图与网络 2020 春

构造法编程报告

周四班 xx 019xxxxxxxxx

2020 年 5 月

一、 问题描述

证明：对有 n 个点的完全图K_n的每条边进行二染色，证明存在一种染色方法，使得染色后的完全图中至多含有$\binom{n}{4}2^{-5}$个同色的K_4。

二、解题思路

1、概率方法证明

一个含有 4 个点的完全图K_4共有 6 条边，每条边都有两种染色的选择，所以这 6 条边同色的概率为$2^{-6}\times 2 = 2^{-5}$，又因为完全图K_n中共有$\binom{n}{4}$个K_4，所以同色K_4数目的数学期望可以表示为$E(NumK_4) = \binom{n}{4}2^{-5}$，所以必定可以找到一种染色方法，使得染色后的完全图中至多含有$\binom{n}{4}2^{-5}$个同色的K_4，得证。

2、去随机化构造

首先，需要对任意一个已经部分染色的(partial colored)完全图K_n定义一个权重函数W。对于K_n中每一个K_4，都有一个权重函数$W(K)$，分为 3 种情况：

① 至少有一条边染成了红色，还有一条边染成了蓝色，这种情况下该K_4最后成为同色K_4的概率为$W(K) = 0$。

② 该K_4中没有一条边被染过色，则该K_4最后成为同色K_4的概率为$W(K) = 2^{-5}$。

③ 如果有$r \geq 1$条边已经被染过色，且都是红色或者都是蓝色，则该K_4最后成为同色K_4的概率为$W(K) = 2^{r-6}$。

根据数学期望的线性性，所有权重之和$W = \sum W(K)$，就是最后染完色后的K_n中所包含同色K_4数目的数学期望$E(NumK_4)$。

假设边$e_1, e_2, \cdots, e_{i-1}$已经被染过色，现在对边$e_i$进行染色，有两种染色选择，如上所述，会出现 3 种情况，因为题目的目标是使最后出现同色K_4的数目最小，所以需要选择一种颜色，使得将e_i染为该色后最后同色K_4数目的数学期望最小。记W_{red}为对边e_i染红色后完全图K_n最后包含同色K_4数目的数学期望，同理记W_{blue}为对边e_i染蓝色后完全图K_n最后包含同色K_4数目的数学期望，根据W的定义，可以得到$W = \frac{1}{2}(W_{red} + W_{blue}) \leq \max\{W_{red}, W_{blue}\}$，所以随着染色的进行，权重函数的值一直不会增加，必有$W_{final} \leq W_0$，而初值W_0就是数边e_0染色后所获得的数学期望，为$W_0 = \binom{n}{4}2^{-5}$，所以$W_{final} \leq \binom{n}{4}2^{-5}$，即按照上述的染色策略进行染色，必定可使染色后的完全图中至多含有$\binom{n}{4}2^{-5}$个同色的K_4，这样便构造出一种染色策略满足题目要求。

三、编程实现

根据上述去随机化方法的解答，我们使用了 Python 语言对该问题进行了编程实现，本节仅对部分代码进行解析并展示最后结果，完整代码见附件中构造法代码 ConstructMethod.py 和随机化代码 RandomMethod.py。

构造法的代码请参见 ConstructMethod.py。函数 *CreateG(n)*的功能是构建一个 n 个点的完全图K_n，并将其每条边初始化染成白色，输入是阶数 n，输出是点 V 和图 G，分别用列表和字典形式表示。函数 *CreateK4(V, G)*的作用是找到图 G 中所有的完全图K_4，输入是点 V 和图 G，输出是该图中所有的K_4，用列表形式表示。函数 *ismonoK4(subG)*的作用判断子图 subG 是否是同色的，输入 subG 是一个K_4，返回一个布尔值，1 表示是同色的，0 表示不是同色的。函数 *CntmonoK4(V, G)*的功能是计数图 G 中同色K_4的个数，输入是点 V 和图 G，返 cnt 表示同色K_4的个数，monoG 存储了所有的同色K_4。函数 *CalWeight(subG)*的作用是计算 subG 的权重值，输入 subG 是一个K_4，返回该K_4的权值 weight。函数 *CalTotalW(V, G)*的作用是计算整个 partial colored Graph 的权重，即最后同色K_4数目的数学期望，输入是点 V 和图 G，输出是期望值 weight。最后，函数 *Iter(V, G)*是对 G 中所有的边$e_1, e_2, \cdots, e_{n\text{选}2}$逐一进行染色，输入是点 V 和图 G，输出是染色后的图 G。

根据上述编好的函数，我们使用构造法对该题进行解答。先从 n 较小的情况入手，当 n=4 时，结果如下：

```
n = 4时，通过构造法，找到一种染色方法使得完全图中共有0个同色的K4
找到一种染色方法使得同色的K4个数不大于n选4*2e-5: 0 <= 0.03125
time:0.0
```

图 1. n=4 时，构造法算得的结果

逐渐增大 n，当 n=10 时，结果如下：

```
n = 10时，通过构造法，找到一种染色方法使得完全图中共有0个同色的K4
找到一种染色方法使得同色的K4个数不大于n选4*2e-5: 0 <= 6.5625
time:0.0399632453918457
```

