Professeur : Michel Bierlaire, Assistant responsable : Nikola Obrenovic, Nourelhouda Dougui

Réseaux et transbordement – corrigé (2 novembre 2018)

Solution de la question 1:

1. A transhipment problem has the form :

$$\min_{x \in \mathbb{R}^n} \sum_{(i,j) \in \mathcal{A}} c_{ij} x_{ij}$$

subject to

$$\sum_{j|(i,j) \in \mathcal{A}} x_{ij} - \sum_{k|(k,i) \in \mathcal{A}} x_{ki} = s_i$$

$$\ell_{ij} \leq x_{ij} \leq u_{ij}, \quad \forall (i,j) \in \mathcal{A},$$

where $s \in \mathbb{R}^m$ represents the supply/demand of flow, $c \in \mathbb{R}^n$ represents the cost of transporting a unit of flow on each arc and $\ell \in \mathbb{R}^n$, $u \in \mathbb{R}^n$ represent the lower and upper capacity of each arc respectively.

We construct a network where each node corresponds to one of the three cities ($m = 3$) and three arcs ($n = 3$) linking Los Angeles to Boston, Los Angeles to New York and New York to Boston. The cost of each arc corresponds to the cost of transporting one unit of honey through that arc.
To calculate s_i for the three nodes, we need to calculate the divergence of the node : for each city we substract the demand of the customers from the production of honey in that city.

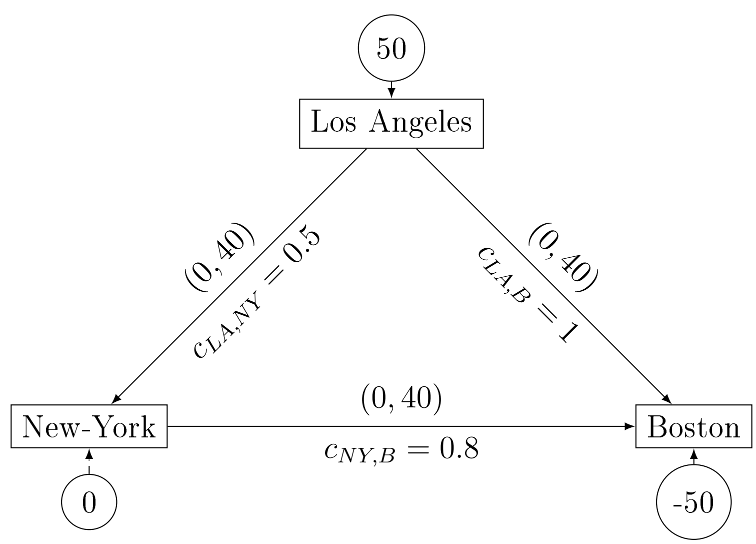
We obtain :
- $s_{\text{LA}} = 50$ is the divergence of node *Los Angeles* : it is a supply node.
- $s_{\text{NY}} = 0$ is the divergence of node *New-York* : it is a transit node.
- $s_{\text{B}} = -50$ is the divergence of node *Boston* : it is a demand node.

We can see that there is no arcs with *Los Angeles* as a downstream node. This is consistent with *Los Angeles* being a supply node. Besides, there is no arcs with *Boston* as an upstream node. This is consistent with *Boston* being a supply node.

The cost vector c representing the cost of transporting a unit of flow on each arc is composed of :
- $c_{\text{LA,B}} = 1$ which is the cost of the arc from node *Los Angeles* to node *Boston*.
- $c_{\text{LA,NY}} = 0.5$ which is the cost of the arc from node *Los Angeles* to node *New York*.
- $c_{\text{NY,B}} = 0.8$ which is the cost of the arc from node *New York* to node *Boston*.

Furthermore, the flow on each arc has upper bound 40 since one can not transport more than 40 units per day on each arc. It has lower bound 0 since negative values make no sense.

We obtain the following network on which the lower and upper bound capacity on each arc are represented by (ℓ_{ij}, u_{ij}) :



The decision variables are :

— $x_{LA,B}$ the flow on the arc from node *Los Angeles* to node *Boston*.
— $x_{LA,NY}$ the flow on the arc from node *Los Angeles* to node *New York*.
— $x_{NY,B}$ the flow on the arc from node *New York* to node *Boston*.

