Homeproblem 1

Anders Segerlund
851017-4934

September 21, 2021

Problem 1.1

1

With the objective function $f(x_1, x_2) = (x_1 - 1)^2 + 2(x_2 - 2)^2$ that should be minimized subject to constraint $g(x_1, x_2) = x_1^2 + x_2^2 - 1 \leq 0$. The function $f_p(\mathbf{x}; \mu)$ is definde as:

$$f_p(\mathbf{x}; \mu) = \begin{cases} (x_1 - 1)^2 + 2(x_2 - 2)^2 + \mu(x_1^2 + x_2^2 - 1)^2, & \text{if } g(x_1, x_2) \geq 0. \\ (x_1 - 1)^2 + 2(x_2 - 2)^2, & \text{otherwise.} \end{cases} \tag{1}$$

2

The partial derivatives of the function $f_p(\mathbf{x}; \mu)$ are obtained as:

$$\nabla f_p(\mathbf{x}; \mu) = \left(\frac{\partial f_p}{\partial x_1}, \frac{\partial f_p}{\partial x_2} \right)^T \tag{2}$$

$$\frac{\partial f_p}{\partial x_1} = \begin{cases} 2(x_1 - 1) + 4x_1\mu(x_1^2 + x_2^2 - 1), & \text{if } g(x_1, x_2) > 0. \\ 2(x_1 - 1), & \text{otherwise.} \end{cases} \tag{3}$$

$$\frac{\partial f_p}{\partial x_2} = \begin{cases} 4(x_2 - 2) + 4x_2\mu(x_2^2 + x_2^2 - 1), & \text{if } g(x_1, x_2) > 0. \\ 4(x_2 - 2), & \text{otherwise.} \end{cases} \tag{4}$$

3

The unconstrained minimum for $\mu = 0$ is obtained by setting the function $\nabla f_p(\mathbf{x}; \mu = 0) = 0$ and solve for x_1 and x_2

$$2(x_1 - 1) = 0 \qquad x_1 = 1 \tag{5}$$
$$4(x_1 - 2) = 0 \qquad x_2 = 2 \tag{6}$$

4

See matlab code provided

5

The result from running the penalty algorithm is displayed in Figure 1 with parameter values $\eta = 0.0001$, $T = 10^-6$ and start values $x^* = (1, 2)^T$. What can be noted is that both x values converge towards a steady value at approximately $\mu = 100$, which can be seen in Figure 2

penalty	x1	x2
0.1	0.6963	1.6419
10	0.33135	0.9955
100	0.31374	0.9553
1000	0.31179	0.9507