图 2. n=10 时，构造法算得的结果

可以看到，当 n=10 时用构造法仍然可以找到一种染色方法，使得染色后的完全图中不包含同色的K_4。

而当 n=11 时，得到结果：

```
n = 11时，通过构造法，找到一种染色方法使得完全图中共有1个同色的K4
找到一种染色方法使得同色的K4个数不大于n选4*2e-5: 1 <= 10.3125
同色的K4为: [{(0, 4): 'red', (0, 6): 'red', (0, 10): 'red', (4, 6): 'red', (4, 10): 'red', (6, 10): 'red'}]
time:0.07597017288208008
```

图 3. n=11 时，构造法算得的结果

可见当 n=11 时，即使每次染色都使期望最小，但是最后必定仍然要存在一个同色的K_4。

继续增大 n，当 n=20 时，可以得到结果如下：

```
n = 20时，通过构造法，找到一种染色方法使得完全图中共有61个同色的K4
找到一种染色方法使得同色的K4个数不大于n选4*2e-5: 61 <= 151.40625
同色的K4为: [{(0, 2): 'red', (0, 6): 'red', (0, 10): 'red', (2, 6): 'red', (2, 10): 'red', (6, 10): 'red'},
time:5.2239766120910645
```

图 4. n=20 时，构造法算得的结果

n=30 时，结果如下：

```
n = 30时，通过构造法，找到一种染色方法使得完全图中共有509个同色的K4
找到一种染色方法使得同色的K4个数不大于n选4*2e-5: 509 <= 856.40625
同色的K4为: [{(0, 2): 'red', (0, 4): 'red', (0, 6): 'red', (2, 4): 'red', (2, 6): 'red', (4, 6): 'red'},
time:73.3657865524292
```

图 5. n=30 时，构造法算得的结果

都得到了比较好的结果，当 n 增大到 100 时，即题目中要求的大小，在运行较长的一段时间后得到结果：

```
E:\Anaconda3\envs\bnq\python.exe F:/课件/图与网络/Homework1/ConstructMethod.py
n = 100时，通过构造法，找到一种染色方法使得完全图中共有106845个同色的K4
找到一种染色方法使得同色的K4个数不大于n选4*2e-5: 106845 <= 122538.28125
```

图 5. n=100 时，构造法算得的结果

所以，由编程的结果可得，可以找到一种染色方法，使得 100 阶的完全图K_{100}经过二染色后，得到染色后的图中包含同色K_4的个数为 106845 个，小于题目中设定的期望值 122538.28125，因为得到的同色K_4过多，不便在这里展示具体的染色方案，考虑 n=100 点边太过密集，也没有进行作图进行展示。

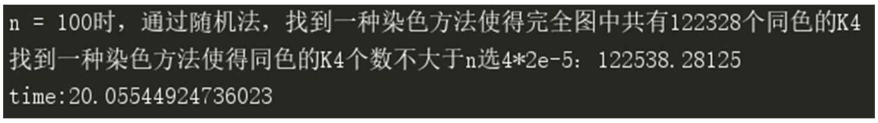
由于构造法生成结果的时间较长，为了尽快获得一个满足条件的染色方案，我们还采用了随机化的方法，即对一个完全图K_n的所有边进行随机染色，具体代码参见 RandomMethod.py。调用 random 包生成 0-1 的随机数，大于 0.5 染红色，否则染蓝色。使用随机法很快就可以得到结果，如下是采用随机法在 n=10，11，20，30 和 100 时获得的结果：

```
n = 10时，通过随机法，找到一种染色方法使得完全图中共有1个同色的K4
找到一种染色方法使得同色的K4个数不大于n选4*2e-5: 6.5625
time:0.0009996891021728516
```

```
n = 11时，通过随机法，找到一种染色方法使得完全图中共有9个同色的K4
找到一种染色方法使得同色的K4个数不大于n选4*2e-5: 10.3125
time:0.0009996891021728516
```

```
n = 20时，通过随机法，找到一种染色方法使得完全图中共有121个同色的K4
找到一种染色方法使得同色的K4个数不大于n选4*2e-5: 151.40625
time:0.01798391342163086
```

```
n = 30时，通过随机法，找到一种染色方方法使得完全图中共有779个同色的K4
找到一种染色方法使得同色的K4个数不大于n选4*2e-5: 856.40625
time:0.08593153953552246
```



n = 100时，通过随机法，找到一种染色方法使得完全图中共有122328个同色的K4
找到一种染色方法使得同色的K4个数不大于n选4*2e-5：122538.28125
time:20.05544924736023

图 6. n=10, 11, 20, 30, 100 时，随机法算得的结果

从结果可以看出，虽然使用随机法可以很快地得到一种满足要求的染色方法，但是使用构造法获得的染色方法，使得染色后的完全图包含同色K_4的数目更小。

经过这一次的编程作业，加深了我对概率方法中去随机化思想的理解，并使我更加熟悉了有关图的编程思想。