2.1 作业 1

Using the **routine simulannealbnd of Matlab**, minimize the following function,

$$f(x) = -e^{-2ln(2)(\frac{x-0.008}{0.854})sin^6(5\pi(x^{0.75}-0.05))}, x \in [0, 1].$$

Plot the **current iteration point**,the **function value**, and the **temperature function**.

使用 matlab 的 simulannealbnd，使用其中的 output function 存储数值并且输出，具体的细节，包括目标函数的图像如下面几张图所示：

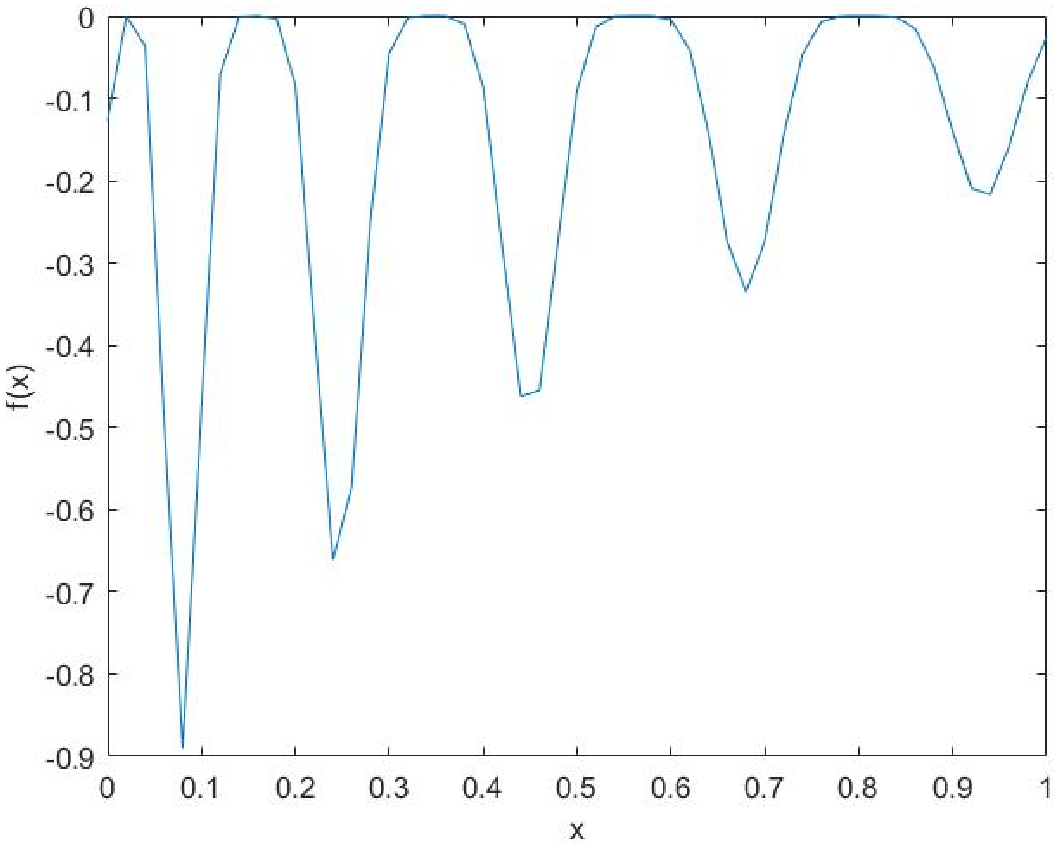

图 1: 目标函数的图像