The objectif is to minimize the total cost, that is :

$$\min_{x\in\mathbb{R}^3} x_{LA,B} + 0.5x_{LA,NY} + 0.8x_{NY,B}$$

Two type of constraints must be satisfied. First, the divergence of each node must correspond to the value of s_i :

$$\begin{aligned} x_{LA,NY} + x_{LA,B} &= 50 \\ x_{NY,B} - x_{LA,NY} &= 0 \\ -x_{NY,B} - x_{LA,B} &= -50 \end{aligned}$$

Second, the value of the flow on each arc must verify the capacity constraints :

$$\begin{aligned} 0 \leq x_{LA,NY} &\leq 40 \\ 0 \leq x_{LA,B} &\leq 40 \\ 0 \leq x_{NY,B} &\leq 40. \end{aligned}$$

Therefore, the minimization problem is :

$$\min_{x\in\mathbb{R}^3} x_{LA,B} + 0.5x_{LA,NY} + 0.8x_{NY,B}$$

subject to

$$\begin{aligned} x_{LA,NY} + x_{LA,B} &= 50 \\ x_{NY,B} - x_{LA,NY} &= 0 \\ -x_{NY,B} - x_{LA,B} &= -50 \\ 0 \leq x_{LA,NY} &\leq 40 \\ 0 \leq x_{LA,B} &\leq 40 \\ 0 \leq x_{NY,B} &\leq 40. \end{aligned}$$

2. We want to represent the problem in standard form using a network transformation.To reach this goal, we need to transform the upper bound constraints into equality constraints. Then, for each arc (i, j) we include a slack variable y_{ij}. We, thus, obtain :

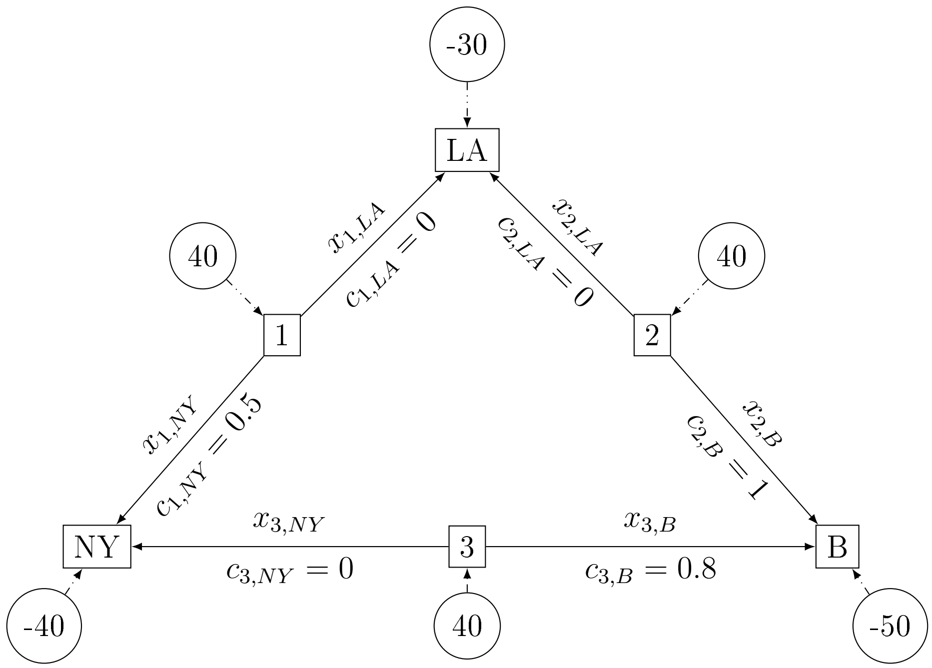
$$\begin{aligned} x_{LA,NY} + y_{LA,NY} &= 40 \\ x_{LA,B} + y_{LA,B} &= 40 \\ x_{NY,B} + y_{NY,B} &= 40. \end{aligned}$$

These constraints can be interpreted as a supply/demand constraint. To do so, we introduce a new node between all pairs of connected nodes : nodes $1, 2$ and 3. Those nodes must have divergence equal to the upper bound of the capacity of the original arc. In our case they all have divergence 40 and thus are supply nodes.

