Assignment 3

MSCI 703 by Prof. Elhedhli

Ali Rafieepouralavialavijeh 20900871

c)

Constraint generation method:

Starting from a random u, table below shows the results through the iterative process of solving the problem:

Iteration number	u	Lagrangian heuristic solution	Heuristic solution gap	LB	UB
1	[10, 15]	[1, 1, 1.0, 0.0, 0.0, 0.0]	115	2.77	120
2	[0.11,0.0]	[1, 0.0, 0.0, 0, 1.0, 1.0]	3.77	5.6	8.7
3	[0.0,0.028]	[0.0, 1.0, 1, 1.0, 0.0, 0.0]	0.17	6	6.17
4	[0.0, 0.0]	[0.0, 1.0, 1, 1.0, 0.0, 0.0]	0	6	6

Sub gradient method:

Since the number of iterations are not easy to be shown in tables, I am providing plots to show the UB and lagrangian heuristic solution gaps:

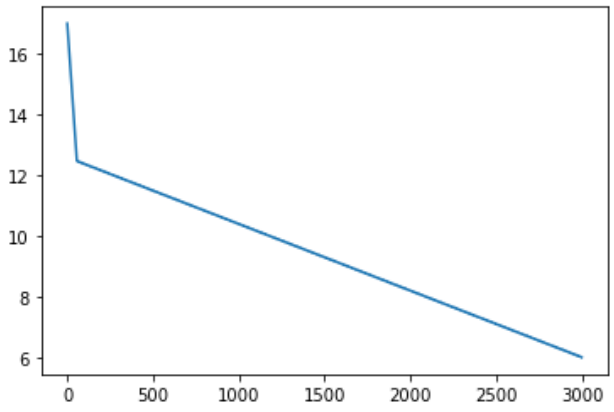

Figure 1

Figure 1 shows the reduction of the upper bound through 3000 iterations until hitting the lower bound 6.

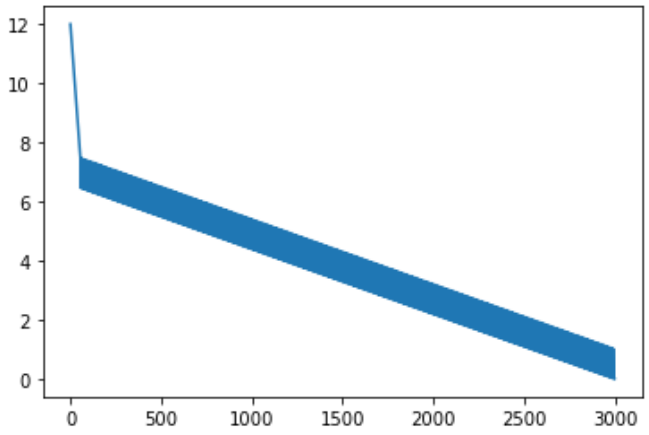

Figure 2

Figure 2 shows the fluctuations of the heuristic solution and therefore the gap through 3000 iterations.

The best heuristic solution is: [0.0, 1.0, 1.0, 1.0, 0.0, 0.0] with an objective value of 6.

e)

Constraint generation method:

Starting from a random u, table below shows the results through the iterative process of solving the problem:

Iteration number	u	Langragian heuristic solution	Heuristic solution gap	LB	UB
1	[10,5,20]	[1, 1, 1, -0.0, -0.0, -0.0]	30	0	35
2	[0,0,0]	[1, 0, 0, 0.0, 1.0, 1.0]	1	6	6

Sub gradient method:

Since the number of iterations are not easy to be shown in tables, I am providing plots to show the UB and lagrangian heuristic solution gaps:

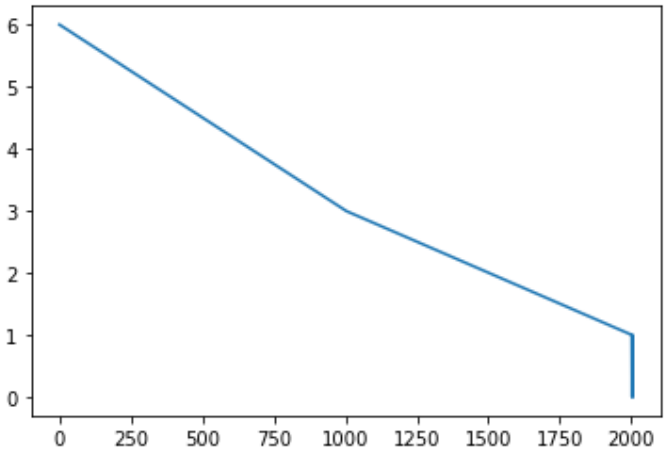

Figure 3

Figure 3 shows the fluctuations of the heuristic solution and therefore the gap through 2000 iterations.

The best heuristic solution is: [0.0, 1.0, 1.0, 1.0, 0.0, 0.0] with an objective value of 6.

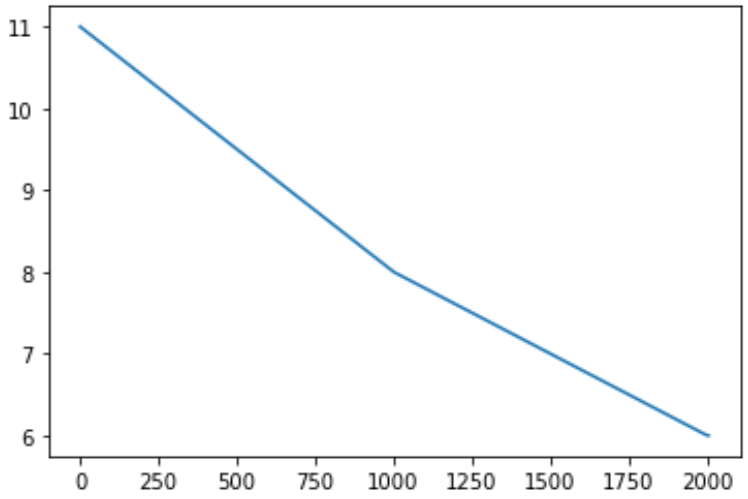

Figure 4

Figure 4 shows the reduction of the upper bound through 2000 iterations until hitting the lower bound 6.

Note:

All details of the iterations, including heuristic solutions, LB, UB, gaps and a lot more information are available during the code run. All models are solved using Gurobi Persistent solver with an academic license.