Programming Project Phase III

# Optimization of Constraint Objective function

# By

# Penalty Function Method

By

Tahir Manuel D'Mello

Roll No.180106055

Aman Kumar

Roll No.214103404

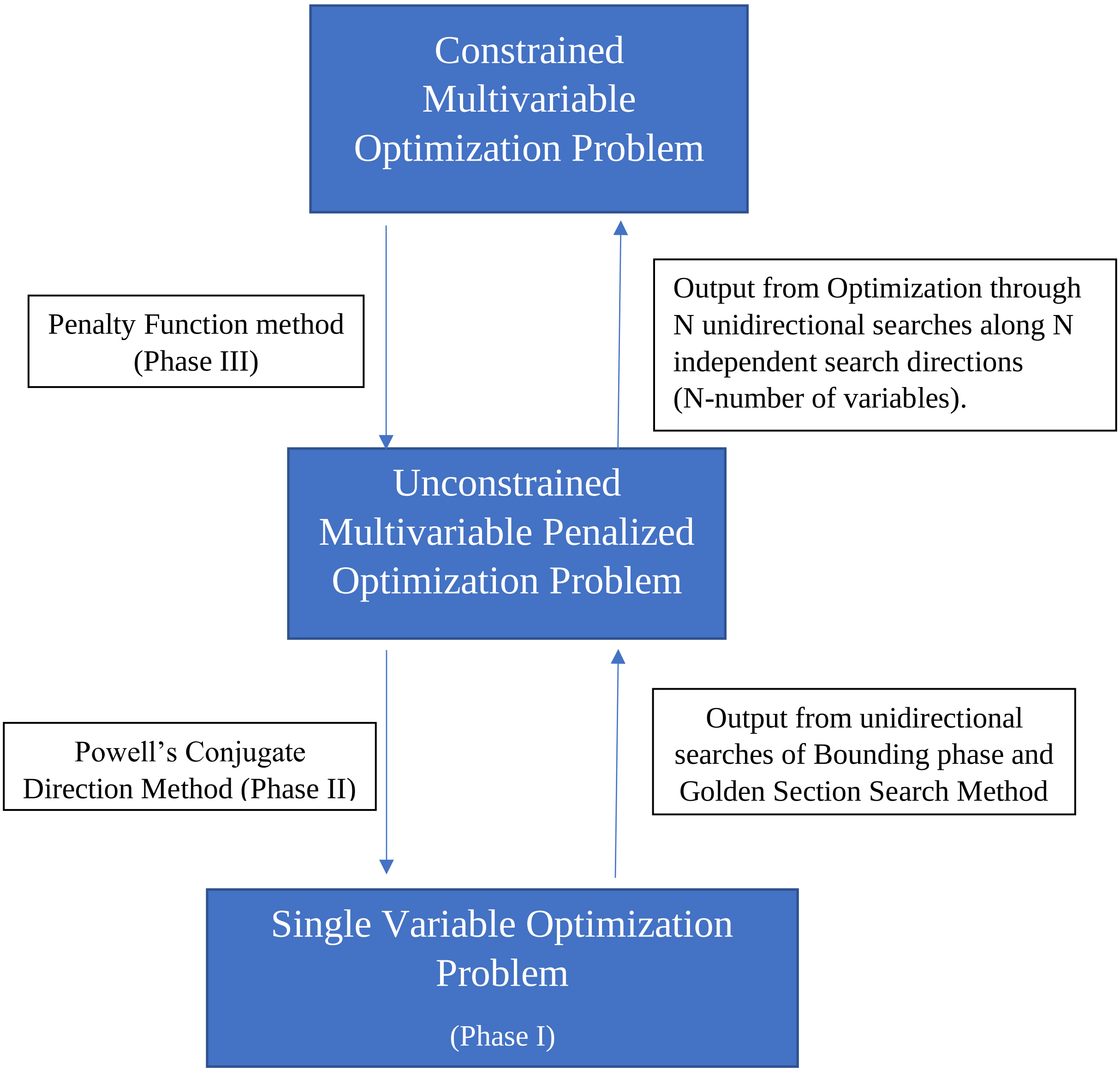
# <u>Introduction</u>

In phase 2 we have seen the use of Powell's Conjugate Direction Method for multivariable optimization problems. Now, when the nature of the problems becomes more general in nature i.e., complexity of the task increases by imposing certain conditions on various parameters dictating a given problem. These conditions are termed as constraints which are the functions of decision variables of a given problem.

