

Simple Problem 解题报告

杭州学军中学 吉如一

1 试题来源

原创。题目可以在这里找到：<http://acm.hdu.edu.cn/showproblem.php?pid=5331>

2 试题大意

给出一棵树，最开始这棵树只有1个编号为0的节点（根节点），接下来进行 n 次操作，第 i 次操作给出一个整数 f_i ($0 \leq f_i < i$)，表示新建一个编号为 i 的点并在它和 f_i 之间连上一条边。

在每一次操作结束之后，你需要输出当前树的最大独立集，即最大化选出的节点数使得所有选出的节点之间两两不存在边。

对于所有数据，有 $n \leq 10^5$

3 算法介绍

3.1 算法一

因为树也是二分图，所以我们可以用二分图最大独立集的算法来求解树的最大独立集。

因为二分图中最大独立集大小等于 n 减去最大匹配，所以我们可以用匈牙利算法来求解每一次询问。

时间复杂度 $O(n^3)$ 。

3.2 算法二

对于树的最大独立集，有一种经典的DP方法。

只考虑 i 所在的子树时，令 f_i 为选了 i 这个节点时的最大独立集， g_i 为不选 i 这个节点时的最大独立集，那么就有转移（其中 j 是 i 的孩子）：

$$\begin{aligned} g_i &= \sum \max(f_j, g_j) \\ f_i &= \sum g_j + 1 \end{aligned}$$

所以对每次询问只需要进行一次 $O(n)$ 的树形DP就行了。
时间复杂度 $O(n^2)$ 。

3.3 算法三

回到算法一的思路，我们先把问题转化成求树上匹配数——这也是有经典的DP方法的：

只考虑 i 所在的子树时，令 f_i 为 i 这个节点已经被匹配时的最大匹配数， g_i 为 i 这个节点没有被匹配时的最大匹配数，那么就有转移（其中 j 是 i 的孩子）：

$$\begin{aligned} g_i &= \sum f_j \\ f_i &= g_i - \min(f_j - g_j) + 1 \end{aligned}$$

这个DP的好处在于，它满足条件 $0 \leq f_i - g_i \leq 1$ ，而最大独立集的DP不满足这一个性质。

证明：这一个不等式对叶子节点显然成立。而对其他节点，考虑它所在的子树，对于每一个匹配了根节点的方案，我们把根节点的匹配从方案中删去，答案最多会减少1，所以有 $f_i - g_i \leq 1$ ；对于每一个没有匹配根节点的方案，如果它的孩子中存在没有被匹配的节点，我们可以直接匹配这两个节点，答案增加了1且根节点被匹配了，否则我们任选一个儿子节点，删去它原来的匹配并把它和根节点匹配，这样答案不变，所以有 $f_i - g_i \geq 0$ 。

于是我们令 $F_i = f_i - g_i$ ，那么就有 $F_i \in [0, 1]$ ，同时有：

$$\begin{aligned} g_i &= \sum g_j + \sum F_j \\ f_i &= g_i - \min F_j + 1 \end{aligned}$$

这样我们就得到了 F 的递推式： $F_i = 1 - \max F_j$ 。

因为所有叶子节点的 g_i 都等于0，所以答案就是 $\sum_{i=1}^n F_i$ 。

到此，问题就转化为了：一棵树，叶子节点的权值为0，如果一个点的所有孩子权值都是1那么它的权值就是0否则它的权值是1。一棵树的权值定义为它的所有节点的权值和。现在每一次加一个叶子，输出整棵树的权值。

如果记录下来每一个节点的权值为0的孩子的个数 A_i ，那么这个节点的权值就是 $[A_i > 0]$ 。考虑加入一个深度为奇数的节点 i ，如果它的祖先 j 的权值发生了变化，当且仅当 i 到 k 的路径上所有深度为偶数的节点 p 都有 $A_p = 0$ ，所有深度为奇数的节点 q 都有 $A_q = 1$ 。插入节点深度为偶数的情况也是大同小异。

所以可以先离线对整棵树树链剖分，在线段树上分别维护深度为奇数，深度为偶数时 A 的最大值，这样在插入一个节点的时候，我们就可以通过在线段树上二分，在 $O(\log^2 n)$ 的时间复杂度内求出权值会发生变化的节点区间。同时，因为最后的答案就是所有节点的权值和，所以只要在线段树上维护一下区间和就行了。

综上，我们得到了一个 $O(n \log^2 n)$ 的做法。同时不难发现，这一个做法同样适用于每一次询问某一个子树的最大独立集——只需要按照在树链剖分的时候记下DFS序，然后对每一个点记录DFS序的区间，这样询问就变成了查询区间和。

3.4 数据构造方法

因为这道题是出到多校联合训练上的，所以不用太关注单个测试点的强度，所以大数据中可以放一个深度 $O(n)$ 的点，一个度数 $O(n)$ 的点以及一个随机的数据。再配合上100个左右的随机小数据，数据强度就已经足够了。

3.5 总结

这道题的难度比较高，在多校赛场上仅有绍兴一中一支队伍通过了这题。

解决这题，首先需要运用一些图论知识把最大独立集问题转化为最大匹配问题，然后要联想到传统的树形DP方法：因为一个很自然的思路是直接上数据结构维护增广路，这样的话问题就变得非常繁琐难以解决。接着要发掘一些关于DP转移的性质，然后再最后转化成一个并不简单的树链剖分问题。

总体来说这道题和目前大部分数据结构题不同，兼备了思维难度和代码难度，题目质量还是比较高的。