|  |  |
| --- | --- |
|  |  |
| Homework 7 Report |  |
|  |  |
|  | 11/30/20  EGR-7040 |
|  | JACK YOUNG |

Table of Contents

[Table of Figures 2](#_Toc57619060)

[Problem Description 3](#_Toc57619061)

[Figure 1: Shows problems outlined by assignment 3](https://raidermailwright-my.sharepoint.com/personal/young_330_wright_edu/Documents/School/Fall%20Semester%202020%20-%20Masters%20Mechanical%20Engineering/EGR%207040/Homework/Homework6/Homework%206%20report.docx#_Toc57619062)

[1 – SLP Method 4](#_Toc57619063)

[Figure 2:Python SLP method optimization case 4](https://raidermailwright-my.sharepoint.com/personal/young_330_wright_edu/Documents/School/Fall%20Semester%202020%20-%20Masters%20Mechanical%20Engineering/EGR%207040/Homework/Homework6/Homework%206%20report.docx#_Toc57619064)

[2 – SQP with GS step size determination 5](#_Toc57619065)

[Figure 3: Python SQP method with GS search 5](https://raidermailwright-my.sharepoint.com/personal/young_330_wright_edu/Documents/School/Fall%20Semester%202020%20-%20Masters%20Mechanical%20Engineering/EGR%207040/Homework/Homework6/Homework%206%20report.docx#_Toc57619066)

[3 – SQP with BFGS hessian updating and GS step size 6](#_Toc57619067)

[Figure 4:Python SQP method with GS step size and BFGS updated hessian approximation 6](https://raidermailwright-my.sharepoint.com/personal/young_330_wright_edu/Documents/School/Fall%20Semester%202020%20-%20Masters%20Mechanical%20Engineering/EGR%207040/Homework/Homework6/Homework%206%20report.docx#_Toc57619068)

[Figure 5: Matlab SQP with GS step size and BFGS updated Hessian 7](https://raidermailwright-my.sharepoint.com/personal/young_330_wright_edu/Documents/School/Fall%20Semester%202020%20-%20Masters%20Mechanical%20Engineering/EGR%207040/Homework/Homework6/Homework%206%20report.docx#_Toc57619069)

[6 – Conclusions 8](#_Toc57619070)

[Appendix A: Code 9](#_Toc57619071)

# Table of Figures

[Figure 1: Shows problems outlined by assignment 3](https://raidermailwright-my.sharepoint.com/personal/young_330_wright_edu/Documents/School/Fall%20Semester%202020%20-%20Masters%20Mechanical%20Engineering/EGR%207040/Homework/Homework6/Homework%206%20report.docx#_Toc57619054)

[Figure 2:Python SLP method optimization case 4](https://raidermailwright-my.sharepoint.com/personal/young_330_wright_edu/Documents/School/Fall%20Semester%202020%20-%20Masters%20Mechanical%20Engineering/EGR%207040/Homework/Homework6/Homework%206%20report.docx#_Toc57619055)

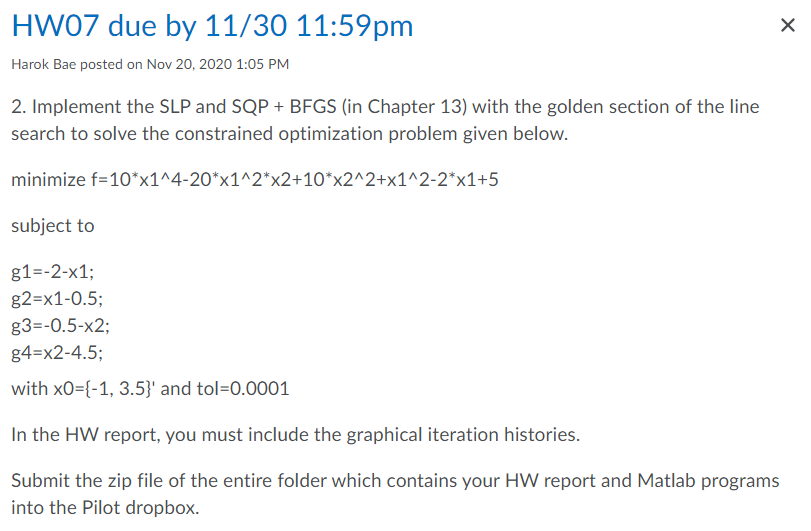
[Figure 3: Python SQP method with GS search 5](https://raidermailwright-my.sharepoint.com/personal/young_330_wright_edu/Documents/School/Fall%20Semester%202020%20-%20Masters%20Mechanical%20Engineering/EGR%207040/Homework/Homework6/Homework%206%20report.docx#_Toc57619056)

