Tutorial 0 : Modelisation

Operations research, 3rd semester.

2024

This tutorial gives you some practical applications of operations research and, this way, is an introduction to what operations research is. For each of the following exercises, the subject is a problem written in natural language. You have to rewrite it in a mathematical way using all the tools you know such that a computer can solve the problem. Some of the problems lack information. In that case, you have to think of what is missing, to make assumptions and, sometimes, simplify the real to focus on the problem you are asked to solve. Note that you do not have to solve the problem : you do have to give the inputs and the outputs of the problem, not how the latter are computed from the former. Note also that there could be multiple possible answers but some of them may be more easily defended. You can use any tool you know : graph, system of equations and/or inequations, set, list, boolean formulas, ...

▶ **Correction**

All corrections are indicative ; you can suggest others. It may be interesting to let the students reflect and argue on the different proposals.

1 Easy modelisations

Each of the following exercises are easy : no information is missing and the adequate tool is almost obvious.

Exercise 1 — *Risk evaluation*

How many links may fail until two users of a telecommunication network, whoever they are, cannot communicate anymore ?

▶ **Correction**

Most obvious solution :
Input : a connected undirected graph $G = (V, E)$
Output : a subset E' of E of minimum size such that removing E' from G disconnects the graph.

Exercise 2 — *Let's play*

Solve a sudoku.

▶ **Correction**

We can propose several solutions : graph coloring, Boolean formula, linear programming. The most trivial being probably the very definition of Sudoku :

We call ε the symbol *empty*.
Input : an integer n, a square matrix A in $\mathcal{M}_{n^2}(\{\varepsilon, 1, 2, \ldots, n^2\})$
Output : a square matrix B in $\mathcal{M}_{n^2}(\{1, 2, \ldots, n^2\})$ such that

- for all $1 \leq i, j \leq n^2$, if $A_{i,j} \neq \varepsilon$ then $A_{i,j} = B_{i,j}$
- for all $1 \leq i, j, k \leq n^2$ such that $j \neq k$, $B_{i,j} \neq B_{i,k}$
- for all $1 \leq i, j, k \leq n^2$ such that $i \neq k$, $B_{i,j} \neq B_{k,j}$
- for all $1 \leq i, j, k, l \leq n^2$ such that $i \neq k$ or $j \neq l$, let q_i, q_j, q_k and q_l be the quotients of $i-1, j-1, k-1$ and $l-1$ by n, if $q_i = q_k$ and $q_j = q_l$ then $B_{i,j} \neq B_{k,l}$.

If such a matrix does not exist, we return nothing.

Exercise 3 — ***Let's be fair-play***

Could you give me the five best olympics countries ?

▶ **Correction**

We need to define *better*. We will use the following definitions :

— a country p is associated with the number of gold medals p_o, silver medals p_a, and bronze medals p_b it has won

— let two countries p and q, we say that $p \preceq q$ if and only if

  — $p_o < q_o$

  — or $p_o = q_o$ and $p_a < q_a$

  — or $p_o = q_o$ and $p_a = q_a$ and $p_b \leq q_b$

Input : a set of countries $\mathcal{P}$.

Output : a subset $\mathcal{Q}$ of 5 countries from $\mathcal{P}$ such that for every country $p \in \mathcal{P}\backslash\mathcal{Q}$ and for every country $q \in \mathcal{Q}$, $p \preceq q$.

Exercise 4 — ***Bier and crackers***

How many channels should I watch or record in order to see every sporting event I would like to see ?

▶ **Correction**

This is a set cover problem. The elements to be covered are the events and the sets are the chains.

Input :

— A set of sporting events $\mathcal{X}$

— A set $\mathcal{S} \subset \mathcal{P}(\mathcal{X})$ of subsets of $\mathcal{X}$

Output : A subset C of $\mathcal{S}$ such that $\bigcup_{s \in C} s = \mathcal{X}$.

2 Intermediate modelisations

The following problems are harder. Try not to forget any important element before modelising.

Exercise 5 — ***Which way ?***

Where and how do we have to put road signs in order to direct people to the main cities ?

▶ **Correction**

The difficulty of this exercise lies in the fact that, contrary to what one might believe, we are not necessarily looking for the shortest path. Returning this solution would create a monstrous disorder in road traffic as everyone would pass through the same place. The aim of this exercise is to make think about this. It may be good to see if it is possible to build a small example that would create problematic traffic with this solution.

Here is a proposal : we associate with each pair of towns/villages an integer corresponding to an estimate of the number of people traveling between these towns, and we estimate for each road the number of people who can pass on these roads simultaneously. The paths we will create between the towns must ensure that on the same road, there are no more people than the maximum number.

For each town w, we create a spanning tree T_w of the network (the set of paths from the other towns to w) ; the road signs will follow the direction of this tree to reach this town. We can then estimate how many people will travel on each section of the road. Given a spanning tree T_w, two nodes (v, w), and an edge e, we define the binary variable $x(T_w, v, w, e)$ as being equal to 1 if and only if e is part of the path from v to w in T_w ; in other words, all the people who want to go from v to w will pass through e.

