

遗传算法

1. 算法原理

遗传算法将择优与随机信息交换结合在一起，在每一代中，使用上一代中最好的，即最适应环境的位或片段，形成新的解集。

遗传算法是一个迭代的过程，在每次迭代中都保留一组候选解，并按某种优劣指标进行排序，然后按某种指标从中选择一些解，利用遗传算子，进行选择、交叉和变异等操作，对其进行运算以产生新一代的一组解。重复上述过程直到满足指定的收敛要求。

2. 算法步骤

(1) 初始化：设置进化代数计数器 $t=0$ ，设置最大进化代数 T ，随机生成 M 个个体作为初始群体 $P(0)$ 。

(2) 个体评价：计算群体 $P(t)$ 中各个个体的适应度。

(3) 选择运算：将选择算子作用于群体。选择的目的是把优化的个体直接遗传到下一代或通过配对交叉产生新的个体再遗传到下一代。选择操作是建立在群体中个体的适应度评估基础上的。

(4) 交叉运算：将交叉算子作用于群体。遗传算法中起核心作用的就是交叉算子。

(5) 变异运算：将变异算子作用于群体。即是对群体中的个体串的某些基因座上的基因值作变动。

(6) 群体 $P(t)$ 经过选择、交叉、变异运算之后得到下一代群体 $P(t+1)$ 。

(7) 终止条件判断：若 $t=T$ ，则以进化过程中所得到的具有最大适应度个体作为最优解输出，终止计算。

3. 算法参数

本次实验中，对于遗传算法步骤的各项参数设定如下

- (1) 群体大小为 20
- (2) 交叉概率为 90%
- (3) 变异概率为 10%
- (4) 基因长度为城市数

4. 精英保留策略

在算法步骤的第（6）步，如果仅用选择、交叉和变异三个遗传算子的标准遗传算法，产生下一代群体，是不能收敛到全局最优的。因为如果只是简单地进行杂交，很可能把较好的基因组合给破坏了，这样就没有达到积累较好基因的目的，反而把原本很好的基因给破坏了。

为了解决这个问题，使得最终结果为全局最优解，可以采用精英保留策略。具有精英保留策略的遗传算法是全局收敛的。精英个体是种群进化到当前为止搜索到的适应度最高的个体，它具有良好的基因结构和优良特性。这样在遗传进化的过程中，迄今为止出现的最优个体不会被遗传操作所破坏。

精英选择方法如下：设到第 t 代时，群体中 x 为最优个体。在新一代 $t+1$ 代群体中，若不存在比 x 更优的个体，就把上一代最优的个体 x 替换掉新一代的最差个体。

5. 遗传操作

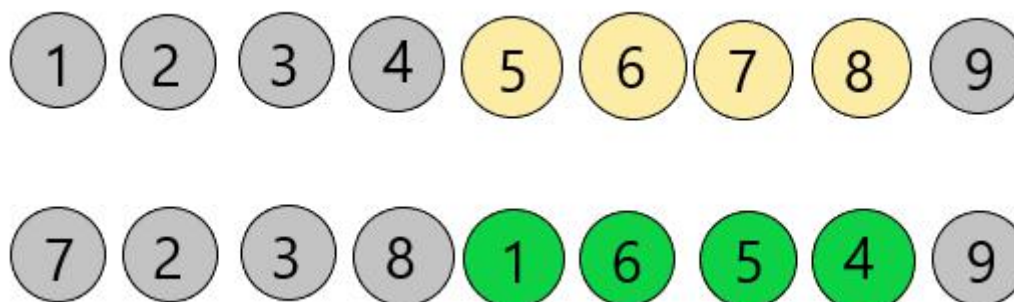
（1）选择

这里采用转盘赌选择法。该策略是先将个体的相对适应值 $\frac{f_i}{\sum f_i}$ 记为 p_i ，然后根据选择概率 $\{p_i, i=1, 2, \dots, N\}$ ，进行选择。

