

人工智能

2020416046

2020级

周晓萌

人工智能实训

计算机学院

**曲阜师范大学实验报告**

**学院 计算机学院 年级、专业 20人工智能 学号 2020416046姓名 周晓萌**

**课程名称 人工智能实训 上课时间 第 1 节 — 第 10 节**

**实验日期 2022 年 9 月 24 日 星期 六 教师签名 雷玉霞 成绩**

**实验四 基于模拟退火算法的旅行商问题**

**1．实验目的**

1. 通过模拟退火算法，解决旅行商问题。

**2．实验设备**

1. 一台装有windows10系统的笔记本电脑。
2. matlab环境。
3. **实验成员**

周晓萌

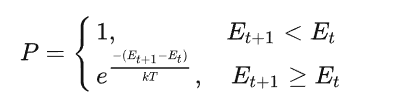
**3. 实验内容**

1. **问题背景**
2. 旅行商问题，即TSP问题(Travelling 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。
3. 本次实验的实验任务是通过模拟退火算法，解决旅行商问题。模拟退火算法(Simulated Annealing,SA)最早的思想是由N. Metropolis等人于1953年提出。1983年,S. Kirkpatrick等成功地将退火思想引入到组合优化领域。它是基于Monte-Carlo 迭代求解策略的一种随机寻优算法，其出发点是基于物理中固体物质的退火过程与一般组合优化问题之间的相似性。
4. **处理过程**

模拟退火算法从某一较高初温出发，伴随温度参数的不断下降,结合一定的概率突跳特性在解空间中随机寻找目标函数的全局最优解，即在局部最优解能概率性地跳出并最终趋于全局最优。这里的“一定的概率”的计算参考了金属冶炼的退火过程，这也是模拟退火算法名称的由来。将温度T当作控制参数，目标函数值f视为内能E，而固体在某温度T时的一个状态对应一个解xi，然后算法试图随着控制参数T的降低，使目标函数f(内能E)也逐渐降低，直至趋于全局最小值（退火中低温时的最低能量状态），就像金属退火过程一样。

1. **模拟退火数学原理**

结合概率突跳特性在解空间中随机寻找目标函数的全局最优解，如果新解比当前解更优，则接受新解，否则基于Metropolis准则判断是否接受新解。接受概率为：



如上公式，假设当前时刻搜索的解为xt，对应的系统能量(目标函数)为Et，对搜索点施加随机扰动，产生新解xt+1，相应地，系统能量为Et+1，那么系统对搜索点从xt到xt+1转变的接受概率就为上公式。

1. **模拟退火流程**

算法实质分两层循环，在任一温度水平下，随机扰动产生新解，并计算目标函数值的变化，决定是否被接受。由于算法初始温度比较高，这样，使E增大的新解在初始时也可能被接受，因而能跳出局部极小值，然后通过缓慢地降低温度，算法就最终可能收敛到全局最优解，具体流程为：

1. 令T=T0,表示开始退火的初始温度，随机产生一个初始解x0,并计算对应的目标函数值E(x0);
2. 令T=kT,其中k取值0到1之间，为温度下降速率；
3. 对当前解xt施加随机扰动，在其邻域内产生一个新解xt+1，并计算对应的目标函数值E(xt+1),计算

ΔE=E(xt+1)−E(xt)

1. 若ΔE<0,接受新解作为当前解，否则按照概率e−ΔE/kT判断是否接受新解；
2. 在温度T下，重复L次扰动和接受过程，即执行步骤3和4；
3. 判断温度是否达到终止温度水平，若是则终止算法，否则返回步骤2.
4. **最终效果**

