人类补完计划 解题报告

安徽师范大学附属中学 罗哲正

1 试题来源

UOJ Round 14 B.人类补完计划

链接：http://uoj.ac/contest/28/problem/193。

2 试题大意

给出一张n个点m条边的基因图，对于一个边集的子集S_e，我们把所有与S_e中**至少**一条边相邻的点形成的集合称为S_v，S_e被称为**好的**当且仅当：

1. $|S_e| = |S_v|$。

2. 对于S_v中不同的两个点x和y，存在一条从x到y的路径使得路径中的边都属于S_e。

设所有与S_e中**大于**一条边相邻的点的数量为w，那么这个边集的**权值**为2^w。一张基因图的权值是所有**好的**边集的权值和。

给出一张无向图求这张基因图的权值和对998244353取模的值。

2.1 限制与约定

每组测试数据的限制与约定如下所示：

<table>
<tr><th>测试点编号</th><th>n的规模</th><th>m的规模</th></tr>
<tr><td>1</td><td rowspan="10">$n \le 11$</td><td rowspan="4">$m \le 20$</td></tr>
<tr><td>2</td></tr>
<tr><td>3</td></tr>
<tr><td>4</td></tr>
<tr><td>5</td><td rowspan="6"></td></tr>
<tr><td>6</td></tr>
<tr><td>7</td></tr>
<tr><td>8</td></tr>
<tr><td>9</td></tr>
<tr><td>10</td></tr>
<tr><td>11</td><td rowspan="4">$n \le 13$</td><td>$m = \frac{n(n-1)}{2}$</td></tr>
<tr><td>12</td><td rowspan="3"></td></tr>
<tr><td>13</td></tr>
<tr><td>14</td></tr>
<tr><td>15</td><td rowspan="6">$n \le 14$</td><td>$m = \frac{n(n-1)}{2}$</td></tr>
<tr><td>16</td><td rowspan="5"></td></tr>
<tr><td>17</td></tr>
<tr><td>18</td></tr>
<tr><td>19</td></tr>
<tr><td>20</td></tr>
</table>

对于所有数据，$1 \le n \le 14, 0 \le m \le \frac{n(n-1)}{2}$。

$1 \le x_i, y_i \le n$，$x_i \ne y_i$，两个点之间最多只有一条边。

时间限制：1s

空间限制：256MB

3 算法介绍

3.1 算法1

对于$m \le 20$的情况，我们可以暴力枚举每条边在不在S_v里，并判断是否符合题目要求，把合法的方案的权值加入答案中。

时间复杂度$O(m2^m)$，期望得分20分。

3.2 算法2

观察题意其实就是枚举所有的基环外向树，w是非叶节点个数，把2^w加入答案。

那么我们首先考虑基环外向树的计数，第一步是环的计数。环的计数可以非常容易的通过状态压缩DP来实现设$cir_{i,S}$表示集合S组成一条以j开头i结尾的链的方案数（令j为S中编号最小的元素），转移时枚举i的下一个元素即可，要求和i直接有边且编号$> j$，最后对于每对i, S，若i, j之间由有边，则将$cir_{i,S}$加入集合S的答案f_S中。注意一个环从j开始有两种不同的顺序遍历，所以最后答案要除以2。

接下来考虑集合V组成的基环外向树计数cnt_V，我们可以枚举环的集合S，然后把环缩成点，直接套用Matrix-Tree定理即可，这样处理所有V的总复杂度是$O\left(3^n n^3\right)$的。

