1

1.1

Minimizing the equation

$$f(x,y) = 100(y-x^2)^+(1-x)^2 \tag{1}$$

using the NelderMead simplex algorithm with the initial simplex vertices (x, y) : $(-1.2, 1.0), (-2, 1), (1, 0)$. The minimum found is points $(1, 1)$ with a function value $8.04756e - 15$ compared to the actual minimum $f(1, 1) = 0$.

1.2

Using the equation

$$f(x,y) = g(x,y)h(x,y) \tag{2}$$

where

$$g(x,y) = 1 + (x+y+1)^2(19-14x+3x^2-14y+6xy+3y^2) \tag{3}$$

and

$$h(x,y) = 30 + (2x-3y)^2(18-32x+12x^2+48y-36xy+27y^2) \tag{4}$$

To find four local minima the NelderMead simplex algorithm was used again with initial simplex vertice guesses based on approximations near the minima from visualization of the function.

Minimum (x,y)	f(x,y)
(-0.499202, -0.499202)	32.6845
(1, -0.334639)	94.6492
(1.8, 0.2)	84
(-0.335, -0.670001)	33.5259

2

Analyzing the data from measurements2.dat with parameters (x, y, σ) by fitting with the 4-parameter nonlinear model

$$f(x) = V_0 + Asin(\omega x - \phi) \tag{5}$$

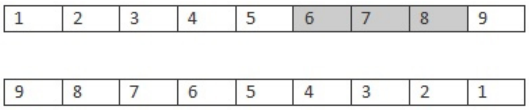
and minimizing the χ^2 using the NelderMead simplex algorthim

$$\chi^2 = \sum_{i=1}^{n}(\frac{f(x_i) - y_i}{\sigma_i})^2 \tag{6}$$

The resulting optimal parameters (V_0, A, ω, ϕ) are $(3.14139, 0.119546, 9.54645, 9.21187)$ with a χ^2 value of 82.9194

3

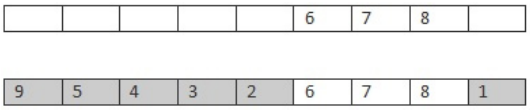
For solving the 130 city traveling salesman problem, the genetic algorithm was used. The implementation uses an arbitrary population size with initially randomized solutions to the problem. The population is randomly sampled from an arbitrary sample size, the two lowest cost solutions are combined to form a child in the next generation of the population. This process is continues until the child generation is of equal size to the preceding parent population. The parent solutions are combined by selecting a random continuous subset of city traversals from the first parent and placing them in the same order and location in the child. Then the non-repeating elements from the second parent are placed into the child. The breeding process is illustrated below.
Parents



1	2	3	4	5	6	7	8	9

9	8	7	6	5	4	3	2	1

Child



					6	7	8	

9	5	4	3	2	6	7	8	1

source: http://www.theprojectspot.com/tutorial-post/applying-a-genetic-algorithm-to-the-travelling-salesman-problem/5

Each crossover is also subject to mutations with an arbitrary rate of mutation. The mutation swaps two random cities in the trip solution.
When the algorithm is run with a population size of 75 and sample size of 20, the consistently achievable lowest cost solutions hover around 39000, but the 1 time optimal solution generated was 37602.8 after 53 generations.