Linear transport problems

formulation and assignment problems

Hugues Talbot

CentraleSupélec
Centre de Vision Numérique (CVN)

May 2, 2018

Outline

Particular linear problems: transport problems

- Some problems in linear programming have a particular structure that can be exploited ;
- These can of course be solved by the simplex algorithm, but there exist methods to solve them very efficiently ;
- Some of these problems are formulated as integer problems as well. However, even in this case the solution is not harder to compute, contrary to standard LP vs. IP.
- We can start with an example:

Electricity distribution

Consider a set of cities powered by some electricity plant. The situation is summarized as follows:

	City 1	City 2	City 3	City 4	Power supplied (GWh)
Plant 1	€8	€6	€10	€9	35
Plant 2	€9	€12	€13	€7	50
Plant 3	€14	€9	€16	€5	40
Demand (GWh)	45	20	30	30	

Here, the costs in the middle of the matrix represent production costs per GWh.

Let us formulate the problem consisting of powering all the cities at the lowest possible cost.

Formulation

- x_{ij} = quantity of electricity in GWh produced at plant i and consumed in city j.
- Transport cost from all the plants = total cost =

$$\begin{aligned} z = \; & 8\,x_{11} + 6\,x_{12} + 10\,x_{13} + 9\,x_{14} \\ & + 9\,x_{21} + 12\,x_{22} + 13\,x_{23} + 7\,x_{24} \\ & + 14\,x_{31} + 9\,x_{32} + 16\,x_{33} + 5\,x_{34} \end{aligned}$$

Formulation : constraints

- Production constraints

$$\begin{aligned} x_{11} + x_{12} + x_{13} + x_{14} &\leq 35 \\ x_{21} + x_{22} + x_{23} + x_{24} &\leq 50 \\ x_{31} + x_{32} + x_{33} + x_{34} &\leq 40 \end{aligned}$$

- Consumption constraints

$$\begin{aligned} x_{11} + x_{21} + x_{31} &\geq 45 \\ x_{12} + x_{22} + x_{32} &\geq 40 \\ x_{13} + x_{23} + x_{33} &\geq 30 \\ x_{14} + x_{24} + x_{34} &\geq 30 \end{aligned}$$

- Usual constraints: ($x_{ij} \geq 0$)

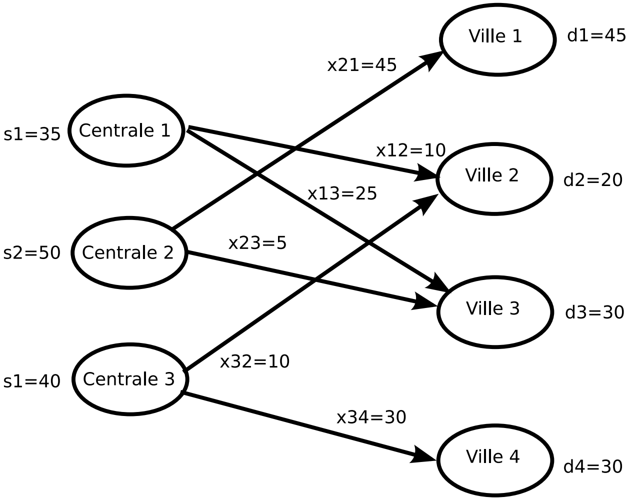
Resolution

- This is a standard LP
- It can be solved using standard tools (e.g. Excel, Matlab, etc)
- We find the following solution:

x_{12}	x_{13}	x_{21}	x_{23}	x_{32}	x_{34}
10	25	45	5	10	30

For a total cost of 1020.

Graphical solution



General form

The general form of a transport problem is the following:

$$\min \sum_{i=1}^{i=m} \sum_{j=1}^{j=n} c_{ij} x_{ij}$$

$$s.t \sum_{j=1}^{j=n} x_{ij} \leq s_i, i \in \{1, \ldots, m\} \qquad \text{(production constraints)}$$

$$\sum_{i=1}^{i=m} x_{ij} \geq d_j, j \in \{1, \ldots, n\} \qquad \text{(consumption constraints)}$$

Terminology

- If the problem is to find a maximum cost, this is still a transport problem:
- If we have
$$\sum_{i=1}^{i=m} s_i = \sum_{j=1}^{j=n} d_j,$$
the problem is *balanced*.
The given example earlier is balanced
- In a balanced problem, all the constraints are equality constraints (why ?).

