力求值和暴力插值也是可行的。当然，这个算法也是过不了的。

满分算法是这样的：注意到对于枚举的每个  $x$ ，第  $k$  大权值不小于  $x \in [1, w]$  的背包转移的形式全部相同，因此可以考虑将它们一起做。可以这么理解：我们把枚举的  $x$  记录到状态中，记录  $f(i, x, k)$  表示在  $i$  子树中权值大于等于  $x$  的节点有  $k$  个。

我们用动态新建节点的线段树维护  $f(i, *, *)$  这个二维数组的第一维。我们需要支持：对应位置相乘（卷积）、所有位置的  $f(i, *, 0)$  加上 1 以及对所有  $x \leq a_i$  的  $f(i, x, *)$  的数组添加一个大小为 1 的必选物品（相当于背包数组后移一位）。第一个操作是线段树合并的标准操作；后两个操作为区间修改，都是线段树的标准操作；因此可以使用线段树合并来优化这个 DP。同样使用前面的暴力求值和暴力插值将卷积优化到  $O(n)$ ，我们就可以以  $O(n^2 \log w)$  的复杂度通过本题。

本题考察了线段树合并的灵活运用、多项式的思想和简单的数学推导，是一道 NOI 较难的题目。正确实现本题满分算法需要两个小时左右的时间，可以作为省选中的压轴题。