

Artificial Intelligence

人工智能实验

高级搜索

中山大学计算机学院
2024年春季

目录

1. 理论课内容回顾

1.1 遗传算法

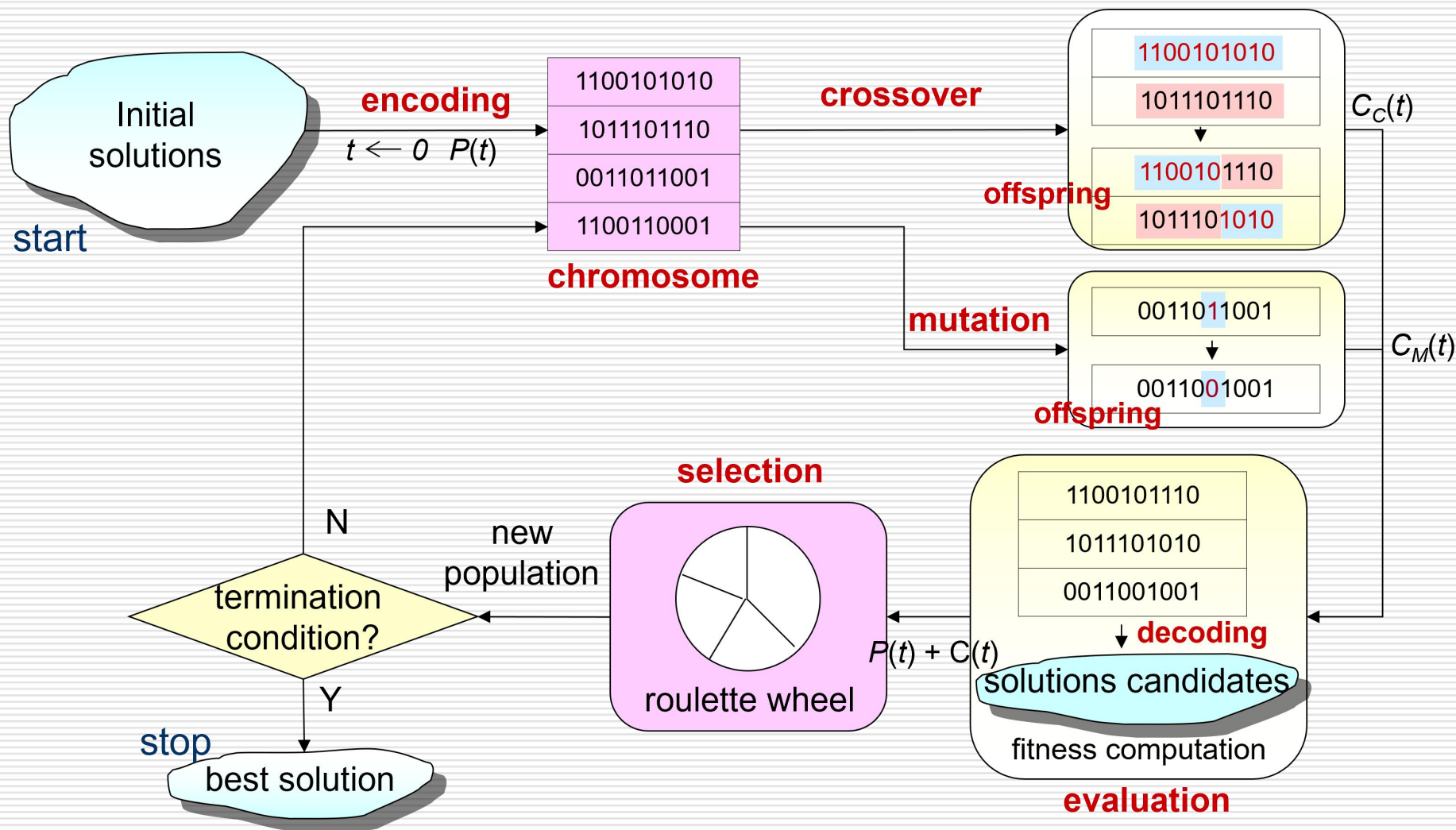
1.2 旅行商问题 (TSP)

