Foundations of Operations Research

Master of Science in Computer Engineering

Roberto Cordone
roberto.cordone@unimi.it

Tuesday 13.15 - 15.15

Thursday 10.15 - 13.15

http://homes.di.unimi.it/~cordone/courses/2014-for/2014-for.html

Example: room-mate selection (statement)

Many U.S. colleges provide two-place rooms to their students, and allow them to formulate preferences about the characteristics of their desired room-mates.

These characteristics can be summarized into a "mutual compatibility" coefficient.

Each room has two places, so that the students must be grouped into pairs (for the sake of simplicity, assume that their number is even).

The college administration aims to achieve the maximum overall compatibility.

Practical advice

Do not try to solve the problem: limit yourselves to describing it

1. Define clearly the objective:
   how to compute it and its unit of measure

2. Define clearly one by one the quantities that make a solution unfeasible: how to compute them and their units of measure

   Describe what is forbidden, not what is legal!

3. Choose the decision variables: what quantities have
   - a value which can be fixed (to some extent)?
   - an influence on the objective function?
   - an influence on the feasibility of the solution?

4. Express the objective through the decision variables

5. Express the constraints through the decision variables

If you fail in step 4 or 5

- modify the decision variables
- go back to steps 1 and 2 and improve the definition

Example: room-mate selection (analysis)

- The objective is overall compatibility, which is expressed as the sum of the compatibilities for the room-mate pairs implemented, to be maximized
- Unfeasibility derives from:
  - assigning more or less than one room-mate to each student
  - assigning fractions of different room-mates to a student (even if these fractions sum up to exactly one room-mate)

Some alternative suggestions for the decision variables

- specify for each student the index of the assigned room-mate
  Indices make bad decision variables: they are not quantitative
  (*they cannot be used to compute the objective and the feasibility*)
  they can be permuted with no real effect on the solution
- specify whether each pair of students are room-mates or not,
  i. e. use a binary variable for each student pair
  (*when fixed, they determine the objective and the feasibility*)

Example: room-mate selection (model)

Denote by

- S the set of students (vertices of an undirected graph)
- P the set of all unordered student pairs (edges of the same graph)
- $c : P \to \mathbb{R}^+$ the compatibility between students i and j

Decision variables (one for each student pair $[i, j] \in P$)

- $x_p = 1$ means that the students of pair p are room-mates
- $x_p = 0$ means that the students of pair p are not room-mates

$$\max f = \sum_{p \in P} c_p x_p$$

$$\sum_{p \in P : i \in P} x_p \leq 1 \qquad i \in S$$

$$x_{ij} \in \{0, 1\}$$

This graph problem is known as Maximum weighted matching problem

Example: freight transportation (statement)

A freight transportation company must service a set of requests.

Each request is characterized by a given number of identical items, each with a given weight (expressed in kg) and volume (expressed in m^3*). Items of different requests usually have different weights and volumes.*

The company has a set of trucks, each able to carry a given maximum weight (expressed in kg) and volume (expressed in m^3*) of merchandise.*

The items of each request can be freely splitted among the trucks.

In general, the demand cannot be fully serviced within the time horizon: the items of some requests could be transported in part, or not at all. The missing items are transported later, paying to the customer a penalty (expressed in Euros), which is the same for all items of a request, but depends on the request.

The company aims to decide how to load the trucks in order to minimize the total penalty for the delayed requests.

Example: freight transportation (data)

In order to start formalizing the problem, we can introduce

- a set of requests R
- a set of trucks T
- a number of items required for each request: $n : R \to \mathbb{N}$
- a weight for each item of a request (in kg/item): $w : R \to \mathbb{R}^+$
- a volume for each item of a request (in m^3/item): $v : R \to \mathbb{R}^+$
- a penalty for each unserviced item of a request (in Euros/item): $p : R \to \mathbb{R}^+$
- a maximum weight which can be loaded on each truck (in kg): $W : T \to \mathbb{R}^+$
- a maximum volume which can be loaded on each truck (in m^3): $V : T \to \mathbb{R}^+$

Example: freight transportation (analysis)

