

2016 数学建模校赛

题目 B

印制电路板焊锡

学 校: 山西大学

队 员: 李恩宾

张红若

张彪

指导教师: 李顺勇

2016 年 9 月 4 日

印制电路板焊锡

摘 要

针对电路板焊锡过程中点胶机的涂锡膏顺序问题，本文将电路板中所有的位点抽象为图中的顶点，将位点间的连边抽象成顶点间的连边，位点间的距离抽象为边上的权值，构建赋权完全图。通过遗传算法和贪婪算法，求解赋权完全图中的最小哈密尔顿圈，即求得最优焊锡顺序和相应的路径长度¹。

关键词： 遗传算法 旅行商问题 贪婪算法 蚁群算法

目录

1	问题的提出	3
2	模型假设	3
3	符号说明	4
4	模型的建立	4
4.1	问题分析	4
4.2	问题一	5
4.2.1	蚁群算法	5
4.2.2	遗传算法	6
4.2.3	改进贪婪算法 ²	7
4.3	问题二	9
4.3.1	一个链式解	10
4.3.2	原理	10
5	模型评价	10

1 问题的提出

焊锡是印制电路板生产过程中的一步后续处理，目的是把贴片元件焊接到电路板表面。人们使用一台装有锡膏点胶机的数控（计算机数字控制，简称 CNC）机械，将锡膏涂布于电路板表面的特定位置。现有一个电路板，大小为 $300mm \times 180mm$ ，共有 280 个位点需要点涂锡膏。为了节约时间以及使机器减少磨损，这里需要设计出一个让点胶机的点击头所走的路程是最少的方案。

问题一：求解点胶机从 1 号位点开始对这 280 个位点进行焊锡，最后回到 1 号位点的最优焊锡顺序，并给出相应路径长。

问题二：不预先设定起点和终点，求解对 1 万张这种类型的电路板进行焊锡的最优顺序，并给出相应路径长，这里假设每张电路板安装点胶机所需时间和所处位置都是相同的。

2 模型假设

1. 假设点胶机的初始位置位于电路板起始位点的正上方。
2. 假设在点胶机进行焊锡的过程中没有焊锡失败的情况，既不需要来回焊锡
3. 每张电路板安装点胶机所需时间和所处位置都是相同的。
4. 假设点胶机的柱头在两点之间是按直线行进的
5. 假设点胶机对印刷电路板的作用力不会使印刷电路板产生移动

3 符号说明

符号	说明
$G(V, E)$	赋权完全图
V	图的点集
E	图的边集
D	节点间距离的邻接矩阵
S	问题的解向量
x_i, y_i	i 的 xy 坐标
$d_{i,j}$	i, j 间的距离

4 模型的建立

4.1 问题分析

通过对电路板焊锡问题的分析，我们将焊锡过程转化为赋权完全图中求最小哈密尔顿圈问题，即 TSP 旅行售货员问题。TSP 问题是典型的 NP-hard 问题³，不存在多项式时间算法求其最优解。我们先使用了蚁群算法，对电路板上的位点进行遍历，最终回到起点，得到最优焊锡顺序，求得相应的路径长。但由于数据量大，蚁群迭代很慢，所得的结果不能满意；我们又构建了遗传算法，遗传算法得到的结果与蚁群相当，仍然不是最优解。最后我们从局部最优解出发构建了一个经过改进的贪婪算法，最终求得了问题的一个满意解。

将问题看做定义在图 $G = (V, E)$ 上的 TSP 问题，其中 $V = 1, 2, \dots, n$ ， $E = (i, j), i, j \in V$ ，图中节点之间的距离矩阵为：

$$D = \begin{bmatrix} 0 & d_{1,2} & \dots & d_{1,n} \\ d_{2,1} & 0 & \dots & d_{2,n} \\ \dots & \dots & 0 & \dots \\ d_{n,1} & \dots & \dots & 0 \end{bmatrix}$$

其中 $d_{i,j} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$ ，那么图 G 的解 $S = (s_1, s_2, \dots, s_n), n \in V$ 。

4.2 问题一

4.2.1 蚁群算法

蚁群算法是通过模拟蚂蚁群体觅食行径而提出来的一种模拟进化算法⁴。经过研究，蚂蚁总能找到从食物源到蚁穴的最短路线。蚂蚁个体之间通过外激素（即信息素）进行信息传递，当蚂蚁发现食物源并返回蚁穴告知同伴的途中，会释放信息素，便于同伴通过感知信息素的存在和其浓度，选择合适路径以找到食物源。蚂蚁在刚找到食物源或者蚁穴时释放的信息素量最多，并随着蚂蚁个体走过距离的增加，释放的信息素越来越少。当遇到图 1 中所示路线时，即 A 点处有两条分支，蚂蚁可选择从经 B 点到达 D 点，也可选择经 C 点到达 D 点，显然 AB 的距离小于 AC 的距离，蚂蚁会根据路径长度，释放不同量的信息素，给同伴此条路径的信息⁵。

