%%交叉操作

%输入变量：pop：父代种群，pc：交叉的概率

%输出变量：newpop：交叉后的种群

function [new\_pop] = crossover(pop, pc)

[px,~] = size(pop);

% 判断路径点数是奇数或偶数

parity = mod(px, 2);

new\_pop = {};

%两个两个交叉

for i = 1:2:px-1

singal\_now\_pop = pop{i, 1};

singal\_next\_pop = pop{i+1, 1};

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% A = [5 3 4 2]; %%

%% B = [2 4 4 4 6 8]; %%

%% [Lia,Locb] = ismember(A,B) %%

%% Lia = 1x4 logical array %%A的每个元素若B中存在则该位为1 否则为零

%% 0 0 1 1 %%

%% Locb = 1×4 %%每个相同的元素在B中的索引

%% 0 0 2 1 %%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

[lia, locb] = ismember(singal\_now\_pop, singal\_next\_pop);%[Lia,Locb] = ismember(A,B)确定 A 的哪些元素同时也在 B 中及其在 B 中的相应位置。

[~, n] = find(lia == 1);%要查找特定的整数值，使用 == 运算符。返回找到的值在lia中的索引

[~, m] = size(n);

%如果随机数小于交叉概率且A中有三个以上路径节点与B中的相同

if (rand < pc) && (m >= 3)

% 生成一个2到m-1之间的随机数，也就是除去开头和结尾，在两条路径的相同节点中随机选取一个节点用于交叉

r = round(rand(1,1)\*(m-3)) +2;%Y = round(X) 将 X 的每个元素四舍五入为最近的整数

crossover\_index1 = n(1, r);%

crossover\_index2 = locb(crossover\_index1);

new\_pop{i, 1} = [singal\_now\_pop(1:crossover\_index1), singal\_next\_pop(crossover\_index2+1:end)];

new\_pop{i+1, 1} = [singal\_next\_pop(1:crossover\_index2), singal\_now\_pop(crossover\_index1+1:end)];

else %否则不交叉

new\_pop{i, 1} =singal\_now\_pop;

new\_pop{i+1, 1} = singal\_next\_pop;

end

%如果有奇数条路径，除最后一条外，其余已按照if的条件进行了是否交叉的处理，所以最后一条仍然不变。

if parity == 1

new\_pop{px, 1} = pop{px, 1};

end

end