山东大学 软件 学院

众智科学与网络化产业 课程实验报告

|  |  |  |  |
| --- | --- | --- | --- |
| 学号：202022300310 | 姓名： 张博宁 | | 班级： 软件20.7 |
| 实验题目：极化关系下网络结构的稳定性 | | | |
| 实验学时：3 | | 实验日期： 2022/6/10 | |
| 实验目的：  课程实验可以验证、巩固和补充课堂讲授的理论知识，是《众智网络和网络化产业》课程中一个重要的、不可或缺的实践环节。其目的是通过实验，使学生能够验证课程中所传授理论，并将理论联系实际解决问题。本课程本身就是一门交叉课程，课程实验可以训练学生利用本专业知识和所学习理论，综合运用经济学、社会学、计算与信息科学以及应用数学的有关概念与方法，讨论社会学和经济学中的计算思维，讨论各类互联网时代众智现象背后的内在机理或规律，培养学生用理工科的方法与工具来分析社会科学和经济学等问题，开阔学生视野，更好地面对未来的挑战。 | | | |
| 硬件环境：  处理器 AMD Ryzen 7 5700U with Radeon Graphics 1.80 GHz  机带 RAM 16.0 GB  系统类型 64 位操作系统, 基于 x64 的处理器 | | | |
| 软件环境：  Windows 11 家庭中文版  Embarcadero Dev-C++ Version 6.3  TDM-GCC 9.2.0 64-bit Release  Gephi 0.9 | | | |
| 实验步骤与内容：  **1、确认任务**  输入：任意极化关系下图的邻接矩阵（注意边有正负）  输出：是否含有负向边圈  **2、分析任务**  所谓“极化关系”，指的是网络中的关系分为“友好”和“敌对”两种，这在人际关系和国际关系的一些特定时期都是显著的。在这样的模型中，结构的稳定性是关注的重点，即一个网络结构中的各个关系性质是趋向于不变，还是趋向于改变（从友好变为敌对，或者反过来）？落实到计算问题上，就是要检测图中是否存在包含奇数个敌对边（负向边）的圈。  **3、代码编写**  数据生成上，与先前实验一样，仍然采用CRD参数控制密度，此实验的友好/敌对关系体现为边权为+1/-1，这两种边的比例使用pos\_div\_neg控制：   1. for(int i=0;i<V;i++) 2. for(int j=0;j<i;j++) 3. if(rand()%100<CRD\*100) 4. G[i][j]=G[j][i]=(rand()%((int)pos\_div\_neg+1)<pos\_div\_neg)?1:-1;   判断奇负环的算法基于如下思路：  1、将正边相连的点缩为一点，若某个因此和为一点的点集中含有负边，说明一定存在一个++-的环，算法结束。  2、对新图进行广度优先搜索，此时图中应只有负边。当且仅当同一层的节点有边相连时该图存在奇数负边环。如下图的第三层有一条内部边，存在奇数负边环BCEDA：    具体算法实现如下：  主体的bfs，对每个节点维护一个深度值：   1. for(int i=0;i<V;i++) 2. if(!d[find(i)]){ 3. q.push(i);d[i]=1; 4. while(!q.empty()){ 5. int o=q.front();q.pop();assemble(o); 6. for(int j=0;j<V;j++){ 7. if((G[o][j]<0&&d[find(j)]==d[find(o)])|| 8. (G[o][j]>0&&d[find(j)]&&d[find(j)]!=d[find(o)])){ 9. flag=1; 10. if(o<j)printf("conflict between %d and %d!\n",o+1,j+1); 11. } 12. } 13. } 14. }   首先，图不一定连通，所以有最外层的for循环应对非连通图。  对于深度优先到的每一个节点，先对其与周围的点进行缩点处理：   1. inline void assemble(int o){ 2. queue<int> qq; 3. qq.push(o); 4. while(!qq.empty()){ 5. int p=qq.front();qq.pop(); 6. for(int i=0;i<V;i++){ 7. if(G[p][i]>0&&!vis[i]){unite(i,p);qq.push(i);vis[i]++;} 8. if(G[p][i]<0&&!d[find(i)]){d[find(i)]=d[find(o)]+1;q.push(i);} 9. if(G[p][i]<0&&d[find(i)]==d[find(o)]){flag=1; 10. if(i<p)printf("conflict between %d and %d!\n",i+1,p+1);} 11. } 12. } 13. }   此过程将所有可以缩到的点赋为同一深度，并对其中出现的负边进行判断。  缩点之后，进行同一层的负边判断，即存在深度相同且与其有负边相连的点，就记录为不稳定。（另一个条件在缩点程序加入后为多余条件，可忽略）  **4、调试与结果分析**  即使使用了正负比4:1的参数，也发现较大的样例中，以随机的方式难以生成稳定的图，故采用小样例（事实上也难以找到稳定的情况），下面是几次稳定的图：      其中绿色为正边。可见他们的节点间，不可能存在奇数负边环。  他们的运行结果：    而更多是这种情况：    此时对应的图：    可见大点内部存在负边，必然存在诸如258这种引起不稳定的环。 | | | |
| 结论分析与体会：  经过本次实验，我熟练了“不稳定”的意义与判定方式，熟练了搜索代码的编写和调试技巧，提高了知识掌握的熟练度，以及运用可视化进行数据分析调试的能力。 | | | |

目录下有生成器与判断器，父目录下还有将邻接矩阵转化为Gephi接受形式的代码目录，编译后运行，与前几次实验相同。