这里把点胶机的点胶头用蚂蚁来代替，将要焊锡的位点用蚂蚁觅食的地方来表示，进而把点胶机焊锡的过程抽象为蚂蚁觅食，以便于利用蚁群算法来求解该问题。

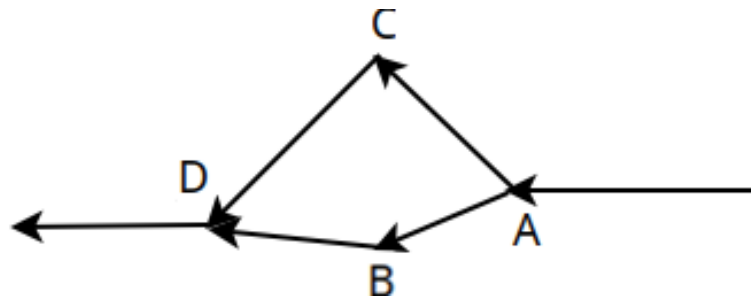


图 1: 蚁群的行为特点

信息素会随时间挥发，因而路上的信息素浓度与蚂蚁经过的个数和释放信息素的量相关。而蚁群个体根据信息素的浓度，优先选择信息素浓度高的路径。

图 2 是在迭代次数为 2000 时的输出图，总长度为 2946.44724482。存在多处两个线相互交叉的现象，有几处的交叉现象明显的增加了总路径的大小。表明了蚁群算法存在的不足之处，进而用遗传算法进行下一步的优化，以便于得到路径更短的输出结果。

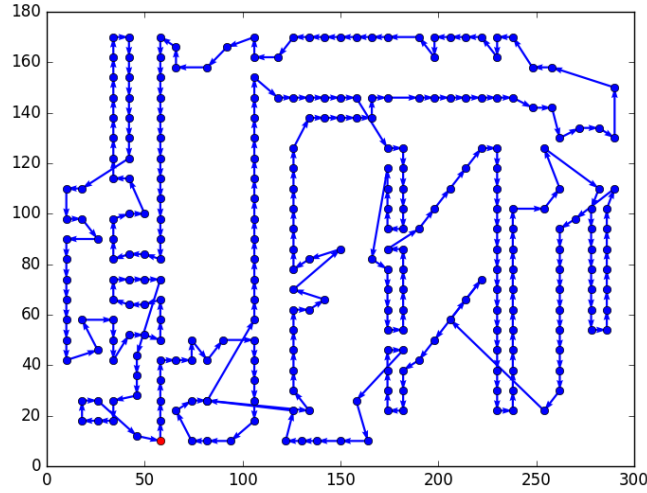


图 2: 蚁群算法得到的结果

4.2.2 遗传算法

遗传算法是模仿生物进化过程的一种算法，它以自然选择和遗传学中的复制、变异和交叉等自然规律为理论依据。利用 GA 算法求解问题，首先需要将模型的解以适当的方式进行编码，并根据模型的目标建立个体适应度函数，然后随机生成初始种群（即一组初始的可行解），再重复对种群进行选择、交叉、变异等遗传操作直到种群成熟，停止优化[?]。对于 NP 困难问题智能算法是有效的选择，我们选用遗传算法对问题一进行建模求解，并且根据实际题目的要求对遗传算法进行微调。