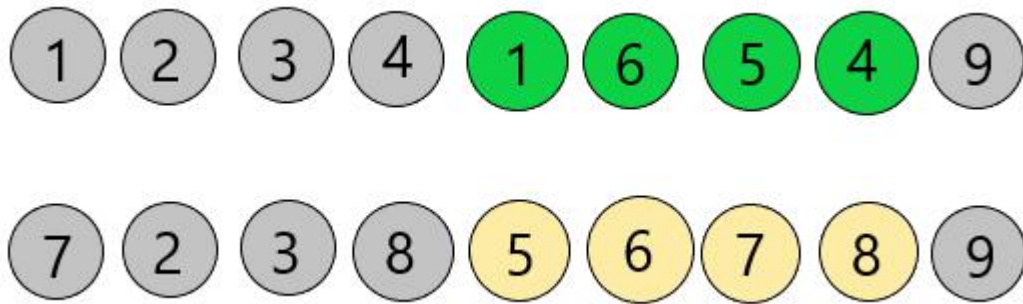
（2）交叉

这里采用的是多点交叉方法。进行交叉操作时要注意，有可能会产生冲突，需要解决冲突。交叉过程如下图所示：

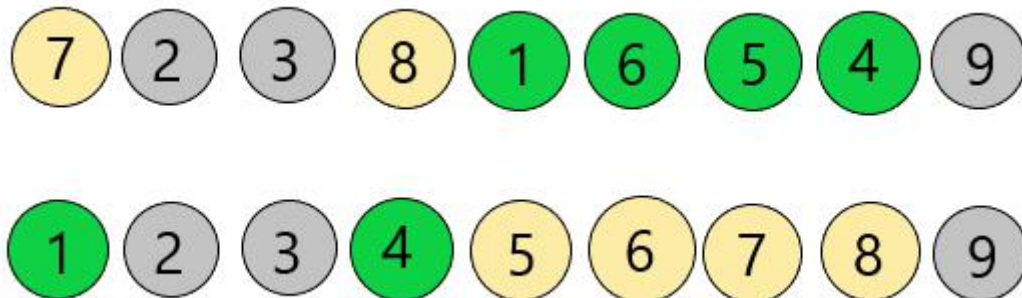
一开始两个个体的基因如下所示：



对应位置交换后得到：

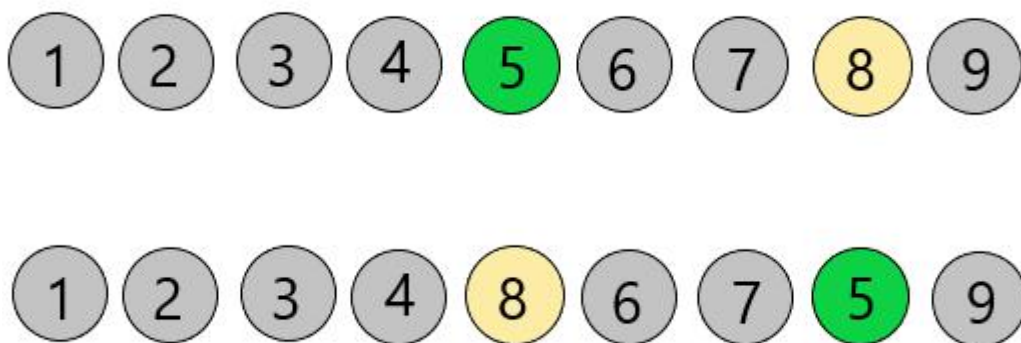


此时发现两个个体的基因出现了重复的，产生了冲突，此时要消除冲突。要解决冲突，就要把刚刚交换基因的部分之外的重复的基因替换成缺少的基因。，如下图所示：



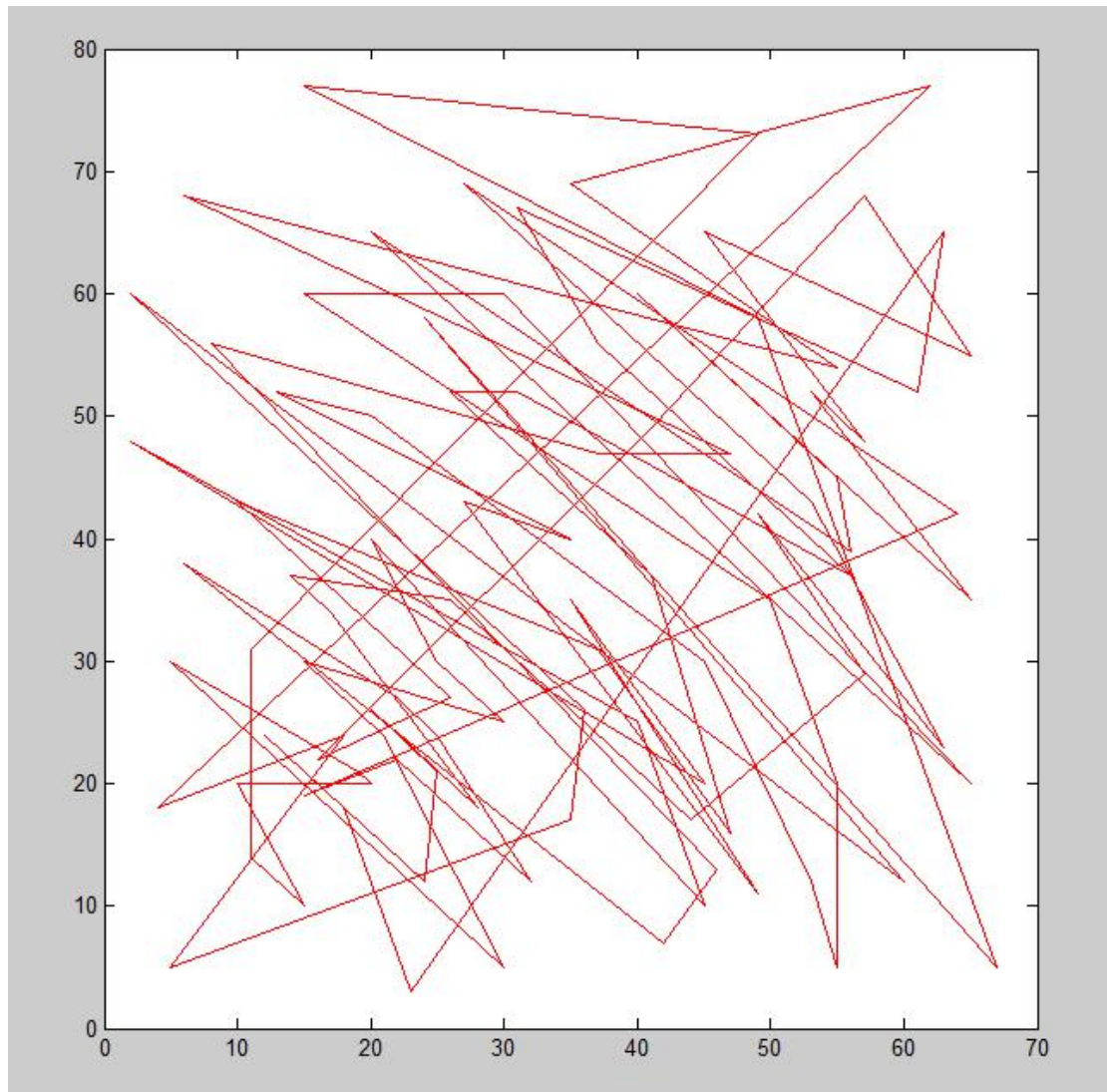
(3) 变异

变异过程就是算计交换两个城市的位置，如下图所示：



6. 实验结果

```
经过的总路径长度为：2579.37  
与最优解的误差为310.074%  
经过的路径为：  
1 85 94 8 53 72 28 88 31 80 70 90 78 35 43 100 61 42 97 99 87 60 83 89 73 54 76 55 77 63 29 9  
1 92 57 95 96 45 41 74 47 69 50 49 81 32 65 86 93 38 2 58 82 36 39 62 67 51 64 66 84 44 14 16  
37 98 17 15 59 5 13 6 18 22 40 46 21 101 75 52 27 48 7 26 56 23 4 12 19 10 25 11 68 3 30 24  
33 79 20 9 34 71 1
```