Then, we have to recalculate the supply of the original nodes using the formula

$$s'_i = s_i - \sum_{(i,j) \in \mathcal{A}} u_{ij}.$$

where i is the upstream node for arc (i, j) and u_{ij} is the upper bound of the capacity of this arc.
— $s'_{LA} = s_{LA} - u_{LA,NY} - u_{LA,B} = 50 - 40 - 40 = -30$,
— $s'_{NY} = s_{NY} - u_{NY,B} = 0 - 40 = -40$,
— $s'_B = s_B = -50$.
Then, we define for each new node :
— an arc from the new node to the downstream node of the original arc, with flow x_{ij} and cost the initial cost.
— an arc from the new node to the upstream node of the original arc, with flow $y_{ij} = u_{ij} - x_{ij}$ and cost zero.
Since the capacities have all lower bound zero and are all above unbounded, we do not draw them and we end up with the following network :



The associated linear optimization problem is :

$$\min x_{2,B} + 0.8x_{3,B} + 0.5x_{1,NY}$$

subject to

$$\begin{aligned}
-x_{1,LA} - x_{2,LA} &= -30\\
x_{1,LA} + x_{1,NY} &= 40\\
-x_{1,NY} - x_{3,NY} &= -40\\
x_{3,NY} + x_{3,B} &= 40\\
-x_{2,B} - x_{3,B} &= -50\\
x_{2,LA} + x_{2,B} &= 40\\
x_{1,LA}, x_{1,NY}, x_{2,LA}, x_{2,B}, x_{3,NY}, x_{3,B} &\geq 0.
\end{aligned}$$

3. The rows of the incidence matrix correspond to the six supply/demand constraints of the problem in standard form, while each column corresponds to one variable. The matrix values are the coefficients of variables in the supply/demand constraints.

Thus, the incidence matrix corresponding to that network is

$$A = \begin{matrix} LA \\ NY \\ B \\ 1 \\ 2 \\ 3 \end{matrix} \begin{pmatrix} -1 & 0 & -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 & 0 & -1 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

We notice that A is not full rank : the sum of all the rows is 0, as there is a 1 and a -1 in each column. This means that there is a redundant constraint. It is induced by the fact that the sum of the divergences of all nodes must be zero. Thus the divergence of the last node, here *Boston*, is uniquely determined by the divergence of all other nodes :

$$\begin{aligned} div(x)_B &= -\text{div}(x)_{LA} - \text{div}(x)_{NY} - \text{div}(x)_1 - \text{div}(x)_2 - \text{div}(x)_3 \\ &= -(-30) - (-40) - 40 - 40 - 40 \\ &= -50. \end{aligned}$$

Solution de la question 2:

A matrix is totally unimodular if the determinant of each square submatrix is 0, 1 or -1. We have to check whether the matrices A, B, C and D satisfy those conditions.

— Matrix A : Since there are two submatrices of size 1 for which the determinant is not 0, 1 or -1 but 2 and 4 respectively, the matrix A is not totally unimodular.

— Matrix B : each square submatrix of size 1 is 0, 1 or -1, and the determinants of the square submatrices of size 2 are

$$\det\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = -1, \quad \det\begin{pmatrix} 0 & 0 \\ 1 & 1 \end{pmatrix} = 0, \quad \det\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = 1.$$

Then, B is totally unimodular.

— Matrix C : each square submatrix of size 1 is 0, 1 or -1, but the determinant of the square submatrix of size 2

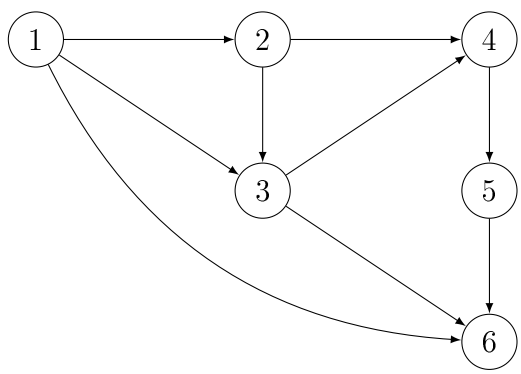
$$\begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}$$

is equal to 2. Then C is not totally unimodular.