2. 实验任务

2.1 用遗传算法求解 TSP 问题

3. 作业提交说明

1.1 遗传算法



1.1 遗传算法

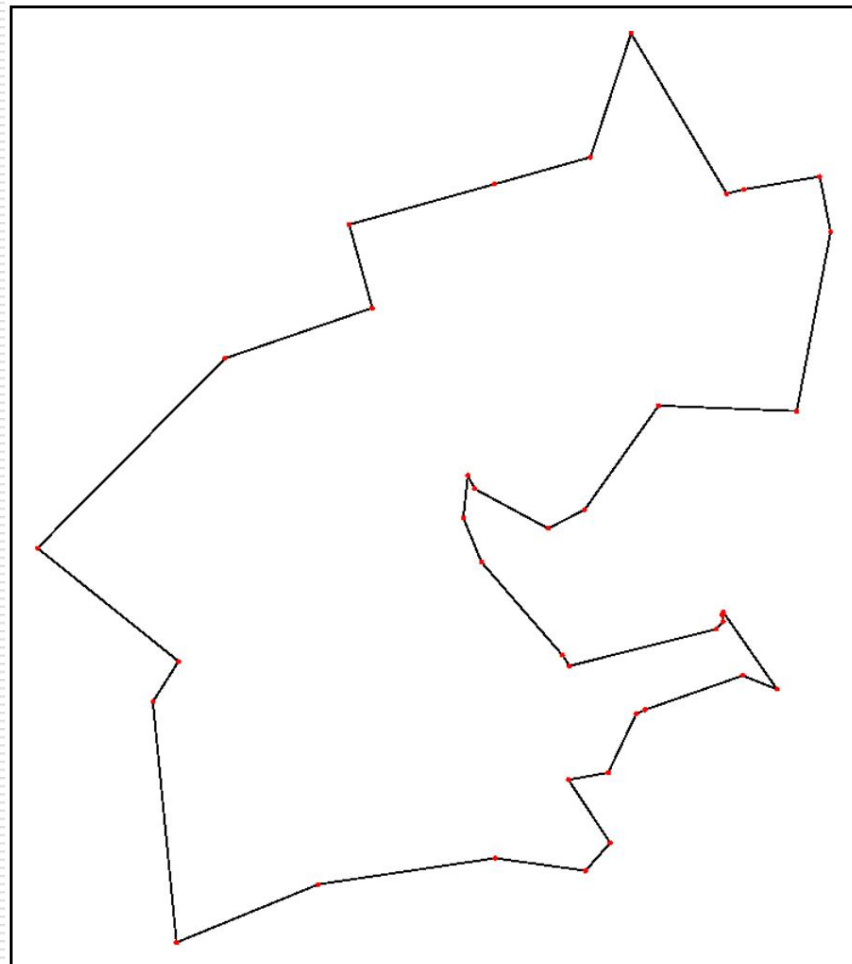
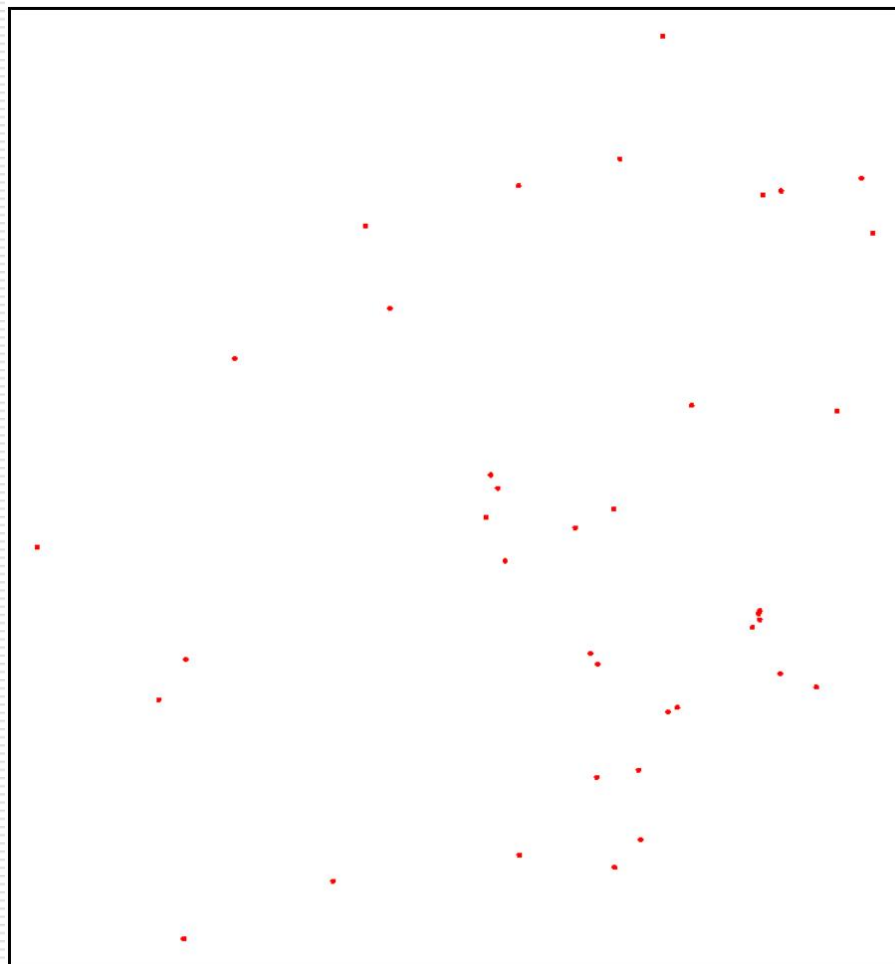
算法 遗传算法解决 TSP