[Figure 4:Python SQP method with GS step size and BFGS updated hessian approximation 6](https://raidermailwright-my.sharepoint.com/personal/young_330_wright_edu/Documents/School/Fall%20Semester%202020%20-%20Masters%20Mechanical%20Engineering/EGR%207040/Homework/Homework6/Homework%206%20report.docx#_Toc57619057)

[Figure 5: Matlab SQP with GS step size and BFGS updated Hessian 7](https://raidermailwright-my.sharepoint.com/personal/young_330_wright_edu/Documents/School/Fall%20Semester%202020%20-%20Masters%20Mechanical%20Engineering/EGR%207040/Homework/Homework6/Homework%206%20report.docx#_Toc57619058)

# Problem Description

The work given for this project can be described by Figure 1.



## Figure : Shows problems outlined by assignment

These requirements involved:

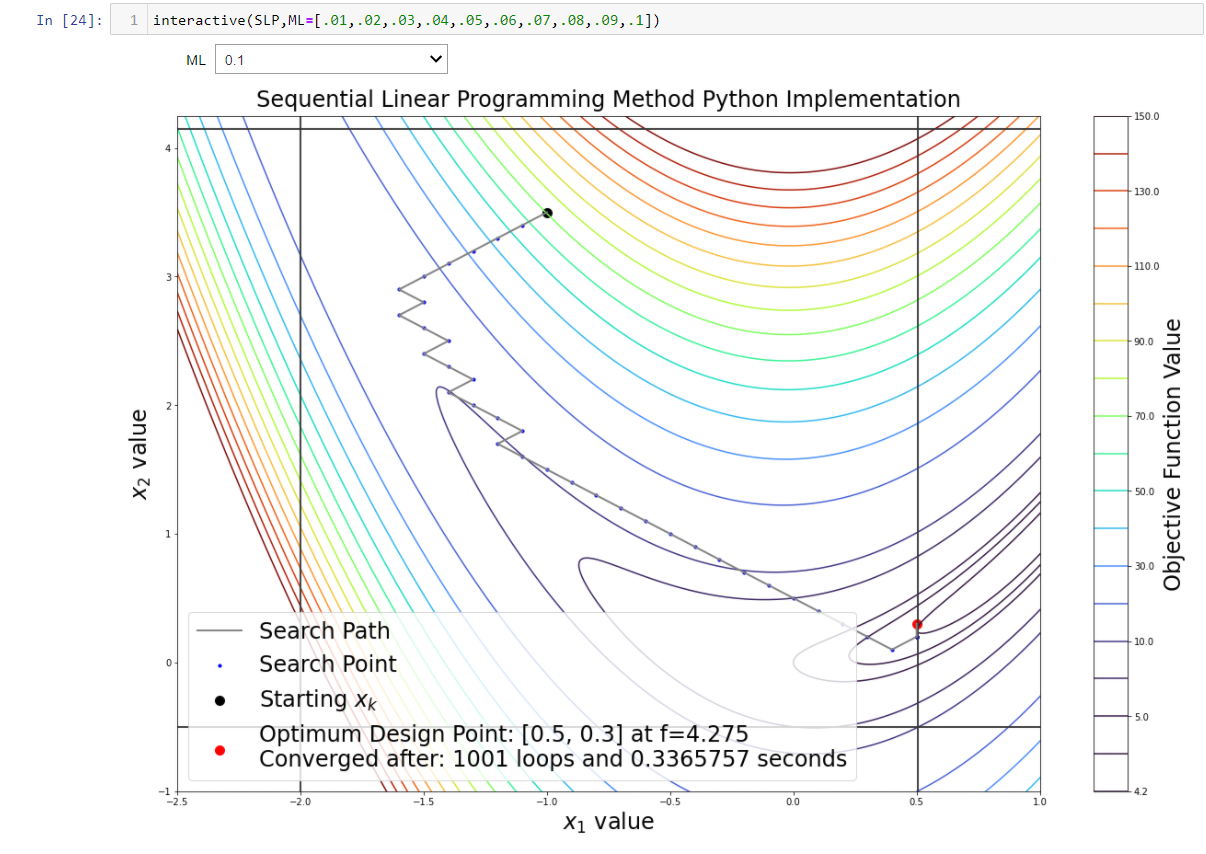
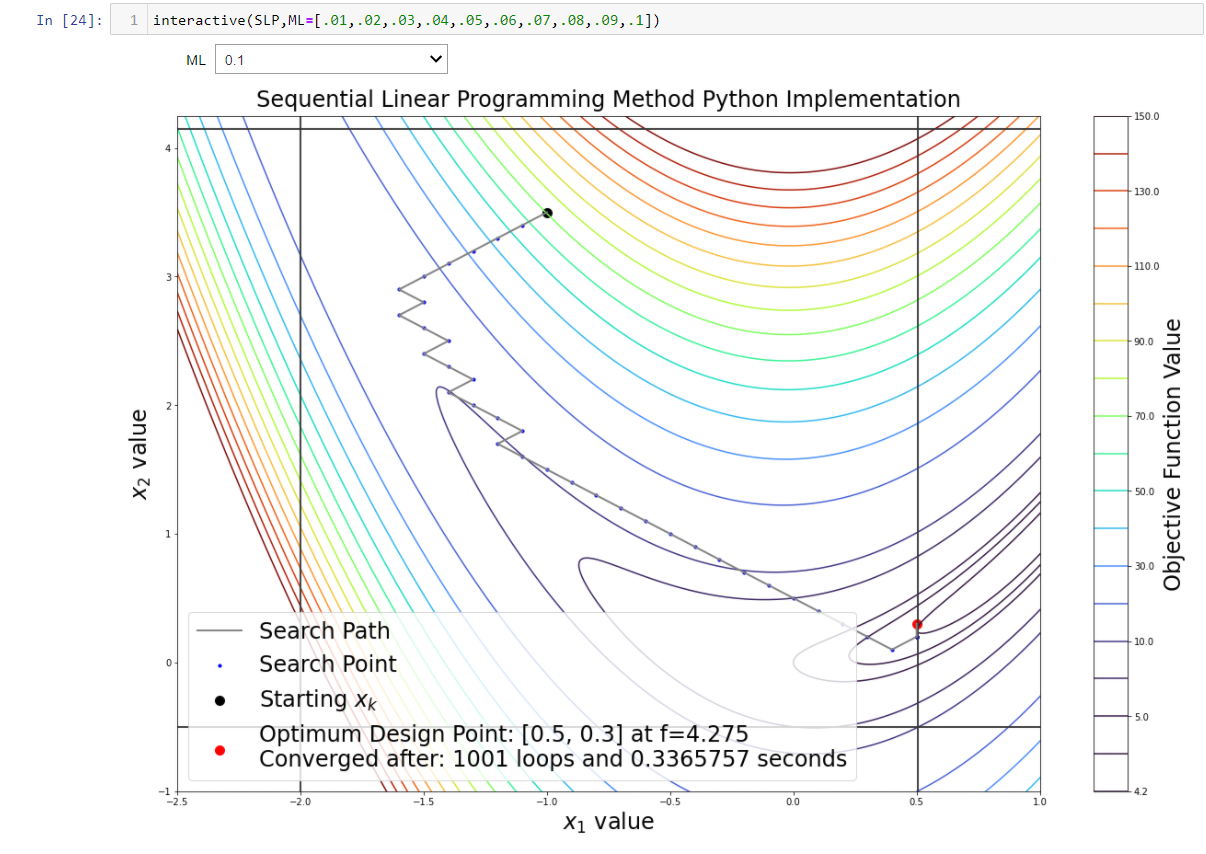
• 1 – Implement sequential linear programming (SLP)

• 2 – Implement sequential quadratic programming (SQP) with golden section (GS) step size

• 3 – Implement Broyden-Fletcher-Goldfarb-Shanno (BFGS) with GS step size

•Use starting x0={-1,3.5} and a tolerance of .0001.