— Matrix D : The dimension of matrix D is (6×9). If we want to check the determinant of each square submatrix of D, we would check all the submatrix from size 1 to size 6 which means :

$$\text{number of submatrix} = \sum_{k=1}^{6} \binom{6}{k} \times \binom{9}{k} = \sum_{k=1}^{6} \frac{6!}{k!(6-k)!} \times \frac{9!}{k!(9-k)!}$$

It is too tedious to try everything. In the other hand, we can see that in each column of matrix D we have a 1, a -1 and the other items are 0. This corresponds to the incidence matrix of the following graph of 6 nodes and 9 arcs :



We know that the incidence matrix of a graph is totally unimodular, then D is totally unimodular.

Solution de la question 3:

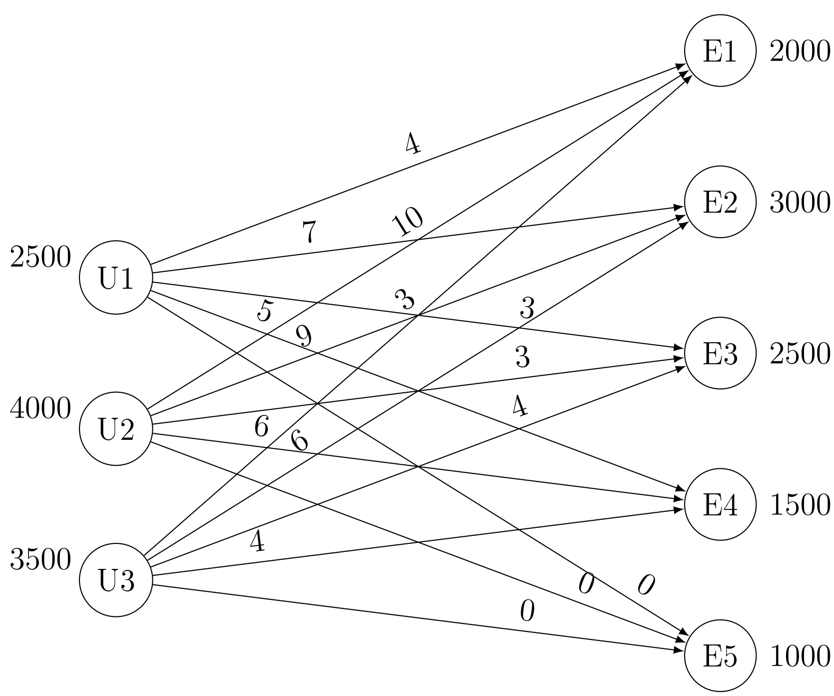
Afin de représenter ce problème sous la forme d'un problème de transbordement, nous devons construire le graphe $\mathcal{G} = (\mathcal{N}, \mathcal{A})$ correspondant. Les noeuds du graphe sont définis ainsi :

— Un noeud (offre) U_i est associé à chaque usine. La divergence de chaque noeud U_i correspond à P_i la capacité de production chaque mois de l'usine i.

— Un noeud (demande) E_j est associé à chaque entrepôt. La divergence de chaque noeud E_j correspond à l'opposé de D_j la demande par mois de chaque entrepôt j.

— Afin d'obtenir un problème de transbordement valide, nous devons avoir $\sum_{k\in\mathcal{N}} s_k = 0$. Comme la somme des divergence des noeuds actuels est égale à 1000, noud devons créer un dernier noeud demande U_5 dont la divergence est égale -1000. Ce noeuds va récupérer le contreplaqué fabriqué et non transporté aux entrepôts.