在一般的遗传算法中，种群个体 j 结构如图 3 所示，变异算子是指让种群中某个个体的某些基因突变的算子，由于本题限定了初始位置与

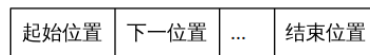


图 3: 算法中的个体

结束位置所以个体基因中的起始位与结束位是固定的不可以被突变算子改变，在种群迭代中其过程如如图 4 所示。

同样的对于算法中的交叉算子，它是使有效基因遗传下去的主要算子，在交叉过程中同样应该遵守第一位与最后一位不可算在交叉过程中的原则，如图 5 所示。

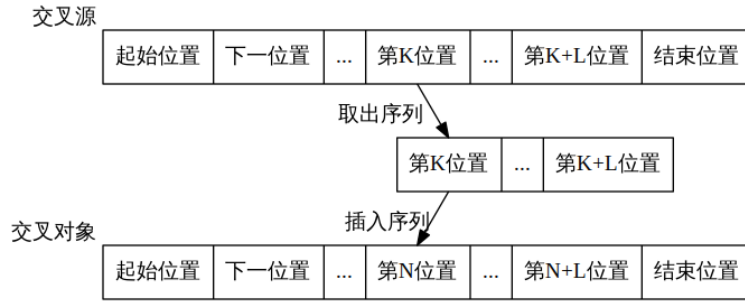


图 4: 交叉算子



图 5: 变异算子

使用上述算法结构对于本问题所给的数据进行种群演化，经过长时间算法运行，得到结果如图 6，总长度为 2775.273329

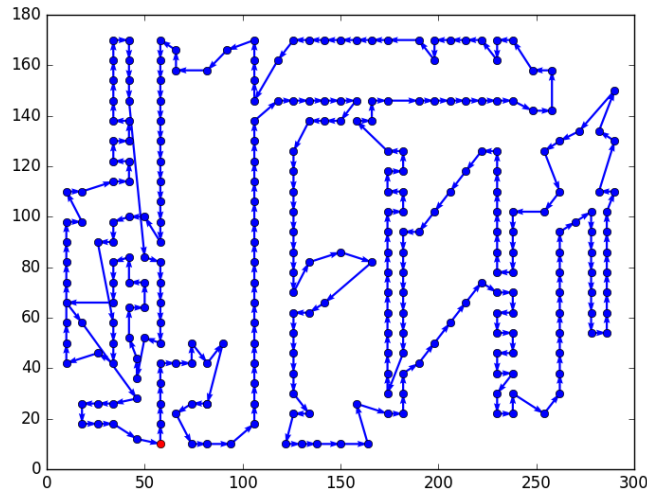


图 6: 遗传算法结果

4.2.3 改进贪婪算法²

贪婪算法是一种构建型启发式算法，其原理是求出任意两位点间欧式距离，建立图 G 的邻接矩阵 D 并为其赋值，其中 $d_{i,j}$ 表示位点 x_i 与

x_j 间的距离。将距离矩阵中两位点间距离按从小到大的顺序排列，从最短边开始向图 G 中添加，直至添加到图 G 成为一条回路为止。在添加过程中算法遵循以下原则：

1. 该节点的链接边数小于 2。
2. 链接的边不会形成边数小于 n 的回路。

但贪婪算法在寻找过程中并没有主动使起点与终点回合，所以会造成最后返回路径非常长，如图 7 所示，这时所得到的总长度为 2861.03928238。分析贪婪算法过程可知，算法最后产生的跨度主要由于算法本身没有确

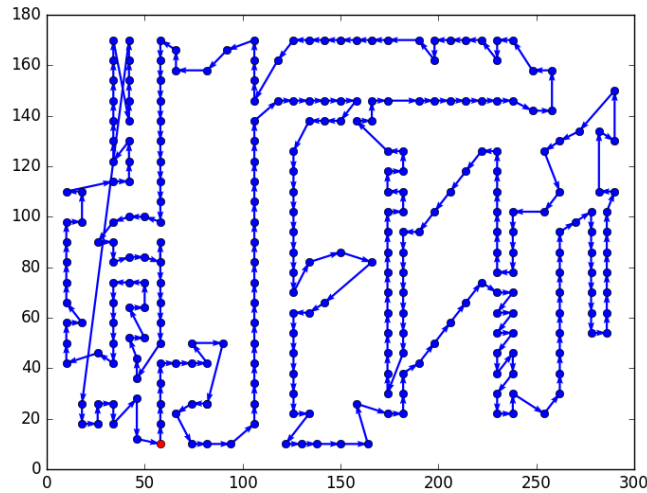


图 7: 贪婪算法得到的结果

定边的拼接方向，依照这种思路对贪婪算法进行改进。根据 Held 与 Karp 提出的 Held-Karp 模型⁶，我们使用一个向量 $\pi = (\pi_1, \pi_2, \dots, \pi_k)$ 对贪婪算法的距离矩阵 D 进行改造，如式 1。

$$D' = \begin{cases} d_{i,j} = d_{i,j} - \pi_i - \pi_j & i \neq j \\ d_{i,j} = M & i = j \end{cases} \quad (1)$$

