clc; clear all; close all;

load('posndis05\_80.mat')

lambda=81;

copy=dis\_pos40;

range=9;

data=copy;

data(copy>=range) =0;

past\_t=5;

for i=1:50

if i==1

diff\_mat(:,:,1)=data(:,:,i)-data(:,:,i);

end

if i~=1

diff\_mat(:,:,i)=abs(data(:,:,i)-data(:,:,i-1));

end

end

sum1=0;

for i=1:past\_t

sum1=sum1+diff\_mat(:,:,i);

avg\_mat(:,:,i)=(sum1/i);

end

for j= past\_t+1:50

sum=0;temp=0;

for k=j-past\_t:j-1

sum=sum+diff\_mat(:,:,k);

end;

sum=sum/past\_t;

avg\_mat(:,:,j)=sum;

for k=j-past\_t:j-1

temp= temp + ( (avg\_mat(:,:,j)-diff\_mat(:,:,k))\*(avg\_mat(:,:,j)-diff\_mat(:,:,k)) );

end;

temp= (temp/past\_t);

temp=abs(sqrt(temp));

std\_mat(:,:,j)=temp;

end

data(data==0)=inf;

avg\_mat(avg\_mat==0) =15;

for i=1:lambda

for j=1:lambda

max = -inf ;

min=inf;

for k=1:50

if diff\_mat(i,j,k)> max

max = diff\_mat(i,j,k);

end

if diff\_mat(i,j,k)<min

min = diff\_mat(i,j,k);

end

end

glbmax(i,j) = max;

glbmin(i,j)=min;

end

end

for i=1:lambda

for j=1:lambda

for k=1:50

if i==j

weight(i,j,k)=0;

continue

end

if glbmax(i,j)== glbmin(i,j)

weight(i,j,k)=1;

continue

end

weight(i,j,k)= ( abs( ( glbmax(i,j) - avg\_mat(i,j,k) ) / (glbmax(i,j)-glbmin(i,j)) ) )/2;

end

end

end

k=50;

end\_index=lambda;

beta= 5;

start\_time=8;

c=300000000;

time =start\_time;

start\_index=1;

[shortestPathsdl, totalCostsdl] = kShortestPath(data(:,:,time),start\_index,end\_index,k);

avg\_dl=0;

final\_path=[];

dl\_final\_path=0;

while start\_index~=end\_index

for i=1:k

dl(i)=0;

end

min=inf;

for i=1:k

cur\_ind=1;

hop\_number=1;

while(shortestPathsdl{1,i}(cur\_ind)~=end\_index)

dl(i)= dl(i) + weight(shortestPathsdl{1,i}(cur\_ind), shortestPathsdl{1,i}(cur\_ind + 1),time)\*( (hop\_number/beta) + (diff\_mat(shortestPathsdl{1,i}(cur\_ind), shortestPathsdl{1,i}(cur\_ind + 1),time)/c )\*100000000) ;

cur\_ind =cur\_ind+1;

hop\_number = hop\_number + 1;

end

if dl(i)<min

min=dl(i);

min\_index=i;

end

end

final\_path= [final\_path;start\_index];

dl\_final\_path = dl\_final\_path + min;

time =time+1;

start\_index=shortestPathsdl{1,min\_index}(2);

[shortestPathsdl, totalCostsdl] = kShortestPath(data(:,:,time),start\_index,end\_index,k);

end

final\_path=[final\_path;end\_index];

[shortestPathsdl, totalCostsdl] = kShortestPath(data(:,:,time),1,end\_index,k);

sum\_min\_dl=0;

sum\_mat\_plr=0;

for q=1:(lambda-1)

xindex=q;

for t=6:40

temp=[];

while xindex~=end\_index

for i=1:k

dl(i)=0;

end

[shortestPathsdl, totalCostsdl] = kShortestPath(data(:,:,t),xindex,end\_index,k);

min=inf;

for i=1:k

cur\_ind=1;

hop\_number=1;

while(shortestPathsdl{1,i}(cur\_ind)~=end\_index)

dl(i)= dl(i) + weight(shortestPathsdl{1,i}(cur\_ind), shortestPathsdl{1,i}(cur\_ind + 1),t)\*( (hop\_number/beta) + (diff\_mat(shortestPathsdl{1,i}(cur\_ind), shortestPathsdl{1,i}(cur\_ind + 1),t)/c )\*100000000) ;

%std\_sum(i) = std\_sum(i) + std\_mat(shortestPathsdl{1,i}(cur\_ind), shortestPathsdl{1,i}(cur\_ind + 1),t);

cur\_ind =cur\_ind+1;

hop\_number = hop\_number + 1;

end

if dl(i)<min

min=dl(i);

min\_index=i;

end

end

temp= [temp;xindex];

xindex=shortestPathsdl{1,min\_index}(2);

end

std\_sum=0;

temp=[temp;end\_index];

delta=t;

alpha=1;

begin = temp(alpha);

while(begin~=end\_index)

std\_sum = std\_sum + std\_mat(temp(alpha),temp(alpha+1),delta);

delta=delta+1;

alpha=alpha+1;

begin=temp(alpha);

end

std\_sum = std\_sum/alpha;

sum\_min\_dl=sum\_min\_dl + min;

sum\_mat\_plr=sum\_mat\_plr + std\_sum;

end

end

sum\_min\_dl = sum\_min\_dl/(35\*(lambda-1));

sum\_mat\_plr=sum\_mat\_plr/(35\*(lambda-1));

disp(sum\_mat\_plr)

disp(sum\_min\_dl)