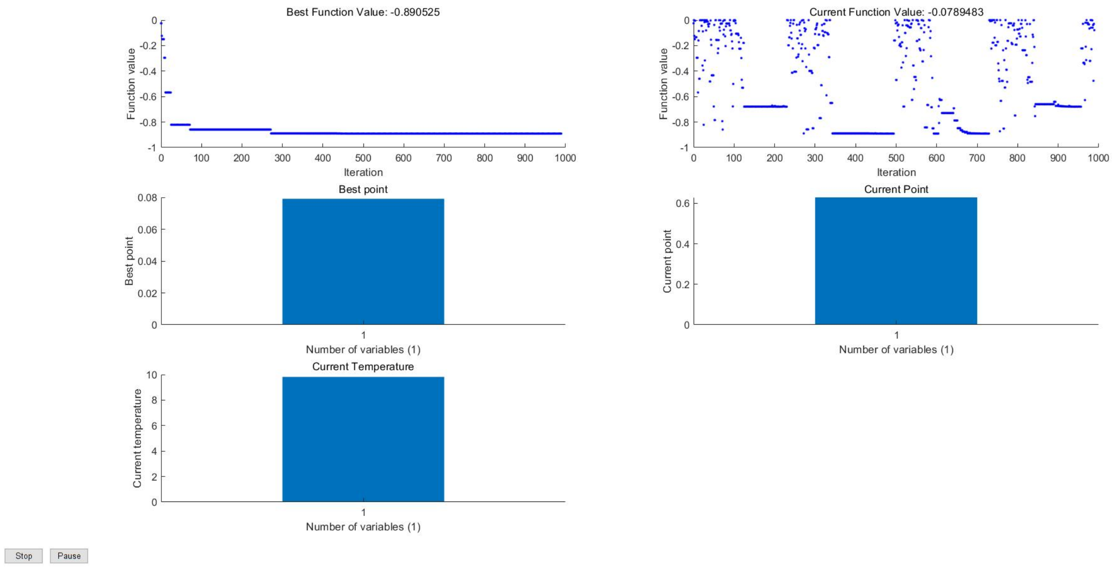

图 2: 模拟退火结果

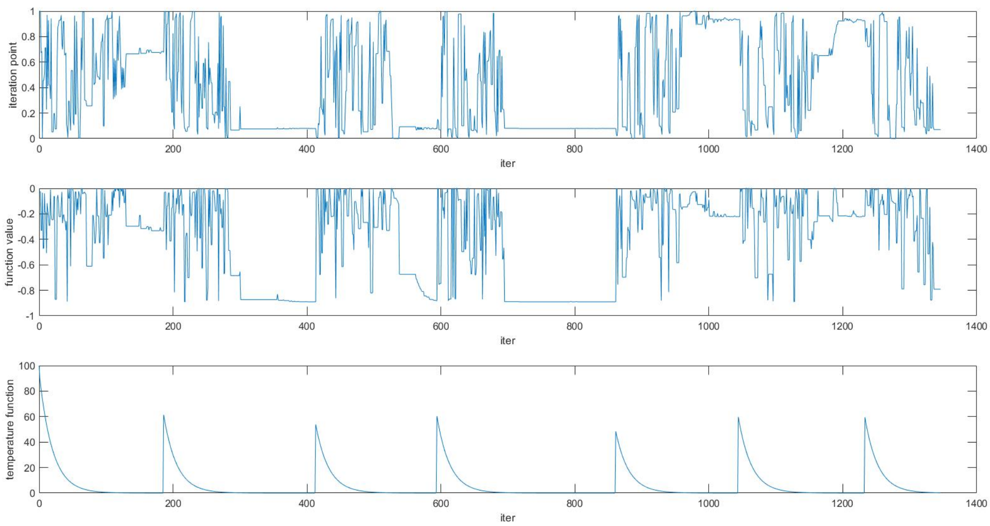

图 3: 模拟退火的 current iteration point,the function value, and the temperature function 图像 (横坐标为迭代次数)

2.2 作业 2

The Himmelblau function has four peaks in the points (3; 2),(-3.799; -3.283),(-2.805; 3.131), and (3.584; -1.848), and it is defined by

$$f(x_1, x_2) = \frac{2186 - (x_1^2 + x_2 - 11)^2 - (x_1 + x_2^2 - 7)^2}{2186}, x_1, x_2 \in [-6, 6].$$