其中 M 为一个足够大的数， π_k 由式 2 构造，其中 α 为比例系数，比例系数越大算法就越倾向于先添加外围的点，其中 len_k 是 k 号点到中心

的距离, \overline{len} 是所有点到中心距离的平均。

$$\begin{cases} \pi = (\pi_1, \pi_2, \dots, \pi_k), \pi_k = \alpha \cdot (len_k - \overline{len}) \\ len_k = \sqrt{(x_0 - x_k)^2 + (y_0 - y_k)^2} \\ \overline{len} = \frac{\sum_{k=0}^{k=n} len_k}{n} \\ x_0 = \frac{\sum_{k=0}^{k=n} x_k}{n}, y_0 = \frac{\sum_{k=0}^{k=n} y_k}{n} \end{cases} \quad (2)$$

根据改进贪婪算法得到最终结果如图 8所示, 总长度为 2791.27883397。

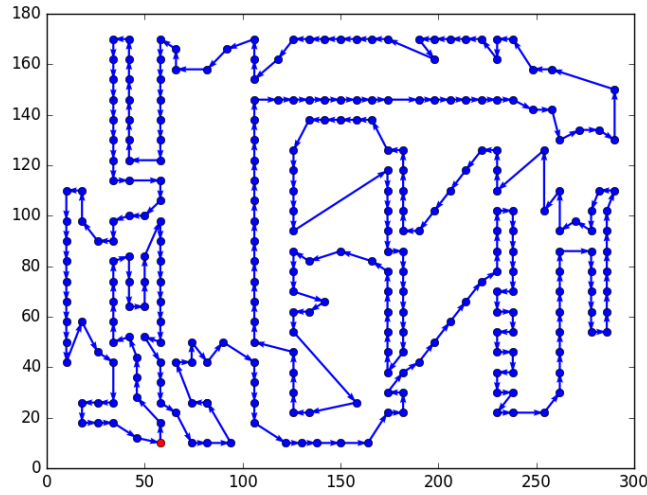


图 8: 改进贪婪算法得到的结果

4.3 问题二

由于此问不要求起点与终点的位置, 而是要求对一万张同样的电路板进行操作, 我们采用如下方案进行求解:

1. 求出一个图 G 的解 S , 首尾不需要相接。
2. 从 S 的起点移动到 S 的终点, 完成一块电路板。
3. 把上一步的终点作为起点, 重复第二步。

4.3.1 一个链式解

遗传算法和蚁群算法可以获得问题的满意解但是其运行过程十分漫长不适用于题目要求，所以利用贪婪算法获得链式满意解，如图 9所示，总长度为 2715.05298165。

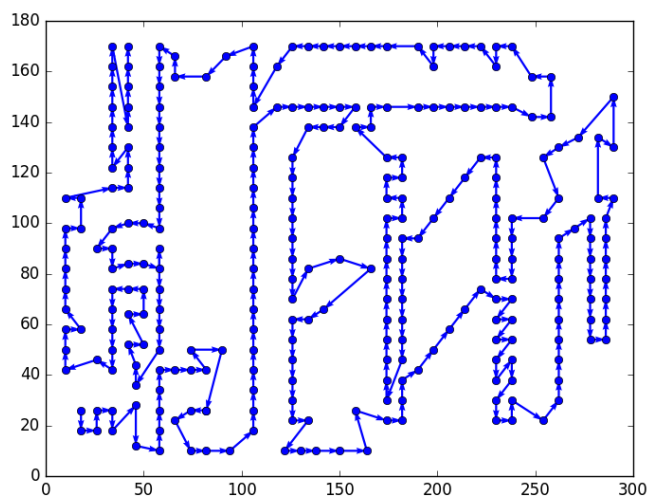


图 9: 改进贪婪算法得到的结果

4.3.2 原理

5 模型评价

参考文献

- [1] 楼建华. 数学建模与数学实验 [J]. 黑龙江高教研究, 2003(3): 126–127.
- [2] 饶卫振, 金淳. 基于求解 TSP 问题的改进贪婪算法 [J]. 运筹与管理, 2012, 21(6): 1–9.
- [3] GUTIN G, PUNNEN A P. The traveling salesman problem and its variations: Vol 12[M]. [S.l.]: Springer Science & Business Media, 2006.

- [4] 士勇, 永强, 研. 蚁群算法及其应用 [M]. [S.l.]: 哈尔滨工业大学出版社, 2004.
- [5] 马良, 项培军. 蚂蚁算法在组合优化中的应用 [J]. 管理科学学报, 2001, 4(2): 32–37.
- [6] HELD M, KARP R M. The traveling-salesman problem and minimum spanning trees[J]. Operations Research, 1970, 18(6): 1138–1162.