Assignment 2

University College Dublin

School of Electronic and Electrical Engineering

Optimisation Techniques- EEEN40580

Student: Aidan O'Sullivan

Student Number: 20209138

19th of Novemeber 2021

1.

Yes, I have read and understood UCD's policy on plagiarism and I hereby declare that this work is my own

Signed: Aidan O'Sullivan **Date:** 19/11/2021

My personal question variables:

$\alpha = 3$

$\beta = 8$

Question 2

The data was modelled as a table, with each person represented on a row and column. Where the two people overlapped a zero was placed if they were not listed as friends and a one if they were friends. This was represented in “Assignment_2_Q2.csv” file.

The variable being minimised in this problem was the number of groups. For the formulation of the problem a constraint was set so that if the data showed that two people were friends, they would not be in the same group. A binary ‘variable U’ decided whether a person belonged in a group or not and ‘f_ij’ was the parameter from the table, stating if two people were friends or not:

$$(U[i,g] + U[j,g])f\_ij \leq 1$$

This is ensured the binary variables would never be one at the same time if both person I and j were friends. The constraint allowed person I and j to belong in the group only if ‘f_ij’ was zero.

Another constraint was added to ensure that a person was not added to more than one group:

$$\sum_{g} U[i,g] == 1$$

This says that the number of groups a person belongs to should equal exactly one.

Another formulation constrained the number of total groups. This was the variable that would ultimately be minimised. Group[g] was a list of possible groups that could be used from 1-11, the solve should only select the minimum amount of these:

$$Number\ of\ groups \geq group[g] * \ U[i,g]$$

As this problem used discrete variables this made it an integer programming. Some variables were not discrete, making it mixed integer. Variables were not multiplied together to keep the problem linear and make it easier to solve.

Finally, through minimization of the objective function, the minimum number of groups was 3. This gave a total minimum cost of 1500 euro to cover the event.

The following table was obtained, with no two friends present in the same group.

Group 1	Group 2	Group 3
Ava	Emma	Jack
Emily	James	William
Lily	Noah	
Lucas	Olivia	
Sophia		

Question 3

In this question the cost of transportation was being minimised. A number of constraints were added to ensure the solver understood the limitations of the logistics model. The first constraint ensured that amount a factory i supplies to its customer j, 'X_ij[i,j', is less than its capacity. A binary variable 'U_ij[i,j]' allows the solver to decide which customers a factory would be most cost efficient at supplying. This binary variable is multiplied by the capacity parameter and not the variable to keep the system linear. The solver would be unable to solve the problem if it became non-linear. This constraint is written as follows:

$$X\_ij[i,j] \ <= \ capacity\_i[i] * U\_ij[i,j]$$

The next constraint ensures that the total amount supplied to a customer from all its factories is greater or equal to its demand. This way the solver understands that it is required for all customer's demands to be met:

$$\sum_i X\_ij[i,j] \ >= \ Demand\_j[j]$$

The final constraint infers that the capacity of the factory must not be exceeded by the amount each factory supplies to all its customers and its own demand. This assumes that each factory supplies its own demand, as this is the most cost effective method when travel cost for goods is considered. This will not be incurred if each factory can use its own capacity to fulfil its demand:

$$\sum_j X\_ij[i,j] \ + \ Demand\_i[i] \ <= \ capacity\_i[i] * U\_i[i]$$

The objective function tried to minimise the sum of all the costs. The distance between each factory and customer was found using the Haversine function. The distance between each connected factory and customer was multiplied by the number of goods transported and its 1 euro per km transport cost. The next factor to consider was the cost of each good produced in the factory multiplied by the amount produced. This amount was the quantity sent to each customer and the quantity produced to meet its own demand. The final cost came from the sum of investment costs for each factory that was chosen to operate by the solver:

$$\sum_{ij}(X\_ij[i,j] * distance\_ij[i,j]) + \sum_{i} (cost\_i[i] * \sum_{j}(sum(model.X\_ij[i,j]) + Demand\_i[i])) + \sum_{i} model.U\_i[i] * model.investment\_i[i]$$

When the problem was solved using Gurobi, the total cost stood at 144348 euro.

As this problem used discrete variables this made it an integer programming. Some variables were not discrete, making it mixed integer. As discussed efforts were made to keep the problem linear by not multiplying variables together.