MATLAB实例：构造网络连接图(Network Connection)及计算图的代数连通度(Algebraic Connectivity)

作者：凯鲁嘎吉 - 博客园 http://www.cnblogs.com/kailugaji/

1. 图的代数连通度(Algebraic Connectivity)

图的代数连通度：Laplace图谱的次小特征值。

$G=(V,E)$是一个简单图，其中 $V=\{v_1,\cdots,v_n\}$为点集，$E=\{e_1\cdots,e_m\}$为边集。若 d_i 为顶点 $v_i(i=1,\cdots,n)$的度，则 G 的 Laplace 矩阵为 $L(G)=D(G)-A(G)$，其中 $D(G)=\mathrm{diag}(d_1,\cdots,d_n)$，$A(G)$为 G 的邻接矩阵。$L(G)$也可以被定义为满足以下等式的二次型

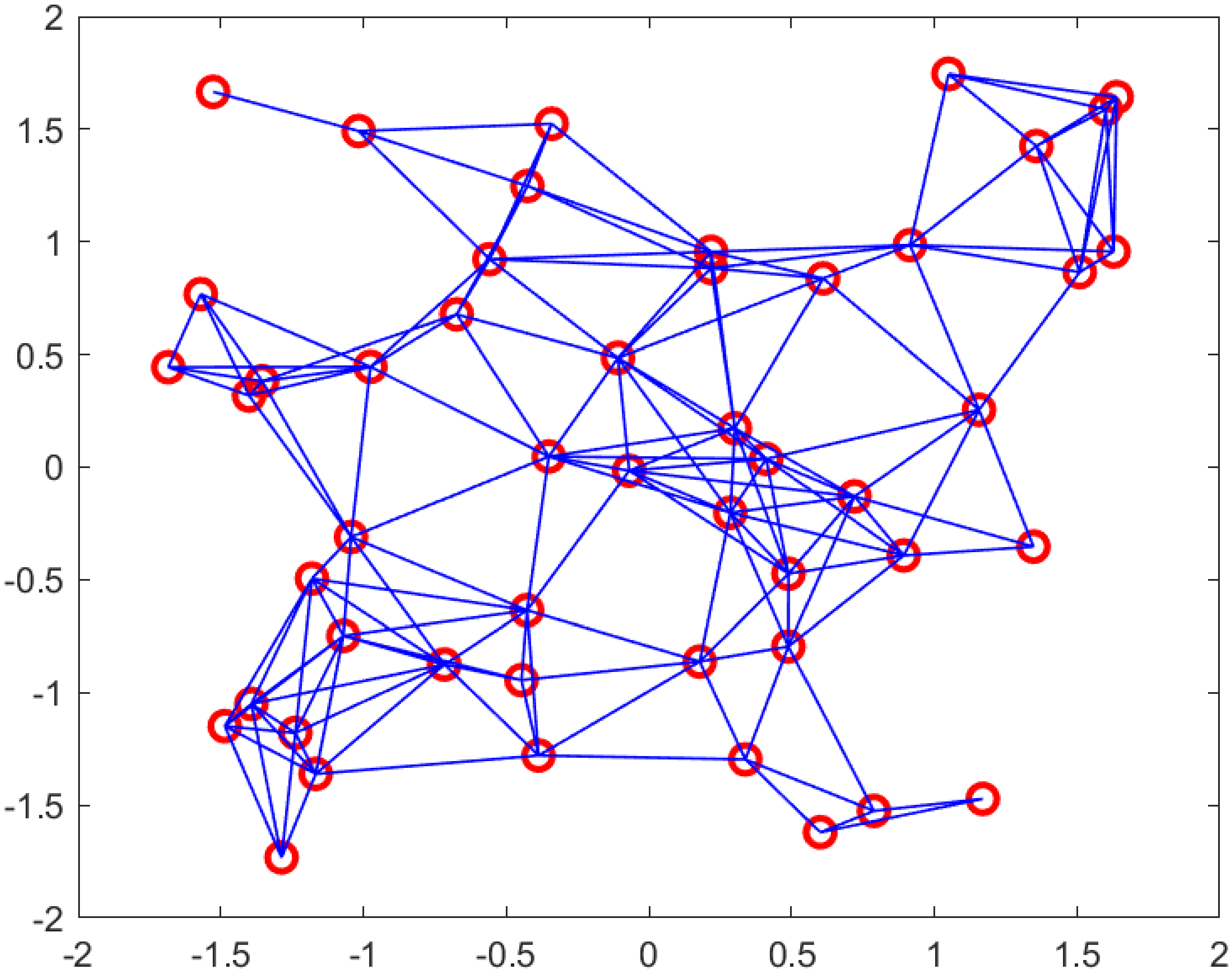
$$x^t L(G)x=\sum_{i<j,v_iv_j\in E}(x_i-x_j)^2,$$

其中 $x=(x_1,\cdots,x_n)^t$，所以 $L(G)$是一个对称的半正定奇异矩阵。

设 $L(G)$的特征值为 $\lambda_1\geqslant\cdots\geqslant\lambda_n=0$。因为 $L(G)$的列和为 0，所以 $e=(1,\cdots,1)^t$ 是 $L(G)$对应于 0 的一个特征向量。众所周知，$\lambda_{n-1}>0$ 当且仅当 G 是连通的。Fiedler[5]称 λ_{n-1}为图 G 的代数连通度，记为 $a(G)$。

