

# SA 实验报告

## 1. 实验介绍

在 TSPLIB (<http://comopt.ifl.uni-heidelberg.de/software/TSPLIB95/>), 多个地址有备份; 其他网站还可以找到有趣的 art TSP 和 national TSP) 中选一个大于 100 个城市数的 TSP 问题,

1. 采用多种邻域操作的局部搜索 local search 策略求解;
2. 在局部搜索策略的基础上, 加入模拟退火 simulated annealing 策略, 并比较两者的效果;
3. 要求求得的解不要超过最优值的 10%, 并能够提供可视化, 观察路径的变化和交叉程度。

## 2. 实验准备

TSP 数据来源: TSPLIB 中的 [ch130.tsp](#) 问题 (最优解 6110)

算法实现: C++

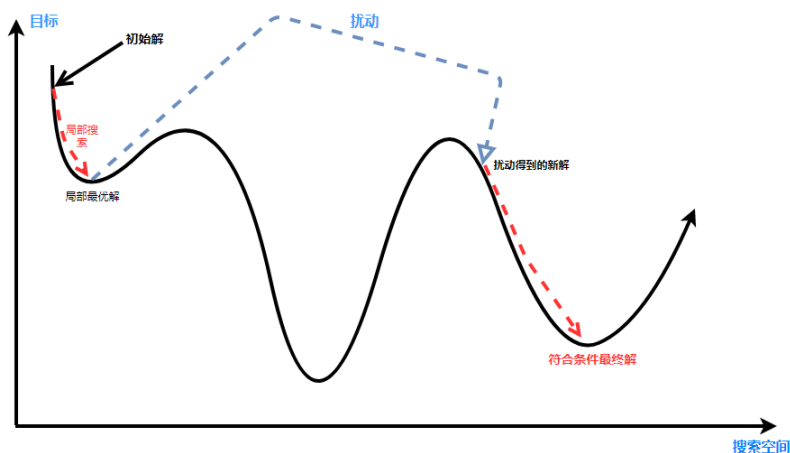
界面实现: QT

## 3. 实验过程

首先使用局部搜索算法来解决 TSP 问题。局部搜索是解决最优化问题的一种启发式算法。因为对于很多复杂的问题, 求解最优解的时间可能是很长的。因此诞生了各种启发式算法来退而求其次寻找次优解, 局部搜索就是其中一种。它是一种近似算法。

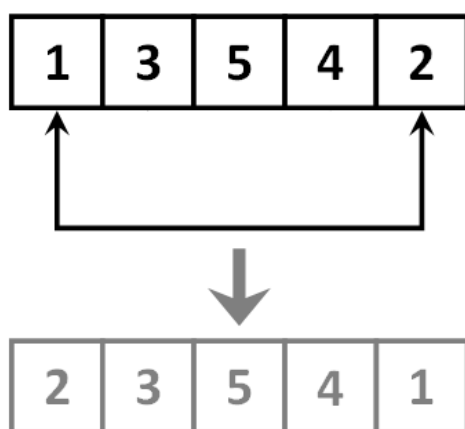
局部搜索算法是从爬山法改进而来的。简单来说, 局部搜索算法是一种简单的贪心搜索算法, 该算法每次从当前解的临近解空间中选择一个最优解作为当前解, 直到达到一个局部最优解。局部搜索从一个初始解出发, 然后搜索解的邻域, 如有更优的解则移动至该解并继续执行搜索, 否则返回当前解。

算法图示:

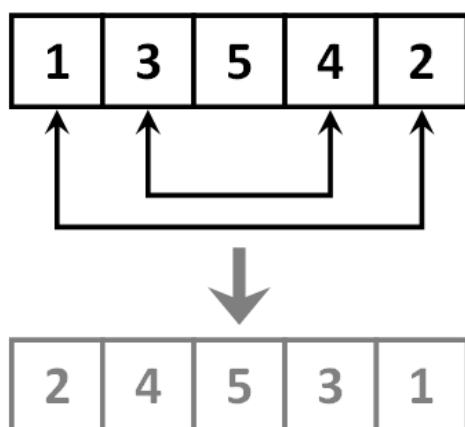


局部搜索算法中，最重要的是邻域操作，即如何从当前解得到邻居解。邻域操作选择的是否足够好会影响局部搜索是否能够跳出局部最优解。这里主要使用两种邻域操作：

### 1. Swap



### 2. Reverse



在本实验中，我的局部搜索算法总共迭代 50000 次。

局部搜索算法主要的问题就是容易陷入局部最优解而不再继续向全局最优前进，为了改进这一缺点接着使用模拟退火算法。

模拟退火算法来源于固体退火原理。

物理退火：将材料加热后再经特定速率冷却，目的是增大晶粒的体积，并且减少晶格中的缺陷。材料中的原子原来会停留在使内能有局部最小值的位置，加热使能量变大，原子会离开原来位置，而随机在其他位置中移动。退火冷却时速度较慢，使得原子有较多可能可以找到内能比原先更低的位置。

模拟退火：其原理也和固体退火的原理近似。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。

模拟退火算法能够跳出局部最优的原因是它会以一定概率来接受一个比当前解要差的解，这个概率由以下公式计算：

$$P(dE) = \exp(dE/(kT))$$

其中  $k$  是一个常数,  $\exp$  表示自然指数, 且  $dE < 0$  (温度总是降低的)。这条公式指明了:

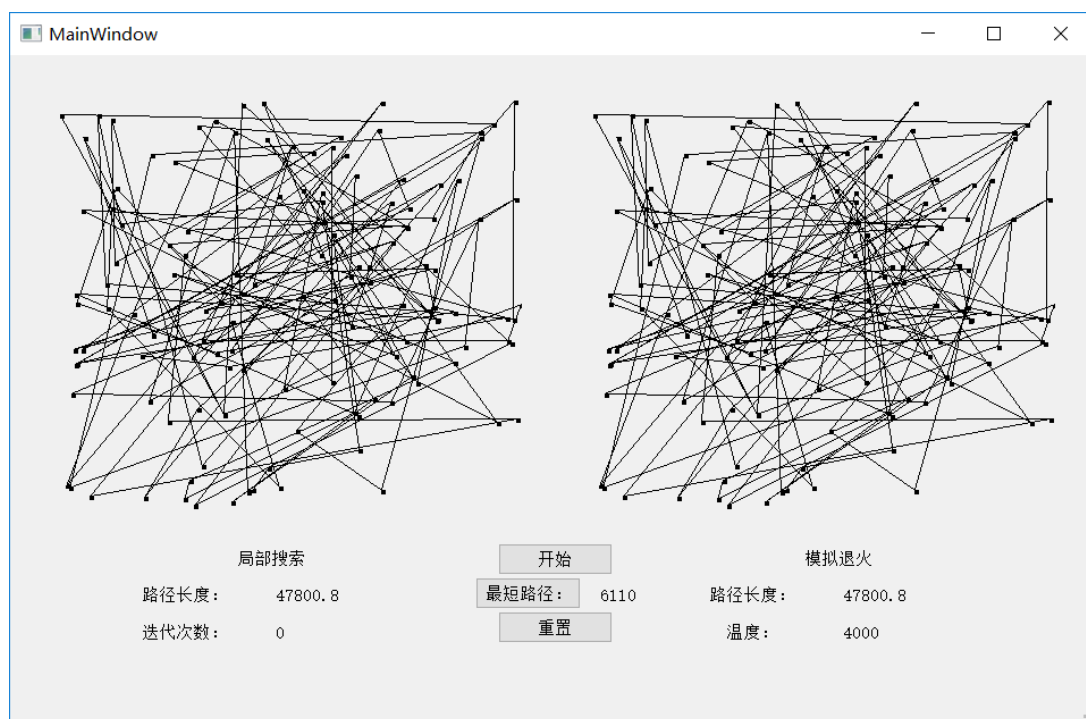
1. 温度越高, 出现一次能量差为  $dE$  的降温的概率就越大;
2. 温度越低, 则出现降温的概率就越小。又由于  $dE$  总是小于 0 (不然怎么叫退火), 因此  $dE/kT < 0$ ,  $\exp(dE/kT)$  取值是  $(0,1)$ , 那么  $P(dE)$  的函数取值范围是  $(0,1)$ 。

随着温度  $T$  的降低,  $P(dE)$  会逐渐降低。我们将一次向较差解的移动看做一次温度跳变过程, 我们以概率  $P(dE)$  来接受这样的移动。也就是说, 在用固体退火模拟组合优化问题, 将内能  $E$  模拟为目标函数值  $f$ , 温度  $T$  演化成控制参数  $t$ , 即得到解组合优化问题的模拟退火演算法: 由初始解  $i$  和控制参数初值  $t$  开始, 对当前解重复“产生新解→计算目标函数差→接受或丢弃”的迭代, 并逐步衰减  $t$  值, 算法终止时的当前解即为所得近似最优解。