Figure 1: Results after running penalty algorithm

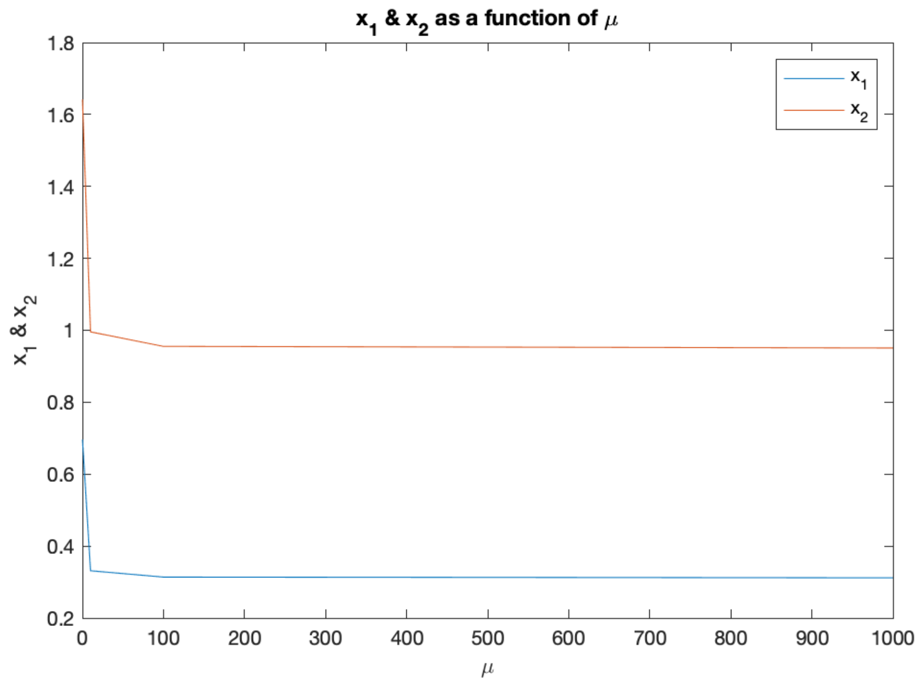

Figure 2: Results after running penalty algorithm

Problem 1.2

a

In order to determine the global min $(x_1^*, x_2^*)^T$ and the function value of $f(x_1, x_2) = 4x_1^2 - x_1x_2 + 4x_2^2 - 6x_2$ we start by first considering the stationary points, taking the partial derivatives of the function $f(x_1, x_2)$ and set it to zero and solve for x_1 and x_2

$$\frac{\partial f}{\partial x_1} = 8x_1 - x_2 = 0 \qquad x_1 = \frac{2}{21} \tag{7}$$

$$\frac{\partial f}{\partial x_2} = -x_1 + 8x_2 - 6 = 0 \qquad x_2 = \frac{16}{21} \tag{8}$$

Thus the first stationary points is $P_1 = (2, 16)^T/21$

The next step is to consider the points on the edges. I will denote the edge between $(0, 0)$ and $(0, 1)$ as Edge a. $(0, 1)$ and $(1, 1)$ as Edge b, and $(0, 0)$ and $(1, 1)$ as Edge c

Edge a

At edge a x_1 is constant at zero and $f(0, x_2)$ and $f'(0, x_2)$ are obtained as.

$$f(0, x_2) = 4(0)^2 - (0)x_2 + 4x_2^2 - 6x_2 \tag{9}$$

$$f'(0, x_2) = 8x_2 - 6 = 0 \qquad x_2 = \frac{3}{4} \tag{10}$$

Thus the next stationary point is $P_2 = (0, 3/4)^T$

Edge b

At edge b $x_2 = 1$, thus $f(x_1, 1)$ and $f'(x_1, 1)$ are obtained as:

$$f(x_1, 1) = 4x_1^2 - x_1(1) + 4(1)^2 - 6(1) \tag{11}$$

$$f'(x_1, 1) = 8x_1 - 1 = 0 \qquad x_1 = \frac{1}{8} \tag{12}$$

Thus the next stationary point is $P_3 = (1/8, 1)^T$

Edge c

At edge c $x_1 = x_2$ thus $f(x_1, x_1)$ and $f'(x_1, x_1)$ are obtained as:

$$f(x_1, x_1) = 4x_1^2 - x_1x_1 + 4x_1^2 - 6x_1 \qquad = 7x_1^2 - 6x_1 \tag{13}$$

$$f'(x_1, x_1) = 14x_1 - 6 = 0 \qquad x_1 = \frac{3}{7} \tag{14}$$

Thus the next stationary point is $P_4 = (3, 3)^T/7$

Finally all points including the corner points $P_5 = (0, 0)$, $P_6 = (0, 1)$ and $P_7 = (1, 1)$ are evaluated by the objective function to determine the minimum of the function $f(x_1, x_2)$ is $P_1 = (2, 16)^T/21$ which can be seen in Table 1

Point	x_1	x_2	$f(x_1, x_2)$
P_1	2/21	16/21	-2,28 *(min)*
P_2	0	3/4	-2,25
P_3	1/8	1	-2,06
P_4	3/7	3/7	-1,28
P_5	0	0	0
P_6	0	1	-2
P_7	1	1	1

Table 1: Points and functional values

b

In order to minimise function $f(x_1, x_2) = 15 + 2x_1 + 3x_2$ subject to constraint $h(x_1, x_2) = x_1^2 + x_1x_2 + x_2^2 - 21$ The Lagrange function $L(x_1, x_2, \lambda)$ is defined as:

$$L(x_1, x_2, \lambda) = 15 + 2x_1 + 3x_2 + \lambda(x_1^2 + x_1x_2 + x_2^2 - 21) \tag{15}$$