2. 网络连接图(Network Connection)的构造

随机生成一个具有50个节点的传感器网络。节点随机放置在3.5 x 3.5方形区域内，通信距离为0.8。如下图所示，共有159条边，其代数连通度为：0.3007。



3. MATLAB程序

demo_Create_Network_Connection.m

```
%创建无向图 网络连接图 Network Connection.
clc;
close all;
clear;

Conf.Square = 3.5;  %方形区域的边长
Conf.NodeNumber = 50;  %节点个数
Conf.CommDist = 0.8;  %最大通信距离

is_create_network = 1;
if is_create_network == 1
    [ Network, Dists ] = CreateNetworksFunc(Conf);
    save Network_1.mat Network
else
    load Network_1.mat
end

nodenum = size(Network.Nodes.loc,1);  %节点个数
lap_matrix = zeros(nodenum);  %节点数*节点数 图的Laplace矩阵：diag(d1,d2,...dn)-邻接矩阵，di为节点i的度
for i=1:nodenum
    idx = Network.Nodes.neighbors{i};  %邻接节点的id
    lap_matrix(i,idx) = -1;   %负的邻接矩阵
    lap_matrix(i,i) = length(idx);  %对角线元素为节点的度
end
eig_val = eig(lap_matrix);  %lap_matrix的特征值
eig_val = sort(eig_val,'ascend');  %从小到大排序，最小特征值为0
algeb_conn = eig_val(2) % algebraic connectivity 代数连通度：lap_matrix的第二小特征值>0，连通图
avg_deg = sum(diag(lap_matrix))/nodenum   % average values 节点度的均值

DrawNetworks(Network);
% DrawNetworks(Network, Dists); %把所有的边的长度（通信距离）都标出来了
print(gcf,'-dpng','Network_1.png');  %保存图片
```

CreateNetworksFunc.m

```
function [ Network, Dists ] = CreateNetworksFunc(Conf)
%  创建无向图 网络连接图 Network Connection.
    num = Conf.NodeNumber;  %节点个数
    square = Conf.Square;  %方形区域的边长
    maxDist = Conf.CommDist;  %最大通信距离

    loc = square*rand(num,2) - square/2;  %num*2的随机数 节点坐标
    Dists = Euclid_Dist(loc(:,1),loc(:,2));  %节点数*节点数，对角线元素为0
```

```
    % without self-loop 不存在节点自己到自己的路径，对角线上的元素为无穷大
    Dists = Dists + 10*maxDist*eye(num);

    Neighbors = cell(num,1);
    maxDegree = 0; %节点的最大度，与节点相邻的最大边数
    edges = 0;  %图的总边的个数，无向图的度/2
    for i=1:num
        Neighbors{i} = find(Dists(i,:)<=maxDist);  %找邻接节点的id
        if length(Neighbors{i}) > maxDegree
            maxDegree = length(Neighbors{i});  %节点的最大度
        end
        edges = edges + length(Neighbors{i});
    end

    Nodes.loc = loc;
    Nodes.neighbors = Neighbors;

    Network.maxDegree = maxDegree;
    Network.edges = edges/2; %% undirected graph
    Network.Conf = Conf;
    Network.Nodes = Nodes;
end

function dist = Euclid_Dist(X,Y)
% 求两两节点之间的距离，输出[节点*节点]的矩阵，距离矩阵
    len = length(X);
    xx = repmat(X,1,len); %节点数*节点数
    yy = repmat(Y,1,len);
    dist = sqrt((xx-xx').^2+(yy-yy').^2);  %节点数*节点数
end
```

DrawNetworks.m

```
function fig = DrawNetworks( Network )
%画无向图 网络连接图 Network Connection.
% function fig = DrawNetworks( Network, Dists )  %把所有的边的长度（通信距离）都标出来了

num = Network.Conf.NodeNumber;  %节点个数
loc = Network.Nodes.loc;  %节点坐标
square = Network.Conf.Square;  %方形区域的边长
Neighbors = Network.Nodes.neighbors;  %邻接节点的id

fig = figure;
plot(loc(:,1),loc(:,2),'ro','MarkerSize',8,'LineWidth',2);  %节点是红色圆圈
side=ceil(square/2);
```

```
axis([-side,side,-side,side]);
for i=1:num
    for k = 1:length(Neighbors{i})
        j = Neighbors{i}(k);
%              c = num2str(Dists(i,j),'%.2f');
%              text((loc(i,1) + loc(j,1))/2,(loc(i,2) + loc(j,2))/2,c,'Fontsize',10);  %把所有的边的长度（通信距离）都标出来了
%              hold on;
        line([loc(i,1),loc(j,1)],[loc(i,2),loc(j,2)],'LineWidth',0.8,'Color','b'); %线是蓝色
    end
end
set(gcf, 'Color', 'w'); %白色
end
```

4. 连通度与代数连通度

图的连通度侧重的是图的结构性质，而代数连通度侧重的是矩阵的代数性质。

- **图的代数连通度**:

  图的Laplace矩阵的次小特征值。

- **点连通度**:

一个具有N个点的图G中，在去掉任意K−1个顶点后(1<=K<=N)所得的子图仍然连通，去掉K个顶点后不连通，则称G是K连通图，K称作图G的点连通度，记作K(G)。

- **边连通度**:

一个具有N条边的图G中，在去掉任意K−1条边后(1<=K<=N)所得的子图仍然连通，去掉K条边后不连通，则称G是K连通图，K称作图G的边连通度，记作K(G)。

5. 参考文献

[1] Hua J, Li C. Distributed variational Bayesian algorithms over sensor networks[J]. IEEE Transactions on Signal Processing, 2015, 64(3): 783-798.

[2] 肖恩利, 束金龙, 闻人凯. 图的代数连通度及其点连通度[J]. 华东师范大学学报(自然科学版), 2003, 2003(4):1-4.

[3] Junhao Hua. Distributed Variational Bayesian Algorithms. Github, 2017.