To handle such complex problems, various methods are devised, one of which is *Penalty Function Method* which we will be discussing here.

It falls under the category of *Transformation Methods* which are simple and most popular optimization methods for handling constraints. In such methods, the constraint problem is transformed into a sequence of unconstrained problems by adding penalty term of each violation. The unconstrained multivariable problem is then solved with techniques we devised in the earlier phases.

The flow of the problem solving can be clearly seen through following flow chart:



# Penalty Function Method

A constrained optimization formulation can be written as

Minimize $f(x)$

Subjected to, $g_j(x) \geq 0, \quad j = 1, 2, \dots, J$

$h_k(x) = 0, \quad k = 1, 2, \dots, K$

$x_i^{(L)} \leq x_i \leq x_i^{(U)}, \quad i = 1, 2, \dots, N$

Penalty function methods work in a series of sequence, each time modifying a set of penalty parameters and starting a sequence with the point obtained in the previous sequence.

$$P(x, R) = f(x) + \Omega(R, g(x), h(x))$$

Where, R a is the set of penalty parameters, Ω is the penalty term chosen to favour the selection of feasible point over infeasible point.

For the selection of Ω and R we have certain methods which are discussed as below:

**Interior penalty methods:** These methods work for feasible points and penalize points that are close to the constraint boundary. To initialise the algorithm, a point in the feasible region is required. For these methods, initially large value of R is taken, which is reduced gradually. Following are some interior penalty terms:

***1)Log Penalty Term***

$$\Omega = -R * \ln[g(x)]$$

It is an interior penalty term as it penalizes only feasible points as domain of log can't be negative so *g(x)>0*.

***2)Inverse Penalty Term***

$$\Omega = R * \left[\frac{1}{g(x)}\right]$$

Here also, *g(x)>0* is to be used for a minimization problem as the infeasible point should increase the objective function it is penalized in case of minimization problem.

**Exterior penalty methods:** These methods penalize infeasible points but not the feasible solutions. In these methods, for the initialization purpose, any of the feasible or infeasible point can be taken. Here, value of $R$ is taken small initially, which increases gradually. Following are some exterior penalty terms:

***1)Perabolic Penalty Term***

$$\Omega = R * \{h(x)\}^2$$

It is used for handling equality constraints only.

***2)Infinite Barrier Penalty Term***

$$\Omega = R * \sum_{j \in J} |g_j(x)|$$

Where, J is set of constrains at a current point.

***3)Bracket Operator Penalty Term***

$$\Omega = R * \langle g(x) \rangle^2$$

Bracket operator works when $g(x) < 0$, i.e., when constraints are violated.

**Effect of R**

- If the optimum point of unconstrained problem is the same as the constrained problem, an initial parameter R = 0 will solve the constrained problem It is because constraints do not exclude the optimum point of the unconstrained problem.

- If constraints make the optimum solution of the unconstrained problem infeasible, we need to apply a series of sequences of the unconstrained optimization using the penalized objective function. When this is the case the generally the optima lie over the boundary of one of the constraints.

- Also, with further iterations of the Penalty Function Method, the value of R changes which depends on interior and exterior penalty terms.

**Algorithm:**

1) Choose two termination parameters $\epsilon_1, \epsilon_2$; an initial solution $x^{(0)}$; a penalty term $\Omega$; and an initial penalty parameter $R^{(0)}$. Choose a parameter c to update R such that $0 < c < 1$ is used for interior penalty terms and $c \geq 1$ is used for exterior penalty terms. Set $t = 0$.
2) Form $P(x^{(t)}, R^{(t)}) = f(x^{(t)}) + \Omega(R(t), g(x^{(t)}), h(x^{(t)}))$.
3) Starting with $x^{(t)}$, find $x^{(t+1)}$ such that $P(x^{(t)}, R^{(t)})$ is minimum for a fixed value of $R^{(t)}$. Use $\epsilon_1$ to terminate the unconstrained search. (Here, we use unidirectional search to find next point.)
4) Is $\left|P\left(x^{(t+1)}, R^{(t)}\right) - P(x^{(t)}, R^{(t-1)})\right| \leq \epsilon_2$? If yes, set $x^{(T)} = x^{(t+1)}$ and **terminate**; else go to Step 5.
5) Choose $R^{(t+1)} = c * R^{(t)}$. Set $t = t + 1$ and go to Step 2.

# Result and Discussion

In this part, the discussion on results obtained via program on the following questions has been made.

