Lab 3 Report

Aakash Banik (16i190010)
IE 684

January 25, 2018

1 Solution 1

1.1 Subpart 4

The optimal solution is 1.99566310588 while the minimum value is 0.114419710898.

1.2 Subpart 5

The routine took 63 iterations.

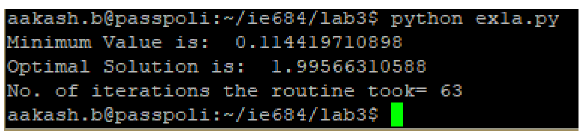

Figure 1: Optimal solution and the no. of iterations the routine took

1.3 Subpart 6

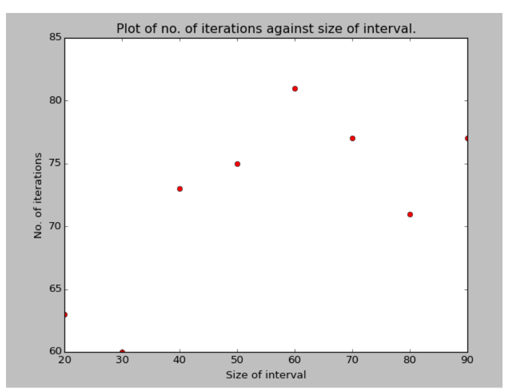

Figure 2: No. of iterations against size of interval

Here we plot the no. of iterations required to find the minima against the size of the interval and try to analyze it. We see a graph with an increasing trend in general but theres no specific pattern to it. This is because the size of the interval affects the no. of iterations sharply till some point around 40 and then stabilizes around a bound.

1.4 Subpart 7

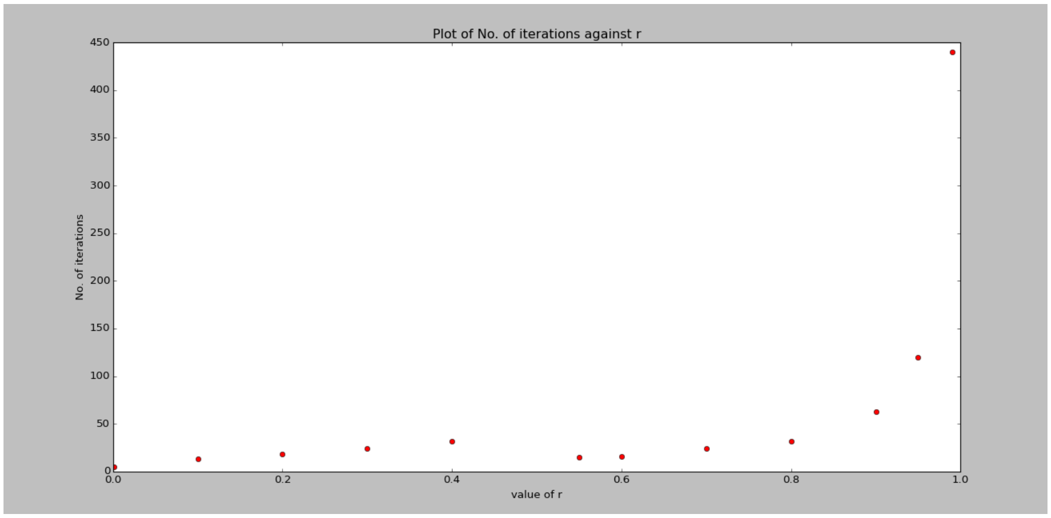

Figure 3: No. of iterations against different values of r

Here we plot the no. of iterations required to find a minima against different values of r. We see an increasing trend in the no. of iterations with increasing r except around the point 0.5 but then increases rapidly when the value of r is close to 1. r basically acts as a scaling parameter and scales the interval between a and b accordingly.