- The objective is the penalty for the delayed items (to be minimized), or equivalently the penalty for the transported items (to be maximized)
  Both are measured in Euros (*Why are they equivalent?*)
- The unfeasibility derives from
  - loading one of the trucks with an excessive weight
  - loading one of the trucks with an excessive volume
  - loading negative amounts of items (*remember: nothing is obvious!*)
  - transporting more items than required
  - transporting fractionary items

This suggests to define integer decision variables

- x_{rt} is the number of items of request $r \in R$ loaded on truck $t \in T$

Defining a binary variable for each single item is also possible, but would be

- unnecessary: different items in the same request cannot be distinguished (*they can be freely exchanged: weight, volume, and penalty are the same*)
- computationally inefficient: more variables and constraints imply exponentially more solutions and computation time (*ILP is $\mathcal{NP}$-complete*)

Example: freight transportation (model)

$$\max f = \sum_{r \in R} \sum_{t \in T} p_r x_{rt}$$

$$\sum_{r \in R} w_r x_{rt} \leq W_t \qquad t \in T$$

$$\sum_{r \in R} v_r x_{rt} \leq V_t \qquad t \in T$$

$$\sum_{t \in T} x_{rt} \leq n_r \qquad r \in R$$

$$x_{rt} \in \mathbb{N}$$

Is it necessary to impose integrality on the decision variables?

Or can we relax the constraint on the basis of an "average" perspective? (repetition of the same solution for several times)

What is a good reason for which we cannot, in this case?

Another model (multi-dimensional knapsack)

You have decided to take a flight, travelling only with hand luggage.

The airline company is very strict on the weight limit, and your luggage is hard-sided. Unfortunately, the things you want to carry do not fit in.

Each object has an intrinsic utility, a weight, and a volume.

You have to choose the objects to carry with you, maximizing the utility of your luggage.

Taxonomy of linear constraints

Classifying the forms of *LP* and *ILP* constraints teaches to recognize

- submodels of a practical problem: this helps the modelling phase
- substructures of the theoretical model: this helps the solving phase

Any linear constraint on variables in $\mathbb{R}^n$ or $\mathbb{N}^n$ falls into three possible classes

1. resource constraints

$$a^T x = \sum_{j=1}^{n} a_j x_j \leq b \text{ with } a_j \geq 0 \text{ for } j = 1, \dots, n$$

   - b describes the available amount of a resource
   - $a^T x$ describes the amount of resource consumed by the solution x

2. demand constraints

$$a^T x = \sum_{j=1}^{n} a_j x_j \geq b \text{ with } a_j \geq 0 \text{ for } j = 1, \dots, n$$

   - b describes the required amount of a good
   - $a^T x$ describes the amount of good provided by the solution x

3. balance constraints

$$\sum_{j=1}^{n'} a_j x_j \leq \sum_{j=n'+1}^{n} a_j x_j + b \text{ with } a_j \geq 0 \text{ for } j = 1, \dots, n$$

   The solution x implies two amounts which must respect a quantitative relation

Taxonomy of variables

Different kinds of decision variables correspond to different kinds of decisions

- continuous variables represent measurable quantities affecting the objective and/or the feasibility of the problem, and freely divisible
- integer variables represent measurable quantities affecting the objective and/or the feasibility of the problem, with indivisible units
- binary variables represent the occurrence of an event or its complement

Variables with multiple indices refer to combinations of independent events

- travelling along street s with vehicle v in day d or not: $x_{svd} \in \{0, 1\}$
- performing operation o on machine m in time slot t or not: $x_{omt} \in \{0, 1\}$
- assigning shift s in ward w to nurse n: $x_{swn} \in \{0, 1\}$

Binary variables and constraints imposed on them can be used to

1. forbid some combinations of events
   - impose requirements on the number of events occurring or not
   - express reciprocal implications or prohibitions among events
2. express some nonlinear relations between variables and constraints

Resource, demand and balance constraints

When considering Combinatorial Optimization problems,
binary variables model the selection of a subset of elements