由于是有四个极值点，事实上在每一次的 GA 算法的输出中，如果将搜索范围设置在 $x_1, x_2 \in [-6, 6]$ 的范围里面，每一次的输出的值几乎都是随机的，为了方便展示，这里将每一次范围进行了收缩，对于 x_1, x_2 的上下限做出了调整，这样寻找到四个特定的值的概率更大一些。在实际的寻找最优解的时候，可以先绘制函数的网格图和等高线图，具体如图 4 图 5 所示，这样的话就可以大致的确定最优解的范围，然后在收缩搜索的范围，就可以得出如图 6 到图 9 所示的相关的输出值。

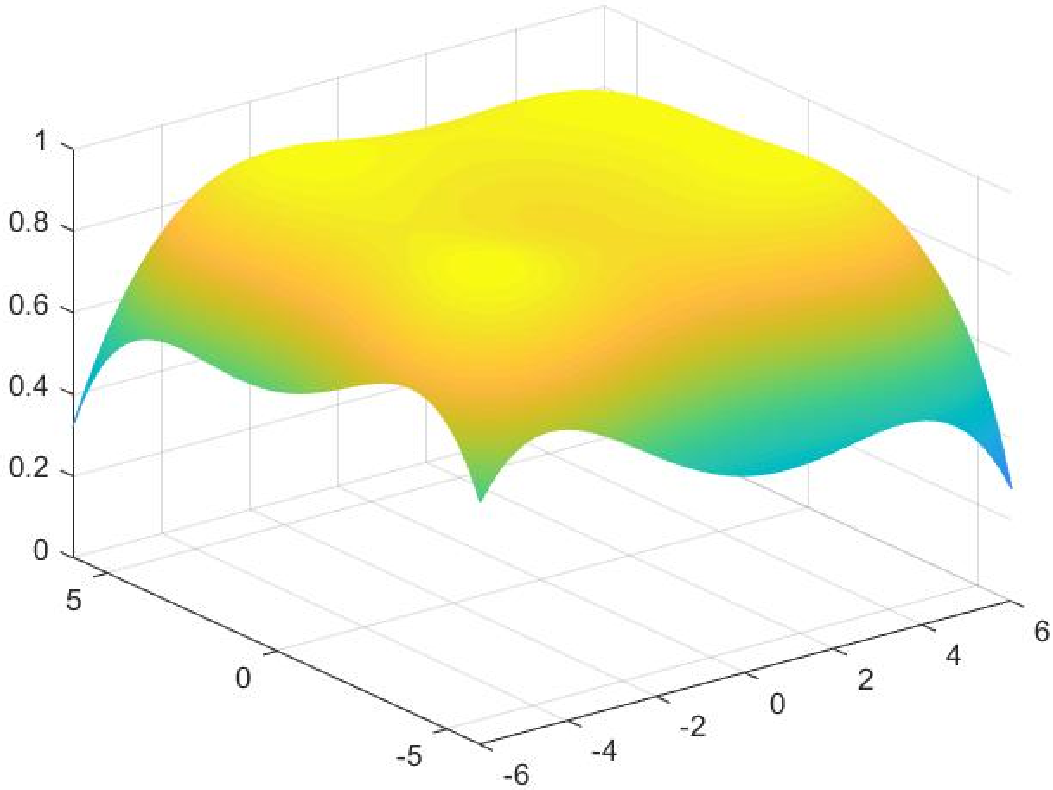

图 4: 函数的网格图

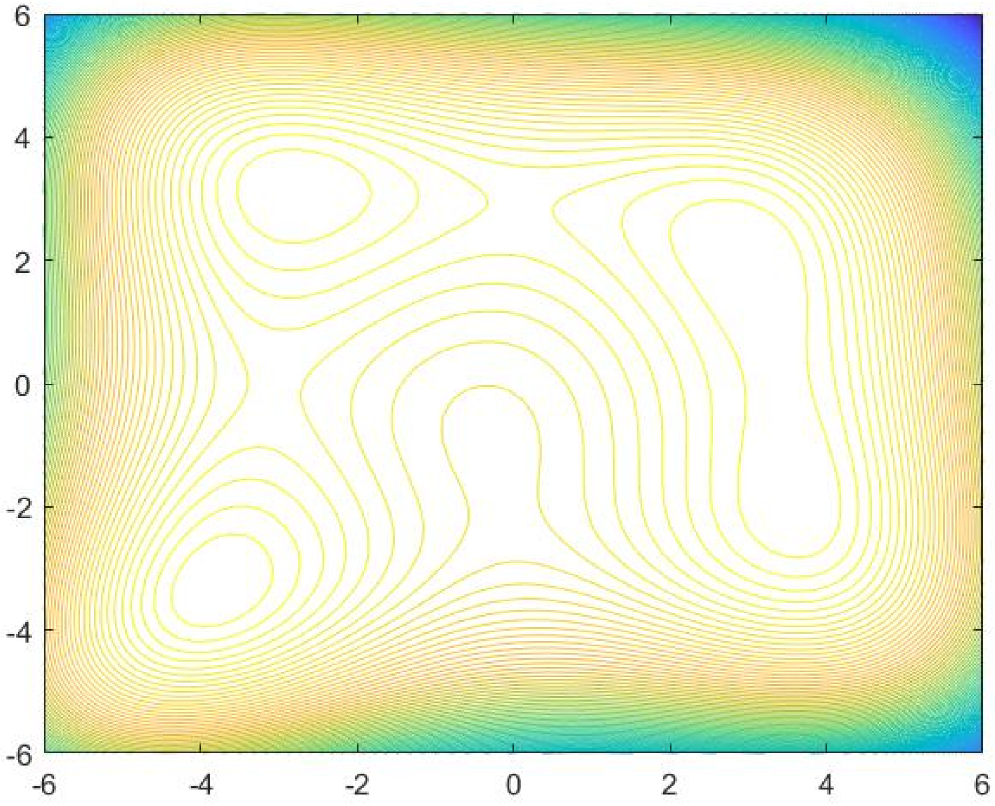

图 5: 函数的等高线图

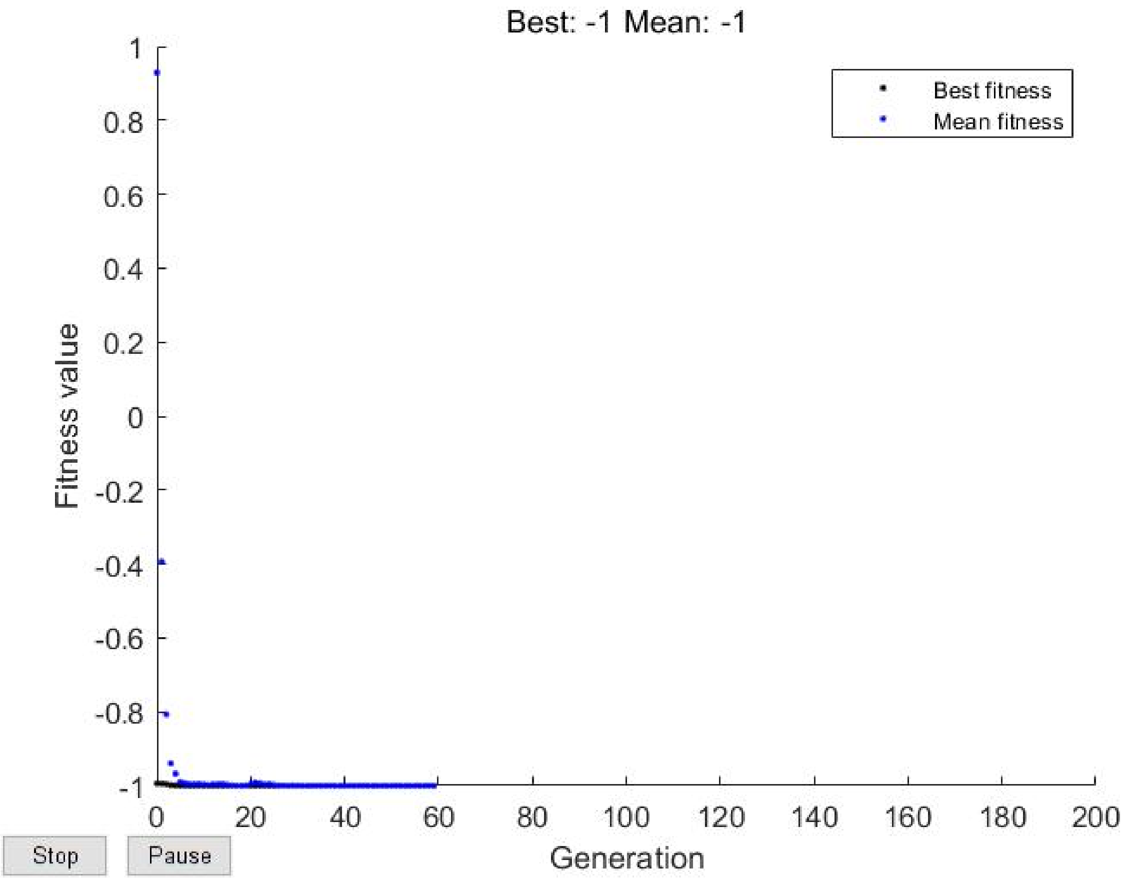