1) *Problem 1*

Minimize $f(x) = (x_1 - 10)^3 + (x_2 - 20)^3$,
Subjected to $g_1(x) = (x_1 - 5)^2 + (x_2 - 5)^2 - 100 \geq 0$,
$g_2(x) = (x_1 - 6)^2 + (x_2 - 5)^2 - 82.81 \leq 0$,
$13 \leq x_1 \leq 20$, $0 \leq x_2 \leq 4$

- Number of Variables: 2 variables

2) *Problem 2*

Maximize $f(x) = \frac{\sin^3(2\pi x_1) * \sin(2\pi x_2)}{{x_1}^3(x_1 + x_2)}$,
Subjected to $g_1(x) = {x_1}^2 - x_2 + 1 \leq 0$,
$g_2(x) = 1 - x_1 + (x_2 - 4)^2 \leq 0$,
$0 \leq x_1 \leq 10$, $0 \leq x_2 \leq 10$

- Number of Variables: 2 variables

3) *Problem 3*

Minimize $f(x) = x_1 + x_2 + x_3$.
Subjected to $g_1(x) = -1 + 0.0025(x_4 + x_6) \leq 0$,
$g_2(x) = -1 + 0.0025(-x_4 + x_5 + x_7) \leq 0$,
$g_3(x) = -1 + 0.01(-x_6 + x_8) \leq 0$,
$g_4(x) = 100x_1 - x_1x_6 + 833.33252x_4 - 83333.333 \leq 0$,
$g_5(x) = x_2x_4 - x_2x_7 - 1250x_4 + 1250x_5 \leq 0$,
$g_6(x) = x_3x_5 - x_3x_8 - 2500x_5 + 1250000 \leq 0$,
$100 \leq x_1 \leq 10000$,
$1000 \leq x_i \leq 10000$, $i = 2, 3$,
$10 \leq x_i \leq 1000$, $i = 4, 5, \dots, 8$

- Number of Variables: 8 variables

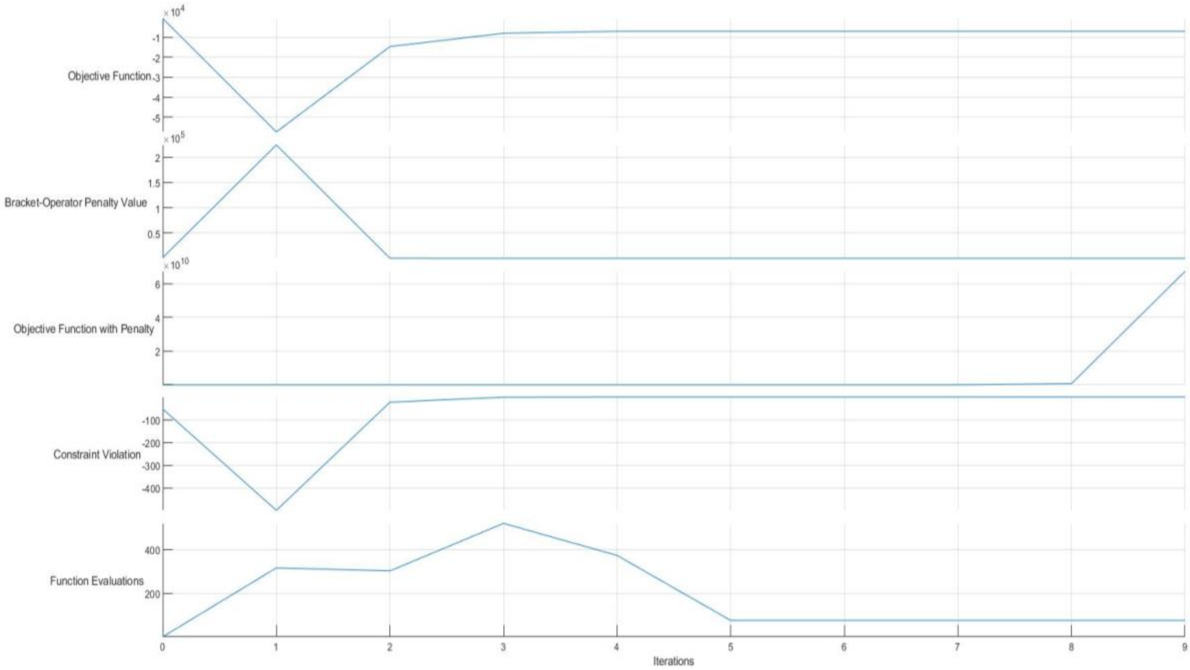
Penalty Function algorithm has been run for 10 times for each problem with varying initial guess and optimum point obtained from each time has been tabled below. In the algorithm we have used exterior penalty term, value of $R$, starting from $R^{(0)} = 0.1$, is increased by factor $c = 100$ after performing each iteration.

