Algorithmic Methods for Mathematical Models

– COURSE PROJECT –

Crime is soaring in Gotham City. Batman feels he is no longer able to cope with it. For this reason he has decided to update his system for monitoring the streets. More precisely, he wants to hide cameras at street crossings. He quickly realizes that he does not need to place a camera at each crossing, as a camera at one of the ends of a street can also cover the other end if it is poweful enough.

There are K models of cameras available in the market, which we will refer to with numbers $1, 2, \ldots, K$. The price in € of a unit of model k (where $1 \leq k \leq K$) is P_k. Other interesting attributes are the *range* R_k (with $1 \leq R_k \leq 49$), the *autonomy* A_k (with $2 \leq A_k \leq 6$), and the *power consumption* C_k per day (with $0 < C_k$, in €). The larger the range, the farther the distance the camera can cover. But to avoid malfunction, a camera cannot be on indefinitely: a camera of model k can be operating at most A_k days without interruption. On the other hand, if a camera starts operating, it must be operating at least 2 days consecutively.

There are N crossings, which are tagged $1, 2, \ldots, N$, respectively. For $1 \leq i, j \leq N$, the value M_{ij} is the minimum range that a camera must have if placed at i to cover j (or symmetrically, to cover i if placed at j). By convention, if $M_{ij} \geq 50$ then it is impossible to cover i with a camera placed at j (and vice versa). Moreover, $M_{ii} = 0$ for all $1 \leq i \leq N$. To ensure that cameras are properly hidden, at most one camera can be placed at a given crossing.

Batman needs to find out in which crossings of the city cameras should be placed, and of which model these should be, so that all crossings are always watched. Moreover, he also wants to have a weekly schedule that shows, for each d of the week (where $1 \leq d \leq 7$, being Monday = 1, ..., Sunday = 7), which cameras should be on. Unfortunately Wayne Corporation is currently going through a very bad financial situation, and as a consequence costs must be minimized.

E.g., let $K = 2$, $(P_1, P_2) = (20, 21)$, $(R_1, R_2) = (1, 1)$, $(A_1, A_2) = (2, 3)$, $(C_1, C_2) = (5, 5)$, $N = 4$ and

$$M = \begin{pmatrix} 0 & 1 & 1 & 50 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 50 & 1 & 1 & 0 \end{pmatrix}$$

A possible solution is to place only two cameras, both of model 1, at crossings 2 and 3. Moreover, the schedule is (a tick indicates that the camera is on):

crossing\day	1	2	3	4	5	6	7
2	✓		✓	✓			✓
3		✓	✓		✓	✓	

It can be checked that all crossings are always covered. Moreover, cameras work as required (note that the camera at crossing 2 operates on Sunday and Monday, and Wednesday and Thursday). The cost of this solution is $2 \cdot 20 + 8 \cdot 5 = 80$. In fact, it is optimal.

On the other hand, now let us place again two cameras at crossings 2 and 3, of models 1 and 2 respectively. Let us also consider the following schedule:

crossing\day	1	2	3	4	5	6	7
2	✓	✓				✓	✓
3			✓	✓	✓		

The cost is $20 + 21 + 7 \cdot 5 = 76$. However, this plan is not valid: the camera at crossing 2 is operating four days in a row (from Saturday to Tuesday).

1. **Work to be done:**
   (a) State the problem formally. Specify the inputs and the outputs, as well as any auxiliary sets of indices that you may need, and the objective function.

   (b) Build an integer linear programming model for the problem and implement it in OPL.

   (c) Because of the complexity of the problem, heuristic algorithms can also be applied. Here we will consider the following:

      i. a greedy constructive algorithm,

      ii. a greedy constructive + a local search procedure,

      iii. GRASP as a meta-heuristic algorithm. You can reuse the local search procedure that you developed in the previous step.

   Design the three algorithms and implement them in the programming language you prefer.

   (d) *Tuning of parameters and instance generation.* Given an instance of input to the problem, the value of $N \cdot K$ is the *size* of the instance.

      i. Implement an instance generator that produces random instances for a given size.

      ii. Tune the α parameter of the GRASP constructive phase with a set of randomly generated instances of large enough size.

      iii. Generate problem instances with increasingly larger size. Solving each instance with CPLEX should take from 1 to 30 min.

   (e) Compare the performance of CPLEX with the heuristic algorithms, both in terms of computation time and of quality of the solutions as a function of the size of the instances.

   (f) Prepare a report and a presentation of your work on the project.

2. **Report:**

   Prepare a report (8-10 pages) in PDF format including:

   - The formal problem statement.
   - The integer linear programming model, with a definition and a short description of the variables, the objective function and the constraints. Do not include OPL code in the document, but rather their mathematical formulation.
   - For the meta-heuristics, the pseudo-code of your constructive, local search, and GRASP algorithms, including equations for describing the greedy cost function(s) and the RCL.
   - Tables or graphs with the tuning of parameters, and with the comparative results.

   Together with the report, you should give all sources (OPL code, programs of the meta-heuristics, instances, generator) and instructions on how to use them, so that results can be reproduced.

3. **Presentation:**

   You are expected to make a presentation of your work at the end of the course (at most 10 minutes long; overtime presentations will be terminated). All group members must participate in the presentation and know all the work in the project. The slots of $\{11, 15, 18\}/12/25$ will be devoted to these presentations. The schedule will be announced in its due time.

   The slides of the presentation in PDF format should be delivered with the report by **09/12/25**.

   The slides can contain plots, equations, algorithms, ... with a very short text that helps to understand them. You are expected to give a full explanation of those contents in the presentation. On the other hand, the report should contain that explanation in a well-organized way as a text.