Many practical problems with two conflicting objectives are modelled as

1. the maximization of a utility under a resource constraint
   - select objects to carry with a maximum capacity constraint
   - select projects to perform with a maximum workforce constraint
   - select investments to perform with a maximum budget constraint
2. the minimization of a cost under a demand constraint
   - select objects to carry with a minimum utility constraint
   - select projects to perform with a minimum quality constraint
   - select investments to perform with a minimum gain constraint

This is why such constraints are called "natural"

Cardinality constraints

A relevant special form of constraints on binary variable is when

$$a_j \in \{0, 1\} \text{ for } j = 1, \ldots, n$$

These cardinality constraints require that the solution include

- at most a given number of events

$$\sum_{j=1}^{n} x_j \leq b$$

- at least a given number of events

$$\sum_{j=1}^{n} x_j \geq b$$

- enough more events of one kind than of another kind

$$\sum_{j=1}^{n'} x_j \leq \sum_{j=n'+1}^{n} x_j + b$$

Examples:

- opening shops
- performing investments (to diversify enough, and not too much)
- assigning room-mates to students (exactly one, for example)
- assigning jobs to machines

Store opening plan (statement)

A chain of stores is going to expand in a new region; the stores can be of different types. A set of potential positions for new stores have been identified; each position has room for a single store.

For each position and type, the yearly amortized opening cost and an expected yearly income have been estimated.

The total available budget for opening new stores has been defined.

The number of stores of each type cannot exceed a maximum (to diversify the offer).

For the sake of modelling, let

- T be the set of store types
- P be the set of potential positions
- $c : T \times P \to \mathbb{R}^+$ be the opening cost of a store of type t in position p
- b be the overall budget
- $\pi : T \times P \to \mathbb{R}^+$ be the yearly income of a store of type t in position p
- $u : T \to \mathbb{N}$ be the maximum number of stores of type t

Store opening plan (analysis)

- The objective is to maximize the difference between the sum of the incomes provided by the open stores and the sum of their opening costs
  So we must compute the total income and the total opening cost
- Unfeasibility derives from
  - exceeding the budget with the total opening cost (resource)
  - exceeding the maximum number of stores of each type (cardinality)
  - exceeding the maximum number of stores in each position (cardinality)
  - opening fractionary stores (integrality)

  So we must compute the total opening cost and the number of open stores for each type and position

All this points towards binary decision variables

- $x_{tp} = 1$ if a store of type $t \in T$ is opened in position $p \in P$
- $x_{tp} = 0$ if no store of type $t \in T$ is opened in position $p \in P$

Why not integer variables?

Store opening plan (model)

Given a subset $X \subseteq T \times P$, consider $\sum_{(t,p) \in X} x_{tp}$

- the sum ignores variables with $(t,p) \notin X$
- the contribution of variables with $x_{tp} = 0$ is zero
- the contribution of variables with $x_{tp} = 1$ is one

The number of variables $x_{tp} = 1$ with $(t,p) \in X$ is $\sum_{(t,p) \in X} x_{tp}$

$$\begin{aligned}
\max f &= \sum_{t \in T} \sum_{p \in P} (\pi_{tp} - c_{tp}) \, x_{tp} \\
\sum_{t \in T} \sum_{p \in P} c_{tp} x_{tp} &\leq b \\
\sum_{t \in T} x_{tp} &\leq 1 \qquad p \in P \\
\sum_{p \in P} x_{tp} &\leq u_t \qquad t \in T \\
x_{tp} &\in \{0,1\} \qquad t \in T, p \in P
\end{aligned}$$

1. maximize the yearly profit f provided by the open stores
2. the total opening cost respects the budget
3. at most one store in each position
4. at most u_t stores of type t
5. the stores are open $(x_{tp} = 1)$ or not $(x_{tp} = 0)$

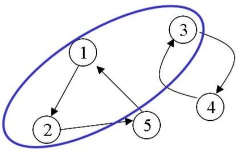
(Sub)tour elimination constraints

Many graph problems require solutions which forbid cycles or circuits

- Minimum Spanning Tree Problem
- Shortest Path Problem (with negative cost circuits)
- Travelling Salesman Problem (this problem allows only circuits including all the nodes, but forbids subtours)