图 6: 输出值为 (3; 2) 的迭代情况

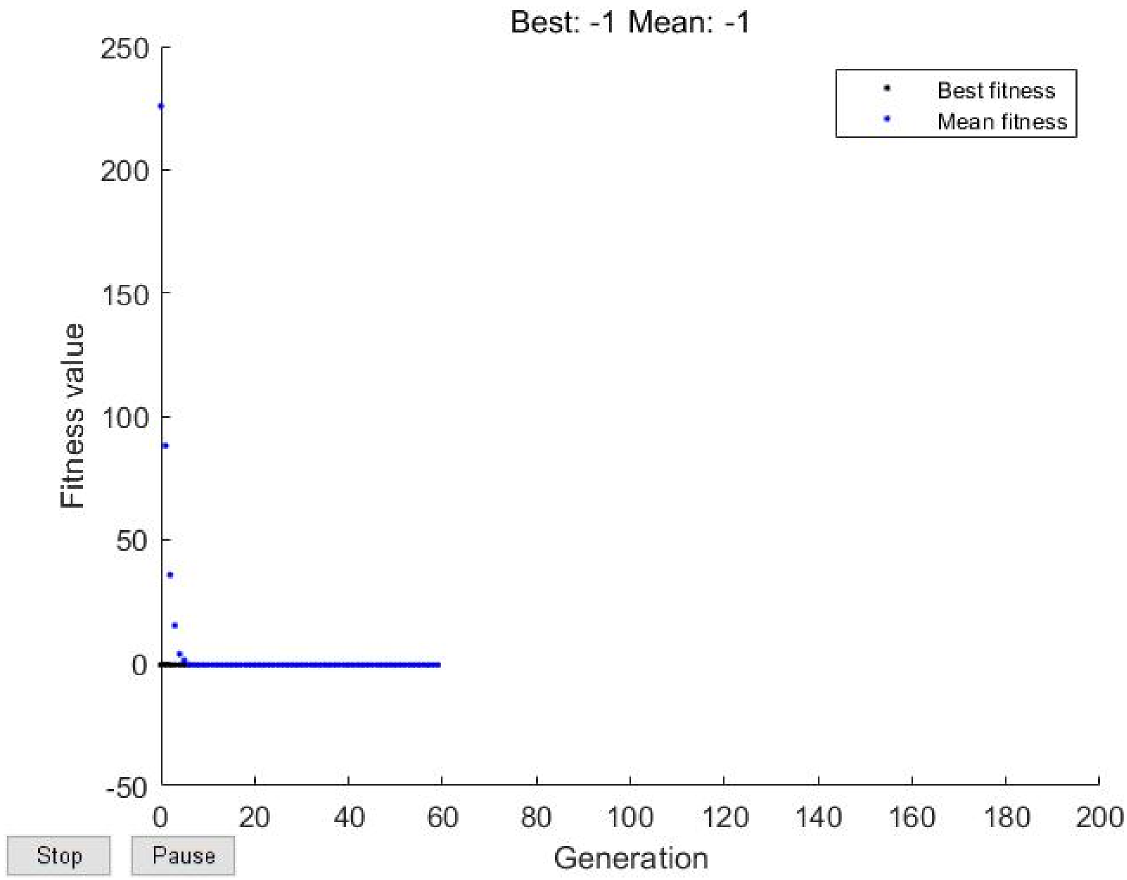

图 7: 输出值为 (-3.799; -3.283) 的迭代情况