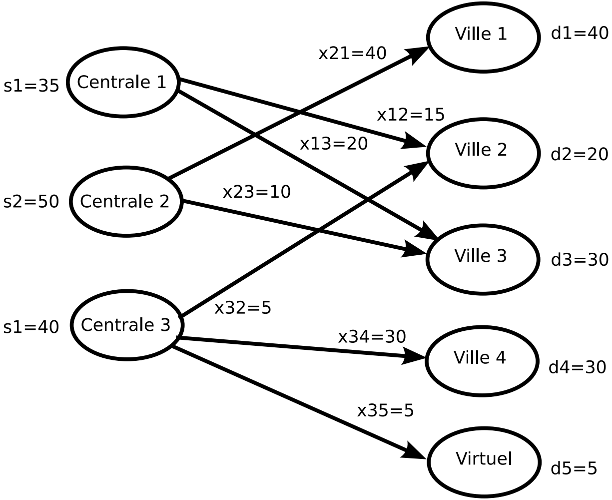
Balanced problems

- It is preferable to only consider balanced problems. Indeed we shall see that these are *comparatively* easier to solve. Contrary to the standard LP case, it is easier to find a starting feasible basis solution in this case.
- Also, in this case, the simplex operations reduce to additions and substractions.

How to balance a transport problem

- To balance a transport problem, for which there is too much production capacity, it is sufficient to create a *virtual consumption node*, for which the demand constraint will be exactly that of the excess production capacity, and for which transport costs are zero (Q: why zero ?).
- The excess, unused capacity will be transported to the virtual node. Naturally, the production path with the (Q: lowest ? highest?) cost will be the one sending its output to the virtual node
- As an example, in the preceeding formulation, assume city one now consumes only 40 GWh. There is an excess of 5 GWh that we can allocated to a virtual consumption point.
- We note the optimal solution is different in this case.

Graphical solution of the non-balanced case



Representation in a tableau

- It is efficient to represent transport problems in a tableau:

	Ville 1		Ville 2		Ville 3		Ville 4		Offre
centrale 1	0	8	10	6	25	10	0	9	35
centrale 2	45	9	0	12	5	13	0	7	50
centrale 3	0	14	10	9	0	16	30	5	40
Demande	45		20		30		30		

- We note the sum of values in rows and columns.

Balancing the problem in the case of excess consumption

- When production is unable to meet consumption, there is no feasible solution. (E.g. if we reduce the capacity of power plant 1 to 30 GWh).
- Sometimes, the model allows for excess production capacity to become available at a higher cost (e.g. buying electricity from another country).
- Example : water production

Water production problem

- Two reservoirs are available to supply three cities with drinkable water. Each reservoir can produce 50 000 m^3 of water a day.
- Water demand for each city is 40 000 m^3/day
- If reservoirs cannot meet demand, there is a penalty per 1000 m^3: €20 for city 1, €22 for city 2 et €23 for city 3.
- Transport costs per 1000 m^3 are summarized here :

De à	City 1	City 2	City 3
Reservoir 1	€7	€8	€10
Reservoir 2	€9	€7	€8

Solution

	City 1		City 2		City 3		Supply
Reservoir 1	20	7	30	8	0	10	50
Reservoir 2	0	9	10	7	40	8	50
Virtual	20	20	0	22	0	23	20
Demand	40		40		40		

Modeling inventory problems

On an example:

- The company NiceSails produces boat sails. This is its demand sheet for the following trimesters:

	1st	2nd	3rd	4th
orders	40	60	75	25

- NS must deliver on time. It has a current inventory of 10 sails and must decide how many sails to produce every trimester at the beginning of each of them. We assume only the sails produced during a particular trimester can be sold.
- Each trimester, NS can produce up to 40 sails at a cost of €400 per sail. By paying it employees overtime, it can produce another 40 sails at a cost of €450 each.
- At the end of each trimester, unsold sails incur an inventory cost of €20.
- We must meet demand and produce the sails at a minimum cost.