输入: n 个城市的坐标

- 1: 随机生成 10 个不同个体的初始种群 $P(0)$
 - 2: **for** $t = 0$ **to** 最大迭代轮数 **do**
 - 3: 初始化后代种群 $C(t)$
 - 4: **for** $i = 1$ **to** 每一代后代数目 **do**
 - 5: 依据适应度, 从 $P(t)$ 中选取一对父母 p_1, p_2
 - 6: 利用交叉操作, 从 p_1, p_2 产生后代 c_1, c_2
 - 7: 以一定概率对 c_1, c_2 进行变异操作
 - 8: $C(t) \leftarrow \{c_1, c_2\}$
 - 9: **end for**
 - 10: 依据适应度, 从 $P(t) \cup C(t)$ 选取优异个体作为 $P(t + 1)$
 - 11: **end for**
 - 12: 依据适应度, 从最后一代种群中选择最优个体作为问题的解
-

1.2 旅行商问题(TSP)

- 旅行商问题(TSP)是最为广泛研究的组合优化问题之一.
- 问题描述: 一个旅行商寻找环游 n 个城市(不重复)的最短路径.



1.2.1 解的表示

□ 表示方法一：随机排列

- 对于n个城市的TSP问题，解表示为1-n的排列

染色体表示

5	4	6	9	2	1	7	8	3
---	---	---	---	---	---	---	---	---

环游顺序: 5-4-6-9-2-1-7-8-3-5

□ 表示方法二：随机数排序

- 生成n个(0,1)范围内的随机数
- 这n个随机数的排序对应旅游城市的顺序

染色体表示

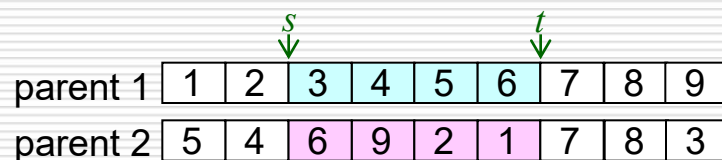
1	2	3	4	5	6	7	8	9
0.23	0.82	0.45	0.74	0.87	0.11	0.56	0.69	0.78

环游顺序: 6-1-3-7-8-4-9-2-5-6

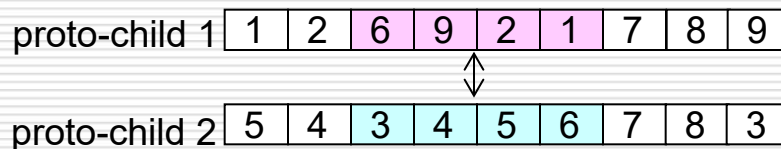
1.2.2 染色体交叉操作

部分映射交叉(Partial-Mapped Crossover, PMX)

