直达距离

load demo02

juzhen=zeros(92);

j=1;

for i=1:140

juzhen(xzuo(i),yzuo(j))=distance(xzuo(i),yzuo(j));

juzhen(yzuo(j),xzuo(i))=distance(xzuo(i),yzuo(j));

j=j+1;

end

%size(sparse(juzhen))

for i=1:92

for j=1:92

if i==j

juzhen(i,j)=0;

end

if i~=j & juzhen(i,j)==0

juzhen(i,j)=Inf;

end

end

end

save D:\matlabcode\2011B\zhida juzhen

zuiduanjuli=floydsuanfa(juzhen)/10;

save D:\matlabcode\2011B\zuiduanjuli zuiduanjuli

距离函数

function distance=jisuan(a,b)

load demo01 !三点坐标

ah=find(zdata==a);

bh=find(zdata==b);

x1=data(ah,1);

x2=data(bh,1);

y1=data(ah,2);

y2=data(bh,2);

distance=sqrt((x1-x2).^2+(y1-y2).^2);

Floyd

function [D,R]=floyd(A)

%用floyd算法实现求任意两点之间的最短路程。可以有负权

%参数D为连通图的权矩阵

D=A;n=length(D);

for i=1:n

for j=1:n

R(i,j)=i;%赋路径初值

end

end

for k=1:n

for i=1:n

for j=1:n

if D(i,k)+D(k,j)<D(i,j)

D(i,j)=D(i,k)+D(k,j);%更新 D(i,j)，说明通过k的路程更短

R(i,j)=R(k,j);%更新R(i,j)，需要通过k

end

end

end

hl=0;

for i=1:n

if D(i,i)<0

hl=1;

break;%跳出内层的for循环

end

end

if(hl==1)

fprintf('有负回路')

break;%跳出最外层循环

end

end

guanxia

load zuiduanjuli %最短距离矩阵

for i=1:20

min\_dis2(:,i)=zuiduanjuli(:,i);

end

min\_dis2=min\_dis2/10;

%zuiduanjuli2 92行20列

save D:\matlabcode\2011B\zuiduanjuli2 min\_dis2

for i=1:92

for j=1:20

if min\_dis2(i,j)>3 %过滤zuiduanjuli2

min\_dis2(i,j)=0;

end

end

end

min\_dis3=min\_dis2;

save D:\matlabcode\2011B\xianzhijuzhen min\_dis3

guanxiatest

load zuiduanjuli %最短距离矩阵

for i=1:20

min\_dis2(:,i)=zuiduanjuli(:,i);

end

min\_dis2=min\_dis2/10;

for i=1:92

for j=1:20

if min\_dis2(i,j)>3 %过滤zuiduanjuli2

min\_dis2(i,j)=10000

end

end

end

min\_dis3=min\_dis2;

%save D:\matlabcode\2011B\guanxiatest min\_dis3

lingojie

load gongzuoyali %cij得到 gongzuoyali

a=zeros(92);

for i=1:92

for j=1:92

if i==j

a(i,j)=1;

end

end

end

Aeq=[a a a a a a a a a a a a a a a a a a a a];

Beq=linspace(1,1,92)';

%size(Beq)

%size(Aeq)

for i=1:20

for j=1:92

b(i,j)=0;

end

end

c=[b b b b b b b b b b b b b b b b b b b b];

%size(c) 20 1840

for i=1:20

for m=1+(i-1)\*92:92\*i

c(i,m)=1;

end

end

A=[c;-c];

%size(A) 40 1840

tmp1=linspace(6,6,20)';

tmp2=linspace(-4,-4,20)';

B=[tmp1;tmp2];

%计算f

a=[];

for i=1:20

a=[gongzuoyalijuzhen(:,i);a];

end

%size(B)

[x,fval]=bintprog(a,A,B,Aeq,Beq);

%xlswrite('b.xlsx',x)

Lingojie2

%load guanxiatest %min\_dis3

%load realfananshuju2 %realfananshuju2

%load zuiduanjuli

%tmp=min\_dis3;

tic

%------计算最距离矩阵 92\*22

% result=zeros(72);

% for i=1:72

% for j=1:72

% if i>j

% hang1=zuiduanjuli(:,i);

% hang2=zuiduanjuli(:,j);

% tmp1=[tmp';hang1';hang2']'; %最短距离矩阵

% gongzuoyalijuzhen=tmp1.\*realfananshuju2;

% result(i,j)=o1jisuan(gongzuoyalijuzhen);

% else

% continue;

% end

% end

% end

% hang1=zuiduanjuli(:,4);

% hang2=zuiduanjuli(:,7);

% tmp1=[tmp';hang1';hang2']'; %最短距离矩阵 92\*22

% gongzuoyalijuzhen=tmp1.\*realfananshuju2;

%

% o1jisuan(gongzuoyalijuzhen)

load result

% for i=1:72

% for j=1:72

% if i<=j

% result(i,j)=10000;

% end

% end

% end

zuixiao=min(min(result));