**Problem 1:**

| S. No | Initial Guess $(x1,x2)$ | $x1T$ | $x2T$ | Function Value |
|---|---|---|---|---|
| 1 | (2,3) | 14.089 | 0.83111 | -6975.2 |
| 2 | (10,1) | 14.089 | 0.83124 | -6975 |
| 3 | (6,2) | 14.089 | 0.83111 | -6975.2 |
| 4 | (3,2) | 14.089 | 0.83113 | -6975.1 |
| 5 | (9,3) | 14.089 | 0.83112 | -6975.2 |
| 6 | (7,1) | 14.089 | 0.83124 | -6975 |
| 7 | (4,13) | 14.089 | 0.83124 | -6975 |
| 8 | (18,2) | 14.089 | 0.83111 | -6975.2 |
| 9 | (5,1) | 14.089 | 0.83124 | -6975 |
| 10 | (11,3) | 14.089 | 0.83111 | -6975.2 |

Final value of R at the termination of algorithm in each case = $10^{15}$

| Solutions | $x1(T)$ | $x2(T)$ | Function Value |
|---|---|---|---|
| Best | 14.089 | 0.83124 | -6975 |
| Worst | 14.089 | 0.83111 | -6975.2 |
| Mean | 14.089 | 0.83116 | -6975.11 |
| Median | 14.089 | .831125 | -6975.15 |
| Standard Deviation | 0.0 | $6.152*10-5$ | 0.094 |




*Plots of various values vs iterations for the best performing combination of parameters for problem 1*

As evident from the plots and table, our algorithm is converging every time, although the theoretical value of function at optima is -6961, the converged value is -6975 which is close to theoretical minima. The slight variation could be because of one or combination of the following reasons:

1)Non-linear nature of the problem.

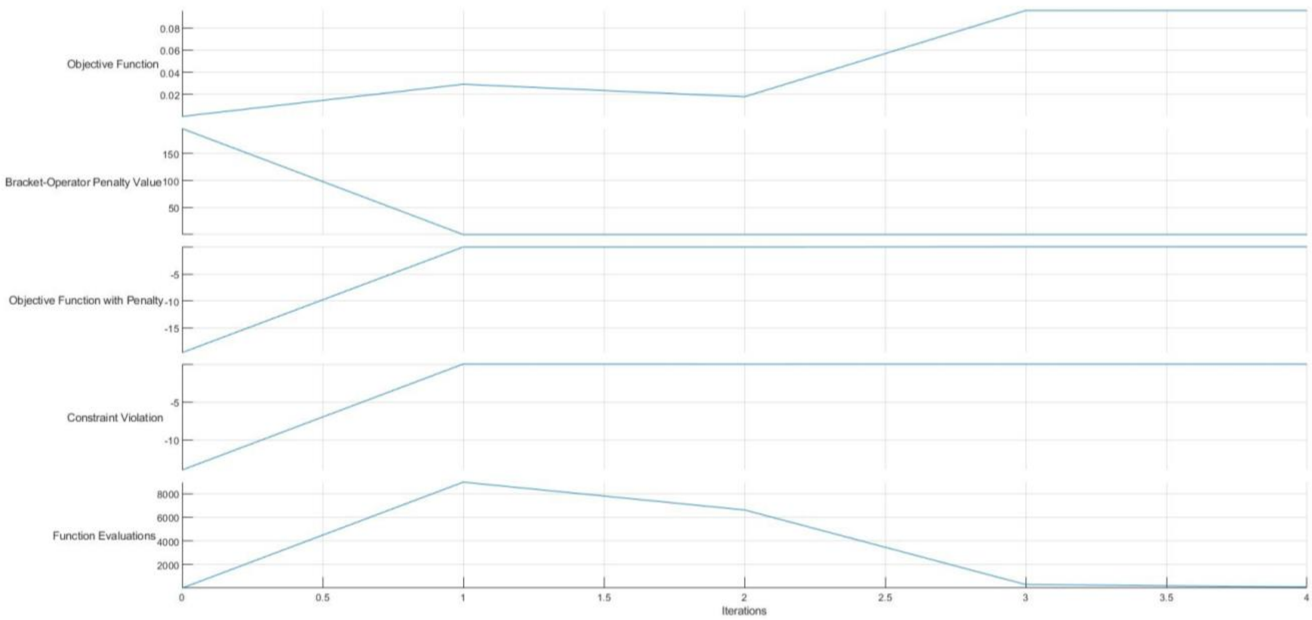
2)It might have stuck to a planar surface where the further improvement of the function is negligible.