Les arcs sont définis ainsi :
Un arc (i, j) relie chaque noeud offre i à chaque noeud demande j. Le coût de l'arc (i, j) est égale à c_{ij} le coût de transport de l'usine i à l'entrepôt j pour $j \in \{1, ..., 4\}$. Le coût des arc $(i, 5)$ est égale à 0. La borne inférieur de chaque arc est 0 car on ne peut pas transporter une quantité négative. De plus, il n' y a pas de borne supérieur sur les arcs.
On obtient le graphe suivant :



A présent, écrivons le problème de transbordement correspondant :

— **Les variables de décision :**
  — x_{ij} qui représente la quantité de contreplaqué transportée de l'usine i à l'entrepôt j pour $i \in \{1, ..., 3\}$, $j \in \{1, ..., 4\}$.
  — x_{i5} qui représente la quantité de contreplaqué qui est restée à l'usine i pour $i \in \{1, ..., 3\}$.

— **Fonction objectif :** l'objectif est de minimiser le coût total du transport. C'est à dire :

$$\min \sum_{i=1}^{3} \sum_{j=1}^{4} c_{ij} x_{ij}.$$

— **Les contraintes :** Les demandes des enrepôts doivent être satisfaites exactement :

$$\sum_{i=1}^{3} x_{ij} = D_j, \forall j \in \{1, \ldots, 4\}.$$

La quantité de contreplaqué produite dans chaque usine est transporté vers les entrepôt ou reste sur place :

$$\sum_{j=1}^{5} x_{ij} = P_i, \forall i \in \{1, \ldots, 3\}.$$

On ne peut pas transporter des quantités négatifs :

$$x_{ij} \geq 0, \forall i \in \{1, \ldots, 3\}, \forall j \in \{1, \ldots, 5\}.$$

On obtient ainsi le problème d'optimisation suivant :

$$\begin{aligned} &\min \sum_{i=1}^{3} \sum_{j=1}^{4} c_{ij} x_{ij} \\ \text{s.c :} &\sum_{i=1}^{3} x_{ij} = D_j, \forall j \in \{1, \ldots, 4\} \\ &\sum_{j=1}^{5} x_{ij} = P_i, \forall i \in \{1, \ldots, 3\} \\ &x_{ij} \geq 0, \forall i \in \{1, \ldots, 3\}, \forall j \in \{1, \ldots, 5\} \end{aligned}$$

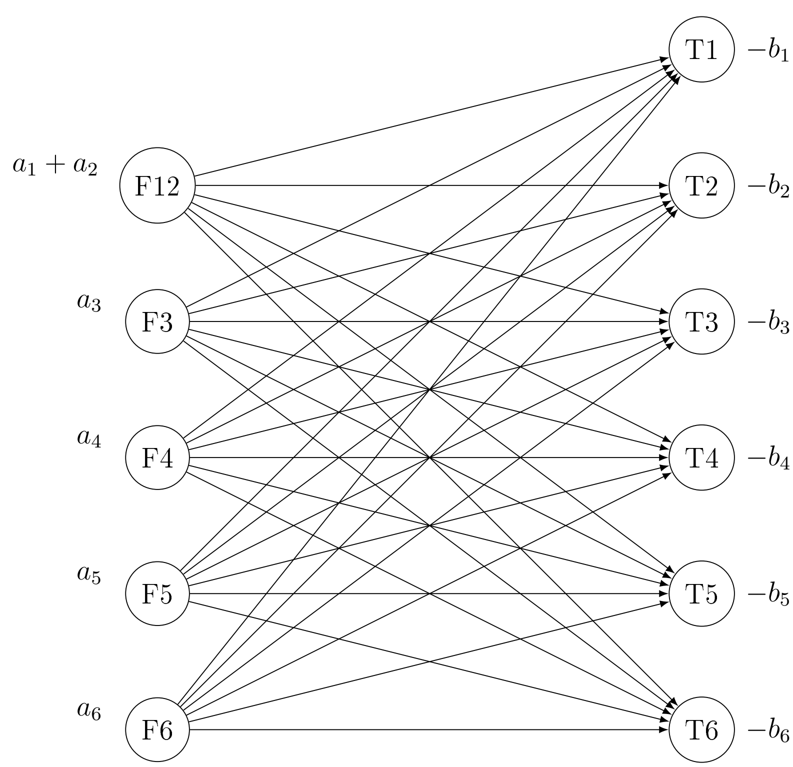
Solution de la question 4:

1. First, we are going to model the problem as a mathematical network $\mathcal{G} = (\mathcal{N}, \mathcal{A})$. We define the set of nodes $\mathcal{N}$ as follows :

- One node per table (and thus q nodes) with a demand equal to the opposite of the number of corresponding seats $-b_j, j \in \{1, \ldots, q\}$.
- One node per family, except for family 1 and 2 for which there is only one node since we do not want them to be seated at the same table. Thus we get $p-1$ nodes with supply $a_1 + a_2$ for the first one and with supply $a_i, i \in \{3, \ldots, p-1\}$ for the others.