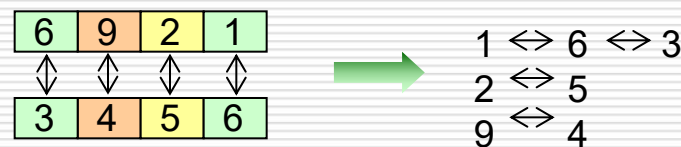
step 1 : 随机选择下标 s, t



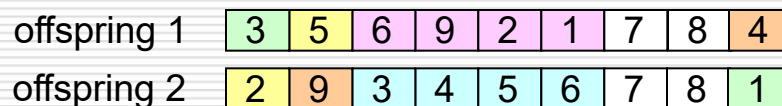
step 2: 交叉子串



step 3: 确定映射关系



step 4: 生成后代染色体



算法 染色体交叉操作

输入: 染色体 v_1, v_2 及其长度 l

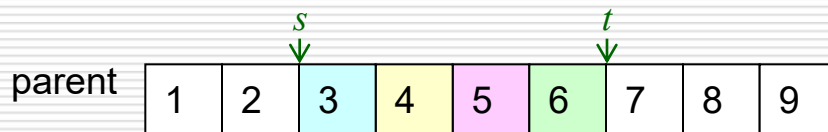
- 1: $R \leftarrow \emptyset$
切割染色体
- 2: 随机生成两个下标 s, t , 满足 $0 < s < t < l - 1$
交叉染色体
- 3: $v'_1 \leftarrow [v_1[0], \dots, v_1[s-1], v_2[s], \dots, v_2[t], v_1[t+1], \dots, v_1[l-1]]$
- 4: $v'_2 \leftarrow [v_2[0], \dots, v_2[s-1], v_1[s], \dots, v_1[t], v_2[t+1], \dots, v_2[l-1]]$
基因映射
- 5: 依据 $[v_1[s], \dots, v_1[t]]$ 与 $[v_2[s], \dots, v_2[t]]$ 建立映射关系
生成后代染色体
- 6: 依据映射关系修改 v'_1, v'_2 非交叉部分

其他交叉操作: order crossover (OX), cycle crossover (CX), position-based crossover, order-based crossover等. 感兴趣的同学可自行尝试.

1.2.3 染色体变异操作

倒置变异(Inversion Mutation)

step 1: 随机选择下标 s, t



step 2: 将 s - t 中间部分倒置



算法 染色体变异操作

输入: 染色体 v 及其长度 l

- 1: $R \leftarrow \emptyset$
切割染色体
- 2: 随机生成两个下标 s, t , 满足 $0 < s < t < l - 1$
倒置
- 3: $v' \leftarrow [v[0], \dots, v[s-1], v[t], v[t-1], \dots, v[s+1], v[s], v[t+1], \dots, v[l-1]]$
- 4: **return** v'

