

复原

卓亮

清华大学

April 8, 2013

目录

1 题目大意

2 自由讨论

3 参考解答

题目大意

假设有一个圆，圆上有 n 条弦。现在给出了每对弦是否相交。相交的弦一共有 m 对。要求给出一个合法的弦的排列，以及求出最多有多少条弦，它们两两不交。

另外，数据均为随机生成。

自由讨论

得分情况

最高分：100

最低分：0 (15人)

平均分：32(全) 50.4(集训队)

得分分布：100, 95, 90, 85, 80, 75, 60, 55, 50, 40, 35, 30, 25, 20, 15, 10, 5, 0

高分选手：陈立杰 胡泽聪 张闻涛 杜卓帆 王悦同 王玉 冯冠宇
梁泽宇 乔明达 张耀楠 黄施霖

问题分析

我们可以建立一张无向图，把弦看成图中的点，而相交情况看成图中的边。

第一问是本题的重点。

最直接的做法就是搜索了。注意到实际上只需要搜索一个每个数都恰好出现2次的圆排列，因而这样做的时间复杂度是 $O(\frac{n^2 \cdot (2n)!}{2n \cdot 2^n})$ ，期望得分10分。

搜索方法

我们可以每次插入一条弦，如果非法就剪枝。容易想到，如果图是有多个连通块，我们可以分别搜索，然后合并起来。为了让剪枝效果明显，最好每次插入的弦跟之前的弦有关联。

另外一种搜索方法是插空。例如，弦的两个端点直接夹的位置的个数，要与与它相交的弦的数目的奇偶性相同。还有，如果还未摆放的弦，无论如何都摆放不了，那么剪枝。

对直接摆放的思考

我们考虑抓住一条弦，把它摆在圆上。凡是与它相交的弦先无视。剩下的弦可能会形成若干连通块，我们递归处理每个连通块，最后，考虑把结果合并？

另一种划分和合并的方法

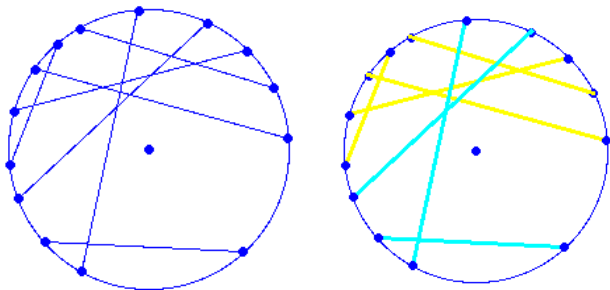


Figure : 弦束的划分

另一种划分和合并的方法

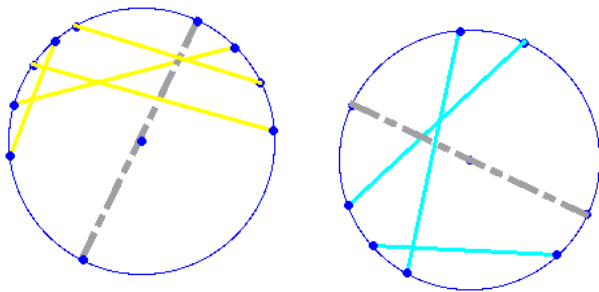


Figure : 分为两个子问题

拆分

我们实际上是要把连通图 $G = (V, E)$ 的点集 V ，划分成4个集合 V_0, V_1, V_2, V_3 。其中， V_0 只与 V_0, V_1 有边， V_1 只与 V_0, V_1, V_2 有边， V_1 与 V_2 两两有边。 V_2 只与 V_1, V_2, V_3 有边， V_3 只与 V_2, V_3 有边。

拆分

我们枚举必在 V_1 的一个点 x ，必在 V_2 的一个点 y 。显然 $(x, y) \in E$ 。接着随便选一个异于 x, y 的点。虽然这个点有可能在 $V_0 \cup V_1$ 或者 $V_2 \cup V_3$ 中，但是只要对这两种情况枚举一遍就可以了。因而不妨设这个点在 $V_0 \cup V_1$ 中。其余所有点都放在 $V_2 \cup V_3$ 中。接下来，枚举每个在 $V_0 \cup V_1$ 中，且异于 x 的点 s 。考虑两种情况。

拆分

$(s, y) \notin E$, 这说明 $s \in V_0$ 。因而凡是与 s 相邻的点都要移动到 $V_0 \cup V_1$ 中。

$(s, y) \in E$, 这说明 $s \in V_1$ 。考虑点 t , 如果 $(x, t) \in E$ 但 $(s, t) \notin E$, 说明 $t \in V_0 \cup V_1$ 。如果 $(x, t) \notin E$ 但 $(s, t) \in E$, 也说明了 $t \in V_0 \cup C_1$ 。

拆分

我们不断用此性质把点从 $V_2 \cup V_3$ 添加到 $V_0 \cup V_1$ 。直到无法添加为止。如果 $|V_2 \cup V_3| > 1$ ，说明我们找到了一种好的划分。我们就采用这种划分方式。这样每部分规模变小了。否则换一组 x, y 。如果对所有 x, y 均没有好的划分，我们就只好采用搜索了。

第一问结果

如果不考虑搜索的部分，复杂度大概是 $O(n^2m^2)$ 。对于随机数据，这个算法运行得比较快。

第二问

有了第一问，第二问就容易了。这实际上是一个区间动态规划的问题。我们用 $F(i, j)$ 表示区间 $[i, j]$ 最多能选几条互不相交的弦。那么转移无非以下几种。如果 i 和 j 是同一条弦的两个端点，那么 $F(i, j) = F(i + 1, j - 1) + 1$ 。否则，我们枚举断点 k ，则 $F(i, j) = \max F(i, k) + F(k + 1, j)$ 。时间复杂度是 $O(n^3)$ 。

第二问

注意 i 和 j 不同属一条弦的情况。可能有以下两种情况。第一种是，放弃一个端点。第二种是取一个端点所对的一条弦，问题分为两个部分。如果我们用 $other(i)$ 表示 i 对应的弦的另一个端点，那么上述讨论可以写

成， $F(i, j) = \max\{F(i+1, j), F(i, j-1), F(i, other(i)) + F(other(i)+1, j), F(i, other(j)-1) + F(other(j), j)\}$ 。这里假设了 $i < other(i), other(j) < j$ 。如果不成立，则不能由该情形转移。时间复杂度是 $O(n^2)$ 。

第二问

我们考虑补图，问题变为求最大团。一般图求最大团，最常用的就是搜索了。如果直接枚举团是由哪些点构成，结合位运算优化，时间复杂度是 $O(2^n)$ ，期望得分40分。另外，也可以考虑对问题折半，用DP+搜索来解决，时间复杂度是 $O(n \cdot 2^{n/2})$ ，期望得分40分。不过这些做法都不是多项式的。

谢谢！