Input :

— a connected undirected graph $G = (V, E)$

— for each pair of nodes v and w, associate an integer $\omega(v, w) \in \mathbb{N}$

— for each edge e, associate an integer $c(e) \in \mathbb{N}$

Output : for each node w, construct a spanning tree T_w of G such that for every edge e, $\sum_{v,w \in V^2} \omega(v,w) \cdot x(T_w, v, w, e) \leq c(e)$.

Exercise 6 — *Are we late ?*

How to decide if a set of tasks of a project may be finished on time and satisfying resource constraints.

▶ **Correction**

The simplest way is probably to define a linear program here. Many constraints are possible, here is one proposal.

Input :

— A set of tasks $\mathcal{T}$ including a starting task $START$ and an ending task END

— An oriented precedence graph without cycles $\mathcal{G} = (\mathcal{T}, A)$ (each arc (t, t') indicates that t must be completed before starting t'), the task $START$ is a source and the task END is a sink

— For each task t, an integer $\omega(t)$ (the number of resources used)

— For each task t, an integer $d(t)$ (the duration of the task) ; $d(START) = 0$ and $d(END) = 0$

— For each task t, a time window $[b(t), e(t)]$ during which the task is to be completed

— A maximum number of resources c

Output : An integer x_t associated with each task such that :

— $x_{START} = 0$

— x_{END} is minimized

— For every task t, $b(t) \leq x_t$ and $x_t + d(t) \leq e(t)$

— For every edge (t, t') in G, $x_t + d(t) \leq x_{t'}$

— For every integer n, $\sum_{t: n \in [x_t, x_t + d(t)]} \omega(t) \leq c$

Exercise 7 — *News*

Where to put law enforcement agents in order to ensure the safety of a crowd ?

▶ **Correction**

It is, roughly speaking, a generalized k-center problem. It is important to differentiate between the set of areas that can be monitored and the set of areas where action can be taken (one can monitor without being able to act and vice versa).

Input :

— A set of positions V

— A set of positions $W \subset V$ where the crowd will be

— An undirected monitoring graph $G = (V, E)$ (each edge (u, v) means that v can be seen from u and vice versa)

— A directed intervention graph $H = (V, F)$ (each arc (u, v) means that intervention at v can be made from u)

— An integer k (the number of teams that can be deployed)

Output : A set U of k nodes from V such that for every node w in W, w has a neighbor in G that is in U and a predecessor in H that is in U.

Exercise 8 — *TMNT !*

How to maximize the number of parts of a pizza by cutting it n times ?

▶ **Correction**

Several questions arise here

— should we make straight cuts ?

— should we go through the center of the pizza ?

— does the pizza have a center ? (it can be square)

— should we make equal slices ?

— is the pizza flat ?

Note : if we make straight cuts without going through the center and without making equal slices, the answer is $\binom{n}{0} + \binom{n}{1} + \binom{n}{2}$ if $n \geq 2$ (On the number of regions in an m-dimensional space cut by n hyperplanes, Chungwu Ho and Seth Zimmerman or `http://mathworld.wolfram.com/CircleDivisionbyLines.html`).
Input : two integers n and m
Output : n hyperplanes in $\mathbb{R}^m$ such that the number of regions formed by these hyperplanes is maximal

Exercise 9 — *Genetic sequencing*

How to rebuild a complete DNA from a set of fragments ?

▶ **Correction**

It requires a minimum of knowledge about sequencing. One method used consists of cutting the DNA into many small pieces because it is the only way to read the nucleotides in order (we cannot do it on the entire DNA). As a result, we end up with many bits of DNA that need to be reassembled.

The advantage is that we did not massacre just one strand of DNA ; we treated many cells at the same time. Each strand was not cut in the same way (the probability of that happening is ridiculous). We will therefore use this to reconstruct the DNA. We look at two pieces and check if they overlap (if the end of one of the pieces is the beginning of the other).

If we can produce a sequence of overlaps using all the pieces, the result is very likely a complete strand of DNA. We obtain a Hamiltonian path problem.
Input : a directed graph $G = (V, A)$, each node is associated with a piece of DNA and two pieces are connected by an arc if the end of the first is the beginning of the second.
Output : a Hamiltonian path of G.

3 Hard modelisations

Each of the following problems gives you too few information. A proper way to modelise them would be to ask questions to experts (those who could almost solve the problem by hand). However, currently, your teacher is the only person you can rely on.

▶ **Correction**

I do not propose any corrections because I think there is really no answer to these exercises. It's too vague. It might be interesting to discuss with the students so they can try to answer this question : what information is missing from me ?

Exercise 10 — *Chemistry problem*

What is the 3D form of a molecule ?

Exercise 11 — *Public transport*

In a city where you have an idea of the visitor numbers thoughout the year, could you adapt the public transport ?

Exercise 12 — *In outer space !*

Will the rocket explode tomorrow ?