Sails production constraints

- Supply nodes:
  1. Initial inventoryl ($s_1 = 10$)
  2. Production for the first trimester : Normal ($s_2 = 40$), extra ($s_3 = 40$).
  3. Production for the second trimester : Normal ($s_4 = 40$), extra ($s_5 = 40$).
  4. Production for the third trimester : Normal ($s_6 = 40$), extra ($s_7 = 40$).
  5. Production for the fourth trimester : Normal ($s_8 = 40$), extra ($s_9 = 40$).

  Total de l'offre: 330

Consumption constraints for the sails

- Demand points:
  1. Demand first trimester ($d_1 = 40$)
  2. Demand second trimester ($d_2 = 60$)
  3. Demande third trimester ($d_3 = 75$)
  4. Demand fourth trimester ($d_4 = 25$)
  5. Virtual node to balance things out ($d_5 = 330 - 200 = 130$).
- Remark : we must make it impossible to produce a sail in the 2nd trimester for sale in the first. Things must remain causal.

Sail tableau

	Trimester 1	Trimester 2	Trimester 3	Trimester 4	Virtual	Supply
Initial	0	20	40	60	0	
T1 Normal	400	420	440	460	0	
T1 Extra	450	470	490	510	0	
T2 Normal	M	400	420	440	0	
T2 Extra	M	450	470	490	0	
T3 Normal	M	M	400	420	0	
T3 Extra	M	M	450	470	0	
T4 Normal	M	M	M	400	0	
T4 Extra	M	M	M	450	0	
Demand	40	60	75	25	130	

Solution for the sail problem

	Trimester 1		Trimester 2		Trimester 3		Trimester 4		Virtual		Supply
Initial	10	0		20		40		60		0	10
T1 Normal	30	400	10	420		440		460		0	40
T1 Extra		450		470		490		510	40	0	40
T2 Normal		M	40	400		420		440		0	40
T2 Extra		M	10	450		470		490	30	0	40
T3 Normal		M		M	40	400		420		0	40
T3 Extra		M		M	35	450		470	5	0	40
T4 Normal		M		M		M	25	400	15	0	40
T4 Extra		M		M		M		450	40	0	40
Demand	40		60		75		25		130		

Finding an initial Feasible Basis Solution

- We consider a transport problem with m supply and n demand nodes. This is a problem with $m+n$ constraints.
- It is usually difficult to find an initial Feasible Basis Solution for general linear programming problems (why?), however it is easy for transport problems.
- An important remark is that in transport problems with $m+n$ constraints, one of these is redundant.

Independent variables

- For instance, in the case of electricity distribution, if we ignore the first equality constraint, we find that it is nonetheless satisfied by the solution.
- Within the $m + n - 1$ remaining constraints, not every subset of $m + n - 1$ variable constitutes a basis.

Loops and bases

- A sequence of 4 or more cells constitutes a basis if and only if:
    - Every consecutive pair of cells are either on the same line or column;
    - No consecutive triplets of cells are on the same line or column;
    - The first and last cell are on the same line or column.
- The following theorem is verified :
  In a balanced transport problem, with m producers and n consumers, a set of $m + n - 1$ variables constitutes a basis if and only if no subset of this variable forms a loop.

Methods for finding an initial FBS

There are three classical methods:

1. The upper-left-corner method ;
2. The minimum cost method ;
3. TheVOGEL method.

We will explain them in the following lecture. In the meantime we will focus on assignment problems, which are a subclass of transport problems.

Example of an assignment problem

- A factory M includes 4 identical machines and 4 different tasks to complete, which can be assigned to any machine.
- The tasks are long and automated, so each machine must accomplish one and only one task. The setup time is given by the following table:

	T1	T2	T3	T4
Machine 1	14	5	8	7
Machine 2	2	12	6	5
Machine 3	7	8	3	9
Machine 4	2	4	6	10

- The factory must associate each machine to a task so that the setup time, which requires manual intervention, is minimal
- Formulate and solve this problem.

Assignment problem formulation

- Assignment problem are special balanced transport problems for which producer and consumer capacity is exactly 1.
- As a transport problem, enforcing an integer solution is free if all constraints are integer.
- Question: can we have assignment $m \times n$ problem with $n < m$? If so, how to describe this situation and how to remedy it ? Can we have the opposite $(n > m)$?

