树上 01 背包*

张晴川
qzha536@aucklanduni.ac.nz

July 3, 2020

1 问题

给一棵 N 个点有根树，每个点 i 有一个物品，体积为 C_i，价值为 W_i。如果选第 i 个物品，必须同时选他的父亲。给定背包容量上限 V，求最大价值。

复杂度要求 $O(NV)$

2 定义

泛化物品 一个泛化物品 f 是一个定义域为 $\{0, 1 \ldots V\}$，值域为实数的函数，也可以看成是长度为 $V+1$ 的数组。

泛化物品的并 给定两个泛化物品 f 和 g，定义泛化物品的并 $(f+g)$ 为 $(f+g)[i] = \max(f[i], g[i])$。

泛化物品的组合 给定两个泛化物品 f 和 g，定义泛化物品的组合 $(f*g)$ 为 $(f*g)[i] = \max_{0 \le j \le i}(f[j] + g[i-j])$。

3 一些特殊泛化物品

加法单位元 0 定义 $\mathbf{0}$ 为 $[-\infty, -\infty, \ldots, -\infty]$，即在所有位置都取 $-\infty$。

乘法单位元 1 定义 $\mathbf{1}$ 为 $[0, -\infty, \ldots, -\infty]$，即在 0 以外都取 $-\infty$。

指示函数 $I_{C,W}$ 定义指示函数 $I_{C,W}$ 为 $[-\infty, \ldots, W, \ldots, -\infty]$，即只 C 处取 W，其余取 $-\infty$。表示一个体积为 C，价值为 W 的物品。

*更多内容请访问：https://github.com/SamZhangQingChuan/Editorials

4 性质

加法单位元 对于任意物品 f，$f + \mathbf{0} = f$。

证明：由定义显然。

乘法单位元 对于任意物品 f，$f * \mathbf{1} = f$。

证明：由定义显然。

交换律 对于任意物品 f, g，$f + g = g + f$，$f * g = g * f$。

证明：由定义显然。

结合律 对于任意物品 f, g, h，$(f+g)+h = f+(g+h)$，$(f*g)*h = f*(g*h)$。

证明：由定义显然。

分配律 对于任意物品 f, g, h，$(f + g) * h = f * h + g * h$。

感性理解：先从 f, g 里面选优的再与 h 组合等价于先分别求 f, g 与 h 的组合再挑优的。

证明：

$$
\begin{aligned}
((f+g)*h)[i] &= \max_{j=0}^{i}((f+g)[j] + h[i-j]) \\
&= \max_{j=0}^{i}(\max(f[j], g[j]) + h[i-j]) \\
&= \max_{j=0}^{i}(\max(f[j], g[j]) + h[i-j]) \\
&= \max_{j=0}^{i}(\max(f[j] + h[i-j], g[j] + h[i-j])) \\
&= \max(\max_{j=0}^{i}(f[j] + h[i-j]), \max_{j=0}^{i}(g[j] + h[i-j])) \\
&= \max((f*h)[i], (g*h)[i]) \\
&= ((f*h) + (g*h))[i]
\end{aligned}
$$

5 算法

设以 r 为根的子树对应的泛化物品为 dp_r，即 $dp_r[C]$ 等于这棵子树中总体积恰好为 C 时的最大价值。这里我们规定必须要拿根 r 对应的物品，即 $dp_r[i < C_r] = -\infty$。假设 r 的儿子们分别为 $\{s_1, s_2, \ldots\}$。

那么不难发现如下关系：

$$dp_r = I(C_r, W_r) * (1 + dp_{s_1}) * (1 + dp_{s_2}) * \cdots$$

由于计算组合每次是 $O(V^2)$ 的，总复杂度为 $O(NV^2)$，无法通过。

现在考虑如何增量加上一个儿子的贡献：$f' = f * (1 + dp_s)$，其中 f 的初值为 $I(C_r, W_r)$，即只选根节点。

展开可以得到如下结果：

$$f' = f * (1 + dp_s) = f * 1 + f * dp_s = f + f * dp_s$$

f 是已知量，所以只需要求出 $f * dp_s$。

利用乘法的结合律，我们可以换一个形式：

$$\begin{aligned} f * dp_s &= f * (I(C_s, W_s) * (1 + dp_{s'_1}) * (1 + dp_{s'_2}) * \cdots) \\ &= (f * I(C_s, W_s)) * (1 + dp_{s'_1}) * (1 + dp_{s'_2}) * \cdots) \end{aligned}$$

这相当于把 s 的初值从 $I(C_s, W_s)$ 替换为 $f * I(C_s, W_s)$，由于 $I(C_s, W_s)$ 形式简单，这相当于强行在 f 中加入一个物品。这一步是 $O(V)$ 的，于是每多一个节点，复杂度只会多 $O(V)$。因此总复杂度是 $O(NV)$。

在 DFS 计算完 $f * dp_s$ 之后，通过 $O(V)$ 的时间更新 $f' = f + f * dp_s$ 即可。

最终答案为 $dp[root]$ 中的最大值。

6 代码

HDOJ 3593