The requirement can be modelled by a family of cardinality constraints, the (sub)tour elimination constraints (*their number is exponential: tricks required!*)

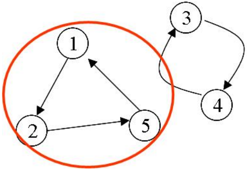
$$\sum_{(i,j)\in A: i,j\in S} x_{ij} \leq |S| - 1 \quad \text{for all } S \subseteq N : |S| \geq 2$$



$S = \{1, 2, 3\}$

$x_{12} + x_{13} + x_{21} + x_{23} + x_{31} + x_{32} \leq 3 - 1$

$1 + 0 + 0 + 0 + 0 + 0 \leq 2$ (satisfied)



$S = \{1, 2, 5\}$

$x_{12} + x_{15} + x_{21} + x_{25} + x_{51} + x_{52} \leq 3 - 1$

$1 + 0 + 0 + 1 + 1 + 0 \leq 2$ (violated)

If circuits on all nodes are allowed, apply the constraint only on $S \subset N$

Packing, partitioning and covering constraints

A relevant special case of cardinality constraint occurs when $b = 1$

- packing constraints: at most one event in J must occur
  (e. g., the matching problem)

$$\sum_{j \in J} x_j \leq 1$$

- partitioning constraints: exactly one event in J must occur
  (e. g., the assignment problem)

$$\sum_{j \in J} x_j = 1$$

- covering constraints: at least one event in J must occur

$$\sum_{j \in J} x_j \geq 1$$

Example: ambulance location (statement)

A health emergency service must distribute the available ambulances around the town. A number of suitable locations have been identified.

The potential users of the service have been sampled with sufficient density, defining their approximate positions on the map of the town.

The law prescribes that an ambulance must be able to reach every user within a given limit time. The users whose travel time from an ambulance location is larger than the given limit are out of reach and must be assigned to other ambulances.

Locate the ambulances so as to keep all the potential users within the reach of at least one ambulance, minimizing the number of ambulances.

Example: facility location (model)

For the sake of modelling, let us define

- I be the set of the potential users (rows)
- J be the set of the potential locations (columns)
- $a_{ij} = 1$ if user i is within reach of location j, $a_{ij} = 0$ otherwise
  The covering matrix derives from the time limit, the travel speed and the lengths of the shortest paths on the street network

and let us introduce the following binary decision variables

- $x_j = 1$ if an ambulance is located in $j \in J$
  (it is useless to locate more ambulances in the same location)
- $x_j = 0$ if no ambulance is located in $j \in J$

This is known as Set Covering problem

$$\min f(x) = \sum_{j \in J} x_j$$

$$\sum_{j \in J} a_{ij} x_j \geq 1 \qquad i \in I$$

$$x_j \in \{0, 1\} \qquad j \in J$$

If necessary, a cost function $c : J \to \mathbb{R}^+$ can be defined on the columns

Logical constraints

The "Merchant of Venice" problem: find the locked case containing Portia's picture (or be banned forever), knowing that

- there are 3 cases
- exactly one of them contains the picture
- each case bears an inscription
- at most one of the inscriptions is true

1	2	3
The picture is in this case	The picture is not in this case	The picture is not in case 1

Logical constraints

1	2	3
The picture is in this case	The picture is not in this case	The picture is not in case 1

Let us define two families of binary variables, related to events

$$x_i = \begin{cases} 1 & \text{if the picture is in case } i \\ 0 & \text{if the picture is not in case } i \end{cases} \qquad i \in \{1, 2, 3\}$$

$$y_i = \begin{cases} 1 & \text{if the inscription on case } i \text{ is true} \\ 0 & \text{if the inscription on case } i \text{ is false} \end{cases} \qquad i \in \{1, 2, 3\}$$

Logical constraints

There are two obvious cardinality constraints

1. one of the cases contains the picture: $x_1 + x_2 + x_3 = 1$
2. at most one of the inscriptions is true: $y_1 + y_2 + y_3 \leq 1$

The events expressed by variables x_i and y_i are linked by logical relations. Correspondingly, the variables must be linked by constraints

5. inscription 1 is true $\Leftrightarrow$ the picture is in case 1
6. inscription 2 is true $\Leftrightarrow$ the picture is not in case 2
7. inscription 3 is true $\Leftrightarrow$ the picture is not in case 1

How to model these relations?