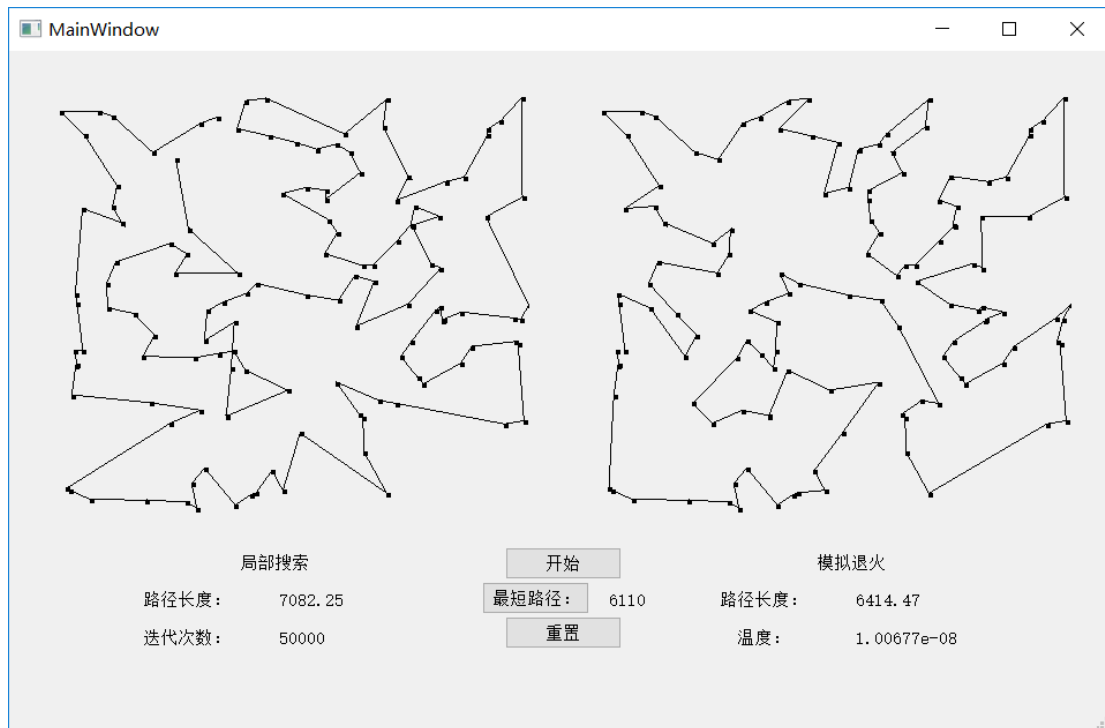
在本实验中, 我的模拟退火算法的初温设置为  $4000^{\circ}\text{C}$ , 退火系数设置为 0.98, 在每个温度下迭代 500 次, 当温度小于  $1\text{e-}8^{\circ}\text{C}$  时停止。

#### 4. 结果分析

应用程序初始界面:



运行后:



十次测试数据:

局部搜索	模拟退火
6948.3	6684.56
7189.06	6709.67
6821.45	6770.01
6966.59	6720.54
6938.48	6730.11
6930.91	6615.81
7089.73	6702.39
6882.35	6630.67
7269.23	6638.96
7082.25	6414.47

平均下来，局部搜索的结果在 6800~7300 间浮动，平均值在 7000 左右，误差在 10%到 15%。而模拟退火的结果最好能达到 6300 左右，一般在 6600~6800 之间，误差在 4%到 10%。

总体来看，模拟退火算法确实能够在一定程度上优化局部搜索，其陷入局部最优的可能性比局部搜索算法更低，能够跳出局部最优。

## 主要参考文献

1. [https://en.wikipedia.org/wiki/Simulated\\_annealing](https://en.wikipedia.org/wiki/Simulated_annealing)
2. [https://en.wikipedia.org/wiki/Local\\_search\\_\(optimization\)](https://en.wikipedia.org/wiki/Local_search_(optimization))
3. <http://doc.qt.io/qt-5/>