2. 实验任务

□ 用遗传算法求解 TSP 问题

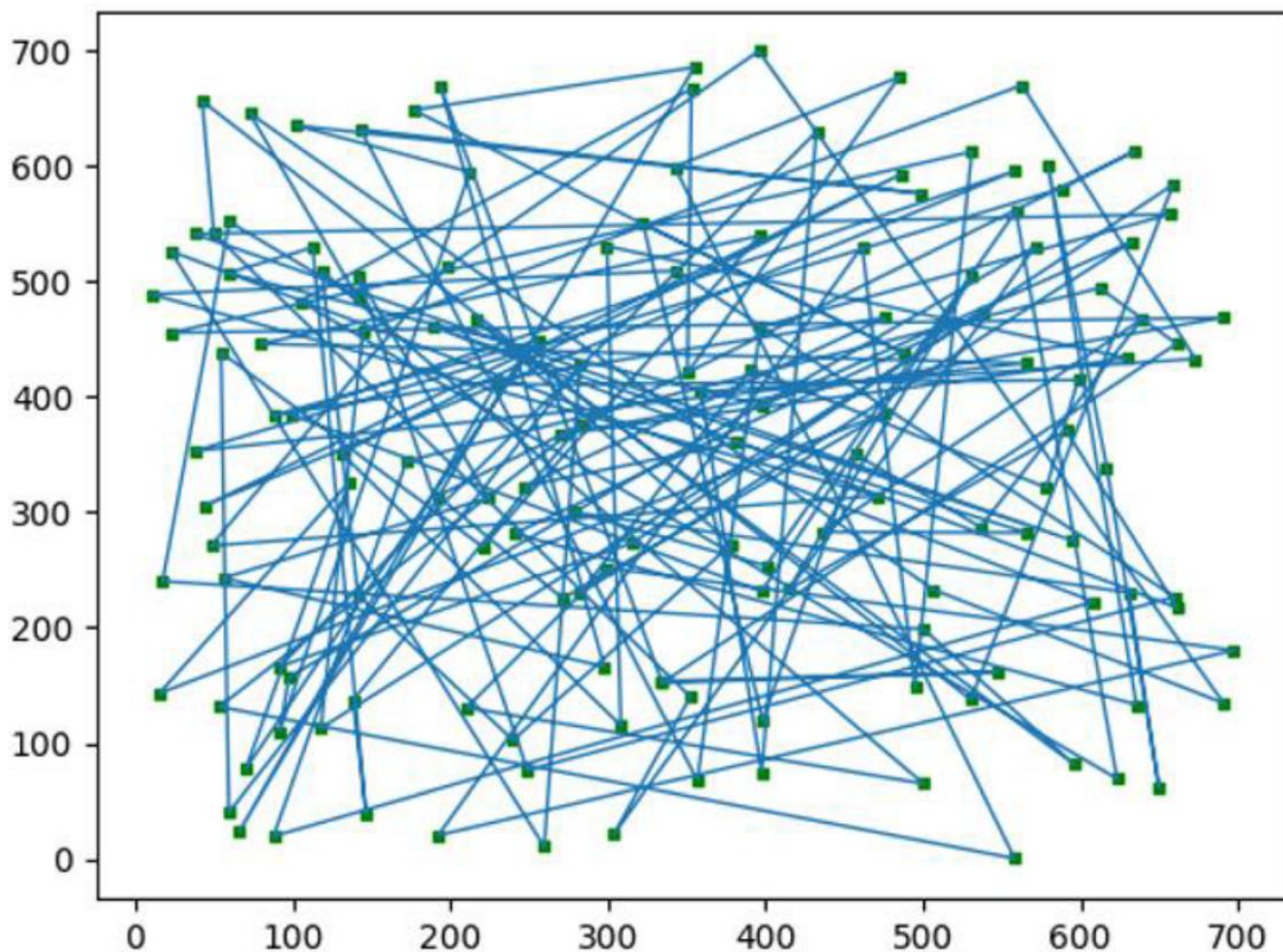
- 在National Traveling Salesman Problems (uwaterloo.ca) (<https://www.math.uwaterloo.ca/tsp/world/countries.html>) 中任选两个 TSP问题的数据集。
- 建议：
 - 对于规模较大的TSP问题, 遗传算法可能需要运行几分钟甚至几个小时的时间才能得到一个比较好的结果. 因此建议先用城市数较小的数据集测试算法正确与否, 再用城市数较大的数据集来评估算法性能。
 - 由于遗传算法是基于随机搜索的算法, 只运行一次算法的结果并不能反映算法的性能. 为了更好地分析遗传算法的性能, 应该以不同的初始随机种子或用不同的参数(例如种群数量, 变异概率等)多次运行算法, 这些需要在实验报告中呈现.