3)Due to the distortion of the contours, it might be converging to an artificial optimum.

**Problem 2:**

| S. No | Initial Guess ($x1$,$x2$) | $x1T$ | $x2T$ | R$^{(T)}$ | Function Value |
|---|---|---|---|---|---|
| 1 | (5,5) | 1.228 | 4.2454 | 1000 | 0.095825 |
| 2 | (5,8) | 1.7341 | 4.7461 | 100000 | 0.029144 |
| 3 | (4,2) | 1.228 | 4.2454 | 100000 | 0.095825 |
| 4 | (3,5) | 2 | 5 | 10 | $2.946*10^{-21}$ |
| 5 | (7,9) | 1.228 | 4.2454 | 1000 | 0.095825 |
| 6 | (2,8) | 1.228 | 4.2454 | 1000 | 0.095825 |
| 7 | (1,9) | 1.228 | 4.2454 | 1000 | 0.095825 |
| 8 | (5,1) | 1.7341 | 4.7461 | 100000 | 0.029144 |
| 9 | (4,3) | 1.228 | 4.2454 | 100000 | 0.095825 |
| 10 | (3,3) | 1.228 | 4.2454 | 10000 | 0.095825 |

| Solutions | $x_1^{(T)}$ | $x_2^{(T)}$ | Function Value |
|---|---|---|---|
| Best | 1.228 | 4.2454 | 0.095825 |
| Worst | 2 | 5 | $2.946*10^{-21}$ |
| Mean | 1.40642 | 4.421 | 0.0729 |
| Median | 1.228 | 4.2454 | 0.095825 |
| Standard Deviation | 0.281 | 0.276 | 0.0358 |



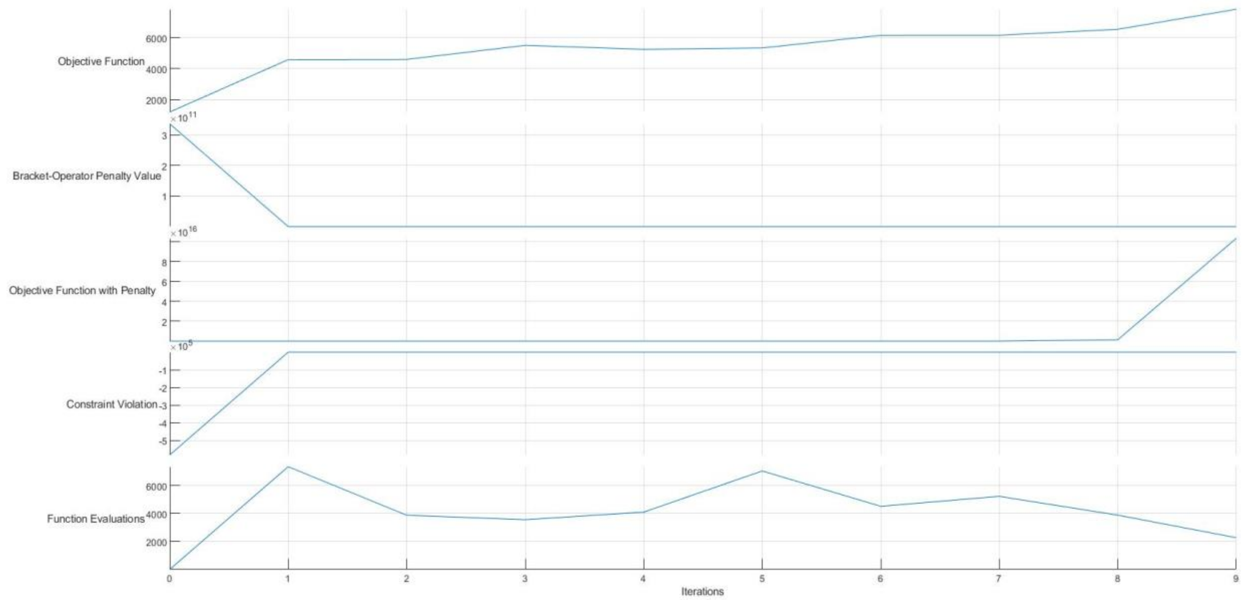

*Plots of various values vs iterations for the best performing combination of parameters for problem 2*

- As can be verified from the table and plots, the algorithm has converged to exact theoretical optimum value except form few points when it was run for 10 different initial guess values.

