Solutions to Some Problems in Lecture 3

范舟
516030910574
zhou.fan@sjtu.edu.cn
上海交通大学 ACM 班

1 随机 Petersen Graph 连通的概率

下面的问题是吴老师在 Lecture 3 中留的思考题.

1.1 问题描述

如图 1，图中每条边有 p 的概率断开，且每条边是否断开的事件是独立的，求整个图连通的概率.

评注 事实上，图 1中的图即为 Petersen Graph，它有 10 个节点，15 条边，每个节点的度数是 3. 在图论的很多问题中，Petersen Graph 提供了一个很好的例子或反例.

1.2 解法

注意到图中共有 15 条边，每条边都有连接/断开两种状态，因此整个图中边的连通性共有 2^{15} 种可能的组合. 对于任意一种组合的情况，若有 k 条边是连接（即未断开）的，那么根据每条边断开事件的独立性，可得这种情况的发生概率为

$$P_k = (1-p)^k p^{15-k}$$

将所有使整个图连通的情况的发生概率相加，即可得到整个图连通的概率. 但可能的情况共有 2^{15} 种，手工通过这种方法计算是不现实的，但是我们可以使用计算机程序很容易地解决这个问题.

我编写了一个 C++ 程序，对于可能的 2^{15} 种情况判断整个图的连通性，并统计在使整个图连通的所有情况中，有 $k \quad (k=0,1,\ldots,15)$ 条边的组合各有多少种. 程序代码如下：

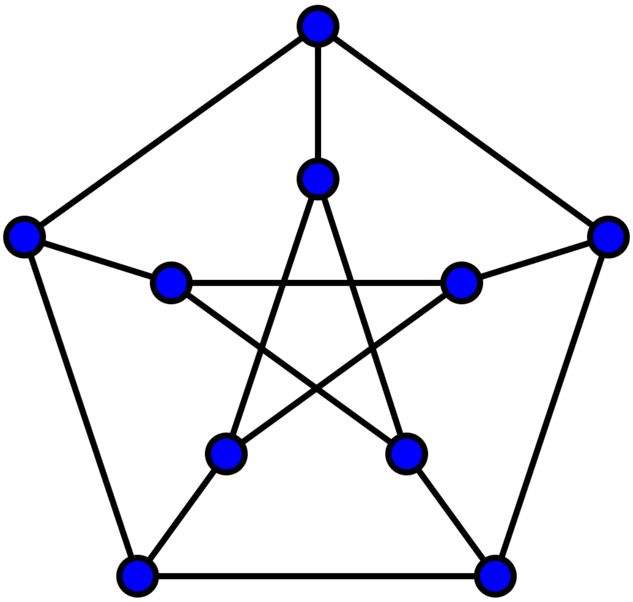

图 1: Petersen Graph

```
#include <iostream>
#include <cstdio>
#include <cstdlib>
#include <algorithm>

using namespace std;

const int NUM_VERTICES = 10, NUM_EDGES = 15;

// edges in the Petersen Graph
const int EDGES[NUM_EDGES][2] = {
    {0, 1}, {1, 2}, {2, 3}, {3, 4}, {4, 1},
    {5, 7}, {6, 8}, {7, 9}, {8, 5}, {9, 6},
    {0, 5}, {1, 6}, {2, 7}, {3, 8}, {4, 9}
};

int f[NUM_VERTICES], size[NUM_VERTICES], connected_case_num_edges[NUM_EDGES + 1];

int Find(int x) {
    int i, j, k;
    j = x;
    while (j != f[j]) j = f[j];
    i = x;
    while (i != j) {
        k = i;
        i = f[i];
        f[k] = j;
    }
    return j;
}

void Merge(int x, int y) {
    int fx = Find(x), fy = Find(y);
```

```
    if (fx == fy) return;
    if (size[fx] < size[fy]) swap(fx, fy);
    size[fx] += size[fy];
    f[fy] = fx;
}

int main() {
    for (int s = 0; s < (1 << NUM_EDGES); ++s) {
        // initialize the union-merge set
        for (int i = 0; i < NUM_VERTICES; ++i) {
            f[i] = i;
            size[i] = 1;
        }
        int edge_cnt = 0;
        for (int i = 0; i < NUM_EDGES; ++i) {
            // if this edge is broken
            if (!(s & (1 << i))) continue;
            // add this edge to the graph
            Merge(EDGES[i][0], EDGES[i][1]);
            ++edge_cnt;
        }
        // check if the graph is connected
        if (size[Find(0)] == NUM_VERTICES)
            ++connected_case_num_edges[edge_cnt];
    }
    for (int i = 0; i < NUM_EDGES + 1; ++i)
        cout << "[number␣of␣edges:␣" << i << "]\tnumber␣of␣connected␣cases:␣" <<
        connected_case_num_edges[i] << endl;
    return 0;
}
```

运行程序，得到程序的输出为：

```
[number of edges: 0]    number of connected cases: 0
[number of edges: 1]    number of connected cases: 0
[number of edges: 2]    number of connected cases: 0
[number of edges: 3]    number of connected cases: 0
[number of edges: 4]    number of connected cases: 0
[number of edges: 5]    number of connected cases: 0
[number of edges: 6]    number of connected cases: 0
[number of edges: 7]    number of connected cases: 0
[number of edges: 8]    number of connected cases: 0
[number of edges: 9]    number of connected cases: 1760
[number of edges: 10]   number of connected cases: 1994
[number of edges: 11]   number of connected cases: 1167
[number of edges: 12]   number of connected cases: 433
[number of edges: 13]   number of connected cases: 104
[number of edges: 14]   number of connected cases: 15
[number of edges: 15]   number of connected cases: 1


--------------------------------
Process exited after 0.3479 seconds with return value 0
```