3. 作业提交说明

- ❑ 提交到课程网站（超算习堂）中对应的“实验作业四”，并注意网站上公布的截止日期
- ❑ 提交格式：提交一个命名为” 学号_姓名.zip” 的压缩包，压缩文件下包含两部分：code文件夹和实验报告pdf文件
- ❑ 实验报告是pdf格式，命名为：学号_姓名.pdf
- ❑ code文件夹：存放实验代码，一般有多个代码文件的话需要有readme
- ❑ “学号_姓名” 样例：20*****_wangxiaoming
- ❑ 如果需要更新提交的版本，则在后面加_v1，_v2。如第一版是” 学号_姓名.zip” ，第二版是” 学号_姓名_v1.zip” ，依此类推
- ❑ 截止日期：2024年4月29日24点

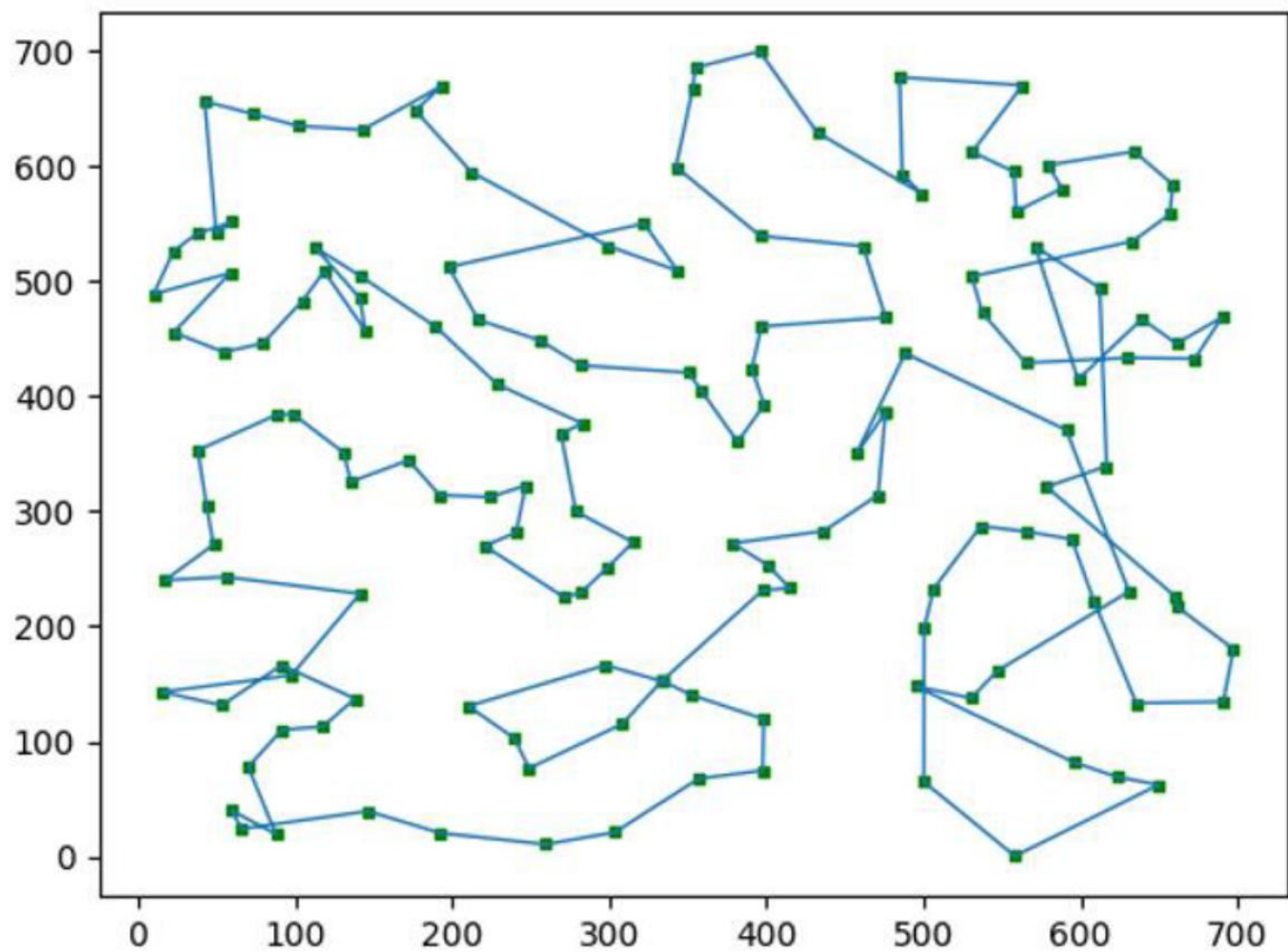
往届学生作品演示

□ 路径可视化



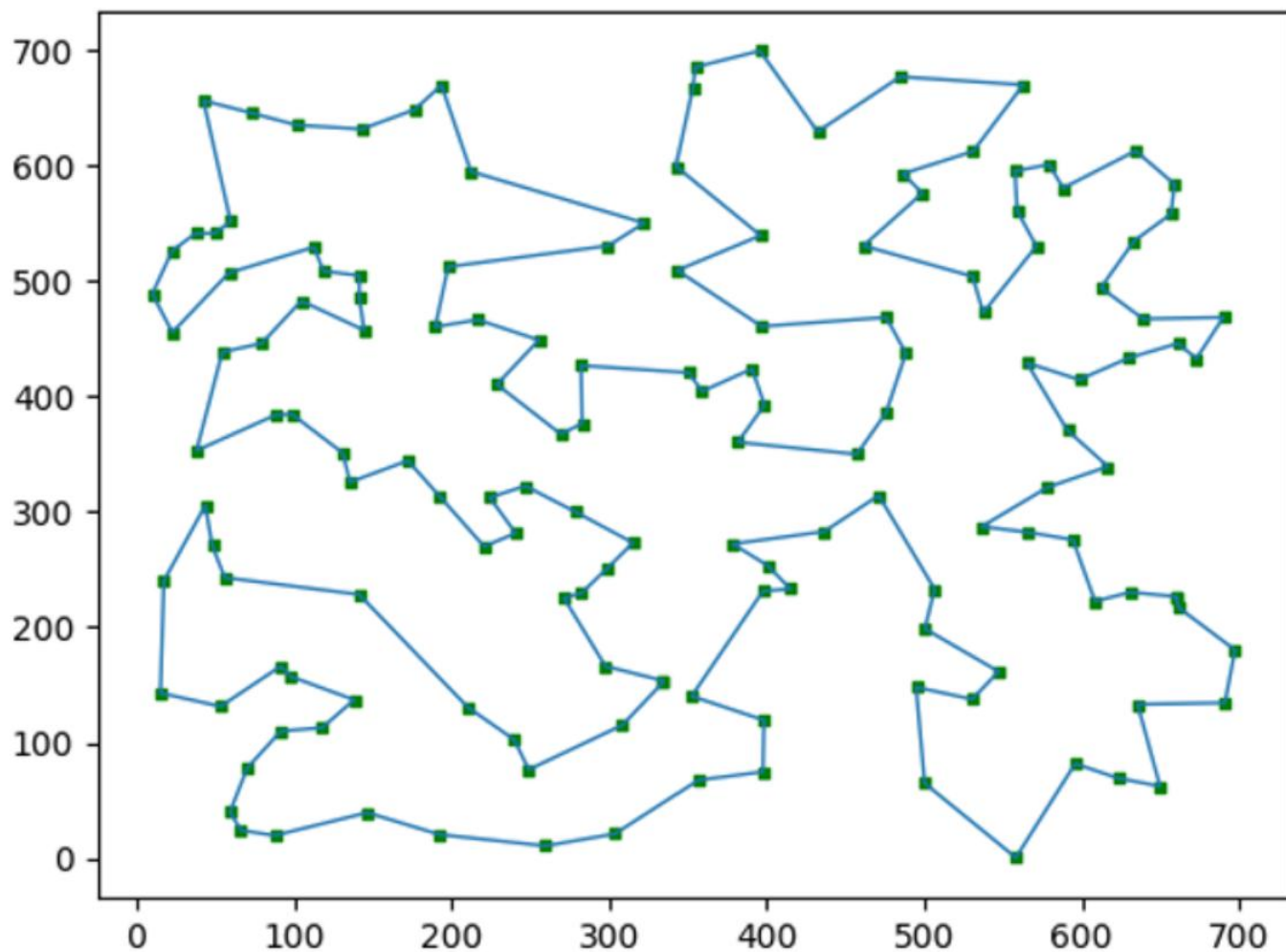
往届学生作品演示

□ 路径可视化



往届学生作品演示

□ 路径可视化



往届学生作品演示

□ 收敛曲线