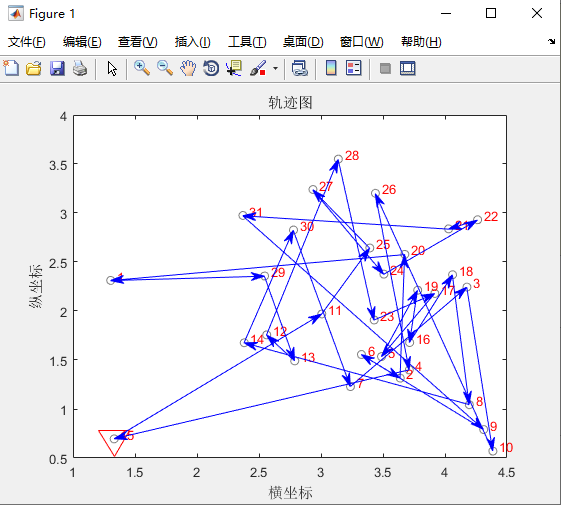


图1 随机解路径图

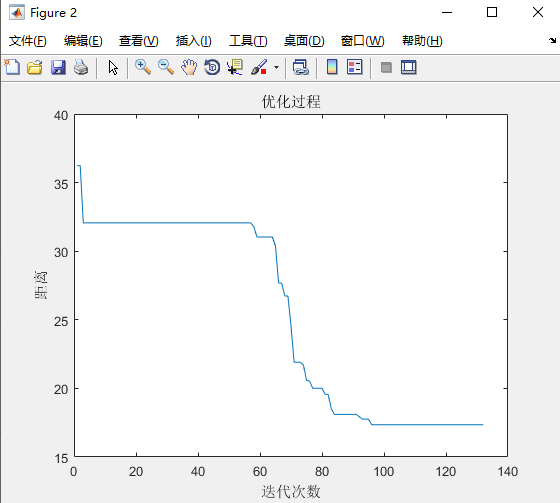


图2 优化迭代结果

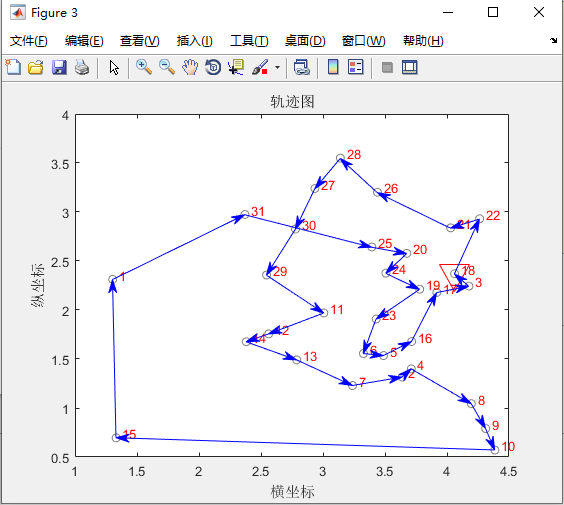


图3 最优解路径图

**4．实验心得**

本次实验考察了如何将人工智能中的模拟退火算法应用于实际问题，通过合适的策略将旅行商问题解决。其实，解决旅行商问题的途径有很多，例如贪心算法，动态规划等等，相较于这些算法，遗传算法具有以下优点：

1：可以突破爬山算法的局限性，获得全局最优解（以一定的概率接受较差解，从而跳出局部最优解）。

2：初始解与最终解都是随机选取的，它们毫无关联，因此具有很好的鲁棒性，即抵御外界不稳定因素的能力。

但同时，遗传算法也具有一些缺点：

1：其最优解常常受迭代次数k的影响，若k值越大，则搜索时间越长，获得的最优解更可靠；k值越小，则搜索时间越短，有可能就跳过了最优解。

2：模拟退火算法受温度冷却速率的影响，若冷却速率较慢，则搜索时间较长，可以获得更优解，但是会花费大量时间；如果冷却速率过快，可以较快的搜索到更优的解，但也有可能直接跳过最优解。

所以，在合适的情况下使用模拟退火算法可以将问题很好的解决，这就是本次实验中我的收获。