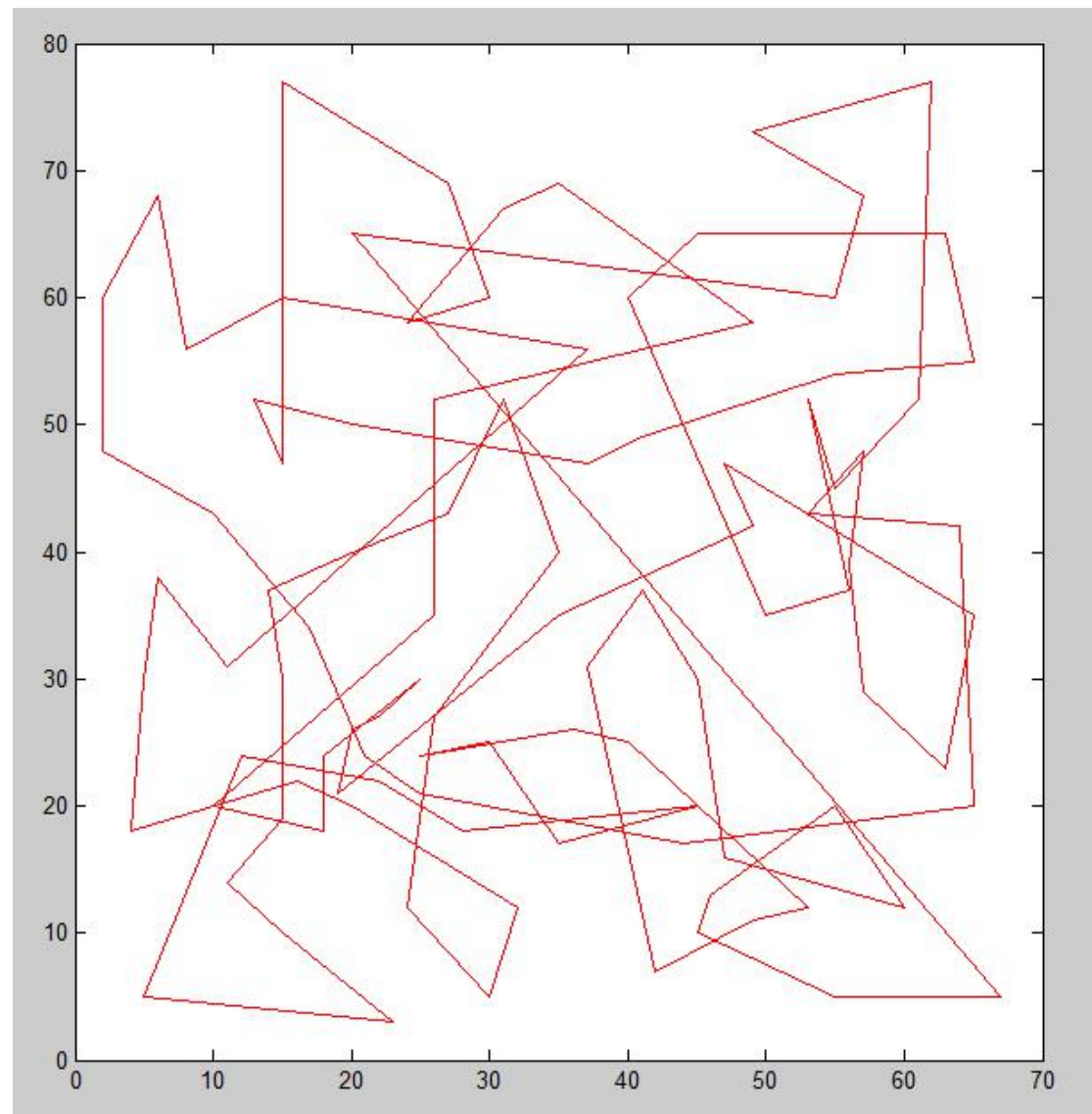
上图为一开始初始解的状态，虽然用了贪心算法，但是还是可以看到贪心算法产生的初始解误差很大，并且画得的路径图交叉程度较高。

经过的总路径长度为：1282.88

与最优解的误差为103.955%

经过的路径为：

```
1 69 7 48 82 64 63 10 62 90 32 51 88 89 16 100 93 6 96 99 98 101 76 50 24 55 54 68 79 77 29 2
5 73 97 59 60 8 46 36 49 47 19 70 84 45 17 86 85 37 57 15 42 94 27 31 52 18 83 5 91 44 14 43
38 61 92 87 21 2 13 95 58 40 56 75 41 53 28 26 72 39 4 74 22 23 67 11 9 71 66 65 78 3 33 80 1
2 30 20 35 34 81 1
```