Logical constraints

Binary variables x_i behave exactly like Boolean variables ξ_i

$$\xi = \mathit{false} \quad \leftrightarrow \quad x = 0$$
$$\xi = \mathit{true} \quad \leftrightarrow \quad x = 1$$

Logical negation $\bar{\xi}$ corresponds to $1 - x$: $\bar{\xi} \leftrightarrow 1 - x$

ξ	$\bar{\xi}$
false	*true*
true	*false*

x	$1 - x$
0	1
1	0

Implication corresponds to $\leq$

$$\xi_1 \Rightarrow \xi_2 \ \leftrightarrow \ x_1 \leq x_2$$

x_1	x_2	$x_1 \leq x_2$
false	*false*	-
false	*true*	-
true	*false*	NO
true	*true*	-

ξ_1	ξ_2	$\xi_1 \Rightarrow \xi_2$
0	0	-
0	1	-
1	0	NO
1	1	-

Consequently, double implication corresponds to $=$

$$\xi_1 \Leftrightarrow \xi_2 \ \leftrightarrow \ x_1 = x_2$$

Logical constraints

The overall formulation of the "Merchant of Venice" problem is

$$
\begin{aligned}
\sum_{i=1}^{3} x_i &= 1 \\
\sum_{i=1}^{3} y_i &\leq 1 \\
y_1 &= x_1 \\
y_2 &= 1 - x_2 \\
y_3 &= 1 - x_1 \\
x_1, x_2, x_3 &\in \{0, 1\} \\
y_1, y_2, y_3 &\in \{0, 1\}
\end{aligned}
$$

- no objective function
  (a single solution is feasible)
- exactly one case contains the picture
- at most one inscription is true
- inscription 1 is true ⇔ the picture is in case 1
- inscription 2 is true ⇔ the picture is not in case 2
- inscription 3 is true ⇔ the picture is not in case 1

So, where is the picture, after all?

Logical constraints

Logical disjunction corresponds to a system of constraints

$$\xi_3 = (\xi_1 \text{ OR } \xi_2) \leftrightarrow \begin{cases} x_3 \geq x_1 \\ x_3 \geq x_2 \\ x_3 \leq x_1 + x_2 \end{cases}$$

ξ_1	ξ_2	ξ_1 OR ξ_2
false	*false*	*false*
false	*true*	*true*
true	*false*	*true*
true	*true*	*true*

x_1	x_2	x_3	$x_3 \geq x_1$	$x_3 \geq x_2$	$x_3 \leq x_1 + x_2$
0	0	0	-	-	-
0	0	1	-	-	NO
0	1	0	-	NO	-
0	1	1	-	-	-
1	0	0	NO	-	-
1	0	1	-	-	-
1	1	0	NO	NO	-
1	1	1	-	-	-

Logical constraints

Logical conjunction corresponds to a system of constraints

$$\xi_3 = (\xi_1 \text{ AND } \xi_2) \leftrightarrow \begin{cases} x_3 \leq x_1 \\ x_3 \leq x_2 \\ x_3 \geq x_1 + x_2 - 1 \end{cases}$$

ξ_1	ξ_2	ξ_1 AND ξ_2
false	*false*	*false*
false	*true*	*false*
true	*false*	*false*
true	*true*	*true*

x_1	x_2	x_3	$x_3 \leq x_1$	$x_3 \leq x_2$	$x_3 \geq x_1 + x_2 - 1$
0	0	0	-	-	-
0	0	1	NO	NO	-
0	1	0	-	-	-
0	1	1	NO	-	-
1	0	0	-	-	-
1	0	1	-	NO	-
1	1	0	-	-	NO
1	1	1	-	-	-