我们考虑给权值一个意义，假设每个节点可以染成黑白两种颜色，叶子节点只能染成白色，那么只要计算染色的总方案数，一个S_e就会被恰好计算w次。

对于染色方案数的计算，我们可以考虑容斥原理，用总方案数减去存在黑色叶子节点的方案数，注意到基环外向树删去叶子之后仍然是基环外向树。

令$val_V = 2^{|V|} cnt_V$，枚举叶子节点集合T，则方案数

$$g_V = \sum_{T \subset V, T \neq \emptyset} (-1)^{|T|} val_{V-T} ways(T, V-T)$$

其中$ways(T, S)$表示T向S连边的方案数，为T中所有节点在S中邻接点个数之积，容斥时暴力计算就好，这部分时间复杂度是$O(3^n n)$的。

于是这个算法总的复杂度就是$O\left(3^n n^3\right)$，期望得分50分。

3.3 算法3

算法2的时间复杂度瓶颈在与求cnt_V上，我们考虑换一种计数方式，把cnt_V的计算和答案的计算放在一起。

首先考虑V集合的权值和计算，对于V的每一个子集T，我们都计算出叶子节点集合为T的基环外向树个数g_T，那么权值和就可以直接计算了。

依旧使用容斥原理，令$h_T = cnt_{V-T} ways(T, V-T)$，即$T$为叶子节点集合的子集的方案数，然后对$h$做子集反演得到$g$：

$$g_S = h_S - \sum_{S \subset T} (-1)^{|T|-|S|} h_T$$

那么显然V集合对答案的贡献就是$2^{|V|} f_V + \sum_{S \subset V, S \neq \emptyset} 2^{|V|-|S|} g_S$。

接下来考虑cnt_V如何计算，对于每个V首先把单独为一个环的情况f_V加入答案，然后既然我们已经求出了以任意集合S为叶子的基环外向树方案数g_S，直接把cnt_V加上所有g_S的和即可。

计算h_T的总复杂度是$O(3^n n)$的，子集反演可以使用$O(2^n n)$的快速算法，总复杂度也是$O(3^n n)$，期望得分70分。

3.4 算法4

我们采用一种新的思考方式，考虑树的purfer编码，每次选取最小的一个叶子删去并将其邻接点加入purfer序列。

具体令$dp_{V,S}$表示V集合组成的S不能为叶子的基环外向树数目。

转移时我们令k为$V - S$中编号最小的节点，我们有两种转移方式：

1.令k为叶子，并将k删去，那么我们在k的邻接点中选取一个节点l，将l设置为可以为叶子。

2.将k设置为不可为叶子。

当$V = S$时可以发现答案一定为一个环，即$dp_{V,S} = f_V$。

注意到这样的统计会有重复计数，因为对于一个基环外向树，令其点集为V'，非叶节点集合为S'，那么对于所有$V = V', S \subseteq S'$都会被$dp_{V,S}$计数一次，共会被计数$2^{|S'|}$次，我们发现这恰好就是这个基环外向树的权值，那么对所有的$dp_{V,S}$求和即可。

时间复杂度也是$O(3^n n)$，期望得分70分。

3.5 算法5

考虑对算法2进行优化，算法2的瓶颈在于计算集合V构成的环套树的方案数，而这个方案数在算法3中我们可以枚举叶子节点的集合进行容斥，而算法3的瓶颈在于莫比乌斯反演。

我们考虑把算法2和算法3结合起来，我们直接枚举**非叶节点**集合S，在枚举S的超集V，枚举V可以使用dfs，在dfs的过程中计算$ways(V-S,S)$，使用如下容斥式子计算答案：

$$g_V=\sum_{S\subset V}(-1)^{|V|-|S|}val_S\,ways(V-S,S)$$

dfs之后统计S对于V的贡献即可，复杂度是$2^{n-|S|}$的，总复杂度就是$O(3^n)$了。

4 总结

这是一道n很小的计数题，这种计数题一般都是采用集合DP并结合容斥等方法，枚举V的叶子节点子集S进行容斥时本题的关键，而子集反演的复杂度n要想省去，需要放弃枚举V再枚举子集S，转而先枚举S再枚举超集V，枚举超集使用dfs可以省掉一个n的复杂度，相当于把很多容斥放在一起计算了。

本题所有的容斥计数方法都较为常见，但是要将这些方法灵活运用，结合起来解决本题，需要较强的计数能力和模型的转换能力。