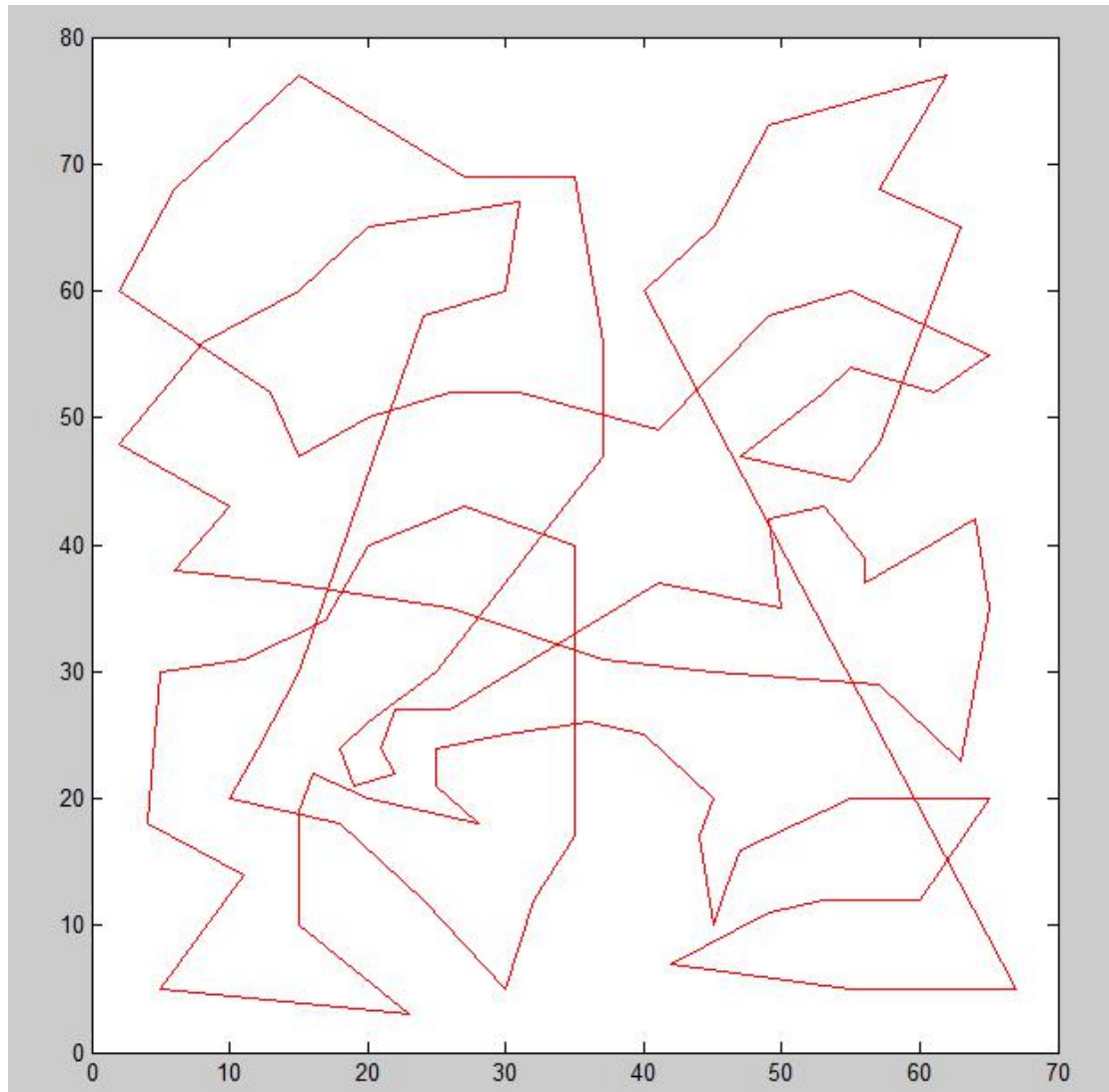
上图是运行了一段时间，误差为 103%时的路径图，此时的路径图交叉程度仍然较高，但相比初始状态已有改善。

经过的总路径长度为：889.8

与最优解的误差为41.4627%

经过的路径为：

1 31 88 7 82 48 36 49 64 63 32 70 69 6 99 93 98 92 59 96 94 28 12 76 77 68 80 29 24 55 54 26
53 89 83 45 8 46 47 19 11 90 10 62 5 61 16 100 42 15 57 2 101 27 52 18 60 84 17 86 44 38 43 1
4 91 85 37 87 97 95 13 58 40 21 73 22 74 72 4 25 39 56 75 41 23 67 30 20 66 65 71 35 79 3 50
33 81 78 34 9 51 1