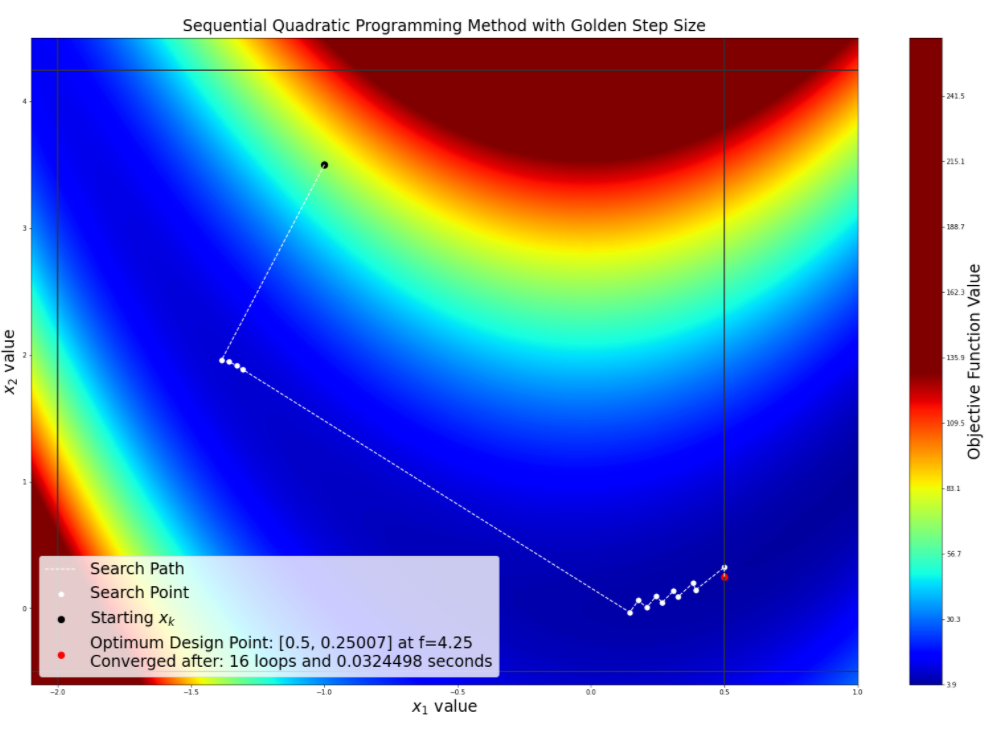
# 1 – SLP Method

 This method was successfully implemented in both Matlab and python. The python results are shown below in Figure 2. This method entails using the sequential linear programming method tableau methodology. This methodology only converged at an x2 of .3 this is due to the step size ML which was selected by the code. Other step sizes were used and showed to converge on the point more closely. This method terminated only after being stopped by a loop counter this is due to the step size being too large to accurately approach the minimum point. In order to implement this in python a function called linprog was imported and used which has the same functionality as in Matlab.