[x,y]=find(result==zuixiao)

%xlswrite('a.xlsx',zhaodao);

% x =68 zuiduanjuli里面的88列

% y =51 zuiduanjuli里面的71列

Toc

O1jisuan

function value=o1jisuan(gongzuo)

%------为计算准备初始化值

a=zeros(92);

for i=1:92

for j=1:92

if i==j

a(i,j)=1;

end

end

end

Aeq=[a a a a a a a a a a a a a a a a a a a a a a];

Beq=linspace(1,1,92)';

%size(Beq)

%size(Aeq)

for i=1:22

for j=1:92

b(i,j)=0;

end

end

c=[b b b b b b b b b b b b b b b b b b b b b b];

%size(c)

for i=1:22

for m=1+(i-1)\*92:92\*i

c(i,m)=1;

end

end

A=[c;-c];

%size(A)

tmp11=linspace(6,6,22)';

tmp22=linspace(-3,-3,22)';

B=[tmp11;tmp22];

%size(B)

%------初始值计算结束

%计算f

a1=[];

for i=1:22

a1=[gongzuo(:,i);a1];

end

%size(a1)

options=optimset('display','off');

%x0=linspace(0,0,2024)';

[x fval]=bintprog(a1,A,B,Aeq,Beq);

value=min(fval);

zuixiao

function zuixiao(a)

%(1)B=sort(A) 对一维或二维数组进行升序排序,并返回排序后的数组,当A为二维时,对数组每一列进行排序.

%eg: A=[1,5,3],则sort(A)=[1,3,5]

%A=[1,5,3;2,4,1],则sort(A)=[1,4,1;2,5,3]

%(2)B=sort(A,dim),对数组按指定方向进行升序排序,

%dim =1,表示对每一列进行排序,,dim=2表示对每一行进行排序.

%(3)B=sort(A,dim,mode),mode为指定排序模式,mode为"ascend"时,进行升序排序,为"descend "时,进行降序排序.

%(4)[B,I]=sort(A,.....),I为返回的排序后元素在原数组中的行位置或列位置.

[B,I]=sort(a);

zuixiao=B(2:4,:);

zuisuo=I(2:4,:);

%zui=[zuixiao;zuisuo];

%save D:\matlabcode\2011B\zuiduanfuwu zuixiao zuisuo

%在保存正数到txt文件时 用save result.txt p -ascii

%dlm('filename',M,'D',R,C)

%将矩阵M的R行和C列用分割符D写入到filename中

Jisuank

load zuiduanjuli

% 1-20 line

% 12 14 16 21 22 23 24 28 29 30 38 48 62 colume

%min\_dis\_matr

% min\_dis\_matr(1:20,12);

a=[12 14 16 21 22 23 24 28 29 30 38 48 62];

b=1;

for i=a

tmp(:,b)=zuiduanjuli(1:20,i);

b=b+1;

end

%tmp为封锁方案矩阵20\*13

%save D:\matlabcode\2011B\fengsuofangan tmp

%把所有的inf换成一个很大的数 把 0 换成 一个很小的数

for i=1:20

for j=1:13

if tmp(i,j)==0

tmp(i,j)=0.0001;

end

end

end

%计算K值

tmp1=tmp;

a=0;

for k=30:-0.001:1

for i=1:20

for j=1:13

if tmp1(i,j)>k

tmp1(i,j)=0;

end

end

end

if rank(tmp1)~=13

a=k+0.001; %注释

break;

end

end

%最优封锁方案

for i=1:20

for j=1:13

if tmp(i,j)>a

tmp(i,j)=0;

end

end

end

zuiyou=tmp;

%rank(zuiyou)

%save D:\matlabcode\2011B\zuiyoufangan zuiyou

%sparse(zuiyou)

Jisuank2

tic

load zuiduanjuli

% 1-20 line

% 12 14 16 21 22 23 24 28 29 30 38 48 62 colume

%min\_dis\_matr

% min\_dis\_matr(1:20,12);

a=[12 14 16 21 22 23 24 28 29 30 38 48 62];

b=1;

%提出前面20行

for i=a

tmp(:,b)=zuiduanjuli(1:20,i);

b=b+1;

end

%提出后面72行

c=1;

for i=a

tmp2(:,c)=zuiduanjuli(21:92,i);

c=c+1;

end

save tmp tmp tmp2

%size(tmp2)

%tmp为封锁方案矩阵20\*13

%tmp2为72\*13

%

% kset=zeros(72);

% %test=zeros(72);

% for i=1:72

% for j=1:72

% if i~=j

% hang1=tmp2(i,:);

% hang2=tmp2(j,:);

% tmp1=[tmp;hang1;hang2];

% %test(i,j)=i\*j;

% kset(i,j)=funck(tmp1);

% else

% continue;

% end

% end

% end

% load kset

% for i=1:72

% kset(i,i)=10000;

% end

% zuixiao=min(min(kset));

% find(kset==zuixiao)

toc