Taking the partial derivatives and set them to zero we obtain:

$$\frac{\partial L}{\partial x_1} = 2 + 2x_1\lambda + \lambda x_2 = 0 \qquad \lambda = \frac{-2}{2x_1 + x_2} \tag{16}$$

$$\frac{\partial L}{\partial x_2} = 3 + x_1\lambda + 2x_2\lambda = 0 \qquad \lambda = \frac{-3}{x_1 + 2x_2} \tag{17}$$

$$\frac{\partial L}{\partial \lambda} = x_1^2 + x_1x_2 + x_2^2 - 21 = 0 \tag{18}$$

By setting eqn (16) = (17) we obtain:

$$x_2 = 4x_1 \tag{19}$$

Using eqn (19) in (18) we obtain

$$x_1^2 + x_1 4x_1 + (4x_1)^2 - 21 = 0 \quad \Rightarrow \quad 21x_1^2 = 21 \quad \Rightarrow \quad x_1 = \pm 1 \tag{20}$$

Thus with (19) we obtain x_2 as:

$$x_2 = \pm 4 \tag{21}$$

Inserting this into the objective function we see that the minimum $f(x_1^*, x_2^*) = P_1 = (-1, -4)$ as the point as can be seen in Table 2

x_1	x_2	$f(x_1, x_2)$
1	4	29
-1	4	25
1	-4	5
-1	-4	1 *(min)*

Table 2: x-values values and corresponding function value

Problem 1.3

a)

A first run of the function indicates that that the steady state point $(x_1^*, x_2^*)^T$ should be found in the vicinity of $(3, 0.5)^T$

The simulation is made with the following parameters:
tournamentSize =2;
tournamentProbability = 0.75;
crossoverProbability = 0.5;
mutationProbability = 0.02;
numberOfGenerations = 2000;

As there were no restrictions on the parameters above, the numbers of generations is set large since this will result in a solution that converges close to the steady point, as this gives the algorithm longer time to improve. In the simulation it was also obvious that reducing the number of generations would result in a function value that is near, but not zero. Secondly the mutation probability is set to optimum value of 0.02 which is related to the Onemax function. (see below). While remaining parameters seem to be pretty well tunes from start and is not changed in my simulation. The result from the simulation can be seen in Figure 3.

b)

The fitness is achieved by running the algorithm 100 times and take the median out of the result for every value of $p_{mut} \in [0, 0.02, 0.1425, 0.265, 0.3875, 0.51, 0.6325, 0.755, 0.8775, 1]$ The median fitness from all iterations along with the median variable value from each run can be seen in Table 4.

The best fitness is achieved at mutation probability of $0.02 = 1/m$ and worst fitness is obtained when no mutation is carried out, or when mutation takes place on all chromosomes.

Considering Darwin's evolution theory this is an expected result, since the process of copying genetic information *sometimes* occur with an error, which may improve the fitness of the next offspring. Which one also can conclude by looking at Figure 4

Iteration	x1	x2	g(x1,x2)
1	3	0.5	0
2	3	0.5	0
3	3	0.5	0
4	3	0.5	0
5	3	0.5	0
6	3	0.5	0
7	3	0.5	0
8	3	0.5	0
9	3	0.5	0
10	3	0.5	0

Figure 3: First simulation of GA algorithm

mutationProbability	Fitness	Function value	Variable values	
0	119	0.008393	3.00014847517456	0.500545367614787
0.01	6869	0.000146	2.99999916553495	0.499999687075606
0.02	118686299	0	3.00000348687182	0.500001028180153
0.03	1927932	1e-06	2.99998828768695	0.500009223819054
0.1425	5325	0.000188	2.99777635925342	0.498935893146273
0.265	1447	0.000691	3.00333166132366	0.502354815672482
0.3875	870	0.001149	3.00744929931907	0.50135643784274
0.51	655	0.001526	3.01158958708017	0.502487286999443
0.6325	617	0.001622	3.01230618394334	0.502825841391857
0.755	699	0.00143	2.99457499368712	0.500288918622998
0.8775	1947	0.000514	2.98317664811542	0.494720354518901
1	158	0.006327	3.01257261671342	0.504160389428151

Figure 4: Result of running GA 100 times

The reason why 0.02 is resulting in the best fitness is due to the Onemax function which find the binary string of a given length that maximizes the sum of its digits.