## Figure :Python SLP method optimization case

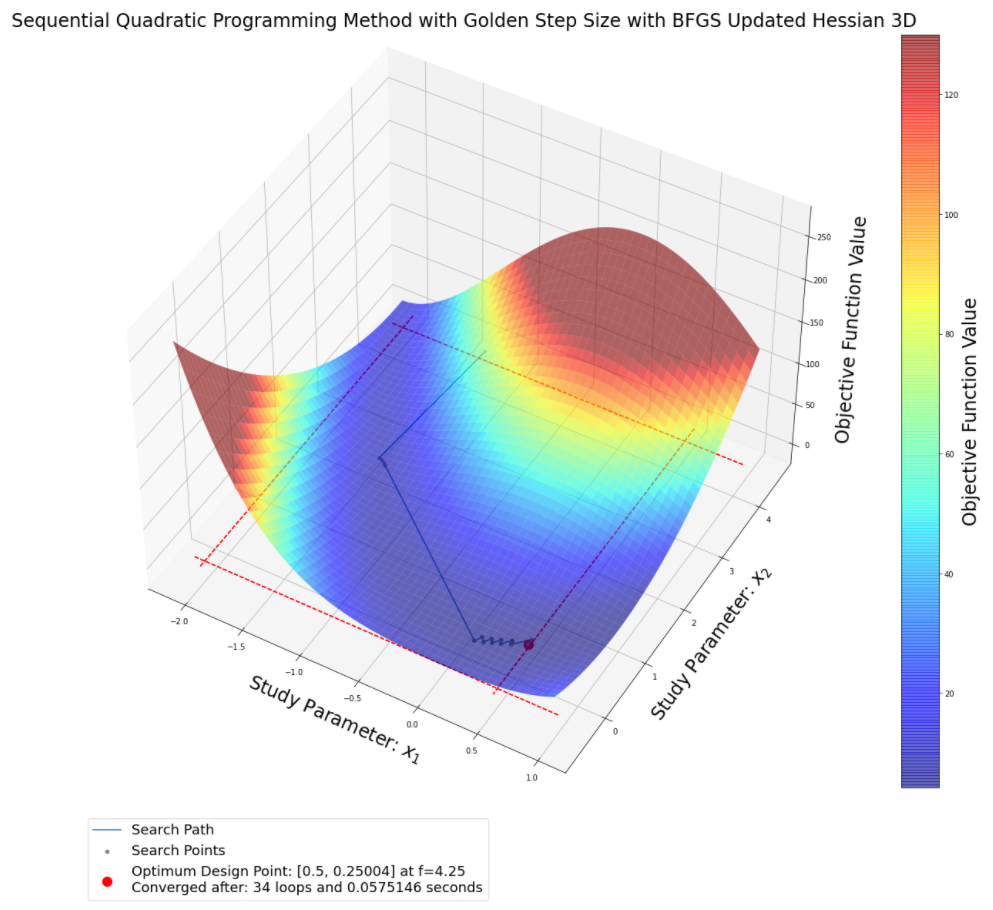
# 2 – SQP with GS step size determination

The SQP method showed much better search behavior than that of SLP. This is due to the dynamic ability of the algorithm to change its step size as well as its search direction. Additonally, the methodology used by sequential quadratic programming is higher order in nature and allows for better choices of the search direction to be made. Once the search direction was chosen by quadprog (in both Matlab and python) the algorithm then determined the step size with the golden section method. The hessian matrix for this method was set as the identity matrix. This method converged on a minimum point in 16 loops and 32 milliseconds. This was the best convergence behavior shown by any of the algorithms tested during this report. The next method involved a more dynamic hessian.

