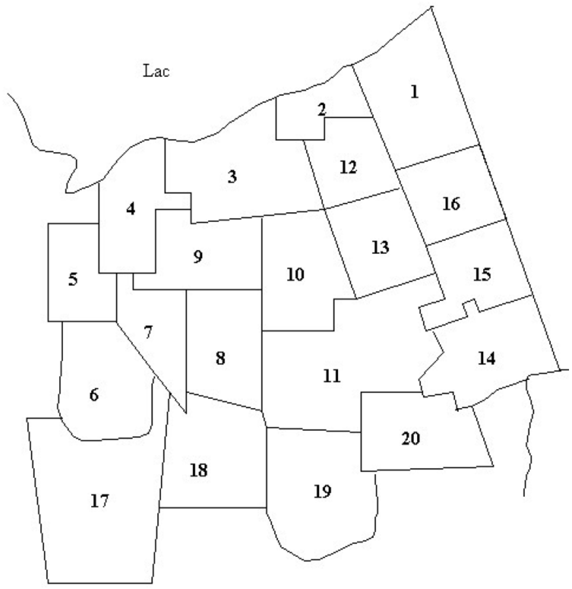
Es 1: A location planning problem

A logistic company is evaluating where to open new warehouses for its products in a new region. The region is composed of 20 districts, as represented in the figure below. Opening a warehouse in one district, means "covering" with its services the district itself and all adjacent ones.



1. Model the problem with a linear programming model (try to write a generalized model, with *N* districts and a generical map (see comments).
2. Implement the model finding the minimal number of warehouses that the firm will need to open in order to cover the whole region.
3. Modify and run the model to answer the following questions:
    a. Consider the following table containing the opening costs for each district. Suppose that you have a budget of 5.5 MEuro. Is it enough for covering all the regions?

district	1	2	3	4	5	6	7	8	9	10
cost M€	1.5	0.9	1.5	1.5	2.5	1.9	3.4	3.9	1.5	1.5
district	11	12	13	14	15	16	17	18	19	20
cost M€	2.5	2.5	2.9	2.5	2.5	4.9	3.9	3.4	1.5	1.5

    b. Suppose that you are given a strict maximum budget of 5.5 MEuro. What is the maximum number of districts that can be covered?

Es1 - Comments

Warehouses.m contains the definition of the map data, in the form of a matrix *CoverMap(I,j)* containing 1 if districts i and j are adjacent, 0 otherwise. It is given as a sparse matrix (the elements not given are = 0). In the same script you'll also find *OpeningCost(i)* containing the opening cost of each district.

Es 2 - A Scheduling problem

Consider the problem of minimizing the total completion time (sum of all jobs completion times) on a single machine environment. Jobs are available for processing from a "release time" rj. The processing times of jobs are pj. This problem is classified as the 1|rj|ΣC problem. Two linear programming models are available for this problem, with a different decision variables choice:

(Jobs completion time variables) Consider Cj as the variable containing the time when job j finishes its operation on the machine. The model includes an objective function, a set of constraints for the release time of jobs and a set of linearized "disjunctive constraint" on the machine:

$$\min \sum_{j=1}^{n} C_j$$

$$\begin{aligned}
&C_j \geq r_j + p_j && \forall j = 1..n \\
&C_i - p_i \geq C_j - My_{ij} && \forall i = 1..n, j = 1..n : i \neq j \\
&C_j - p_j \geq C_i - M(1 - y_{ij}) && \forall i = 1..n, j = 1..n : i \neq j \\
\\
&C_j \in R && \forall j = 1..n \\
&y_{ij} \in \{0,1\} && \forall i = 1..n, j = 1..n
\end{aligned}$$

(Positional completion time variables) Consider $C_{[j]}$ as the decision variable containing the completion time of the job in j-th position, and $x_{ij} \in \{0,1\}$ equal to 1 if job i is in position j of the sequence, zero otherwise. Then, it is possible to write a model where the constraints are: a job is chosen for each position in the sequence; each job is processed only once; completion time of all jobs must consider their release times; for each job, its start time must be after the completion of its preceding operations.

$$\min \sum_{j=1}^{n} C_{[j]}$$

$$\begin{aligned}
&\sum_{i=1}^{n} x_{ij} = 1 && \forall j = 1, \dots, n \\
&\sum_{j=1}^{n} x_{ij} = 1 && \forall i = 1, \dots, n \\
&C_{[1]} = \sum_{i=1}^{n} (p_i + r_i) x_{i1} \\
&C_{[j]} \geq C_{[j-1]} + \sum_{i=1}^{n} p_i x_{ij} && \forall j = 2, \dots, n \\
&C_{[j]} \geq \sum_{i=1}^{n} (p_i + r_i) x_{ij} && \forall j = 2, \dots, n \\
&x_{ij} \in \{0,1\}, \quad C_{[j]} \geq 0
\end{aligned}$$

Implement both the models and test them on the data contained in the file *Scheduling.m*, finding the optimal solution.

Es 2 – Comments

- The implementation of the first model is already given as an example in the course material. Pay attention to the definition of a feasible value for the big-M constant.
- The optimal solution value is 13764 and clearly is the same for both models.
- Using the first model, you can try different values of "big-M" for the disjunctive constraints. What is the effect of having a larger or smaller value?
- Which is the most performing model? Is the result on one instance to be considered valid in general?
- You can try to run Gurobi instead of the default Matlab solver to see how much more efficient it is.