实验报告

【实验名称】 实验三 基于遗传算法的TSP问题求解

【实验目的】

学习遗传算法

【实验原理】

1.详细说明

遗传算法（Genetic Algorithm）是模拟达尔文生物进化论的自然选择和遗传学机理的生物进化过程的计算模型，是一种通过模拟自然进化过程搜索最优解的方法。遗传算法是从代表问题可能潜在的解集的一个种群（population）开始的，而一个种群则由经过基因（gene）编码的一定数目的个体(individual)组成。每个个体实际上是染色体(chromosome)带有特征的实体。染色体作为遗传物质的主要载体，即多个基因的集合，其内部表现（即基因型）是某种基因组合，它决定了个体的形状的外部表现，如黑头发的特征是由染色体中控制这一特征的某种基因组合决定的。因此，在一开始需要实现从表现型到基因型的映射即编码工作。由于仿照基因编码的工作很复杂，我们往往进行简化，如二进制编码，初代种群产生之后，按照适者生存和优胜劣汰的原理，逐代（generation）演化产生出越来越好的近似解，在每一代，根据问题域中个体的适应度（fitness）大小选择（selection）个体，并借助于自然遗传学的遗传算子（genetic operators）进行组合交叉（crossover）和变异（mutation），产生出代表新的解集的种群。这个过程将导致种群像自然进化一样的后生代种群比前代更加适应于环境，末代种群中的最优个体经过解码（decoding），可以作为问题近似最优解。

1. 问题描述与解决过程

旅行商问题，即TSP问题（Traveling Salesman Problem）又译为旅行推销员问题、货郎担问题，是数学领域中著名问题之一。假设有一个旅行商人要拜访n个城市，他必须选择所要走的路径，路径的限制是每个城市只能拜访一次，而且最后要回到原来出发的城市。路径的选择目标是要求得的路径路程为所有路径之中的最小值。

求出从第一个城市经过n-1个城市，最后会到第一个城市的有效序列

适应度函数:f(s) =

生成一个含有N个城市的顺序序列，生成另一个含有N个城市的序列

交叉方式：在两个序列中**随机**选一串对应的城市对进行交换

交换后，必定产生冲突，再将产生冲突的城市对互换，如下图所示：

冲突解决方式：循环在交换序列中查找产生冲突的对应城市编号，直到循环结束，找到两个城市，在交换串之外进行交换

**enum{A=0,B,C,D,E,F,G,H,I,J};**

序列1： A G C B E F D H A

序列2： A H D C B E G F A

序列1： A H D C B F D H A

序列2： A G C B E E G F A

交叉

序列1： A H D C B F D H A

序列2： A G C B E E G F A

序列1： A H D C B F E G A

序列2： A G C B E D H F A

变异方式：

序列: A F C D E B A

随机选取序列中除开头节点和结尾节点之外的两个节点进行互换，如：选中F和D节点，变换之后的节点为：

序列: A D C F E B A

序列的选择条件：在node[2]和node[3]中适应度较小的，进行随机性变异

淘汰：原序列产生两个交换后的序列，对新产生的两个序列中的适应度较小的序列进行判断——是否会产生变异，随后，再对4个序列进行适应度大小进行从大到小的排序，淘汰掉两个适应度值小的序列，剩下两个适应度大的序列

输出：输出第一序列和其适应度与路程长度。（在牺牲了一定最优性的情况下，换取了程序快速得解的能力）

1. 算法流程

初始化种群

评估对种群所有个体的优劣进行判断

记录最优个体

选择淘汰掉较劣的个体

进化、交叉产生后代、突变

更新种群

将最优个体作为最终结果

是否结束

【实验结果】