1.5 Subpart 8

Function where this algorithm fails to find the correct minimum is: $f(x) = x^2$

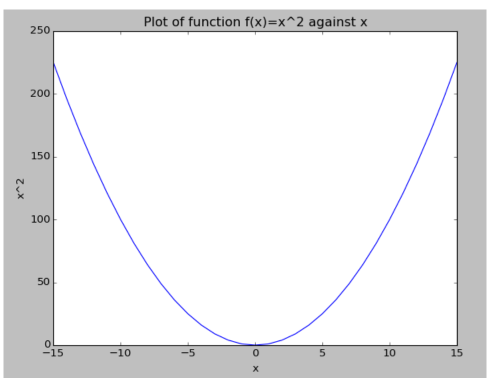

Figure 4: Plot of f(x)=x^2 against x

The algorithm fails in this case because the stopping criteria of the algorithm is satisfied after the first iteration. This is because the algorithm compares the functional value at points 'a' and 'b' and if the functional value at these two points are very close to each other, the algorithm returns either of 'a' or 'b' as the minima. In this case since $f(x) = x^2$ is a symmetric function around zero so any choice of 'a=k' and 'b=-k', where k $\in \mathbb{R} \setminus \{0\}$ will satisfy the stopping criteria without giving the actual minima as an answer.

2 Solution 2

2.1 Subpart 2

```
aakash.b@passpoli:~/ie684/lab3$ python ex2a.py
Objective Value:  30.8760970388
Optimal Solution:  0.487218665644 4.93909766679 4.94518790012
No. of iterations:  71
```

Figure 5: Objective value and optimal solution

2.2 Subpart 3

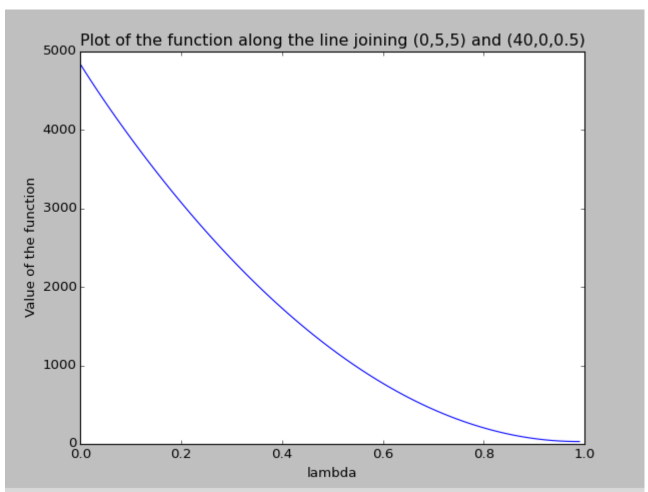

Figure 6: Plot of the function along the line joining the points (0,5,5) and (40,0,0.5)

We can see from the plot that the function has a minima when lambda is close to 1.This means that the optimal solution of the function is $\lambda(0, 5, 5) + (1 - \lambda)(40, 0, 0.5)$ with the value of $\lambda = 1$; which given us an optimal solution of (0,5,5). The optimal value calculated by the algorithm is close to this value. So the algorithm worked well in this example.

3 Solution 3

3.1 Subpart 2

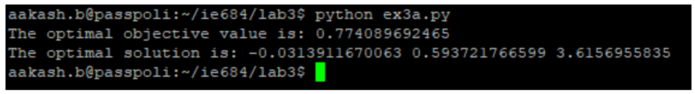

Figure 7: Objective value and Optimal solution as obtained from the algorithm

The objective value is: 0.774089692465 The optimal solution is: -0.0313911670063 0.593721766599 3.6156955835

3.2 Subpart 3

Algorithm:

r $\leftarrow$ 0.9
initialize $y, d, \lambda = 1$
initialize l=(l1,l2,l3)=y
while $f(y + \lambda d) < f(y)$ **do**
    $\lambda = \lambda + 1$
initialize $u = (u1, u2, u3) = y + \lambda d$
while *not converged* **do**
    $b1 \leftarrow l1 + r(u1 - l1)$
    $a1 \leftarrow l1 - r(u1 - l1)$
    $b2 \leftarrow l2 + r(u2 - l2)$
    $a2 \leftarrow l2 - r(u2 - l2)$
    $b3 \leftarrow l3 + r(u3 - l3)$
    $a3 \leftarrow l3 - r(u3 - l3)$
    if $f(a) < f(b)$:
        u1=b1
        u2=b2
        u3=b3
    else: l1=a1
        l2=a2
        l3=a3
return Objective value= $f(l1, l2, l3)$; Optimal Solution= $l1, l2, l3$