**Problem 3:**

| S. No | $x_1^T$ | $x_2^T$ | $x_3^T$ | $x_4^T$ | $x_5^T$ | $x_6^T$ | $x_7^T$ | $x_8^T$ | $R^{(T)}$ | Function Value |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 5707.2 | 1000 | 6750.6 | 247.8 | 157.8 | 853 | 135.3 | 999.5 | $10^{15}$ | 13458 |
| 2 | 15086 | 1000 | 1563 | 161.7 | 10 | 660.1 | 195.6 | 799.4 | $10^{15}$ | 4071.6 |
| 3 | 100 | 1000 | 3037.1 | 86.6 | 10 | 313.4 | 10 | 413.3 | $10^{15}$ | 4137.1 |
| 4 | 1089.8 | 4972.3 | 1761.1 | 179.1 | 392.7 | 438.4 | 232.8 | 552.3 | $10^{15}$ | 7823.2 |
| 5 | 100 | 1000 | 2029.6 | 10 | 11.1 | 506.1 | 38.9 | 613.3 | $10^{15}$ | 3129.6 |
| 6 | 100 | 1000 | 2522.2 | 10 | 10 | 395.4 | 40 | 495.7 | $10^{15}$ | 3622.2 |
| 7 | 939.5 | 1500 | 1518.2 | 92 | 9.7 | 672.1 | 133 | 817.1 | $10^{15}$ | 3957.7 |
| 8 | 1827 | 1000 | 1780 | 612.6 | 358.5 | 481.7 | 294.9 | 617.7 | $10^{15}$ | 4607 |
| 9 | 10000 | 1000 | 1046.3 | 530.9 | 317.2 | 604 | 263.8 | 758.6 | $10^{15}$ | 12046 |
| 10 | 897.8 | 1339.2 | 1638.2 | 151 | 10 | 611.7 | 152.1 | 757.8 | $10^{15}$ | 3875.1 |

| Solutions | $x_1^T$ | $x_2^T$ | $x_3^T$ | $x_4^T$ | $x_5^T$ | $x_6^T$ | $x_7^T$ | $x_8^T$ | Function Value |
|---|---|---|---|---|---|---|---|---|---|
| Best | 1089.8 | 4972.3 | 1761.1 | 179.1 | 392.7 | 438.4 | 232.8 | 552.3 | 7823.2 |
| Worst | 5707.2 | 1000 | 6750.6 | 247.8 | 157.8 | 853 | 135.3 | 999.5 | $10^{15}$ |
| Mean | 3584.73 | 1481.15 | 2364.63 | 208.17 | 128.7 | 553.59 | 149.64 | 682.47 | 6072.75 |
| Median | 1014.65 | 1000 | 1770.55 | 156.35 | 10.55 | 555.05 | 143.7 | 687.75 | 4104.35 |
| Standard Deviation | 4877.5 | 1175.97 | 1554.24 | 195.54 | 155.95 | 149.78 | 93.50 | 166.44 | 3565.89 |




*Plots of various values vs iterations for the best performing combination of parameters for problem 3*

- As it can be seen from the plots and table, the algorithm is not converging for any of the initial guess. The very close value to the thoeretical optima(7049.33) we got is 7823.2 which has got significant difference from the desired value.
- One of the important reasons for this behaviour could be the distortoion of the actual contours during the implementation of penalty function method,which is when coupled with non-linearity and more number of variables, could further create complexities that might generate large number of artificial optimum values.

# Conclusion

- For problem 1 & 2, we are getting almost very accurate and precise values which are very close to theoretical optima in 1 and equals to that in 2.

- For problem 3, almost every iteration converges to some artificial optimum value which is significantly different in comparison to theoretical optimum value. From this problem it can be said that when the number of variables is increased, the chances of getting the optimum solution reduces due to hight distortion of the actual contours.

- The non-convergence issues of the problems can be thought of due to following reasons:
    - Non-linearity of the problem might create local optimum regions.
    - It might be due to the nature of the surface which could be flat at certain points where the algorithm might get stuck and generate local optima.
    - Due to the distortion of the contours while implementation of Penalty function method, it might be converging to an artificial optimum.