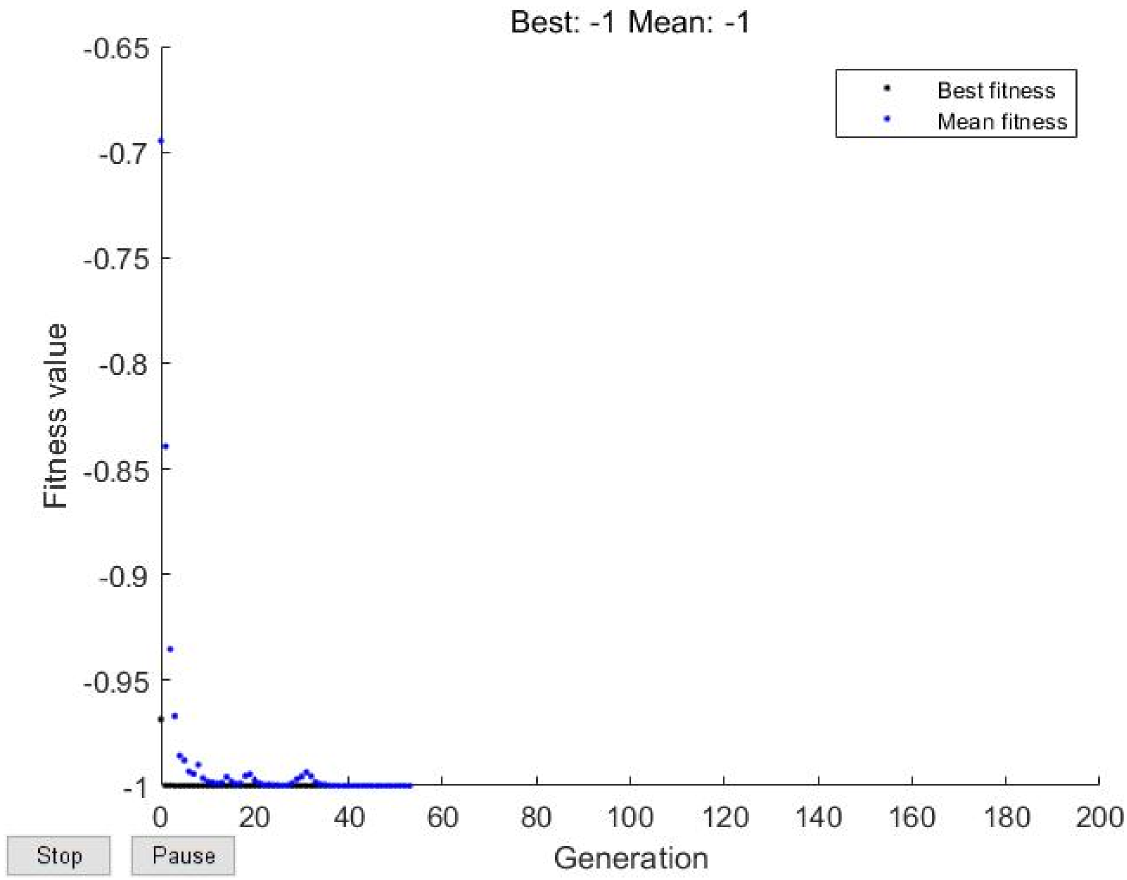

图 8: 输出值为 (-2.805,3.131) 的迭代情况

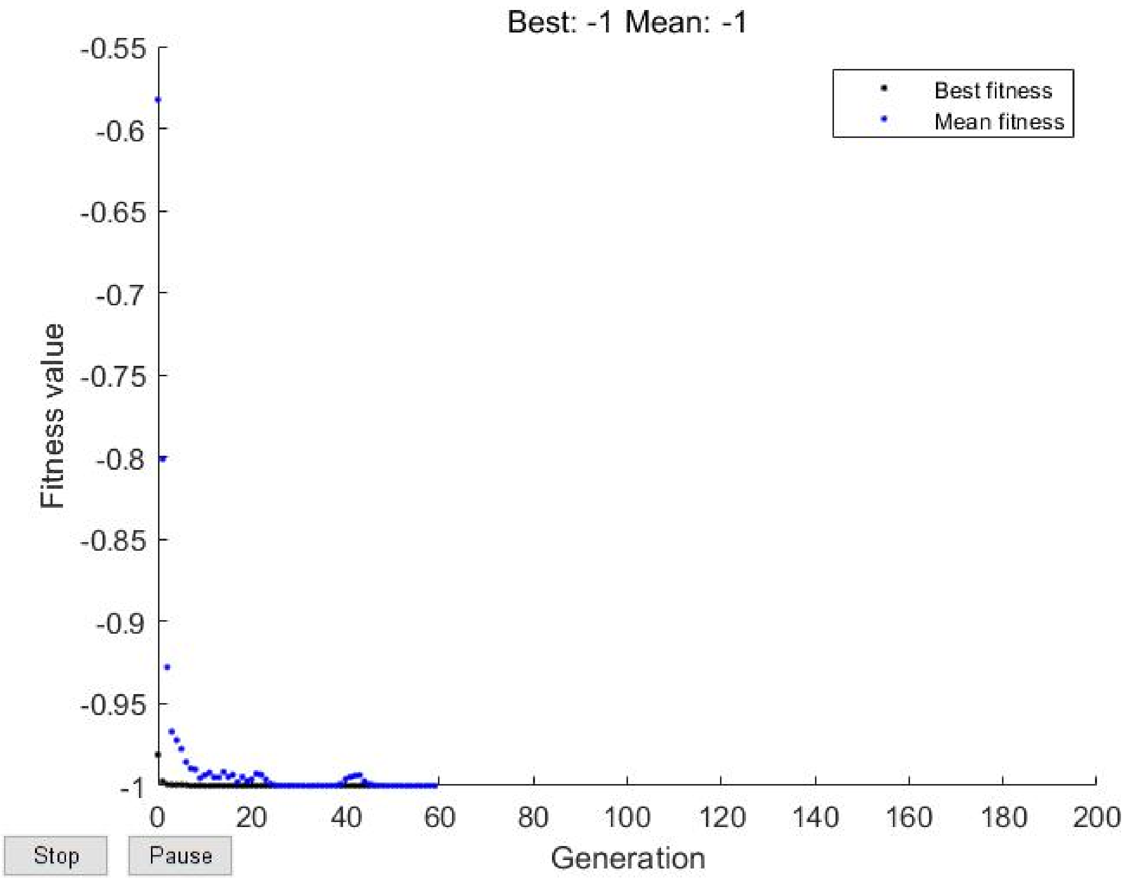

图 9: 输出值为 (3.584; -1.848) 的迭代情况