从运行结果中，我们得到了每种情况的数量，将其记作 $N_k \quad (k = 0, 1, \ldots, 15)$，表示使整个图连通的所有可能情况中，包含 k 条边的组合共有 N_k 种. 可以看到当 $k < 9$

时，$N_k=0$，这表示要使整个图连通，至少需要 9 条边. 并且有

$$N_9=1760,\quad N_{10}=1994,\quad N_{11}=1167,\quad N_{12}=433,\quad N_{13}=104,\quad N_{14}=15,\quad N_{15}=1$$

整个图连通的概率即为

$$\begin{aligned}
P=&\sum_{k=0}^{15} N_k P_K\\
=&\sum_{k=0}^{15} N_k(1-p)^k p^{15-k}\\
=&1760(1-p)^9p^6+1994(1-p)^{10}p^5+1167(1-p)^{11}p^4+\\
&433(1-p)^{12}p^3+104(1-p)^{13}p^2+15(1-p)^{14}p+(1-p)^{15}
\end{aligned}$$

1.3 图像

使用 Matlab 程序画出整个图连通的概率随每条边断开的概率 p 变化的图像，得到图 2.

Matlab 代码如下：

```
p = 0 : 0.01 : 1;
prob_connected = f(p);
plot(p, prob_connected, 'LineWidth', 2);
xlabel('每条边断开的概率');
ylabel('整个图连通的概率');

function prob_connected = f(p)
    prob_connected = 1760 .* (1 - p).^9 .* p.^6 + 1994 .* (1 - p).^10 .* p.^5 + ...
        1167 .* (1 - p).^11 .* p.^4 + 433 .* (1 - p).^12 .* p.^3 + ...
        104 .* (1 - p).^13 .* p.^2 + 15 .* (1 - p).^14 .* p + (1 - p).^15;
end
```

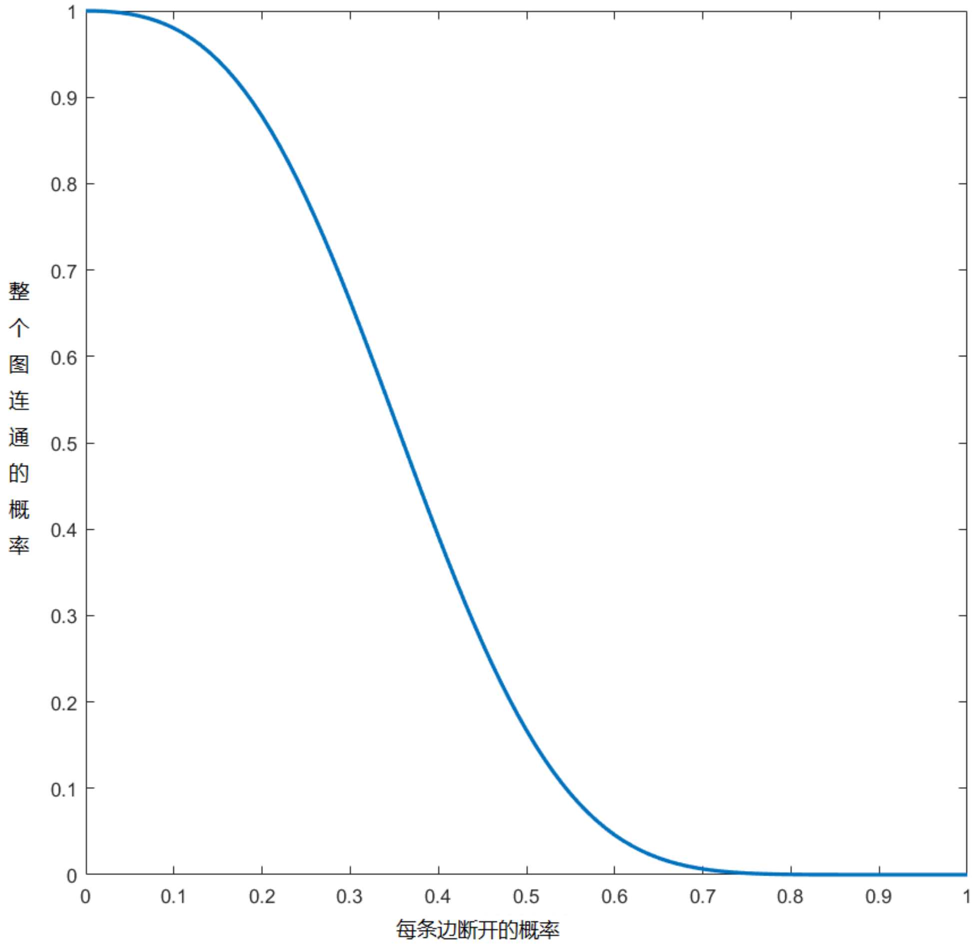

图 2: 整个图连通的概率随每条边断开的概率 p 变化的图像