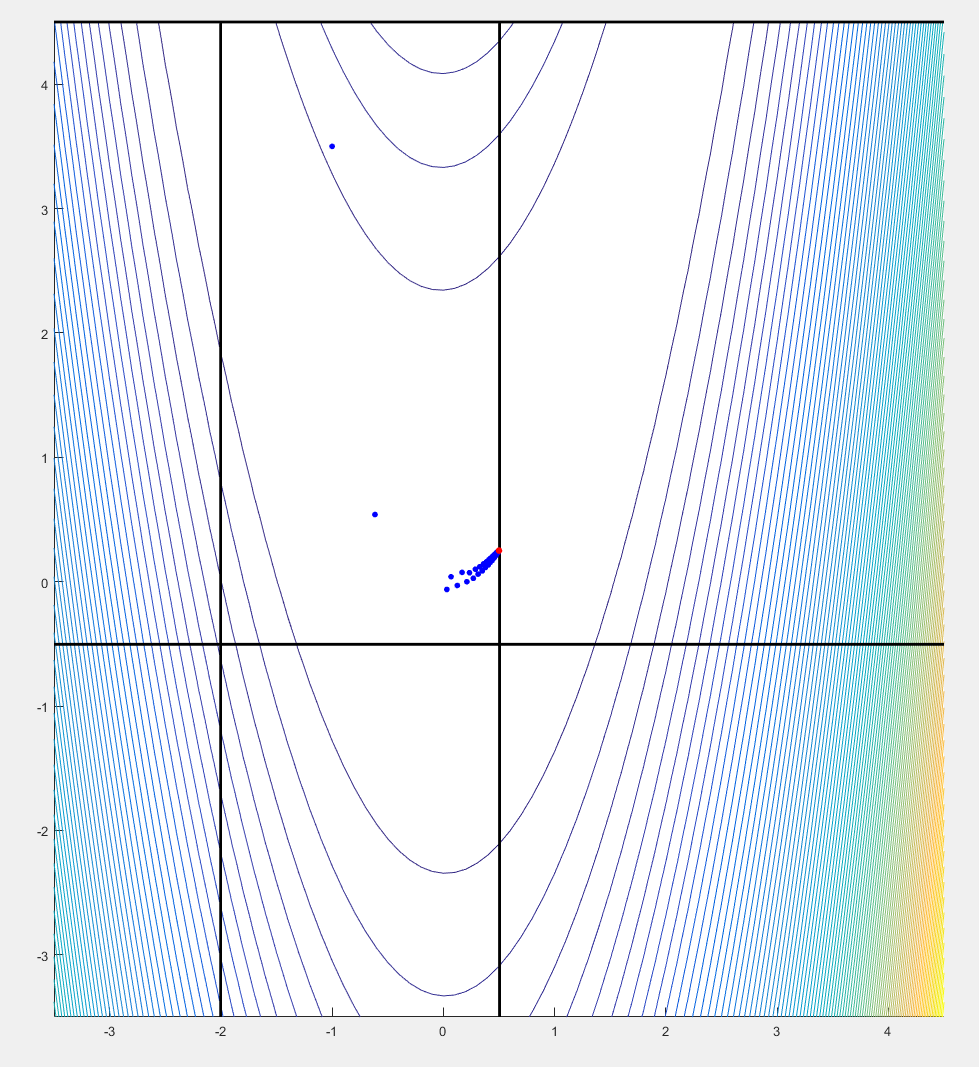


## Figure : Python SQP method with GS search

# 3 – SQP with BFGS hessian updating and GS step size

 This methodology gave some trouble during the python implementation as the updating of the hessian matrix went through iterations. However, the values of the updated hessian were still calculated and unused in order to weakly simulate the increase in time to implement this methodology would take to impement in a python scenario. The time to convergence was about double and the number of iterations increased a similar amount.

## Figure :Python SQP method with GS step size and BFGS updated hessian approximation

 This figure shows the search behavior for the Matlab implementation of the SQP-GS-BFGS methodology. This method converged quickly and the hessian updated for the BFGS section of the methodology correctly unlike the python implementation in its current form.

## Figure : Matlab SQP with GS step size and BFGS updated Hessian

# 6 – Conclusions

In conclusion, of the methodologies studied they all were able to converge on the minimum point of X= [.5, .2500]. The convergence to the correct minimum point reflects the consistent nature of the methodologies studied. From a practical standpoint from the data shown in:

Figure 2:Python SLP method optimization case

Figure 3: Python SQP method with GS search

Figure 4:Python SQP method with GS step size and BFGS updated hessian approximation

Figure 5: Matlab SQP with GS step size and BFGS updated Hessian

We can say that the best methodology studied was that of the SQP method. This ideal behavior among the methodologies is shown by this method because in the implementations studied it converges with the best behavior over time. The other methods all converged to the same minimum point except for SLP at larger step sizes. This homework overall improved my understanding for how the SLP, SQP, and BGFS methodologies are implemented using GS method for step size determination.

# Appendix A: Code

A GitHub repository containing the code for this project can be accessed at the link provided:

<https://github.com/Jyoung31/Homework-7---EGR7040---Jack-Young.git>

Python was used to write the code for the project and the files are stored as .ipynb file extension.