The algorithm returns the highest fitness for the mutation probability = 0.02 which corresponds to $1/numberOfGenes$ and lowest for 0 and 1 which can be seen in Figure 5

c)

In order to analytically prove the actual minimum of the function $g(x_1^*, x_2^*)$ we start by assuming that the minimum is is in the vicinity of $(x_1^*, x_2^*) = (3, 0.5)^T$ after looking in Figure 4. In order for $(3, 0.5)^T$ to be a minimum value the following equation must be satisfied:

$$\nabla g(x_1, x_2) = 0 \tag{22}$$

The function:

$$g(x_1, x_2) = (1.5 - x_1 + x_1 x_2)^2 + (2.25 - x_1 + x_1 x_2^2)^2 + (2.625 - x_1 + x_1 x_2^3)^2 \tag{23}$$

Can for simplicity be written as

$$g(x_1, x_2) = g_1(x_1, x_2) + g_2(x_1, x_2) + g_3(x_1, x_2) \tag{24}$$

Thus:

$$\nabla g(x_1, x_2) = \nabla g_1(x_1, x_2) + \nabla g_2(x_1, x_2) + \nabla g_3(x_1, x_2) \tag{25}$$

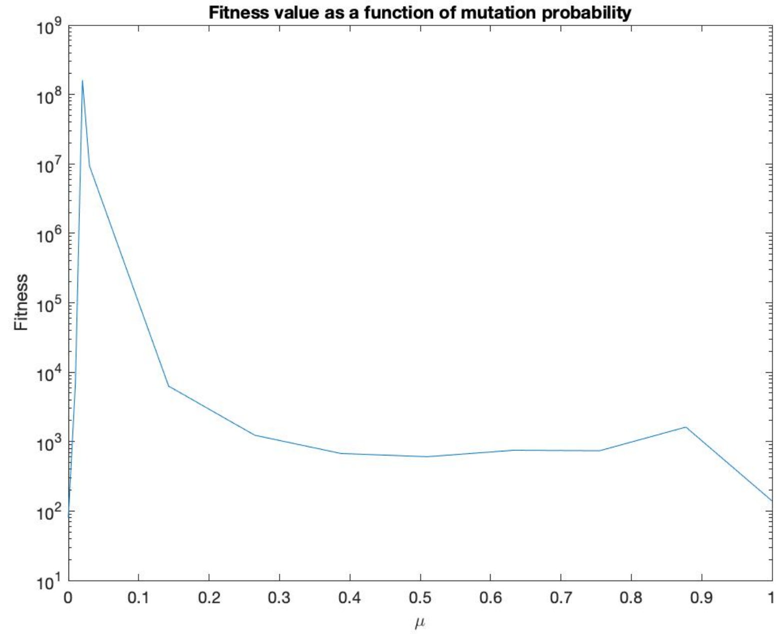

Figure 5: Fitness as a function of mutation probability

$$\nabla g_1(x_1,x_2) = \begin{bmatrix} 2(1.5 - x_1 + x_1x_2)(x_2 - 1) \\ 2(1.5 - x_1 + x_1x_2)x_1 \end{bmatrix} \Rightarrow \nabla g_1(3, 0.5) = (0,0)^T \tag{26}$$

$$\nabla g_2(x_1,x_2) = \begin{bmatrix} 2(2.25 - x_1 + x_1x_2^2)(x_2 - 1) \\ 2(1.5 - x_1 + x_1x_2^2)(2x_1x_2) \end{bmatrix} \Rightarrow \nabla g_2(3, 0.5) = (0,0)^T \tag{27}$$

$$\nabla g_3(x_1,x_2) = \begin{bmatrix} 2(2.625 - x_1 + x_1x_2^3)(x_2^3 - 1) \\ 2(2.625 - x_1 + x_1x_2^3)(3x_1x_2^2) \end{bmatrix} \Rightarrow \nabla g_3(3, 0.5) = (0,0)^T \tag{28}$$

As all the gradient terms are zero we can conclude that $(x_1^*, x_2^*) = (3, 0.5)^T$ is a stationary point of $g(x_1, x_2)$