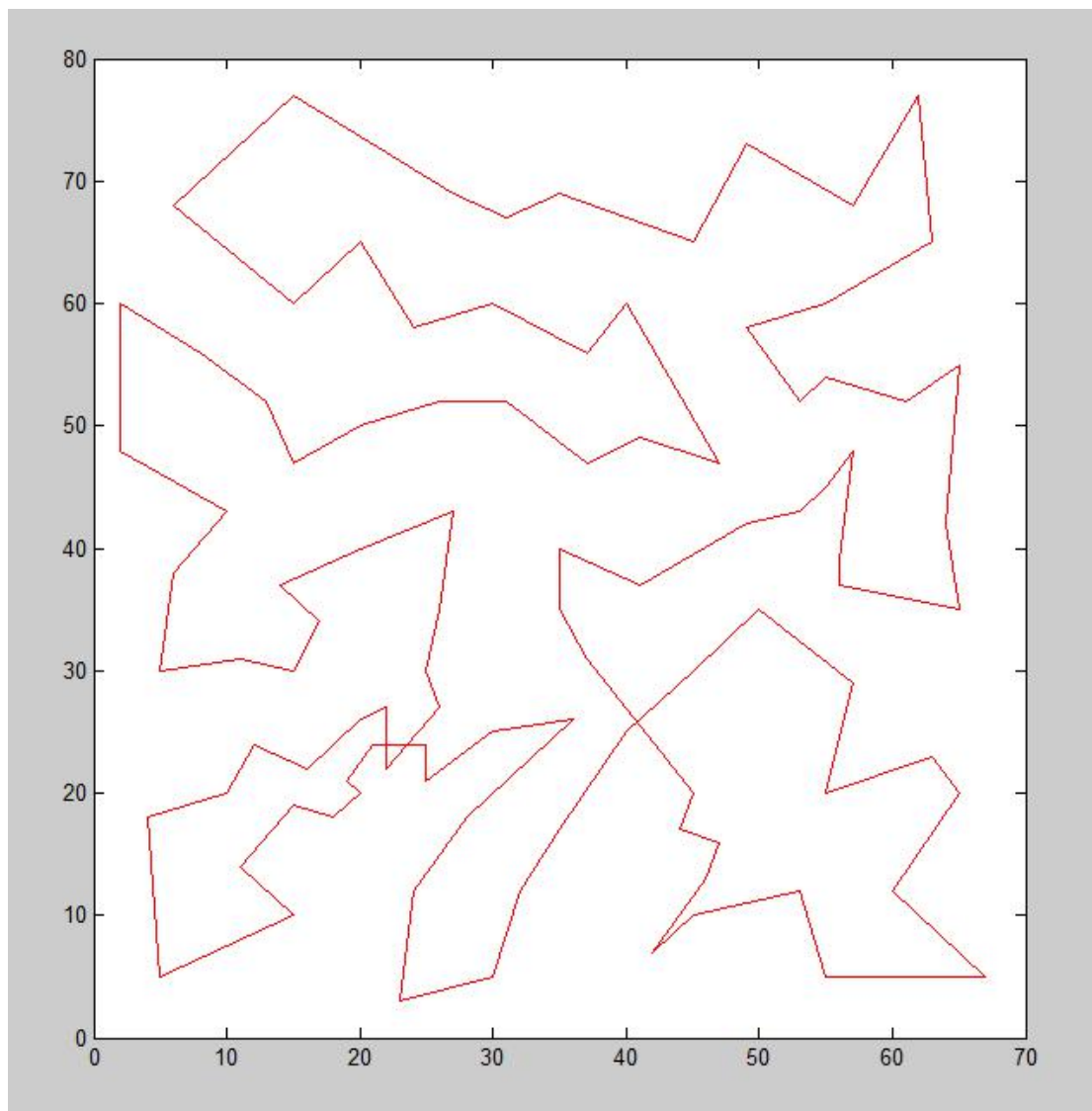


上图是运行了更长时间的截图，此时可以看到路径图的交叉程度已经有较大程度的改善，此时解的效果与最优解的误差为41%

```

经过的总路径长度为：687.692
与最优解的误差为9.33108%
经过的路径为：
1 50 30 70 10 62 11 19 49 64 63 90 32 20 66 71 65 35 9 51 33 81 78 34 29 24 80 68 79 3 77 76 28 27
101 53 21 73 72 74 41 22 75 56 23 67 39 25 55 4 54 12 26 40 2 57 15 43 42 87 58 13 97 95 59 98 37
100 91 44 14 38 86 16 61 85 93 99 96 92 94 6 89 52 18 83 60 5 84 17 45 8 46 36 47 48 82 7 88 31 6
9 1

```



上图是程序的最终运行结果得到的截图，此时路径图还是有一点交叉，解的效果与最优解的误差达到了 **9%**，但是此时的遗传算法陷入了局部最优，很难在获得较好的解。多次运行程序，得到的最终解的效果误差在 **10%**左右徘徊。