Finding a solution

- In any $m \times m$ assignment problem, we will always have, m variables with value 1 and $m \times m - 1$ with value 0 (why ?)
- We can find an initial FBS and we can solve it using the transport simplexe that we will see in the next lecture, however, many basis variables are degenerate and so the simplexe method is not well adapted.
- There exist a specific method that is well adapted to balanced assignment problems, the “Hungarian” algorithm.

History of the "Hungarian" algorithm

- The assignment problem was studied by Jacobi in the 19th century and the solution reprinted here was described by him and published posthumously in latin in 1890[1]
- Two Hungarian mathematicians studied it independently : Dénes König and Jenó Egerváry,
- in 1955 Harold Kuhn developed it and published it, naming it the "Hungarian" method.
- James Munkres reviewed the algorithm and showed that it was strongly polynomial with a complexity of $O(n^4)$.
- Edmonds and Karp reduced the complexity to $O(n^3)$.
- Ford and Fulkerson adapted the method to general transport problems.

[1] "De aequationum differentialum systemate non normali ad formam normalem revocando", reproduced in C.G.J. Jacobi's gesammelte Werke, fünfter Band, herausgegeben von K. Weierstrass, Berlin, Bruck und Verlag von Georg Reimer, 1890, p. 485-513

The “Hungarian” algorithm

In spite of this distinguished history, the method is simple:

1. Find the minimum element in every line of the matrix $m \times m$. Construct a new matrix by subtracting this minimum from every number in the line.
2. In this new matrix, find the minimum in each column. Subtract this minimum from all the element in each column.
3. Find the minimum number of lines that can cover all the zeros in this matrix. If fewer than m lines are sufficient, go to step 4, else stop.
4. From all the elements not covered by a line in step 3 (so none of them are zeros), find the minimum k. Subtract this minimum k from all the non-covered elements and *add* k to all the elements that are covered twice.
5. go to step 3.

An example of resolution

1- Line minima

	Task 1	Task 2	Task 3	Task 4	Min
Machine 1	14	5	8	7	5
Machine 2	2	12	6	5	2
Machine 3	7	8	3	9	3
Machine 4	2	4	6	10	2

An example of resolution

2- Column minima

	Task 1	Task 2	Task 4	Task 4
Machine 1	9	0	3	2
Machine 2	0	10	4	3
Machine 3	4	5	0	6
Machine 4	0	2	4	8
Minimum	0	0	0	2

An example of resolution

4- Line covering: now 3 lines are sufficient to cover all the zeros.

	T1	T2	T3	T4
M1	9	0	3	0
M2	0	10	4	1
M3	4	5	0	4
M4	0	2	4	6

An example of resolution

4- The minimum in the non-covered cells is 1, value that we remove from the non-covered cells and add to the doubly-covered ones:

	T1	T2	T3	T4
M1	10	0	3	0
M2	0	9	3	0
M3	5	5	0	4
M4	0	1	3	5

Optimal basis choice

- A the end of the algorithm, m zeros are present and covered in the matrix.
- There exist at least one set of variable forming a basis constituted of zeros. This basis is an initial FBS.

	T1	T2	T3	T4
M1	10	(0)	3	0
M2	0	9	3	(0)
M3	5	5	(0)	4
M4	(0)	1	3	5

- This FBS is optimal for the assignment
- Here : $x_{12} = x_{24} = x_{33} = x_{41} = 1$.

Justification

- If a constant k is added to any line or column in a balanced transport problem, this does not change the solution. It only adds this constant k to the cost.
- Similarly, step 3 of the Hungarian method does not change the optimum because it is equivalent to doing simultaneously:
  - add k to each line covered by a horizontal line ;
  - subract k to each line not covered by a vertical line.
- Step 1 and 2 create at least one zero per line or column. Step 3 creates at least one more zero each time.
- Throughout the procedure, costs remain non-negative.
- Eventually when the algorithm finishes, the optimum remains the same, and must have zero cost. It consists of choosing a basis among the zeros.

Conclusion

- In this lecture we have introduced transport (also known as transportation) problems;
- These are special case of LP problems that are easy to solve, even if all constraints and solutions are integer.
- We have particularly looked at a solution for assignment problems.
- Next time we will introduce a generic solution called the transportation simplex.