

# 人工智能基础实验报告

刘砺志

(2014 级计算机 1 班 22920142203873)

本文是人工智能基础——用遗传算法求解 TSP 问题的实验报告。

## 1 实验概述

本次实验要求实现遗传算法来解决旅行商问题 (TSP)，即已知  $n$  个城市之间的相互距离，现有一个推销员必须遍访这  $n$  个城市，并且每个城市只能访问一次，最后又必须返回出发城市。如何安排他对这些城市的访问次序，可使其旅行路线的总长度最短。

## 2 实验原理

遗传算法是一种模拟自然界进化的概率搜索算法。在遗传算法中，问题的状态空间被描述为个体 (individuals) 的集合。每一个个体都被表示为一个字符串或矩阵，它们被称为染色体 (chromosomes)。使用遗传算法的目的就是为了在状态空间中找到具有最佳基因的个体。个体的基因质量又评估函数 (evaluation function) 度量。被测试的搜索空间部分被称为种群 (population)。粗略的遗传算法流程由如下算法描述。

### ABSTRACT-GENETIC-ALGORITHM

- 1 Make initial population at random.
- 2 **while** not stop
- 3     Select parents from the population.
- 4     Produce children from the selected parents.
- 5     Mutate the individuals.
- 6     Extend the population adding the children to it.
- 7     Reduce the extended population.
- 8 Output the best individual found.

在我的实现中，我采用了路径表达 (path representation) 来表示遍历的结点顺序。一次遍历被表示为  $n$  个结点 (从 0 到  $n-1$  标号) 的一个排列。例如，依次游历 3-2-4-1-7-5-8-6，

则被表达为

$$(3\ 2\ 4\ 1\ 7\ 5\ 8\ 6)$$

## 2.1 选择 (Selection)

为了评估不同游历顺序的优劣，我定义了个体的适应度函数 (fitness function) 进行评估。适应度被定义为这条环路路径长度倒数的适度放缩，即对于路径

$$(c_0, c_1, c_2, \dots, c_{n-2}, c_{n-1})$$

其适应度函数为

$$fitness = \frac{\alpha}{\sum_i dist(c_i, c_{i+1})}$$

其中， $\alpha$  为某个合适的正常数。

接着通过轮盘赌来产生父代。事实上，具有高适应度的个体具有更高的概率被挑选为父代。

## 2.2 交叉 (Crossover)

这里我们选用部分映射交叉算子 (partially-mapped crossover operator, PMX)，其算法原理详见文献 [2]。

## 2.3 变异 (Mutation)

此处我们选用基因交换来实现变异，即随机选取染色体中的两个位点，交换之。例如，对于个体

$$(1\ 5\ 3\ 7\ 2\ 4\ 6\ 8)$$

我们选择基因 5 和 4，交换它们，则后代为

$$(1\ 4\ 3\ 7\ 2\ 5\ 6\ 8)$$

## 2.4 复制 (Copy)

我们还选取了一小部分个体直接进入下一代，而不进行交叉或变异。

# 3 实验结果

我采用 Python 3.6.0 编写了实验程序 `tsp.py`。我设定初始种群规模为 500，最大代数为 2000 代，适应度函数缩放系数  $\alpha = 100$ ，交叉概率  $P_c = 0.7$ ，变异概率  $P_m = 0.005$ 。为了进

行测试，我选取 TSPLIB 库 [3] 中 burma14 数据集，它包括了 14 个城市的坐标。最后，我使用 matplotlib 库将路径和坐标绘制出来。一次测试的结果如图1所示，路径总长度为 30.8785。从图上看，这个结果比较优秀。

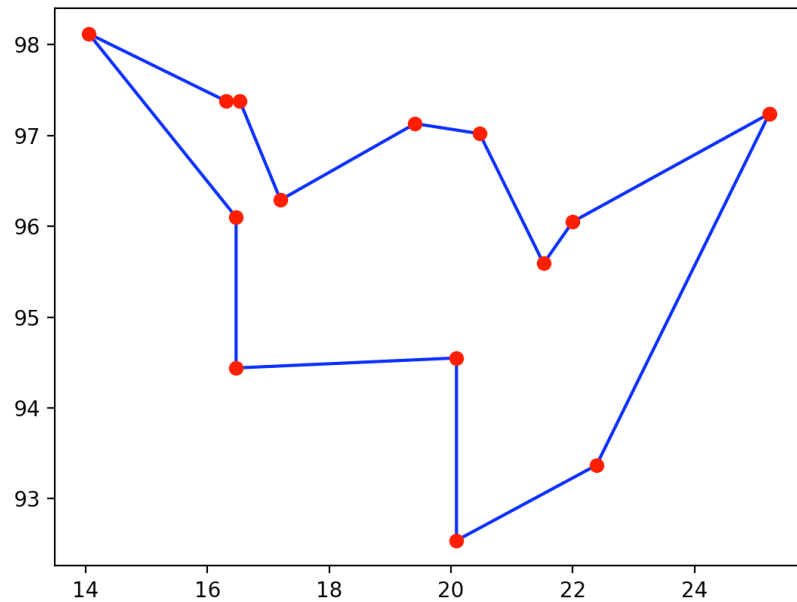


图 1 遗传算法求解旅行商问题测试结果

## 参考文献

- [1] 人工智能, Nils J. Nilsson 著, 郑扣根, 庄越挺译, 潘云鹤校, 北京: 机械工业出版社, 2000 年 9 月
- [2] [http://www.dca.fee.unicamp.br/~gomide/courses/EA072/artigos/Genetic\\_Algorithm\\_TSPR\\_eview\\_Larranaga\\_1999.pdf](http://www.dca.fee.unicamp.br/~gomide/courses/EA072/artigos/Genetic_Algorithm_TSPR_eview_Larranaga_1999.pdf)
- [3] <http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95>