Let's define the set of arcs $\mathcal{A}$. We introduce an arc between each pairs of family/table nodes. Those arcs have lower bound zero since we cannot have a negative number of persons seated, and an upper bound 1 since we cannot have two persons of the same family seated at the same table. Note that with this configuration, the two first families cannot be seated at the same table.
We associate arbitrarily a cost of 1 to each arc.

The goal is to find a flow vector that take into account the supply and demand of each node.
We obtain the following network for $p = q = 6$:



2. We are looking the necessary conditions for the existence of a solution.
   - A necessary condition for the feasibility of a transhipment problem is $\sum_{i=1}^{m} s_i = 0$, with m the number of nodes and s_i the supply/demand of each node. This condition is satisfied for this problem as the number of persons is equal to the number of seats.
   - The supply/demand of each node must be $\leq$ to its degree, indeed :
     - If there is a family with more member than tables, that is $\exists i \in \{1, \ldots, p\}$ such that $a_i > q$, then at least 2 members of this family will be seated in the same table and consequently there is no solution.
     - If the total number of member in the first and second one is greater than the number of tables, that is $a_1 + a_2 > q$, then there is no solution.
     - If there is a table with more seats than $p - 1$, that is $\exists j \in \{1, \ldots, q\}$ such that $b_j > p - 1$, then at least 2 members of the same family or one member of the first family and one member of the second family will be seated in this table and consequently there is no solution.
3. Lets write the problem as a transhipment problem using the above network formulation.
   - **Decision variables** Lets x_{ij} be a binary variable that is 1 if one person of family i seats to table j, and 0 otherwise. Note that family 1 and 2 are merged in the same family we denote by $12^{\star}$. The other families are still denoted from 3 to p.
   - **Objective function** The cost c_{ij} to attribute to family i a seat in table j correspond to the cost of the arc in the network which is 1. The objective function is then

$$\min \sum_{(i,j)\in\mathcal{A}} c_{ij} x_{ij} = \min \sum_{(i,j)\in\mathcal{A}} x_{ij}.$$

   - **Constraints** We must give a seat to each person of each family :

$$\sum_{j} x_{12^{\star},j} = a_1 + a_2$$

$$\sum_{j} x_{ij} = a_i, \forall i \in \{3, \ldots, p\}.$$

Each seat of each table must be assigned to one person :

$$\sum_i x_{ij} = b_j, \forall j \in \{1, \dots, q\}.$$

Two persons from the same family cannot be assigned to the same table :

$$x_{ij} \leq 1, \forall j \in \{1, \dots, q\}, \forall i \in \{12^\star, 3, \dots, p\}.$$

To finish, negative values do not make sense :

$$x_{ij} \geq 0, \forall j \in \{1, \dots, q\}, \forall i \in \{12^\star, 3, \dots, p\}.$$

Since, if we write the constraint in the form $Ax = b$, we have A totally unimodular and b integer, then the condition $x_{ij} \in \{0, 1\}$ is not needed.

We end up with the following optimization problem :

$$\min \sum_{(i,j)\in\mathcal{A}} x_{ij}$$

subject to

$$\begin{aligned}
\sum_j x_{12^\star,j} &= a_1 + a_2 \\
\sum_j x_{ij} &= a_i, \forall i \in \{3, \dots, p\} \\
\sum_i x_{ij} &= b_j, \forall j \in \{1, \dots, q\} \\
x_{ij} &\leq 1, \forall j \in \{1, \dots, q\}, \forall i \in \{12^\star, 3, \dots, p\} \\
x_{ij} &\geq 0, \forall j \in \{1, \dots, q\}, \forall i \in \{12^\star, 3, \dots, p\}.
\end{aligned}$$